

Міністерство освіти і науки України
Чернігівський національний технологічний університет

Ткач Ю.М., Трунова О.В., Мехед Д.Б.,
Базилевич В.М., Мурашківська В.П., Петренко Т.А.,
Гур'єв В.І., Фірсова І.В.

**СУЧАСНІ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ
ТЕХНОЛОГІЇ У НАВЧАННІ
МАТЕМАТИЧНИХ ДИСЦИПЛІН
В ЗАКЛАДАХ ВИЩОЇ ОСВІТИ**

Монографія

2016

УДК 004
ББК 22.1
С 91

Затверджено вченою радою Чернігівського національного технологічного університету (протокол № 3 від 28.03.2016 р.)

Рецензенти:

Корченко О.Г., завідувач кафедри безпеки інформаційних технологій Національного авіаційного університету, д.т.н., професор,

Дутка Г.Я., проректор Львівського інституту економіки і туризму, д.п.н., професор,

Горошко Ю.В., завідувач кафедри інформатики та обчислювальної техніки Чернігівського національного педагогічного університету ім. Т.Г.Шевченка д.п.н., професор.

**С 91 Сучасні інформаційно-комунікаційних технологій у навчанні математичних дисциплін в закладах вищої освіти / за заг. ред.Ткач Ю.М.; [Ткач Ю.М., Трунова О.В., Мехед Д.Б., Базилевич В.М., Мурашківська В.П., Петренко Т.А., Гур'єв В.І., Фірсова І.В.]. Ніжин: ФОП Лук'яненко В.В. ТПК «Орхідея», 2016. – 358 с.
ISBN 978-966-2185-82-9**

У монографії висвітлено теоретичні засади застосування сучасних інформаційно-комунікаційних технологій у навчанні математичних дисциплін (вищої математики, теорії ймовірностей та математичної статистики, економетрики, оптимізаційних методів та моделей тощо) та розкрито практичні аспекти використання різноманітних інформаційно-комунікаційних технологій у навчальному процесі, а саме, вільного (Open Office Calc, GeoGebra, Gran1, Gran2D, SciLab) та пропріетарного (Excel, MathCat, MatLab) програмного забезпечення.

Запропонована монографія буде сприятиме оволодінню студентами та викладачами сучасними інформаційно-комунікаційними технологіями (ІКТ), їх застосуванню у навчальному процесі закладів вищої освіти, виробленню стійких вмінь та навичок розв'язати професійні задачі засобами ІКТ та підвищенню ефективності навчального процесу в цілому. Тим самим забезпечуватиметься *фундаменталізація* професійної підготовки фахівців різних спеціальностей.

Монографія буде корисною викладачам математичних дисциплін вищих навчальних закладів та іншим науково-педагогічним працівникам при викладанні фахових дисциплін.

УДК 004
ББК 22.1

ISBN 978-966-2185-82-9

Ткач Ю.М., Трунова О.В., Мехед Д.Б.,
Базилевич В.М., Мурашківська В.П., Петренко Т.А.,
Гур'єв В.І., Фірсова І.В.

**СУЧАСНІ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНІ
ТЕХНОЛОГІЇ У НАВЧАННІ
МАТЕМАТИЧНИХ ДИСЦИПЛІН
В ЗАКЛАДАХ ВИЩОЇ ОСВІТИ**

Монографія

В авторській редакції

Відповідальний за випуск – Лук'яненко В.В.

Підписано до друку 28.11.2016 р.
Формат 60x 84/16. Друк офсетний. Папір офсетний.
Гарнітура Times New Roman. Обл.-вид. арк. 28,60.
Ум. друк. арк. 20,81. Тираж 300 прим.
Зам. № 483

Віддруковано з оригінал-макету замовника

Видавець - ФОП Лук'яненко В.В. ТПК «Орхідея»

*Свідоцтво про внесення суб'єкта видавничої справи
до державного реєстру видавців, виготівників
і розповсюджувачів видавничої продукції
серія ДК № 3020 від 02.11.2007 р.*

Друк - ФОП Черниш Ю.В.

16600, Чернігівська обл., м. Ніжин, вул. Небесної сотні, 13 а.
Тел.: (04631) 5-15-05,
E-mail: holdingvv@gmail.com

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ВИКОРИСТАННЯ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ У НАВЧАННІ МАТЕМАТИКИ У ЗАКЛАДАХ ВИЩОЇ ОСВІТИ В УМОВАХ ФУНДАМЕНТАЛІЗАЦІЇ ПРОФЕСІЙНОЇ ОСВІТИ.....	9
1.1. Стратегічні напрями розвитку вищого навчального закладу.....	9
1.2. Інформатизація навчального процесу у ВНЗ	12
1.3. Аналіз понятійного апарату з проблеми інформаційно-комунікаційних технологій у навчальному процесі ВНЗ.....	16
1.4. Проблеми застосування інформаційно-комунікаційних технологій у навчанні математики у ВНЗ	20
1.5. Програмні засоби навчального призначення у навчанні математики	27
1.6. Інтеграція математики, інформаційно-комунікаційних технологій та фахових дисциплін	40
1.7. Моніторинг навчальних досягнень студентів за допомогою інформаційно-комунікаційних технологій	50
РОЗДІЛ 2. ВИКОРИСТАННЯ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ У ПРОЦЕСІ НАВЧАННЯ МАТЕМАТИЧНИХ ДИСЦИПЛІН У ВНЗ.....	53
2.1. R-програмування	53
2.1.1. Історичні факти	53
2.1.2. Розвиток мови R	54
2.1.3. Особливості мови R	54
2.1.4. Недоліки мови R.....	55
2.1.5. Встановлення R.....	55
2.1.6. Встановлення R-Studio.....	60
2.1.7. Огляд RStudio.....	61
2.1.8. Бібліотеки	62
2.1.9. Використання R як калькулятора	65
2.1.10. Числа, вектори, матриці та масиви.....	65
2.1.11. Логічні вирази.....	72
2.1.12. R- пакети	77
2.1.13. Використання R за чотири кроки	77
2.1.14. Основи	78
2.1.15. X-Y графіки.....	79
2.1.16. X-Y графіки – умовна побудова	82
2.1.17. Темпи росту зоопланктону.....	84
2.1.18. Зображення та контурні графіки.....	84
2.1.19. Побудова математичних функцій.....	85
2.1.20. Робота з матрицями.....	87

2.1.21.	Корінь функції однієї змінної	88
2.1.22.	Інтерполяція, згладжування	90
2.1.23.	Диференціальні рівняння	91
2.1.24.	Статистика.....	93
2.2.	MATLAB	105
2.2.1.	Основи роботи з MatLab	106
2.2.2.	Операції з векторами і матрицями в системі MatLab	111
2.2.3.	Програмування в середовищі MatLab	117
2.2.4.	Робота з графікою засобами MatLab	123
2.2.5.	Розв'язання типових завдань алгебри і аналізу.....	129
2.2.6.	Апроксимація та інтерполяція даних	137
2.2.7.	Методи розв'язання звичайних диференціальних рівнянь	139
2.3.	Mathcad	143
2.3.1.	Призначення Mathcad.....	144
2.3.2.	Основні прийоми роботи з пакетом Mathcad	145
2.3.3.	Особливості інтерфейсу користувача	146
2.3.4.	Головне меню MathCad.....	147
2.3.5.	Використання інструментальних і вбудованих панелей MathCad.....	147
2.3.6.	Робота з формульним редактором.....	149
2.3.7.	Робота з текстовим редактором	154
2.3.8.	Масиви	155
2.3.9.	Побудова графіків в Mathcad	156
2.3.10.	Побудова графіків у декартовій системі координат	158
2.3.11.	Трьохвимірні графіки	160
2.3.12.	Форматування трьохвимірних графіків	161
2.3.13.	Робота з матрицями і векторами в Mathcad.....	163
2.3.14.	Функції визначення матриць та операції з блоками матриць.....	166
2.3.15.	Розв'язування систем лінійних рівнянь	167
2.3.16.	Оператори в Mathcad.....	169
2.3.17.	Алгебраїчні рівняння та системи.....	172
2.3.18.	Інтегрування.....	176
2.3.19.	Диференціювання.....	177
2.3.20.	Розв'язування диференціальних рівнянь	180
2.3.21.	Застосування MathCad для розв'язання задач оптимізації	182
2.3.22.	Програмування у MathCad	186
2.3.23.	Створення програмного модуля	187
2.3.24.	Оператори програмування.....	187
2.3.25.	Переваги використання Mathcad для студентів	190
2.3.26.	Переваги використання Mathcad для викладачів.....	190
2.3.27.	Mathcad Prime	190
2.4.	Scilab	191
2.4.1.	Арифметичні обчислення	191
2.4.2.	Змінні	193
2.4.3.	Системні змінні.....	194
2.4.4.	Функції.....	194

82. Brian S. Everitt and Torsten Hothorn (2005). A Handbook of Statistical Analyses Using R, London and Erlangen, <http://www.R-project.org>.

83. Hohenwarter M. Combination of Dynamic Geometry, Algebra and Calculus in the Software System GeoGebra [Electronic resource] / Markus Hohenwarter and Karl Fuchs // Computer Algebra Systems and Dynamic Geometry Systems in Mathematics Teaching Conference 2004. Pecs, Hungary. – Mode of access :http://www.geogebra.at/publications/pecs_2004.pdf , с. 123.

84. J. H. Maindonald (2001). Using R for Data Analysis and Graphics An Introduction. Statistical Consulting Unit of the Graduate School, Australian National University.

85. Jim Lemon. Kickstarting R, <http://cran.R-project.org/other-docs.html>.

86. John M. Chambers (2008). Software for Data Analysis. Programming with R, Springer Science+Business Media, LLC, USA, ISBN: 978-0-387-75935-7.

87. John Verzani. simpleR — Using R for Introductory Statistics, <http://cran.R-project.org/other-docs.html>.

88. Michael J. Crawley (2007). The R Book, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester. ISBN-13: 978-0-470-51024-7.

89. Phil Spector (2008). Data Manipulation with R, Springer Science+Business Media, LLC, USA, ISBN 978-0-387-74730-9.

90. Program complex QUIK: [Electron. resource]. - Mode of access: <http://www.quik.ru/about/>.

91. R Development Core Team (2009). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL <http://www.R-project.org>.

92. Stefan Steinhaus Comparison of mathematical programs for data analysis (Edition 4.4).— Munchen/Germany. – 60p. Режим доступу: <http://www.scientificweb.de/ncrunch/> (дата звернення 20.05.2016).

93. Tkach Yu. Mathematization of knowledge - the core of fundamentalization of professional training of the future economists // Science and Education a New Dimension. Pedagogy and Psychology, IV(40), Issue 81, 2016. – P.70-73.

94. Ukrainian Exchange / Index description: [electrons. resource]. - Mode of access: <http://www.ux.ua/ru/index/ux/description.aspx>.

95. Yosef Cohen and Jeremiah Y. Cohen (2008). Statistics and Data with R, John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, ISBN 978-0-470-75805-2.

2.4.5.	Функції користувача	195
2.4.6.	Логічні оператори.....	197
2.4.7.	Побудова двовимірних графіків	201
2.4.8.	Задачі математичного аналізу	209
2.4.9.	Інтегрування функції, заданої формулою.....	210
2.4.10.	Інтегрування таблично заданої функції.....	210
2.4.11.	Інтегрування по квадратурі	212
2.4.12.	Обчислення похідної.....	213
2.4.13.	Частинна похідна.....	214
2.4.14.	Нелінійні рівняння та системи.....	214
2.4.15.	Алгебраїчні рівняння	216
2.4.16.	Трансцендентні рівняння.....	219
2.4.17.	Системи рівнянь	226
2.4.18.	Завдання оптимізації.....	229
2.4.19.	Локальний мінімум функцій однієї змінної	229
2.4.20.	Побудова тривимірних графіків	232
2.4.21.	Пошук мінімуму	233
2.4.22.	Масиви і матриці	234
2.4.23.	Формування одновимірних масивів	234
2.4.24.	Формування матриці	236
2.4.25.	Дії над векторами	238
2.4.26.	Дії над матрицями	240
2.4.27.	Лінійна і векторна алгебра	241
2.4.28.	Матричні розкладання	244
2.4.29.	Система лінійних алгебраїчних рівнянь (СЛАР).....	246
2.4.30.	Рішення СЛАР методом зворотної матриці	246
2.4.31.	Рішення СЛАР методом Гауса.....	248
2.5.	GeoGebra	249
2.5.1.	Використання для побудови текстового поля введення	252
2.5.2.	Використання для побудови поля введення тексту і миші.....	253
2.5.3.	Дотичні без даної команди	254
2.5.4.	Інститути GeoGebra.....	257
2.5.5.	Використання GeoGebra у шкільному курсі математики	257
2.5.6.	Веб-сервіси "GeoGebraTube" та "Бібліотека комп'ютерних моделей".....	261
2.5.7.	Інсталяція GeoGebra.....	262
2.5.8.	Запуск GeoGebra	263
2.5.9.	Вікно програми GeoGebra	263
2.5.10.	Панель інструментів.....	264
2.5.11.	Меню GeoGebra	278
2.5.12.	Практикум з розв'язування математичних задач та створення комп'ютерних моделей.....	281
2.6.	Gran.....	291
2.6.1.	GRAN 1.....	292
2.6.2.	GRAN – 2D	299
2.7.	MS Excel.....	305

2.7.1. Побудова парної економетричної моделі із використанням можливостей табличного процесору EXCEL.....	305
2.7.2. Побудова багатофакторної економетричної моделі за допомогою можливостей табличного процесора excel.....	313
2.7.3. Побудова багатофакторної лінійної економетричної моделі на основі матричного оператора Імнк.....	323
2.7.4. Використання стандартної програми «линейн».....	326
2.7.5 Дослідження мультиколінеарності.....	328
2.7.6. Точковий прогноз.....	336
2.8. OpenOffice Calc.....	337
2.8.1. Загальні відомості про OpenOffice Calc.....	337
2.8.2. Використання OO calc для вирішення математичних задач.....	342
ДОДАТОК 4.....	345
Список авторів.....	350
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	351

позабіржових торговельних систем в режимі реального часу // [Электронный ресурс]. – Режим доступа: www.kinto.com/ukr/itrade/software/programm.

66. Просветов Г.И. Математические методы и модели в экономике: задачи и решения: Учебно-практическое пособие.- М.: Издательство «Альфа-Пресс», 2008.– 344 с.

67. Руссо Ж.-Ж. Исповедь. Мысль о науке [Текст]/ Ж.-Ж.Руссо. – М.: Педагогика, 1981. – 81 с.

68. Смеричевская С.В. Макроэкономические аспекты формирования управленческих стратегий в инновационном развитии системы образования Украины // Стратегія інноваційного розвитку системи вищої освіти України: Матеріали міжнародної науково-практичної конференції / Голов. ред. С.В. Смеричевська. - Донецьк: Видавничий дім «Кальміус», 2007.-С. 8-12.

69. Співаковський О. В. Теорія і практика використання інформаційних технологій у процесі підготовки студентів математичних спеціальностей. / О. В. Співаковський. – Херсон: Айлант, 2003. – 229 с.

70. Субетто А.И. Проблемы фундаментализации и источников формирования содержания высшего образования: грани государственной политики. - Кострома: Костр. пед. ун-т, 1995. - 332 с.

71. Сясєв А.В. Вступ до системи MathCad: Навч. посіб.-Д: Вид-во Дніпропетр. Ун-ту, 2004. – 108 с.

72. Тихомиров О.К. Стратегия и тактика компьютеризации / О.К.Тихомиров // Вестник высшей школы. – 1988. - №3. – С. 15-17.

73. Трайнев В.А. Информационные коммуникационные педагогические технологии (обобщения и рекомендации) : учеб. пособ. / В.А. Трайнев, И.В. Трайнев. – М. : Дашков и Ко , 2008. – 280 с.

74. Триус Ю.В. Використання систем комп'ютерної математики при вивченні і розв'язуванні задач оптимізації // Проблеми сучасного підручника: Зб. наук. праць / Ред. кол.– К.: Педагогічна думка, 2004. – Вип. 5.– Ч.ІІ.– С. 191-200.

75. Триус Юрій Васильович. Комп'ютерно-орієнтовані методичні системи навчання математичних дисциплін у вищих навчальних закладах : автореф. дис... д-ра пед. наук: 13.00.02 / Національний педагогічний ун-т ім. М.П.Драгоманова. - К., 2005. – 61 с.

76. Усе (універсальний словник-енциклопедія) // [Электронный ресурс]. – Режим доступа: <http://slovo.org.ua/29/53407/18910.html>.

77. Ушинский К.Д. Сочинения. Т.5. – 355 с.

78. Чепиков М.Г. Интеграция науки(Филос. Очерки) [Текст]/М.Г.Чепиков. – М.: Мисль, 1981. – 276 с.

79. Энциклопедический словарь / Гл. ред. Б.А. Введенский. В 3-х т. – М.: Изд. «Большая сов. энцикл.», 1953. – 1955 с.

80. Яковлев И.П. Интеграция высшей школы с наукой и производством [Текст]/И.П.Яковлев. – Л: Изд-во ЛГУ, 1987. – 202 с.

81. Ястребов А.В. Дуалистические свойства математики и их отражение в процессе преподавания // Ярославский педагогический вестник. - №1, 2001. – С. 48-53 <http://www.km.ru/referats/D6A3B95449CF4FF5AC11BF231357EE7F>.

49. Куклін О.В. Концептуальні засади інноваційного розвитку вищих навчальних закладів // Електронне наукове фахове видання "Ефективна економіка" включено до переліку наукових фахових видань України з питань економіки (Наказ Міністерства освіти і науки України від 29.12.2014 № 1528) №6, 2012.

50. Лазарев Ю.Ф. Моделирование процессов и систем в MATLAB. Учебный курс. – СПб.: Питер; Киев: Издат. группа BHV, 2005. – 512 с.

51. Лазарев Ю.Ф. Довідник з MATLAB / Електронний навчальний посібник з курсового і дипломного проектування. – К.: НТУУ "КПІ", 2013. – 132 с.

52. Левина М.М. Основы технологии профессионального педагогического образования / М.М. Левина // М-во образования Рос. Федерации. Моск. Пед. Гос. ун-т, М-во образования Респ Беларусь. Акад. Последиплом. Образования. - Минск, 1998. - 344с.

53. Мартынов Г. Г., Иванов А. П. MATLAB 5.x, вычисления, визуализация, программирование. - М.: «Кудиц-образ», 2000. - 332 с.

54. Машбиц Е.И. Психолого-педагогические проблемы компьютеризации обучения / Е.И.Машбиц. – М.: Педагогіка, 1988. – 192 с.

55. Меморандум международного симпозиума ЮНЕСКО // Высшее образование в России. – 1994. - №4. – С.4-6.

56. Національна доктрина розвитку освіти України у ХХІ ст. // [Електронний ресурс]. – Режим доступа: <http://zakon5.rada.gov.ua/laws/show/347/2002>.

57. Павельциг Г. Интегрция – дифференциация – прогресс / Интегративные тенденции в современном мире и социальный прогресс / Под ред. М.А.Розова. – М.: МГУ, 1989. – С. 27-42.

58. Песталоцци И.Г. Избранные педагогические сочинения. – М., 1963. – Т.2. – 175 с.

59. Пилипчук А.Ю. Реформування освіти та інформатизація: основні проблеми і підходи до їх вирішення <http://www.ime.edu-ua.net/em5/cont.html>

60. Пойа Д. Математическое открытие. Решение задач: основные понятия, решение и преподавание / Пойа Д. ; пер. с англ. – М.: Наука, 1970. – 452 с.

61. Попович Н. М. Вплив інформаційно-комунікаційних технологій на якість підготовки фахівців у ступеневій педагогічній освіті / Попович Н. М. // Вісник Житомирського державного університету імені Івана Франка (педагогічні науки). – 2009. – № 47. – С. 95–99.

62. Постанова кабінету міністрів України «Про затвердження державної програми «Інформаційні та комунікаційні технології в освіті і науці» на 2006-2010роки» від 7 грудня 2005 року, №1153 // [Електронний ресурс]. – Режим доступа: <http://zakon4.rada.gov.ua/laws/show/1153-2005-п>.

63. Постанова КМУ «Про затвердження загальних вимог до програмних продуктів, які закупаються та створюються на замовлення державних органів» від 12 серпня 2009 р. № 869.

64. Потемкин В. Г., Рудаков П. И. MatLAB 5 для студентов. - 2-е изд., испр. и дополн. - М.: ДИАЛОГ-МИФИ, 1999. - 448 с.

65. Програмний комплекс для організації доступу до біржових і

ВСТУП

Суспільство на сучасному етапі розвитку характеризується високими темпами науково-технічного прогресу, збільшенням цінності інформації й питомої ваги інформаційних послуг у всіх сферах життєдіяльності людини, разом з тим, відбувається процес інформатизації різних галузей науки, освітньої зокрема.

Необхідність використання інформаційно-комунікаційних технологій (ІКТ) у навчанні фундаментальних дисциплін, зокрема математики, сьогодні вже не потребує підтвердження та пропагування. ІКТ надають можливість візуалізації навчального матеріалу, економії часу на занятті, дозволяють зробити освітній процес більш наочним та ефективним, подолати перепони, що були утворені традиційним, надто формалізованим та абстрактним викладом багатьох університетських дисциплін.

У закладах вищої освіти хоча і усвідомлюють потребу у проведенні занять із використанням сучасних ІКТ, але комплексного дидактичного забезпечення із застосування різноманітних програмних засобів у навчанні поки що не створено, математичні дисципліни не виняток.

Галузеві стандарти (освітньо-кваліфікаційні характеристики, освітньо-професійна програма) більшості спеціальностей, які є основою та визначають зміст професійної підготовки, передбачають впровадження такої системи освіти, яка забезпечить не тільки оволодіння фахівцем поглибленими знаннями з галузі, а і вміннями інноваційного характеру, навичками науково-дослідної та управлінської діяльності. Це потребує вирішення відповідних методологічних, теоретичних і технологічних проблем, пов'язаних з: проектуванням структури та змісту професійної підготовки студентів вказаних спеціальностей; використанням принципу подвійного випередження (на основі фундаменталізації та методологізації змісту професійної освіти); реалізацією принципу гуманізму; відбором форм та методів професійної підготовки, що враховують інтереси освітньої системи та самих майбутніх фахівців; професійним відбором і діагностикою професійної придатності тощо.

Згідно «Основних засад розвитку інформаційного суспільства в Україні на 2007-2015 роки», затверджених Законом України № 537-в від 9 січня 2007 року, однією з основних цілей розвитку інформаційного суспільства в Україні є забезпечення комп'ютерної та інформаційної грамотності населення, насамперед шляхом створення системи освіти, орієнтованої на використання новітніх інформаційно-комунікаційних технологій (ІКТ) у формуванні всебічно розвиненої особистості [32, 62]. Тому підходи до навчання в цілому, математики зокрема, потребують суттєвих змін.

Враховуючи все вище сказане, навчальний процес має відбуватись із використанням сучасних інформаційно-комунікаційних технологій, що також сприятиме розв'язанню проблем фундаменталізації професійної підготовки майбутніх фахівців різних галузей, активізації розумової діяльності студентів та розвитку їхніх інтелектуальних умінь.

На думку більшості дослідників, основним педагогічними завданнями використання інформаційно-комунікаційних технологій у навчанні є:

- підвищення наочності навчального матеріалу та полегшення його сприйняття завдяки компактному і чіткому поданню інформації;
- розвиток творчого потенціалу суб'єктів навчання, їх здібностей комунікативних дій, умінь експериментально-дослідницької діяльності; культури навчальної діяльності, підвищення мотивації навчання;
- інтенсифікація всіх рівнів навчально-виховного процесу, підвищення його ефективності та якості;
- розширення та поглиблення змісту навчання з дисципліни, що вивчається;
- засвоєння повного спектру понять, операцій і функцій, вільне оперування якими передбачено змістом навчальної дисципліни;
- реалізація соціального замовлення, зумовлена інформатизацією сучасного суспільства [61].

До основних шляхів застосування ІКТ в освіті ми відносимо:

- створення педагогічних програмних засобів (ППЗ),
- застосування інформаційно-комунікаційних технологій під час навчання,
- застосування інформаційно-комунікаційних технологій до моніторингу рівня навчальних досягнень студентів.

Проте шляхи використання інформаційно-комунікаційних технологій у навчальному процесі не обмежується лише тими, що ми перерахували вище. Крім того, ІКТ має місце під час впровадження дистанційного навчання, а також час підготовки та проведення занять викладачем (мультимедійні засоби навчання) тощо.

[Электронный ресурс]. – Режим доступа: <http://zakon.rada.gov.ua/cgi-bin/laws/main.cgi>.

33. Закон України «Про основні засади розвитку інформаційного суспільства в Україні на 2007–2015 роки» від 9 січня 2007 року № 537-V.– <http://zakon.rada.gov.ua/cgi-bin/laws/main.cgi>.

34. Закон України «Про Основні засади розвитку інформаційного суспільства в Україні на 2007–2015 роки» від 9 січня 2007 року, № 537-v // [Электронный ресурс]. – Режим доступа: <http://zakon2.rada.gov.ua/laws/show/537-16>.

35. Захарова И.Г. Информационные технологии в образовании : учеб. пособ. для студ. высш. пед. учеб. завед. / И.Г. Захарова. – М. : Академия, 2003. – 192 с.

36. Зубов А.В. Информационные технологии в лингвистике : учеб. пособ. для студ. лингв. фак-тов высш. учеб. завед. / А.В. Зубов, И.И. Зубова – М. : Академия, 2004. – 208 с.

37. Инновационные технологии в гуманитарном вузе / [Носков В.И., Кальянов А.В., Мирошниченко О.В. и др.]. - Донецк : ООО "Лебедь", 2002. - 288 с.

38. Иванчук М.Г. Основы технологии интегрированного навчання в початковій школі. – Чернівці. – 2000. – С. 36-62

39. Інформатизація // [Электронный ресурс]. – Режим доступа: <http://uk.wikipedia.org/wiki/%D0%86%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%B7%D0%B0%D1%86%D1%96%D1%8F>.

40. Інформаційні технології // [Электронный ресурс]. – Режим доступа: <http://studopedia.org/2-162857.html>.

41. Інформаційно-комунікаційні технології // [Электронный ресурс]. – Режим доступа: <http://klasnaocinka.com.ua/uk/article/informatsiino-komunikatsiini-tekhnologiyi--yak-suc.html>.

42. Інформаційно-комунікаційні технології // [Электронный ресурс]. – Режим доступа: <http://foldoc.org/Information+and+Communication+Technology>.

43. Кадемія М. Ю., Шахіна І. Ю. Інформаційно-комунікаційні технології в навчальному процесі : Навчальний посібник / Кадемія М. Ю., Шахіна І. Ю. / – Вінниця, ТОВ «Планер». - 2011. – 220 с.

44. Кедров Б.М. Синтез современного научного знания. – М.: Наука, 1973. – 224 с.

45. Кетков, Ю.Л. MATLAB 6.x . Программирование численных методов / Ю. Л. Кетков, А. Ю. Кетков, М.М. Шульц.– СПб.: БХВ-Петербург, 2004.–672 с.

46. Коменський Я.А. Избранные педагогические сочинения. – М., 1955. – 320 с.

47. Комп'ютерна грамотність // [Электронный ресурс]. – Режим доступа: http://uk.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D1%8E%D1%82%D0%B5%D1%80%D0%BD%D0%B0_%D0%B3%D1%80%D0%B0%D0%BC%D0%BE%D1%82%D0%BD%D1%96%D1%81%D1%82%D1%8C.

48. Кудрявцева С.П., Колос В.В. Міжнародна інформація. Навчальний посібник для студентів вищих навчальних закладів. 2-е видання. – К.: Видавничий Дім „Слово”, 2008. – 400 с.

15. Державна національна програма «Освіта» («Україна ХНІ століття») // [Электронный ресурс]. – Режим доступа: <http://zakon3.rada.gov.ua/laws/show/896-93-п>.

16. Дзюбенко А.А. Новые информационные технологии в образовании / А.А. Дзюбенко. – М., 2000. – 104 с.

17. Дистервег Ф.А.В. О природосообразности и культуросообразности в обучении // Избр. пед. соч. М.: АПН РСФСР, 1956. – С. 227.

18. Довгопол И.И. Современные образовательные и педагогические технологии / И.И. Довгопол, Т.А. Ивкова. – Симферополь, 2006. – 336 с.

19. ДСТУ 2938-94 Основні поняття. Терміни та визначення // [Электронный ресурс]. – Режим доступа: <http://document.ua/sistemi-obroblennja-informaciyi.-osnovni-ponjattja.-termini--std1033.html>.

20. Дутка Г.Я. Принцип фундаменталізації та його реалізація у математичній підготовці майбутніх економістів: дис... д-ра пед. наук: 13.00.04 / Г.Я. Дутка; АПН України; Ін-т пед. освіти і освіти дорослих. – К., 2009. – 412 с.

21. Дьяконов В. П. Компьютерная математика. Теория и практика. – М.: Нолидж, 2001.– 1296 с.

22. Дьяконов, В.П. MATLAB: учеб. курс/ В.П. Дьяконов. – СПб.: Питер, 2001.– 560 с.

23. Дьяконов, В.П. Simulink 4. Специальный справочник / В.П. Дьяконов.– СПб.: Питер, 2002.– 528 с.

24. Ершов А.П. Компьютеризация школы и математического образования / А.П.Ершов // Математика в школе. – 1989. - №1. – С. 14-30.

25. Жалдак М. І. Комп'ютер на уроках математики : посіб. [для вчителів] / М. І. Жалдак. – К. : Техніка, 1997. – 304 с.

26. Жалдак М. І. Комп'ютер на уроках математики : посібник [для вчителів] / М. І. Жалдак. – К. : Техніка, 1997. – 304 с.

27. Жалдак М. І. Педагогічний потенціал комп'ютерно-орієнтованих систем навчання математики // Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць / Редкол. – К.: НПУ імені М. П. Драгоманова. – Вип. 7. – 2003. – С. 3–16

28. Жалдак М. І. Проблеми інформатики та інформатизації / М. І.Жалдак // Комп'ютер у школі та сім'ї. – 1998. – № 1. – С. 3-4.

29. Жалдак М.І. Педагогічний потенціал комп'ютерно-орієнтованих систем навчання математики // Комп'ютерно-орієнтовані системи навчання: Зб. наук. праць / Редкол. – К.: НПУ імені М. П. Драгоманова. – Вип. 7. – 2003.– С. 3–16.

30. Жильцов О. Б. Развитие познавательной деятельности учащихся 7 классов средней школы при изучении математики с использованием ИТ : дис. ... канд. пед. наук : 13.00.02 / Жильцов Олександр Борисович; УДПУ ім. М. П. Драгоманова. – К., 1994. – 227 с.

31. Зайцева И.В. Математическое моделирование самоорганизующихся экономических процессов : дис. канд. физ.-мат. наук : 05.13.18 / Зайцева Ирина Владимировна. - Ставрополь, 2005. - 142 с.

32. Закон України «Про основні засади розвитку інформаційного суспільства в Україні на 2007–2015 роки» від 9 січня 2007 року № 537-V //

РОЗДІЛ 1. ТЕОРЕТИЧНІ ОСНОВИ ВИКОРИСТАННЯ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ У НАВЧАННІ МАТЕМАТИКИ У ЗАКЛАДАХ ВИЩОЇ ОСВІТИ В УМОВАХ ФУНДАМЕНТАЛІЗАЦІЇ ПРОФЕСІЙНОЇ ОСВІТИ

1.1. Стратегічні напрями розвитку вищого навчального закладу

Сучасний етап розвитку світової та трансформації вітчизняної економіки передбачає створення суспільства, яке базується на знаннях, де соціальний та економічний добробут країни визначається наукоємкими технологіями, інноваційною спрямованістю та рівнем інтелектуального розвитку соціуму. В умовах глобалізації, посилення конкуренції фундаментального значення для національної економіки набуває розвиток ринку освітніх послуг [68].

Тому, освітні тенденції 90-х рр. минулого століття, а саме диференціація та спеціалізація навчання, що покликані були знизити навчальне навантаження та зменшити термін навчання для найшвидшого включення молоді людини у суспільно-економічні стосунки, залишаються у минулому.

Перехід до нової освітньої парадигми, в основі якої лежить фундаменталізація освіти, передбачає якісно нові цілі освіти (навчання), дидактичні основи та закономірності, нові принципи відбору та систематизації знань, такі що не тільки розширюють та поглиблюють професійні знання, а і визначають різні взаємозв'язки між знаннями, уміннями та навичками, сприяють формуванню системного, цілісного теоретико-методологічного знання, а також наукового, дослідницького та творчого стилю мислення студентів.

Так, у Меморандумі міжнародного симпозіуму ЮНЕСКО зазначено, що фундаментальна цілісна природничо-наукова і гуманітарна освіта, покликана відіграти ключову роль у формуванні особистості і забезпеченні стійкого розвитку суспільства, має розглядатись як самостійна і найважливіша галузь інтелектуальної діяльності: надзвичайно важливо привернути увагу світової наукової і культурної спільноти до проблеми створення інтелектуального підґрунтя сучасної системи освіти, до розвитку наукових досліджень з логіки, методології, філософії, історії, соціології і психології науки, а також до розробки сучасних технологій навчання в галузі фундаментальної освіти; інтереси формування єдиного освітнього простору в рамках світової спільноти можуть бути забезпечені шляхом розробки загальних підходів до міжнародних освітніх стандартів, вимог до фундаментальної освіти». За висновками ЮНЕСКО, фундаментальна освіта оцінюється в усьому світі як один з основних чинників національної безпеки, стійкого розвитку країни, забезпечення її високого статусу в світовому співтоваристві. Під фундаментальністю освіти мають на увазі спрямованість її змісту на методологічні, інваріантні елементи знань, які сприяють ініціації, розвитку і реалізації інтелектуального і творчого потенціалу учнів [55].

Фундаменталізація вищої професійної освіти сьогодні складає один з пріоритетних напрямів державної освітньої політики та наукових досліджень в галузі педагогіки. Оскільки саме фундаменталізація спрямована на становлення

цілісної наукової картини оточуючого світу та інтелектуальний розквіт особистості та сприяє істотному підвищенню якості освіти і освітнього рівня людей шляхом відповідної зміни змісту дисциплін, що вивчаються, і методології навчального процесу.

На думку А.І.Субетто [70, с. 11] у професійній освіті фундаменталізація гарантує:

- системний рівень пізнання дійсності, здатність бачити та досліджувати механізми самореалізації та саморозвитку явищ та процесів;

- формування найсуттєвіших, довготривалих знань, що лежать в основі цілісного сприйняття сучасної картини світу,

- формування цілісного енциклопедичного погляду на сучасний світ та місце людини у ньому;

- оволодіння основам єдиної людської культури у її природничонауковій та гуманітарній площинах;

- створення бази професійної культури та професійної майстерності.

Головним ядром фундаменталізації А.І.Субетто вважає універсалізацію, а одним з найважливіших аспектів фундаменталізації він визначив математизацію знань.

Ми погоджуємось з його думкою, оскільки на сучасному етапі розвитку суспільства більшість видів діяльності неподільно пов'язані із формалізацією, ключовим елементом якої є математика та її засоби.

Математика важлива для фахівців будь-якого профілю та спеціальності, оскільки з її допомогою вони можуть пізнавати навколишній світ, з'ясувати сутність явищ та процесів, моделювати їх діяльність та прогнозувати подальший розвиток або здійснювати наукові дослідження в цілому. Крім того, математика є базою для подальшого вивчення фахових дисциплін, сприяє розумовому розвитку студентів та формуванню у них наукової цілісної картини світу. Тому дуже складно знайти золоту середину між фундаментальністю викладу матеріалу і доступністю для сприйняття.

Невелика кількість годин, що виділяються для вивчення курсу, значні обсяги знань у галузі математики сьогодні не дозволяють у повному обсязі викласти матеріал студентам протягом занять з математики. Це спонукає до пошуків шляхів інтенсифікації навчального процесу.

Одним із шляхів вирішення проблеми інтенсифікації навчального процесу з математичних дисциплін є застосування інформаційно-комунікаційних технологій у освітньому процесі.

Таким чином, математика є потужним інструментарієм аналізу та прогнозування будь-яких явищ та процесів. У поєднанні з величезними можливостями інформаційно-комунікаційних технологій вони дають новий поштовх до удосконалення та розвитку вищої освіти в цілому.

Опитання, яке ми провели серед роботодавців в різних галузях показало, що випускники вищих навчальних закладів мають володіти:

- фундаментальними (теоретичними та практичними) знаннями, що сприятиме зменшенню витрат підприємства на перепідготовку та підвищення кваліфікацію,

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Алексашина И.Ю. Интегративный подход в естественнонаучном образовании / И.Ю. Алексашина // Народное образование. - 2001. - № 1. - С. 161
2. Алексеев Е.Р., Чеснокова О.В. Основы работы в математическом пакете MathCad. Учебное пособие.-Издательство: ДонНТУ, 2012. – 187с.
3. Антонов Н.С. Интегративная функция обучения [Текст]/Н.С.Антонов //Современные проблемы методики преподавания математики. – М.: Просвещение, 1985. С.25-30.
4. Безрукова В.С. Словарь нового педагогического мышления. – Екатеринбург: Свердловськ. Обл. ИСУ, 1992. – 94 с.
5. Берулава М.Н. Интеграция содержания образования [Текст] / М.Н.Берулава. – Бийск: Изд-во БиГПИ, 1993. – 172 с.
6. Васяк Л.В. Формирование профессиональной компетентности будущих инженеров в условиях интеграции математики и спецдисциплин средствами профессионально ориентированных задач : дис. канд. пед. наук: 13.00.02 / Васяк Любов Владимировна. – Чита, 2007. – 170 с.
7. Веб-сайті Біржі / Futures Contract / Definitions [Electron. resource]. - Mode of access: http://uk.m.wikipedia.org/wiki/%D0%A4%D1%8E%D1%87%D0%B5%D1%80%D1%81%D0%BD%D0%B8%D0%B9_%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%B0%D0%BA%D1%82
8. Гершунський Б.С. Компьютеризация в сфере образования / Б.С.Гершунский. – М.: Педагогіка, 1987. – 264 с.
9. Глушков В.М. Основы безбумажной информатики / В.М.Глушков. – М.: Наука, 1987. – 552 с.
10. Говорухин В., Цибулин В. Компьютер в математических исследованиях.–СПб.: Питер, 2001.– 624 с.
11. Григорьев С.Г. Использование информационных и коммуникационных технологий в общем среднем образовании / С.Г. Григорьев, В.В. Гриншкун // [Электронный ресурс]. – Режим доступа: <http://www.ido.rudn.ru/nfpk/ikt/vved.html>.
12. Гультьев, А. П. Визуальное моделирование в среде MATLAB: учеб. курс/ А. П. Гультьев.– СПб.: Питер, 2000.– 432 с.
13. Гур'єв В.І., Фірсова І.В. Моделювання транспортної задачі за допомогою Mathcad // Математичне та імітаційне моделювання систем. МОДС 2015: тези доповідей 10 міжнародної науково-практичної конференції (Чернігів, 22-26 червня 2015 р.). – Чернігів: ЧНТУ, 2015. – 472 с. – С. 229-233.
14. Гур'єв В.И., Фирсова И.В. Решение оптимизационных экономических задач средствами mathcad // Историко-правовые, социально-экономические и технологические аспекты развития общества : междунар. науч.-практ. конф., 25 марта 2015 г., г. Чернигов: материалы докл. и выступлений/ Укр.-Рос. ин-т (фил.) Федер. гос. бюджет. образоват. учреждения высш. проф. образования «Моск. гос. машиностроит. ун-т (МАМИ)» – Чернигов: Десна Полиграф, 2015. – 164 с. – С. 88-92.

Список авторів

ПІБ	Розділ (пункт) монографії
Ткач Ю.М.	Розділ 1, п.2.6. 2.7
Трунова О.В.	п.2.2
Мехед Д.Б.	п.2.4
Базилевич В.М.	п.2.8
Мурашківська В.П.	п.2.1
Петренко Т.А.	п.2.5
Гур'єв В.І., Фірсова І.В.	п.2.3

- інформаційно-комунікаційними технологіями (це і програми Microsoft Office, вільне програмне забезпечення (ВПЗ), спеціальні програмні продукти – 1С: бухгалтерія, 1С: підприємство тощо),

- професійними якостями особистості (ініціативність, самостійність в роботі, організаторські здібності, здатність до саморозвитку, готовність швидко та без значних фізичних зусиль та матеріальних затрат змінити напрям або профіль діяльності,

- іноземними мовами (найбільш часто вживаними є англійська, німецька, французька та китайська).

Самі випускники закладів вищої освіти зазначають, що їм не вистачає практичного аспекту підготовки, навичок роботи із різними програмними продуктами, зокрема вільним та пропрієтарним програмним забезпеченням, недостатньо проведені паралелі між фундаментальними дисциплінами та дисциплінами фахового циклу підготовки тощо.

Тому, одним із шляхів вирішення проблеми реформування вищої освіти (у контексті її фундаменталізації) це фундаменталізація та інформатизація професійної підготовки майбутніх фахівців всіх профілів.

Таким чином, інтеграція ІКТ у освітній процес є необхідною умовою та стратегічним напрямом модернізації системи вищої освіти. Навички володіння різноманітними програмними засобами сьогодні стали невід'ємною складовою професійних компетентностей майбутнього фахівця будь-якої галузі.

Стрімке зростання ролі інформаційно-комунікаційних технологій та відповідного програмного забезпечення, як знаряддя інтелектуальної діяльності, висувають якісно нові вимоги до освітнього процесу у сучасному закладі вищої освіти.

Підтвердженням цьому є те, що у Концепції інноваційного розвитку України визначені такі стратегічні пріоритети розвитку вищої освіти:

а) забезпечення інноваційної спрямованості системи освіти шляхом: підвищення рівня комп'ютеризації вищих навчальних закладів; активізації науково-технічної та інноваційної діяльності ВНЗ та збільшення обсягу її бюджетного фінансування; утворення інноваційних структур в системі освіти; реформування системи вищої освіти з урахуванням вимог європейських стандартів і збереження культурних та інтелектуальних національних традицій;

б) підвищення результативності вітчизняного сектору наукових досліджень і розробок з метою посилення його ролі у забезпеченні інноваційного розвитку національної економіки шляхом: інтеграції університетського сектору наукових досліджень і розробок до світової інноваційної системи, сприяння розвитку міжнародного партнерства у сфері науково-технічної та інноваційної діяльності; збільшення частки наукових та науково-технічних досліджень у ВНЗ, спрямованих на створення нових видів інноваційної продукції; врахування кон'юнктури світового ринку під час визначення пріоритетних напрямів розвитку науки і техніки та інноваційної діяльності;

в) забезпечення розширеного відтворення знань на основі інтеграції діяльності вищих навчальних закладів, академічних та галузевих наукових установ шляхом: підвищення рівня фондоозброєності державного сектору наукових досліджень і розробок; концентрації ресурсів на пріоритетних напрямках розвитку науки і техніки та інноваційної діяльності, створенні умов для організації високотехнологічного виробництва в Україні; посилення взаємодії освіти та науки, створення університетів дослідницького типу на базі провідних ВНЗ; стимулювання процесу підвищення кваліфікації, розвиток системи дистанційного навчання [49].

Шляхами удосконалення навчального процесу ВНЗ є інтеграція (вбудовування) інформаційно-комунікаційних технологій у діючі дидактичні системи, гармонійне поєднання традиційних і комп'ютерно-орієнтованих технологій навчання, не заперечування і відкидання здобутків педагогічної науки минулого, а, навпаки, їх удосконалення і посилення, в тому числі і за рахунок використання досягнень у розвитку комп'ютерної техніки і засобів зв'язку [29].

Отже, система вищої освіти зазнає суттєвих змін. Кожен рівень та кожен компонент ВНЗ потребує корегування. У тому числі освітні програми, дидактичні принципи, інноваційні освітні технології тощо.

Сучасний інноваційний підхід щодо формування та розвитку кваліфікаційного рівня майбутніх фахівців являє собою не підсумковий перелік нормативних знань, вмінь, навичок та якостей особистості. Він втілює модель інтегральної особистісної освіти, яка регулюється через системоутворюючі якості. У цьому випадку логіка освітнього процесу передбачає рух від цілісних інтегративних характеристик фахівця до конкретних засобів діяльності, знань, вмінь та навичок з урахуванням принципу індивідуального підходу в рамках суб'єкт-суб'єктної моделі навчання [37].

Таким чином, одним з основних напрямів реформування вищих навчальних закладів, поряд із підвищенням якості освіти та забезпеченням конкурентоспроможності випускників закладів вищої освіти, є використання нових методів і технологій навчання, у першу чергу це стосується застосування сучасних інформаційно-комунікаційних технологій у процесі вивчення однієї з фундаментальних дисциплін - математики.

1.2. Інформатизація навчального процесу у ВНЗ

Інформатизація (*Informatisation*) — сукупність взаємопов'язаних організаційних, правових, політичних, соціально-економічних, науково-технічних, виробничих процесів, що спрямовані на створення умов для задоволення інформаційних потреб громадян та суспільства на основі створення, розвитку і використання інформаційних систем, мереж, ресурсів та інформаційних технологій, які побудовані на основі застосування сучасної обчислювальної та комунікаційної техніки [39].

Інформатизація — це процес широкомасштабного використання ІКТ у всіх сферах соціально-економічного, політичного і культурного життя

Графіка

plot(x)	побудова значень x (на осі y), впорядкованих на осі x
plot(x, y)	двовимірний графік побудови значень x (на осі x) і значень y (на осі y)
hist(x)	побудова гістограми послідовності x
barplot(x)	побудова гістограми значень x; використовуйте <code>horiz=FALSE</code> для горизонтальних смуг
pie(x)	кругова діаграма
boxplot(x)	«коробочка з вусами»
symbols(x, y, ...)	Побудова символу (коло, квадрат, прямокутник, зірка ...) в заданих координатах (x, y)

Індексація списків

$x[n]$	список з елементами n
$x[[n]]$	n-й елемент списку
$x[["name"]]$	елемент списку з ім'ям "name"
xname$	елемент з ім'ям "name" у списку

Індексація матриць

$x[i,j]$	елемент матриці i-го рядка, j-го стовпчика
$x[i,]$	i-й рядок
$x[,j]$	j-й стовпчик
$x[,c(1,3)]$	стовпчики 1 та 3
$x["name",]$	рядок з ім'ям "name"

Математичні функції

sin,cos,tan,asin,acos,atan,atan2,log,log10,exp

$\max(x)$	максимум елементів x
$\min(x)$	мінімум елементів x
$\text{sum}(x)$	сума елементів x
$\text{mean}(x)$	середнє значення елементів x
$\text{median}(x)$	медіана елементів x
$\text{quantile}(x,\text{probs}=\text{=})$	вибірковий квантиль, відповідний заданому значенню ймовірності probs
$\text{rank}(x)$	ранг елементів x
$\text{sd}(x)$	стандартне відхилення x
$\text{round}(x, n)$	округлення елементів x до n десяткових знаків
$\text{log}(x, \text{base})$	обчислення логарифму x за основою base
$\text{Re}(x)$	дійсна частина комплексного числа
$\text{Im}(x)$	уявна частина комплексного числа
$\text{Mod}(x)$	модуль x
$\text{abs}(x)$	модуль x
$\text{Arg}(x)$	аргумент комплексного числа (в радіанах)

Матриці

$t(x)$	транспонування
$\text{diag}(x)$	діагональ матриці x
$\%*\%$	множення матриць
$\text{solve}(a,b)$	розв'язання $a\%*\%x = b$ для x
$\text{solve}(a)$	обернена матриця

суспільства з метою підвищення ефективності використання інформації і знань для управління, задоволення інформаційних потреб громадян, організацій і держави і створення передумов переходу країни до інформаційного суспільства. Інформатизація суспільства — це створення високоорганізованого інформаційного середовища, яке обумовлює вирішення всього комплексу завдання, що висувуються суспільством. Інформаційне середовище — сукупність технічних і програмних засобів зберігання, обробки і передачі інформації, а також політичні, економічні і культурні умови реалізації процесів інформатизації [48].

Суспільство етапу інформатизації використовує інформацію у якості засобу, за допомогою якого воно може усвідомлювати себе та функціонувати як єдине ціле. Одним із багатьох означень поняття «інформація» є: це нові відомості, які прийняті, зрозумілі і оцінені її користувачем як корисні. Основними вимогами до інформації є її достовірність та доступність. Тобто, сьогодні відбувається формування інформаційного суспільства.

За визначенням Комісії Європейського Союзу *інформаційне суспільство* — це суспільство, в якому діяльність людей здійснюється на основі використання послуг, що надаються за допомогою інформаційних технологій та технологій зв'язку [75].

Освіта в умовах формування інформаційного суспільства не може залишатись осторонь.

Інформатизація освіти — це процес забезпечення освітньої галузі методологією та практикою розробки та оптимального використання сучасних інформаційно-комунікаційних технологій (ІКТ) [31].

Інформатизація освіти — це створення і використання інформаційних технологій для підвищення ефективності видів діяльності, що здійснюються в системі освіти [59].

Ми вважаємо, що цей процес сприяє:

- удосконаленню механізмів управління системою освіти на основі використання різних програмних засобів,
- створенню комп'ютерно-орієнтованих методичних систем навчання дисциплін, математики зокрема,
- формуванню умінь самостійно здобувати знання та навички,
- здійсненню різноманітних видів діяльності з обробки інформації.

Позитивним фактором у створенні інформаційного суспільства та інформатизації освіти в Україні є прийнятий Закон України «Про основні засади розвитку інформаційного суспільства в Україні на 2007–2015 роки» (09.01.2007 р. за № 537-V Верховною Радою України) [32].

У цьому документі зазначено, що розвиток інформаційного суспільства в Україні та впровадження новітніх ІКТ в усі сфери суспільного життя і в діяльність органів державної влади та органів місцевого самоврядування визначається одним з пріоритетних напрямів державної політики. Одним із основних стратегічних цілей розвитку інформаційного суспільства в Україні є створення загальнодержавних інформаційних систем, насамперед у сферах охорони здоров'я, освіти, науки, культури, охорони довкілля.

Крім того, серед основних напрямками розвитку інформаційного суспільства в Україні законом виокремлено:

- надання кожній людині можливості для здобуття знань, умінь і навичок з використанням ІКТ під час навчання, виховання та професійної підготовки;

- створення умов для забезпечення комп'ютерної та інформаційної грамотності усіх верств населення, створення системи мотивацій щодо впровадження і використання ІКТ для формування широкого попиту на такі технології в усіх сферах життя суспільства [32].

Отже, інформатизація суспільства не може відбуватись без інформатизації освіти.

Зауважимо, що інформатизація освіти має випереджати інформатизацію інших сфер та галузей діяльності людини, оскільки у ній формуються соціальні, загальнокультурні, психологічні та професійні засади інформатизації суспільства. Інформатизація освіти є головною умовою успішної інформатизації суспільства.

Метою вищої освіти сьогодні є підготовки висококваліфікованих фахівців. Однією з вимог до майбутніх фахівців має стати комп'ютерна грамотність. Це стосується і студентів економічних спеціальностей університетів. Під комп'ютерною грамотністю будемо розуміти (англ. *computer literacy*) – це оволодіння мінімальним набором знань і навичок роботи на персональному комп'ютері [39].

Досягнути цієї мети можна шляхом: якісних змін змісту, удосконалення методів та організаційних форм навчання, впровадження сучасних освітніх технологій у навчальний процес, зокрема інформаційно-комунікаційних технологій (із урахуванням нових досягнень у певній галузі). При цьому мають бути враховані психолого-педагогічні закономірності та дидактичні принципи. Такий підхід обумовлений об'єктивною необхідністю інтеграційних процесів, зокрема між математикою та інформатикою. Тобто має здійснюватись інформатизація навчального процесу.

Ми поділяємо думку М. І. Жалдака, що інформатизація навчального процесу - це створення і широке впровадження в повсякденну педагогічну практику нових комп'ютерно-орієнтованих методичних систем навчання (КОМСН) на принципах поступового і неантагоністичного, без руйнівних перебудов і реформ, вбудовування інформаційно-комунікаційних технологій у діючі дидактичні системи, гармонійного поєднання традиційних і комп'ютерно-орієнтованих технологій навчання, не заперечування і відкидання здобутків педагогічної науки минулого, а, навпаки, їх удосконалення і посилення, в тому числі і за рахунок використання досягнень у розвитку комп'ютерної техніки і засобів зв'язку. Інформатизація навчального процесу сприятиме підвищенню якості професійної підготовки майбутніх фахівців, активізації навчально-пізнавальної і науково-дослідної діяльності студентів ВНЗ, розкриттю їхнього творчого потенціалу, збільшенню ролі самостійної та індивідуальної роботи [29].

Етапи інформатизації освіти [43]:

rep(x,times)	повторити x разів; Використовуйте each= щоб повторити "кожен" елемент з x each раз; Rep (C (1,2,3), 2) буде 1 2 3 1 2 3; Rep (C (1,2,3), each = 2) буде 1 1 2 2 3 3
data.frame(...)	створення блоку даних з поіменованими або безіменними аргументами; data.frame(v=1:4,ch=c("a","B","c","d"),n=10); короткі вектори використовуються повторно, до довжини найдовшого вектора
list(...)	створення списку з поіменованими або безіменними аргументами; list(a=c(1,2),b="hi",c=3i);
array(x, dim=)	створення масиву з даними x; розмірність вказується, наприклад як dim=c(3,4,2); елементи x використовуються повторно, якщо x не достатньо довгий
matrix(x,nrow=, ncol=)	матриця; елементи x використовуються повторно
factor(x, levels=)	кодує вектор x в якості фактора
gl(n,k, length=n*k, labels=1:n)	створення рівнів (факторів), вказавши шаблон їх рівнів; k – число рівнів; n – число повторень;
expand.grid()	data.frame від усіх даних векторів або факторів
rbind(...)	об'єднання аргументів по рядках для матриць, масивів даних та ін.
cbind(...)	аналогічно, по стовпчикам

Доступ до даних

Індексація векторів

x[n]	n-й елемент
x[-n]	всі крім n-ого елементу
x[1:n]	перші n елементів
x[-(1:n)]	елементи від n+1 до кінця
x[c(1,4,2)]	вказані елементи
x["name"]	елемент з ім'ям "name"
x[x > 3]	всі елементи, значення яких більші 3
x[x > 3 & x < 5]	всі елементи, значення яких від 3 до 5
x[x %in%("a", "and", "the")]	елементи у вибраній множині

<code>read.fwf(file,widths,header=FALSE,sep="",as.is=FALSE)</code>	зчитування таблиці фіксованої ширини відформатованих даних в «data.frame»; widths – цілий вектор, що задає ширини полів фіксованої ширини
<code>save(file,...)</code>	зберігаються задані об'єкти (...) в платформи незалежному бінарному форматі XDR
<code>save.image(file)</code>	зберігає всі об'єкти
<code>cat(..., file="", sep=" ")</code>	друкує аргументи після приєднання до символу; sep – символ роздільник між аргументами
<code>print(a, ...)</code>	друкує свої аргументи; тобто в загальному, може мати різні методи для різних об'єктів
<code>format(x,...)</code>	форматування об'єкта R для красивого друку
<code>write.table(x,file="",row.names=TRUE,col.names=TRUE,sep=" ")</code>	друк x після перетворення в data.frame; якщо quote = TRUE, символ або фактор стовпців береться в лапки ("); sep – роздільник полів; eol – роздільник кінця рядка; na – це стрічка для відсутніх значень, використовуйте col.names=NA, щоб додати порожню назву стовпця, для того, щоб отримати заголовки стовпців правильно вирівняні для введення таблиць

Більшість функцій введення / виводу мають аргумент файлу. Це часто може бути символічний рядок назви файлу або з'єднання. `file=""` означає стандартний ввід або вивід. З'єднання можуть включати в себе файли, канали, стислі файли, і R змінні.

Для взаємодії з базою даних, дивіться пакети RODBC, DBI, RMySQL, RPostgreSQL, і ROracle. Подивіться всі пропозиції XML, hdf5, NetCDF для читання інших форматів файлів.

Створення даних

<code>c(...)</code>	узагальнена функція поєднання аргументів у вектор; якщо recursive=TRUE, спускається по списку, об'єднуючи всі елементи в один вектор
<code>from:to</code>	генерує послідовність; ":" має пріоритет оператора; 1: 4 + 1 буде вектор "2,3,4,5"
<code>seq(from,to)</code>	генерує послідовність by= вказує приріст; length= визначає потрібну довжину
<code>seq(along=x)</code>	формує 1, 2, ..., length(along); корисно для for циклів

1) етап характеризується наступними ознаками: початок масового впровадження засобів інформаційних технологій і в першу чергу комп'ютерів; проводиться дослідницька робота з педагогічного освоєння засобів комп'ютерної техніки і відбувається пошук шляхів її застосування для інтенсифікації процесу навчання; суспільство йде по шляху усвідомлення суті і необхідності процесів інформатизації; відбувається базова підготовка в галузі інформатики на всіх ступенях безперервної освіти,

2) етап характеризується наступними ознаками: активне освоєння і фрагментарне впровадження засобів ІТ в традиційні початкові дисципліни; освоєння педагогами нових методів і організаційних форм роботи з використанням комп'ютерної техніки; активна розробка і початок освоєння педагогами навчально-методичного забезпечення; постановка проблеми перегляду змісту, традиційних форм і методів навчально-виховної роботи,

3) етап характеризується наступними ознаками: наскрізне використання засобів сучасних ІКТ у навчанні; перебудова змісту всіх ступенів безперервної освіти на основі його інформатизації; зміна методичної основи навчання і освоєння кожним педагогом широкого кола методів і організаційних форм навчання, що підтримуються відповідними засобами сучасних інформаційних технологій.

Сьогодні, як зазначають автори посібнику [43], і ми з ними цілком погоджуємось, відбувається активний перехід з другого на третій етап.

У зв'язку з цим та в умовах інформатизації всього навчального процесу в останні роки почали вибудовуватись нові тенденції та підходи до математичної освіти. Разом з тим, проявили себе і протиріччя, які формуються і розвиваються в процесі її змін.

Ю.В. Триус [75] виокремив такі протиріччя:

1. Породжений бурхливим розвитком науки і техніки ХХ століття "інформаційний бум" спричинив необхідність перебудови вищої освіти в цілому, що обумовило виникнення протиріччя між змістом вищої освіти і реальними потребами суспільства в її результатах.

2. Протиріччя між можливостями студентів, більшість з яких володіє загальними прийомами роботи в сучасних інформаційних середовищах, та методами, засобами й організаційними формами навчання, що їм пропонуються у вищих навчальних закладах.

3. Сучасні педагогічні технології, методи розвиваючого і особистісно-орієнтованого навчання недостатньо використовуються в практиці навчання математичних дисциплін у ВНЗ, тому що вимагають для їх впровадження набагато більше інтелектуальних і фізичних зусиль викладачів, використання нових засобів створення навчальних інформаційних ресурсів, порівняно з традиційними підходами і технологіями навчання. Отже, існує протиріччя між загальними цілями вищої освіти та методами і засобами досягнення цих цілей, що використовуються у навчальному процесі більшості ВНЗ.

4. Вивчаючи математичні дисципліни, студенти опрацьовують великий обсяг теоретичного матеріалу, здобувають необхідні знання, уміння і навички щодо розв'язування типових математичних задач. Однак, потрапляючи до

реального середовища професійної діяльності, студенти, як правило, не можуть застосувати отримані знання про існуючі методи і алгоритми пошуку оптимальних розв'язків професійних задач. Невідповідність великого обсягу теоретичного матеріалу умінню використовувати його в нестандартних ситуаціях усе більше загострює протиріччя між репродуктивними і розвиваючими способами навчання.

5. Дидактичні засоби підтримки навчального процесу є одним з найважливіших інструментів у роботі викладачів математичних дисциплін. Кількісна недостатність і мала варіативність цих засобів обмежують бажання викладачів у доборі навчального матеріалу. Так виникає протиріччя між існуючими формами зберігання й передавання методичного та педагогічного досвіду і можливостями, що відкриваються на основі використання сучасних інформаційно-комунікаційних технологій.

Важливе місце у подоланні зазначених вище суперечностей займає формування єдиного інформаційного простору. Однак цей процес йде дуже повільно, безсистемно, що викликано низкою складнощів, а саме:

1. У цілому у закладах вищої освіти досить багато комп'ютерів (згідно із санітарними нормами 12 студентів на 1 комп'ютер), однак дуже часто вони вже давно морально та технічно застарілі. Тому ефективність їх використання у навчальному процесі є вкрай низкою.

2. Значна частина дидактичних та навчальних матеріалів, накопичених у викладачів, є нерозповсюджені серед інших, тобто не відбувається у повній мірі обмін та накопичення досвіду, а тим самим уповільнюється процес інформатизації навчального процесу.

3. Існуючі засоби навчання (різноманітні програмні засоби, мультимедійні презентації) не використовуються або використовуються не в повній мірі, без врахування індивідуальних особливостей студентів, а отже знижують мотивацію студентів до навчання.

Таким чином, модернізація системи вищої освіти має відбуватись на основі гармонійного поєднання (інтеграції) традиційних технологій навчання та сучасних інформаційно-комунікаційних технологій. Це буде сприяти формуванню інформатизації освіти та суспільства в цілому.

1.3. Аналіз понятійного апарату з проблеми інформаційно-комунікаційних технологій у навчальному процесі ВНЗ

Поняття “інформаційно-комунікаційні технології” (ІКТ) не є однозначним.

У науковій літературі зустрічаються різноманітні аналоги даного поняття, а саме “комп'ютерні технології”, “інформаційні технології” тощо.

І.І. Довгопол стверджує, що комп'ютерні технології навчання – це процеси підготовки та передавання інформації учню, засобом здійснення яких є комп'ютер. Тобто, комп'ютерні технології засновані на формалізації знань, залученні до навчального процесу діалогу засобів штучного інтелекту та застосуванні спеціальних пакетів прикладних програм, у тому числі тих, що

ДОДАТОК 4.

Довідкова карта по R

Отримання довідки

Більшість функцій R мають потужну онлайн документацію.

help(topic)	документація по темі topic
?topic	документація по темі topic
help.search("topic")	пошук довідкової системи
apropos("topic")	імена всіх об'єктів у списку пошуку, що відповідають регулярному виразу "topic"
help.start()	відкриття HTML версії допомоги
str(a)	відображає внутрішню структуру об'єкта R
summary(a)	дає "резюме" а, зазвичай як статистичне резюме, але і як загальне значення, має різні операції для різних класів а
ls()	показує об'єкти в шляху пошуку; вкажіть pat="pat" для пошуку за шаблоном
ls.str()	str() для кожної змінної в шляху пошуку
dir()	показує файли в поточній директорії
methods(a)	S3 методи а
methods(class=class(a))	перераховані всі методи для обробки об'єктів класу а

Введення і виведення даних

load()	завантажує набори даних, які записані за допомогою save
data(x)	завантажує вказаний набір даних
library(x)	завантаження додаткових пакетів
read.table(file)	читає файл у форматі таблиці і створює data.frame з нього; роздільник за замовчуванням sep=""; використовуйте header=TRUE, щоб зчитувати перший рядок як заголовок імен стовпців; використовуйте as.is=TRUE, щоб запобігти перетворення векторів символів у фактори; використовуйте comment.char="", щоб запобігти інтерпретації "#" як коментаря; використовуйте skip=n, щоб пропустити n рядків перед читанням даних;
read.csv("filename", header=TRUE)	аналогічно, але з встановленим за замовчуванням зчитуванням розділених комами файлів
read.delim("filename", header=TRUE)	аналогічно, але з встановленим за замовчуванням зчитуванням розділених табуляцією файлів

Екранні форми, завдання змінних, цільової функції, обмежень і граничних умов двоіндексного завдання і її рішення представлені на рис. 2.170, 2.171 і в таблиці 2.25.

Рис. 2.170 – Екранна форма одноіндексної завдання (курсор в цільовій комірці F15)

Таблиця 2.25

Формули екранної форми завдання OpenOffice.org Calc

Об'єкт математичної моделі	Вираз в OpenOffice.org Calc
Змінні завдання	C3:E6
Формула в цільовій комірці F15	=SUMPRODUCT(C3:E6;C12:E15)
Обмеження по рядках в комірках F3, F4, F5, F6	=SUM(C3:E3) =SUM(C4:E4) =SUM(C5:E5) =SUM(C6:E6)
Обмеження по стовпцях в комірках C7, D7, E7	=SUM(C3:C6) =SUM(D3:D6) =SUM(E3:E6)
Сумарні запаси і потреби в осередках H8, G9	=SUM(H3:H6) =SUM(C9:E9)

Рис. 2.171 – Екранна форма після отримання рішення задачі (курсор в цільовій комірці F15)

зорієнтовані на викладача, який не є програмістом. Але ж деякі фахівці вважають термін “комп’ютерні технології” невдалим (буквально – обчислювальні технології), тому що розуміння комп’ютера як обчислювальної машини (від англ. computer – обчислювач) стало вже анахронізмом [18].

Термін “інформаційна технологія” вперше був застосований в 1958 році у статті Х.Лівітта та Т.Уіслера “Менеджмент у 80-х”. Вони трактували його як технологія збирання, обробки, зберігання та розповсюдження інформації за допомогою комп’ютерних та телекомунікаційних засобів.

Інформаційні технології (ІТ) - сукупність методів і програмно-технічних засобів, об’єднаних в технологічний ланцюг, що забезпечує збір, обробку, зберігання і відображення інформації з метою зниження трудомісткості її використання, а також для підвищення її надійності і оперативності. Інформаційні технології зазвичай є функціональними компонентами інших видів технологій (виробничих, організаційних, соціальних) і виконують роль інтелектуального ядра останніх. Використання інформаційних технологій дозволяє значно підвищити ефективність цих та ін. технологій, скорочуючи при цьому витрати різних інших видів ресурсів суспільства [43].

Міжнародні організації ISO та ІЕС, створили технічний комітет із стандартизації інформаційних технологій JTC1 (Joint Technical Committee, 1987), в статутних документах комітету JTC1 дав таке означення поняттю “інформаційні технології (ІТ)”: “Інформаційні технології включають специфікацію, проектування і розробку систем і засобів, що мають справу із збором, уявленням, обробкою, безпекою, передачею, організацією, зберіганням і пошуком інформації, а також обміном і управлінням інформацією” [40].

Пізніше до поняття «інформаційні технології» почали додавати слова «комунікаційні», оскільки в умовах стрімкого розповсюдження всесвітньої мереж Internet комунікації (спілкування, поширення та розповсюдження) стали відігравати надзвичайно важливу роль у інформаційної та освітньому просторах.

У Вікіпедії зазначено, що інформаційно-комунікаційні технології (ІКТ, від англ. *Information and communications technology, ICT*) - часто використовується як синонім до інформаційних технологій (ІТ), хоча ІКТ це загальніший термін, який підкреслює роль уніфікованих технологій та інтеграцію телекомунікацій (телефонних ліній та бездротових з’єднань), комп’ютерів, підпрограмного забезпечення, програмного забезпечення, накопичувальних та аудіовізуальних систем, які дозволяють користувачам створювати, одержувати доступ, зберігати, передавати та змінювати інформацію. Іншими словами, ІКТ складається з ІТ, а також телекомунікацій, медіа-трансляцій, усіх видів аудіо і відеообробки, передачі, мережеских функцій управління та моніторингу [42].

Інформаційно-комунікаційні технології або ІКТ — засоби, пов’язані зі створенням, збереженням, передачею, обробкою і управлінням інформації. Цей широко вживаний термін включає в себе всі технології, що використовуються для спілкування та роботи з інформацією. Концепція інформаційних технологій була додана до елементу комунікації і виникла у 1980-ті роки. Наразі ІКТ

включають апаратні засоби (комп'ютери, сервери, тощо) та програмне забезпечення (операційні системи, мережеві протоколи, пошукові системи, тощо) [41].

А.А. Дзюбенко трактує інформаційні комунікаційні технології навчання як сукупність програмних, технічних, комп'ютерних і комунікаційних засобів, а також способів та новаторських методів їхнього застосування для забезпечення високої ефективності й інформатизації освітнього процесу [16].

С.Г.Григор'єв визначив ІКТ як узагальнене поняття, яке описує різноманітні методи, способи та алгоритми збору, накопичення, обробки, подання й передаван- ня інформації [11].

В.А. Трайнев до складу ІКТ відносить сукупність методів та програмно-технічних засобів, що об'єднують в технологічний ланцюг, який забезпечує збір, обробку, збереження та відображення інформації з метою зниження трудомісткості її використання, а також для підвищення її надійності й оперативності [73].

І.Г. Захарова тлумачить ІКТ як конкретний спосіб роботи з інформацією: це і сукупність знань про способи та засоби роботи з інформаційними ресурсами, і спосіб та засоби збору, обробки та передавання інформації для набуття нових відомостей про об'єкт, що вивчається" [35].

А.В. Зубов у своїй праці [36] визначає такі компоненти ІКТ:

1. Теоретичні засади (основу складають найважливіші поняття й закони інформатики (інформатика як наука, об'єкт та предмет інформатики; поняття інформації, її властивостей та особливостей, до яких відносять цінність, повноту, актуальність, компактність, достовірність та логіч- ність; різноманітні класифікації інформації; основні інформаційні процеси, типи ін- формаційних ресурсів, види інформаційної діяльності, принципи функціонування комп'ютерної техніки, алгоритми інформаційного моделювання, використання ІКТ).

2. Методи вирішення завдань (моделювання, системний аналіз, системне проекту- вання, методи передачі, збору, продукування, накопичення, збереження, обробки, передачі та захисту інформації).

3. Засоби вирішення завдань: апаратні (персональний комп'ютер і його основні складові, локальні та глобальні мережі, сучасне периферійне обладнання); програмні (системні, прикладні, інструментальні).

Схематично ці компоненти інформаційно-комунікаційних технологій представлено на рис. 1.1.

Програмне забезпечення (програмні засоби) (ПЗ; англ. *software*) — сукупність програм системи обробки інформації і програмних документів, необхідних для експлуатації цих програм [19].

Розрізняють системне програмне забезпечення (зокрема, операційна система, транслятори, редактори, графічний інтерфейс користувача); прикладне програмне забезпечення, що використовується для виконання конкретних завдань, наприклад, статистичне програмне забезпечення; інструментальне програмне забезпечення (комп'ютерні програми, призначені для проектування,

завжди пробіл в записі формули - це невинне порожнє місце. Іноді це оператор перетину діапазонів.

Пропуск в синтаксисі створює деякі неоднозначні ситуації в самому записі. Ось приклад дуже схожих зовні, але дуже різних по суті записів Excel:

= SUM(A1, A2, A3)

= SUM (A1, A2, A3)

У першому випадку це виклик функції SUM зі списком параметрів з трьох елементів. У другому випадку це бінарний оператор пробіл між двома параметрами, першим з яких є строковий ідентифікатор, який розцінюється як іменованій діапазон комірок, а другий - це список параметрів з трьох елементів. Другий випадок є вірним записом з точки зору синтаксису мови формул OOXML. І результатом буде кумулятивний результат перетину між іменованим діапазоном і списком параметрів.

Варто згадати, що ось такий запис має сенс: '= SUM A1'. Наприклад, SUM є іменованим діапазоном A1:B2, в цьому випадку результатом наведеного прикладу буде значення з комірки A1.

Зважаючи на відмінності описані вище та інші спробуємо вирішити декілька завдань за допомогою OpenOffice Calc.

1 Вирішення транспортної задачі.

Транспортна задача відноситься до двохіндексних завдань лінійного програмування. Двохіндексні завдання ЛП вводяться і вирішуються в OpenOffice.org Calc аналогічно одноіндексним завданням. Специфіка введення умови двохіндексного завдання ЛП полягає лише в зручності матричного завдання змінних завдання і коефіцієнтів.

Розглянемо рішення двохіндексного завдання, суть якого полягає в оптимальній організації транспортних перевезень штучного товару зі складів до магазинів (таблиця 2.24).

Таблиця 2.24

Вихідні дані транспортної задачі

Тариф грн./шт.	1-й магазин	2-й магазин	3-й магазин	Запаси, шт.
1-й склад	2	9	7	25
2-й склад	1	0	5	50
3-й склад	5	4	100	35
4-й склад	2	3	6	75
Необхідність, шт.	45	90	50	-

Цільова функція і обмеження даного завдання мають вигляд:

$$L(X) = 2X_{11} + 9X_{12} + 7X_{13} + X_{21} + 5X_{23} + 5X_{31} + 4X_{32} + 100X_{33} + 2X_{41} + 3X_{42} + 6X_{43} \rightarrow \min;$$

$$\begin{cases} X_{11} + X_{12} + X_{13} = 25, \\ X_{21} + X_{22} + X_{23} = 50, \\ X_{31} + X_{32} + X_{33} = 35, \\ X_{41} + X_{42} + X_{43} = 75, \\ X_{11} + X_{21} + X_{31} = 45, \\ X_{12} + X_{22} + X_{32} = 90, \\ X_{13} + X_{23} + X_{33} = 50, \\ \forall X_{ij} \geq 0, \forall X_{ij} - \text{целые} (i = \overline{1,4}; j = \overline{1,3}). \end{cases}$$

Одна з них - циліндрична з глибиною, що є різновидом 3-М гістограми. Для кращого огляду вона була трохи повернена. Інша є звичайною кругової 3-М діаграмою. Залишилися дві відносяться до виду 2-М діаграм: лінія з символами, отриманими з Галереї, і гістограма, що використовує різні заливки і фон.

2.8.2. Використання OO calc для вирішення математичних задач

Популярні стандарти

На сьогоднішній день існують два найбільш популярних стандарти, які використовуються в табличних редакторах. З одного боку це редактор електронних таблиць MS Excel і його стандарт ECMA OOXML. З іншого боку розташовуються сімейство редакторів LibreOffice / OpenOffice Calc і стандарт OpenFormula.

Для звичайного користувача формули в редакторі таблиць асоціюються з викликом функцій. З урахуванням всіх версій редакторів, функцій налічується близько чотирьох з половиною сотень. Їх умовно поділяють на групи по застосуванню: інженерні, для роботи з базами даних, дати і часу, фінансові, логічні, інформаційні, математичні і тригонометричні, статистичні, текстові і т.д. Але незалежно від сфери застосування всі функції підпорядковуються одним і тим самим правилам. Вони можуть працювати з певними типами вхідних даних або ж використовувати результат роботи один одного.

Відмінності стандартів в редакторах електронних таблиць

Що стосується функцій, в обох стандартах досить багато відмінностей. Вони можуть мати різну кількість параметрів. У свою чергу параметри можуть мати різні типи вхідних значень. Може відрізнитися навіть точність обчислень. Також багато відмінностей в самому синтаксисі формул. Нижче в таблиці наведемо буквально декілька прикладів відмінностей записи для англійської локалізації (формат запису формул в різних локалізаціях також може відчутно відрізнитись):

	MS Excel	OpenOffice Calc
1		
2	Масив значень ={1,2,3; 3,2,1}	= {1,2,3 3,2,1}
3	Посилання =Sheet1!A1	=Sheet1.A1
4	Діапазон =Sheet1!A1:B5	=Sheet1.A1:Sheet1.B5
5	Виклик функцій =SUM(1,2,3)	=SUM (1;2;3)
6	Оператор перетину діапазонів: =A1:B5 B5:C6	=A1:B5!B5:C6

В останньому прикладі пробіл в синтаксисі MS Excel між двома діапазонами є значущим і це бінарний оператор перетину діапазонів.

У синтаксисі формул дійсно є відмінності, але вони не настільки великі, щоб перешкодити розробити єдине внутрішнє представлення для обох записів мови формул. Внутрішнє представлення - це синтаксичне дерево розбору, в якому вже не існує відмінностей між різним синтаксисом формул. Звичайно, не всі конструкції можна однозначно конвертувати з одного запису в іншу і є певний рівень несумісності.

Крім відмінностей між записами, існують проблеми використання токенів в рамках однієї записи. Йдеться про оператор пробілу в OOXML. Не

розробки, адміністрування і супроводження системного та прикладного програмного забезпечення) [76].



Рис. 1.1 - Компоненти інформаційно-комунікаційних технологій

Прикладним програмним засобом навчального призначення називають таке програмне забезпечення, в якому відображається деяка предметна галузь, тією чи іншою мірою реалізується технологія її вивчення, забезпечуються умови для здійснення і комп'ютерної підтримки різних видів навчальної діяльності [83].

Програмний продукт (англ. *programming product*) – це:

- 1) програмний засіб, програмне забезпечення, які призначені для постачання користувачеві (покупцеві, замовникові),
- 2) програма, яку може запускати, тестувати, виправляти та змінювати будь-яка людина. Вона може використовуватись в різних операційних системах та з різними наборами даних [63].

Педагогічний програмний засіб (ППЗ) – прикладні програмні продукти для організації та підтримки навчального діалогу користувача з комп'ютером. Призначений для представлення навчальної інформації та організації навчання з урахуванням індивідуальних можливостей студентів. Передбачає наявність зворотного зв'язку викладача зі студентом [76].

Розглянутий вище понятійний апарат найчастіше використовується у науковій та методичній літературі з проблеми застосування ІКТ у навчанні математики.

Враховуючи все вище сказане, можна зробити висновок про те, що ІКТ є одним із засобів навчання, що сприяє реалізації педагогічної ідеї.

1.4. Проблеми застосування інформаційно-комунікаційних технологій у навчанні математики у ВНЗ

Істотним недоліком у професійній підготовці майбутніх фахівців є їх недостатній професіоналізм у використанні ІКТ. Це зумовлено низьким рівнем застосування інформаційно-комунікаційних технологій у навчальному процесі, що в свою чергу спричинено необізнаністю значної частини викладачів у сфері комп'ютерно-орієнтованих методик викладання або (та) невмінням використовувати їх у навчанні.

Отже, сьогодні одним із головних завдань вузів стає формування комп'ютерно-орієнтованих професійних компетентностей студентів. Оскільки випускники закладів вищої освіти мають не тільки володіти знаннями в галузі комп'ютерної техніки, а й бути фахівцями із застосування ІКТ у своїй професійній діяльності, вміти спілкуватися в інформаційному середовищі.

Математика як одна з фундаментальних дисциплін професійної підготовки у ВНЗ не може залишатись осторонь таких важливих змін у освітній сфері.

Дослідженню питання використання сучасних інформаційно-комунікаційних технологій у навчанні як загальноосвітньої так і вищої шкіл присвячені роботи Н.В.Апатової, Р. Вільямса, А. П. Єршова, М.І.Жалдака, В.М.Касаткіна, В.І.Клочка, А.А.Кузнецова, М.П.Лапчика, К. Макліна, В.М.Монахова, Н.В.Морзе, С.А.Ракова, Ю.С.Рамського, Ю.В.Триуса та інших.

Дидактичні та психологічні аспекти застосування ІКТ у навчанні висвітлювали А. М. Алексюк, Ю. К. Бабанський, В.П.Беспалько, Л. В. Занков, В.П.Зінченко, В.С.Ледньов, Е.І.Машбиц, А.Н.Леонт'єв, І. Я. Лернер, М. І. Махмутов, В.В.Рубцов, В.Ф.Паламарчук, Л.Н.Прокопенко, В. В. Серіков, С. Д. Смірнов, Н.Ф.Тализіна, О.К.Тихомиров та інші.

Проблеми впровадження інформаційно-комунікаційних технологій у навчальний процес математики досліджувались у роботах В. П. Дьяконова, М. І. Жалдака, В. І. Клочка, О. Г. Мордковича, Н. В. Морзе, С. А. Ракова, Ю. С. Рамського, О. В. Співаковського, Ю. В. Горошка, О. Б. Жильцова, Т. В. Зайцевої, Ю. Г. Лютюка, А. В. Пенькова, С. О. Семерікова, Ю.В. Триус та інших.

Для представлення даних у вигляді діаграми їх слід попередньо виділити (разом із заголовками, якщо такі є), після чого з меню Вставка вибрати пункт Діаграма. У відкритих вікні Автоформат діаграми потрібно виконати ряд дій, що дозволяє вибрати тип діаграми, вказати розташування рядів даних, задати заголовок діаграми, найменування осей і т. д. Якщо потрібно вставити діаграму в документ, що складається з декількох таблиць, то можна встановити, в яку таблицю повинна бути вставлена діаграма. Після заповнення потрібних полів натисніть на клавішу Готово і діаграма буде розміщена на аркуші.

Зауважимо, що при складанні діаграм допускається багаторазове виділення, тобто виділені дані не зобов'язані розташовуватися в таблиці у вигляді безперервної області. Багаторазове виділення здійснюється при натиснутій клавіші Ctrl. Якщо ви використовуєте багаторазове виділення, то переконаєтесь, що цей набір розрізнених осередків доцільно використовувати для складання діаграми.

OpenOffice Calc дозволяє змінювати окремі елементи діаграми. Виділення діаграми або її частини здійснюється клацанням миші, при цьому з'являються вісім зелених квадратиків, розташованих по периметру.

Курсор, потрапляючи у виділену таким чином область, набуває вигляду хрестика. Натискання на ліву кнопку миші дозволяє переміщати виділений об'єкт. Для видалення виділеного об'єкта виберіть з контекстного меню пункт Вирізати.

Клацання правою кнопкою миші призводить до появи меню форматування діаграми, пункти якого дозволять відкоригувати вигляд практично будь-якій частині діаграми: встановити межі діапазонів на осях діаграми і масштаб, змінити прозорість області діаграми, задати будь-якої фон, управляти відображенням сітки і багато чого іншого.

Якщо ви помістили діаграму на задній план електронної таблиці, то виділити її просто клацанням миші вже не вдасться.

Відкрийте панель Опції малювання і виберіть інструмент Виділення. Тепер клацанням виділіть діаграму.

При створенні лінійчатих діаграм (графіків) допускається використання різних символів, які можуть бути обрані з файлів з малюнками або з так званої Галереї. У ній міститься велика кількість малюнків для оформлення кордонів осередків, фонових зображень, маркерів і «тривимірних» об'єктів.

Тип діаграми можна змінити в будь-який момент. У діалозі, який з'явиться завдяки виділенню діаграми і викликом команди Формат - Тип діаграми, представлені різні типи діаграм.

Перегляньте в цьому діалозі всі типи діаграм: двовимірні (2-М) і тривимірні (3-М). У 3-М-діаграмах можна налаштувати напрям освітлення, навколишній світ і колірний фільтр. Такі діаграми можна обертати і нахилити за допомогою миші.

Нижче наведено чотири діаграми, що відповідають одному і тому ж набору даних, отриманому в результаті останнього коректного багаторазового виділення з розглянутого вище прикладу.

Для завдання суми чисел, що знаходяться в стовпці або рядку слід використовувати кнопку. У комірці з'явиться формула виду = СУММ (...). Програма намагається здогадатися, який інтервал підсумовування. Якщо вас не влаштовує запропонований діапазон, то виділіть лівою кнопкою миші потрібну область осередків так, щоб навколо неї з'явилася червона рамка. Те ж саме можна зробити і вручну, вказавши діапазон клітинок у рядку формул. Відзначимо, що порожні клітинки при підсумовуванні трактуються як містять нульові значення.

Познайомимося детальніше з функцією ДАТА, що забезпечує введення дат в електронну таблицю. OpenOffice Calc зберігає такі дані у вигляді чисел, але відображає у клітинці у форматі дати, вирівнюючи їх (аналогічно числах) по правому краю. Звичайно, можна відформатувати клітинку, яка містить дату, і в числовому форматі. Синтаксис функції - ДАТА (Рік; Місяць; День). Рік - ціле число від 1600 до 3000, при введенні від 0 до 29 додається 2000, а при введенні числа від 30 до 99 - 1900. Місяць - число від 1 до 12, що задає номер місяця. День - число від 1 до 31, яке встановлює день місяця. Коли значення місяця і дня більше допустимих, вони перераховуються на наступну позицію (рік, місяць) з переповненням. Формула = ДАТА (00; 12; 31) дає 31.12.2000, а при введенні = ДАТА (00; 13; 31) вийде дата 31.01.2001.

У програмі OpenOffice Calc можна просто ввести дати в форматі «місяц.число.год» (без вказівки лапок, інакше введення буде інтерпретований як текст), наприклад, «5.17.2» для 17 травня 2002 року. У цьому випадку будь введення зі значеннями, що виходять за межі допустимих, трактується не як дата, а як текст. Функція ТДАТУ () повертає дату і час відповідно до системним часом комп'ютера, які оновлюються при кожному перерахунку документа.

Для того щоб задати ім'я області (комірки), необхідно цю область спочатку виділити і за допомогою пункту Імена - Поставити меню Вставка (або за допомогою комбінації клавіш Ctrl + F3) викликати діалогове вікно Присвоїти ім'я. Ім'я повинно починатися з літери, відрізнятися від стандартних імен осередків і не може містити пропусків. Після введення імені слід натиснути кнопку Додати. У цьому ж діалозі можна задати імена інших областей, ввівши спочатку ім'я, а потім виділивши клітинки аркуша, які повинні отримати це ім'я. У ньому можна присвоїти імена навіть часто використовуваним формулами або елементам формул.

Потужним інструментом при роботі з електронними таблицями є підбір параметра. За допомогою цього інструменту можна дізнатися значення, яке при підстановці в формулу дає бажаний результат. Для того щоб скористатися підбором параметра потрібно мати формулу з декількома постійними значеннями і одним змінним.

Діаграми

Всі розглянуті нами програми підготовки електронних таблиць в тій чи іншій мірі можуть відображати результати розрахунків у вигляді діаграм. Однак жодна з них не може на даний момент зрівнятися з програмою OpenOffice Calc, що пропонує багатий вибір різних типів діаграм, з яких можна вибрати найбільш відповідні для демонстрації структури ваших даних.

У дослідженнях Н.В.Макарової, А.І.Мішеніна, Л.В.Смоліної та інших доведено, що стійкі вміння та навички інформаційної діяльності у фахівців економічної галузі сприяють розв'язанню професійних задач, ефективному аналізу економічних даних, оцінці тенденції розвитку різних об'єктів економічної діяльності.

Однією з перших монографій з інформаційних технологій була робота академіка В.М. Глушкова [9]. У цій роботі вперше у вітчизняній літературі було використано термін «інформаційна технологія». На думку автора, інформаційні технології – це процеси, які пов'язані із переробкою інформації. На нашу думку, за таким означенням можна зробити висновок про те, що інформаційні технології є невід'ємною частиною освіти оскільки навчання передбачає процес передачі інформації від викладача до студента.

А.П.Єршов [24] запропонував принципи комп'ютеризації (інформатизації) математичної освіти. Б.С.Гершунський [8] дослідив загально педагогічні аспекти інформатизації освіти. Стратегічні та тактичні завдання інформатизації освіти висвітлено у роботах О.К.Тихомірова [72]. Е.І.Машбиць [54] розглянув психолого-педагогічні проблеми інформатизації освіти.

Цілий ряд дисертаційних досліджень містять розробки щодо впровадження ІКТ у навчальний процес різних дисциплін, математики зокрема.

Сучасні найбільш ґрунтовні дослідження із застосування ІКТ проводили Ю.Горошко (система інформаційного моделювання у підготовці майбутніх учителів математики та інформатики), М.Жалдак (система підготовки учителя до використання інформаційної технології у навчальному процесі), Н.Морзе (система методичної підготовки майбутніх вчителів інформатики в педагогічних університетах), Т.Крамаренко (формування особистісних якостей школяра у процесі комп'ютерно-орієнтованого навчання математики), С.Раков (формування математичних компетентностей учителя математики на основі дослідницького підходу в навчанні з використанням інформаційних технологій), О.Співаковський (теоретико-методичні основи навчання вищої математики майбутніх вчителів математики з використанням інформаційних технологій), С.Семеріков (активізація пізнавальної діяльності студентів при вивченні чисельних методів у об'єктно-орієнтованій технології програмування), Ю.Триус (комп'ютерно-орієнтовані методичні системи навчання математичних дисциплін у вищих навчальних закладах) та інші.

Жалдак М.І. зазначив, при використанні ІКТ в навчанні математики на перший план повинно висуватись з'ясування проблеми, постановка задачі, розробка відповідної математичної моделі, матеріальна інтерпретація отриманих за допомогою комп'ютера результатів [28].

Характерною особливістю використання ІКТ у навчанні математики:

- інтегрованість (математики, ІКТ та фахових дисциплін майбутніх економістів);
- гнучкість (добору освітніх технологій навчання);
- інтерактивність (здатність взаємодіяти або знаходитися в режимі діалогу з ким-небудь (людиною) або з чим-небудь (наприклад, комп'ютером)).

Педагогічні цілі використання програмних засобів:

1) розвиток особистості студента на основі використання ІКТ:

- розвиток мислення (логічного, алгоритмічного тощо);
- розвиток комунікативних здібностей (робота у колективі, парі, малій групі тощо);
- розвиток дослідницьких здібностей (комп'ютерне моделювання);
- формування інформаційної культури (уміння здійснювати обробку інформації);

2) інтенсифікація навчально-виховного процесу:

- застосування ІКТ у процесі підготовки до заняття та у процесі проведення самого заняття (візуалізація навчального матеріалу, наочність тощо);
- встановлення міжпредметних зв'язків за рахунок використання ІКТ під час вивчення різних навчальних дисциплін;

3) удосконалення інформаційно-методичного забезпечення педагогічної діяльності:

- розширення можливостей спілкування та співпраці на основі комп'ютерних засобів комунікації;
- надання можливостей безперервного підвищення кваліфікації та перепідготовки незалежно від віку, географії проживання та часу;
- створення єдиного інформаційно-освітнього середовища на основі активного використання комп'ютерних мереж різного рівня (глобальних, корпоративних, локальних).

Сучасні інформаційно-комунікаційні технології надають практично необмежені можливості для самостійної та спільної творчої діяльності викладача та студента.

Функціональні властивості сучасних інформаційно-комунікаційних технологій сприяють реалізації наступних можливостей:

- необмежені можливості збору, зберігання, передачі, перетворення, аналізу та застосування різноманітної за своєю природою інформації;
- підвищення доступності освіти з розширенням форм отримання освіти;
- забезпечення безперервності отримання освіти та підвищення кваліфікації протягом всього активного періоду життя;
- розвиток особистісно-орієнтованого навчання, додаткової і випереджаючої освіти;
- значне розширення і вдосконалення організаційного забезпечення освітнього процесу (віртуальні школи, лабораторії, університети інше);
- підвищення активності суб'єктів в організації та веденні навчального процесу;
- створення єдиного інформаційно-освітнього середовища навчання і не тільки одного регіону, але країни і світового співтовариства в цілому;
- незалежність освітнього процесу від місця і часу навчання;
- значне вдосконалення і збагачення методичного та програмного забезпечення освітнього процесу;

червоним, то бажаного результату досить просто досягти за допомогою умовного форматування. Є дві можливості зв'язати автоматичне присвоєння формату комірки з деякими умовами.

Перша з них - присвоєння формату формулою. Функція СТИЛЬ може бути додана до існуючої формулою у клітинці. Таким чином, разом з функцією поточного ви можете задати, наприклад, колір клітинки залежно від значення. Використання формули $=...+ \text{СТИЛЬ} (\text{ЯКЩО} (\text{Поточна} ()) > 3; \text{«червоний»}, \text{«зелений»})$ викликає фарбування осередки в червоний колір, якщо значення більше 3, і призначає стиль з іменем «зелений» в іншому випадку (зрозуміло, якщо зазначені стилі визначені).

Інша можливість - використання умовного формату. За допомогою пункту Умовне форматування в меню Формат у діалозі допускається завдання до трьох умов, які повинні бути виконані, щоб осередок чи група виділених осередків отримала певний формат. При зміні даних стилі форматування будуть змінюватись автоматично.

Формули

При завданні формул поряд з основними арифметичними операціями, OpenOffice Calc надає безліч спеціальних функцій, які можна ввести в інтерактивному режимі за допомогою Автопілота функцій. OpenOffice Calc підтримує, зокрема, багато статистичні методи: від регресійних розрахунків до довірчих інтервалів. Особливо цікава можливість змінювати окремі параметри в обчисленнях, що залежать від багатьох факторів, і простежувати, як це впливає на результат. Це так звані розрахунки «що було б, якщо». Наприклад, при розрахунку кредиту шляхом простої зміни періоду, процентної ставки або сум виплати можна відразу ж побачити, як змінюються інші фактори.

Як і всі інші засоби роботи з електронними таблицями, OpenOffice Calc дозволяє використовувати відносні та абсолютні посилання. Перед кожним значенням, яке має використовуватися як абсолютне, ставиться знак долара \$. Для перетворення поточної посилання, в якій знаходиться курсор у рядку введення, з відносною в абсолютну і навпаки, слід використовувати комбінацію клавіш Shift + F4.

Практично всі функції (за винятком математичних та деяких статистичних) в програмі OpenOffice Calc локалізовані, тобто використовують російськомовні імена. До таких належать і всі функції з розділу Дата & Час (однак функція ПАСХАЛЬНОЕВОСКРЕСЕНЬЕ () визначає дату католицької, а не православної Пасхи).

Для введення функцій у клітинку можна скористатися майстром функцій. Виберіть клітинку, в яку треба ввести функцію, і натисніть кнопку Майстер функцій, яка знаходиться на панелі інструментів, або виберіть команду Функція ... з меню Вставка. Перед вами з'явиться вікно Автопілот функцій, в якому треба виділити функцію і натиснути кнопку Далі>> або ОК, після чого з'явиться вікно введення аргументів вибраної функції. Функцію можна вводити не тільки за допомогою майстра функцій, але і вручну, якщо ви пам'ятаєте, як вона називається і скільки у неї параметрів.

осередку (якщо в меню Довідка включена опція Підказка). Про наявність коментаря свідчить невеликий червоний квадратик у верхньому правому куті клітинки. У контекстному меню можна відзначити пункт Показати примітки для постійного їх відображення. Клацання в полі коментаря дозволяє приступити до його редагування.

Форматування даних та осередків

При створенні практично будь-якого документа в тій або іншій формі використовується форматування. Електронні таблиці не є винятками і будь-яка програма для їх створення підтримує можливості форматування. Особливістю цієї програми є як можливість використання спочатку заданих форматів, так і створення своїх власних стилів клітинок.

Для того щоб приступити до форматування клітинки або осередків їх слід виділити, потім за допомогою пункту Осередок ... з меню Формат або пункту Формат ячеек контекстного меню відкрити вікно Атрибути осередку, яке містить вкладки, що дозволяють задати параметри форматування. На вкладці Числа можна вибрати формат числа. За допомогою вкладки Шрифт можна встановити тип, розмір і колір шрифту.

Розглянемо кілька детальніше вкладку Вирівнювання. Вона призначена для управління вирівнюванням вмісту клітинки. Крім того, тут установлюється відстань від ліній сітки і відправлення листа. Перемикачі По горизонталі та По вертикалі дозволяють задати вирівнювання вмісту осередку в горизонтальному і вертикальному напрямку.

Якщо перемикач По горизонталі встановлений в режим Стандарт, то використовуються стандартні правила вирівнювання: числа вирівнюються по правому краю, а текст - по лівому.

Кругла кнопка дозволяє за допомогою миші плавно змінювати кут нахилу листи. Вертикальна кнопка встановлює відображення вмісту клітинки по вертикалі, як би в стовпчик.

Якщо вказано опцію Розрив рядка, то буде дозволений автоматичний розрив рядка на краю осередку. Відзначимо, що натискання комбінації клавіш Ctrl + Enter в будь-якому місці тексту також призведе до розриву рядка.

При роботі з осередками електронної таблиці, також як і з текстовими документами, можна використовувати як жорстке, так і м'яке форматування: або безпосередньо у клітинці задати певний розмір шрифту, або створити стиль з необхідним розміром шрифту і застосувати його до осередку. Для документів, з якими доводиться працювати часто, і які при цьому повинні виглядати однаково, рекомендується застосовувати м'яке форматування, тобто скористатися стилем, а при роботі з документами, які необхідно тільки швидко надрукувати, припустимо застосування жорсткого форматування. Стили клітинок забезпечують особливу зручність і дозволяють отримувати деякі цікаві ефекти.

OpenOffice Calc дозволяє застосовувати стилі осередків як в ручному режимі, так і автоматично - в залежності від певних умов. Якщо ви хочете особливим чином виділити деякі дані в таблиці, наприклад, всі значення вище середнього виділити зеленим кольором, а значення нижче середнього -

- забезпечення можливості вибору індивідуальної траєкторії навчання;
- розвиток самостійної творчо розвиненої особистості;
- розвиток самостійної пошукової діяльності студента, розвиток нових видів діяльності;
- підвищення мотиваційної сторони навчання та інше.

Інформаційно-комунікаційні технології є засобом за допомогою, якого викладач може:

- якісно змінювати (доповнити, удосконалити) технології навчання,
- здійснювати особистісний підхід до студентів,
- забезпечувати динамічне оновлення організації навчального процесу,
- формувати цілісну картину навчання.

Практика використання ІКТ під час навчання математики у вищих навчальних закладах сьогодні знаходиться у стадії становлення. Це частково пояснюється тим, що необхідні для впровадження ІКТ програмні засоби стали доступними навчальним закладам лише в останні роки. Разом з тим, ще однією причиною недостатньо інтенсивного впровадження ІКТ у навчальний процес математики є, безумовно, незначна кількість методичних розробок щодо ІКТ у математиці вищого навчального закладу та невідповідність більшості викладачів математики ВНЗ до використання педагогічних програмних засобів (ППЗ) у навчальному процесі.

Цілі використання ІКТ у навчальному процесі:

- розвиток творчої активної особистості;
- підготовка студента до життєдіяльності в умовах інформаційного суспільства;
- реалізація соціального замовлення, який обумовлений інформатизацією суспільства;
- інтенсифікація всіх рівнів навчально-виховного процесу;
- створення якісно нових умов та засобів для наукового пошуку.

Щодо навчання математики, то ІКТ, крім зазначеного вище, дає можливість:

- економії часу на занятті, як наслідок більше часу викладач може виділити для мотивації навчання студентів, розв'язування прикладних задач та організації дослідницької і самостійної роботи (у поза аудиторний час),
- індивідуалізації та диференціації навчального процесу за умови збереження його цілісності,
- стимулювання пізнавальної активності,
- здійснення контролю, корекції та самоконтролю, самокорекції,
- включення до навчального процесу нових форм роботи, наприклад, моделювання економічних процесів та явищ за допомогою різного програмного забезпечення,
- опрацювання значних обсягів інформації за менший час,
- створення стійкого пізнавального мотиву тощо.

У своїй роботі Клочко В.І. показав, що застосування нових інформаційних технологій у навчанні дозволяє:

- привести зміст підготовки у відповідність до вимог сучасного життя;
- поєднати традиційні технології і сучасні інформаційні технології навчання, підвищити інтенсивність навчально-пізнавальної діяльності студентів;
- поглибити інтегрований вплив на особистість студента за рахунок доступу до сучасних інформаційних технологій, що дає можливість розширити і поглибити оволодіння вміннями та навичками, необхідними студенту в подальшому житті;
- урізноманітнити форми участі студентів у навчальній діяльності та прийоми опрацювання навчального матеріалу, активніше формувати професійно значимі знання, вміння і навички.;
- залучити студентів до науково-пізнавальної діяльності, що забезпечує систематичність, поглиблення та розширення теоретичної бази знань, розкриття творчого потенціалу студентів;
- підвищити мотивацію вивчення теоретичного матеріалу і курсу в цілому» [47].

Жалдак М.І. вважає, що широке впровадження в навчальний процес нових інформаційних технологій навчання, що базується на комп'ютерній підтримці навчально-пізнавальної діяльності, відкриває далекосяжні перспективи щодо розширення і поглиблення теоретичної бази знань та надання результатам навчання практичної значущості, інтеграції навчальних предметів і диференціювання навчання та активізації навчально-пізнавальної діяльності, посилення спілкування студентів і викладача між собою, збільшення частки самостійного дослідницького характеру навчальної діяльності, розкриття творчого потенціалу студентів і викладача з урахуванням їхніх позицій, уподобань, специфіки забезпечення та перебігу навчального процесу [27].

Метою вивчення математики (згідно з Галузевим стандартом) є формувати системи теоретичних знань і практичних навичок з основ математичного апарату, основних методів кількісного вимірювання випадковості дії факторів, що впливають на будь-які процеси, засад математичної статистики, яка використовується під час планування, організації та управління виробництвом, оцінювання якості продукції, системного аналізу різноманітних структур та технологічних процесів тощо.

Основними завдання навчання математики у закладах вищої освіти:

- вивчення основних принципів та інструментарію математичного апарату, який використовується для розв'язування прикладних задач, математичних методів систематизації, опрацювання та застосування статистичних даних для наукових та практичних висновків,
- ознайомлення студентів із роллю математики у сучасному житті,
- навчання студентів основним теоретичним положенням, що необхідні для подальшого вивчення фахових дисциплін,

2.8. OpenOffice Calc

2.8.1. Загальні відомості про OpenOffice Calc

OpenOffice Calc — електронні таблиці, застосунок з офісного пакету OpenOffice.

Calc за своїми можливостями приблизно відповідає Microsoft Excel, та може відкривати і зберігати файли в форматі Excel. Додатково Calc має низку особливостей, не представлених в Excel, наприклад, він має систему, яка автоматично визначає послідовності графіків, побудованих на основі даних користувача. Calc також вміє писати таблиці прямо в форматі PDF.

Форматом по замовчуванню для OpenOffice.org 2.0 Calc може бути встановлений або формат Microsoft Excel, або OASIS Open Document Format (ODF). Додатково Calc підтримує низку інших форматів, зі здатністю читати і писати в них.

Calc може зберігати максимум 65536 рядків та 256 стовпчиків в кожній таблиці, в кожному файлі максимум до 256 таблиць. До версії 2.0 обмеження на кількість рядків становила 32000.

Як і решта застосунків з пакету OpenOffice.org, Calc здатний працювати на різних платформах, включаючи Mac OS X, Microsoft Windows, Linux, FreeBSD та Solaris. Випускається під ліцензією GNU Lesser General Public License, отже Calc є вільним програмним забезпеченням.

Електронні таблиці OpenOffice Calc володіють на даний момент найбільшими можливостями серед всіх вільно розповсюджуваних програм подібного класу. Ця програма є частиною проекту OpenOffice, метою якого є надати користувачеві аналог комерційного продукту Microsoft Office, і практично не відрізняється від MS Excel по функціональності. Детальна вбудована документація та зручна система довідки дозволяють користувачеві швидко освоїти всі особливості роботи з цим програмним продуктом.

Запуск OpenOffice Calc здійснюється командою soffice. Після старту програми в меню Файл слід вибрати пункт Відкрити, якщо ви збираєтеся редагувати вже існуючий файл, або в пункті Створити вибрати опцію Документ електронної таблиці. Робоча книга за замовчуванням містить 3 аркуші, але якщо кількість аркушів у книзі або їх назва вас не влаштовує, то ви можете легко додати, видалити або перейменувати їх. Подвійний клацання в області заголовків листів книги призводить до появи меню, що дозволяє виконати зазначені операції.

Контекстні меню програми, що з'являються при натисканні правої кнопки миші, пов'язані з певними об'єктами програми, такими як елементу таблиці, заголовки рядків, стовпців або листів і т. д. Процес введення, редагування даних, створення формул у програмі OpenOffice Calc практично ідентичний процесу роботи з уже розглянутими електронними таблицями.

OpenOffice Calc може розмішати в осередках своїх таблиць числа (використовуючи кому для відділення дробової частини), формули і текст. До кожного осередку може бути доданий коментар (меню Вставка, пункт Примітки), який автоматично відображається при підведенні курсору до

методів тощо). Всі необхідні обчислення та перетворення можна виконувати у табличному процесорі Excel, використовуючі при цьому розглянуті вище операції та функції.

2.7.6. Точковий прогноз

Виконати прогноз прибутку фірми на майбутні, приміром, чотири місяці можна за допомогою функції ТЕНДЕНЦИЯ.

Для цього треба виконати наступні дії:

- 1) виділіть область порожніх комірок, яка складається з одного стовпчика та 4 рядків (за кількістю прогнозних періодів);
- 2) активізуйте Мастер функцій;
- 3) виберіть Категорію Статистические, функцію ТЕНДЕНЦИЯ. Клацніть на ОК;
- 4) заповніть аргументи функції (рис. 2.169):

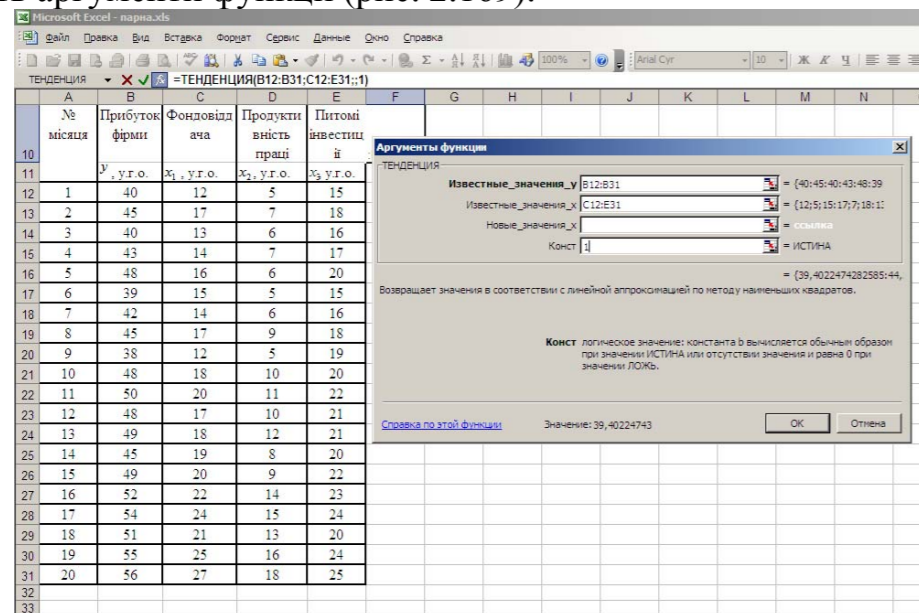


Рис. 2.169 – Приклад точкового прогнозу

- *Известные значения Y* – діапазон, який містить дані, що характеризують результуючу;
 - *Известные значения X* – діапазон, який містить дані, що описують всі незалежні змінні;
 - *Новые значения X* – діапазон комірок, який містить значення змінних, які необхідно підставити у рівняння для визначення очікуваних значень y ;
 - *Константа* – логічне значення, яке вказує на наявність або відсутність вільного члена у рівнянні (*Константа* = 1). Клацніть на ОК.
- 5) у лівій верхній комірці виділеної області з'явиться перший елемент результуючого масиву даних; для того, щоб вивести всі дані, клацніть <F2>, а потім – комбінацію клавиш <Ctrl>+<Shift>+<Enter>.

Зауважимо, що інтервальний прогноз є більш реалістичним, ніж точковий. Оскільки вказує не одне значення, а діапазон значень, якому із вказаною ймовірністю повинно належати майбутнє значення результуючої.

- розвиток логічного та алгоритмічного мислення у студентів,
- формування навичок застосовувати отримані теоретичні знання на практиці,
- формування навичок самостійності у роботі та здобутті знань й вмінь.

Знання, які були набуті під час вивчення математики, нададуть можливість майбутньому фахівцю самостійно розв'язувати актуальні прикладні завдання засобами математики.

Вища математика є продовженням шкільної математики та фундаментом для вивчення у подальшому фахових дисциплін.

У зв'язку з тим, що сьогодні активно відбувається інформатизація всіх сфер життя людини, і освіти у тому числі, тому необхідно модернізувати цілі та завдання навчання всіх математичних дисциплін у ВНЗ.

Таким чином, до мети варто додати, наприклад:

- формування вмінь автоматизованих математичних обчислень,
- формування вмінь застосувати відповідне програмне забезпечення до побудови математичних моделей явищ та процесів.

Згідно із запропонованими цілями треба доповнити завдання вивчення математики:

- навчання всіх розділів курсу математики здійснювати із застосуванням інформаційно-комунікаційних технологій та на основі інтеграції математики та фахових дисциплін.
- навчання студентів вчитись,
- формування інтересу до вивчення математики.

ІКТ можна класифікувати за наступними критеріями:

1. За педагогічними завданнями:

- теоретична підготовка (електронні енциклопедії, підручники, посібники тощо),
- практична підготовка (тренажери, електронні підручники та посібники з вбудованими підказками для виконання завдань тощо),
- контроль рівня набутих знань (програмні продукти для тестування та перевірки рівня знань студентів),
- виховна робота (аудіо-, відео- матеріали з патріотичного виховання та виховання у молоді важливих для суспільства якостей (толерантність, повага тощо)),
- комплексне навчання (дистанційне навчання)

2. За функціями у освітньому процесі:

- навчальні,
- інформативні,
- пошукові.

3. За способом застосування ІКТ у освітньому процесі:

- аудиторні,
- позааудиторні.

4. За формою взаємодії:

- «online»,

- «offline».

Однак ефективність та якість навчально-виховного процесу, значною мірою буде залежати від доцільного застосування інформаційно-комунікаційних технологій навчання. Використання ІКТ на певному етапі навчання (на певному уроці) є доцільним, якщо:

а) «вища, ніж при використанні традиційних засобів навчання, ефективність»;

б) неможливість реалізації певних засобів навчання у вигляді матеріальних об'єктів (оригінали у природних умовах, оригінали у штучних умовах, модельні еквіваленти оригіналів - фізичні моделі);

в) недостатня наочність та зрозумілість або надлишкова складність відповідних вербально-знакових, графічних (статичних та динамічних), знакових, логічно-математичних моделей» [37].

До можливих негативних наслідків застосування ІКТ ми відносимо:

- ізоляції студентів,
- невміння спілкуватись з однолітками (відсутність комунікативної складової навчання або її недостатність).

Використання ІКТ на занятті створює умови для моделювання (для створення математичної моделі), для заощадження часу, залишаючи технічну сторону розв'язання задачі комп'ютеру. Для навчально-виховного процесу це дає значні переваги, оскільки більшість задач, які студенти розв'язують, є задачі з прикладним змістом, що в свою чергу вимагає від них вміння перекладати умову задачі на мову математики, тобто моделювати.

При використанні ІКТ у навчанні математики треба з'ясувати проблему дослідження, визначити завдання, розробити відповідну математичну модель та розв'язати й інтерпретувати результати за допомогою ІКТ.

Відомий математик і методист Д.Пойа говорив, що коли вчитель математики заповнить відведений йому навчальний час "натаскуванням" тих, хто навчається в шаблонних вправах, він знищить їх інтерес, загальмує їх розумовий розвиток та втратить свої можливості [60]. Від ефективності використання задач у навчанні математики значною мірою залежить не тільки якість навчання, виховання й розвитку студентів, але й ступінь їх практичної підготовленості до наступної професійної діяльності.

З метою економії часу на уроці, підвищення ефективності уроку, розвитку раціонального мислення та формування у студентів основ комп'ютерної грамотності доцільно використовувати різноманітні програмні засоби навчального призначення (педагогічні програмні засоби - ППЗ).

Педагогічні програмні засоби відіграють надзвичайно важливу роль у організації навчального процесу, зокрема з математики.

Ми вважаємо, що педагогічні програмні засоби (ППЗ) такі як Gran1, Gran-2D, Gran-3D, динамічна геометрія (DG), динамічна математика GeoGebra, різноманітні системи комп'ютерної алгебри (CAS) Mathcad, Derive, Maple, Mathematica, база знань та набір обчислювальних алгоритмів Wolfram Alpha, табличний процесор Microsoft Excel, вільне програмне забезпечення (ВПЗ)

Визначення «нормалізованих» оцінок параметрів.

Для цього скористаємося функцією ЛИНЕЙН без визначення параметру b_0^*

(при заповненні робочого поля функції в вікні КОНСТ позначити «ЛОЖЬ» або «0»)

b_3^*	b_2^*	b_1^*	
0,3103	0,3024	0,382	0
0,1491	0,2009	0,214	#Н/Д
0,9269	0,2858	#Н/Д	#Н/Д
71,878	17	#Н/Д	#Н/Д
17,612	1,3885	#Н/Д	#Н/Д

Істинні оцінки параметрів моделі визначаються за формулами:

$$b_1 = b_1^* \times \left(\frac{\sigma_y}{\sigma_{x_1}} \right), \quad b_2 = b_2^* \times \left(\frac{\sigma_y}{\sigma_{x_2}} \right), \quad b_3 = b_3^* \times \left(\frac{\sigma_y}{\sigma_{x_3}} \right),$$

$$b_0 = \bar{Y} - b_1 \bar{X}_1 - b_2 \bar{X}_2 - b_3 \bar{X}_3$$

Для зручності зведемо всі необхідні для розрахунків дані в допоміжну таблицю 2.23.

Таблиця 2.23

Показник	y	x_1	x_2	x_3
Середнє значення	46,85	18,05	9,60	19,80
Середнє квадратичне відхилення, σ	5,38	4,26	3,97	3,04

Істинні параметри:

$$b_1 = 0,382 \times \left(\frac{5,38}{4,26} \right) = 0,482, \quad b_2 = 0,302 \times \left(\frac{5,38}{3,97} \right) = 0,409, \quad b_3 = 0,310 \times \left(\frac{5,38}{3,04} \right) = 0,549.$$

$$\text{Звідси, } b_0 = 46,85 - 0,482 \times 18,05 - 0,409 \times 9,60 - 0,549 \times 19,80 = 23,353$$

Модель прибутку без колінеарності має вигляд:

$$Y = 23.353 + 0.482x_1 + 0.409x_2 + 0.549x_3.$$

Оцінка достовірності побудованої моделі проведемо за допомогою визначення коефіцієнту кореляції та коефіцієнту детермінації.

Коефіцієнт кореляції визначимо за допомогою функції КОРРЕЛ, категорія «Статистичні»

$$r_{yy_r} = 0,9628.$$

$$\text{Коефіцієнт детермінації } R^2 = (0,9628)^2 = 0,9269$$

Примітка. Дослідження масиву на наявність гетероскедастичності та автокореляції здійснюється за допомогою відповідних тестів (алгоритмів,

Таблиця 2.22

Нормалізація залежної змінної	
Прибуток фірми Y, у.г.о.	Y*
40	-1,273
45	-0,344
40	-1,273
43	-0,715
48	0,2136
39	-1,458
42	-0,901
45	-0,344
38	-1,644
48	0,2136
50	0,5852
48	0,2136
49	0,3994
45	-0,344
49	0,3994
52	0,9567
54	1,3283
51	0,771
55	1,514
56	1,6998

Тоді, масив нормалізованих змінних для визначення оцінок параметрів нормалізованої моделі буде наступним (таблиця 2.23):

Таблиця 2.23

Вхідні дані для побудови «нормалізованої моделі»

№ місяця	Y*	X ₁ *	X ₂ *	X ₃ *
1	-1,273	-1,41989	-1,15996	-1,58071
2	-0,344	-0,24643	-0,65563	-0,59276
3	-1,273	-1,1852	-0,9078	-1,25139
4	-0,715	-0,9505	-0,65563	-0,92208
5	0,2136	-0,48112	-0,9078	0,065863
6	-1,458	-0,71581	-1,15996	-1,58071
7	-0,901	-0,9505	-0,9078	-1,25139
8	-0,344	-0,24643	-0,1513	-0,59276
9	-1,644	-1,41989	-1,15996	-0,26345
10	0,2136	-0,01173	0,100866	0,065863
11	0,5852	0,45765	0,353032	0,72449
12	0,2136	-0,24643	0,100866	0,395176
13	0,3994	-0,01173	0,605198	0,395176
14	-0,344	0,222958	-0,40347	0,065863
15	0,3994	0,45765	-0,1513	0,72449
16	0,9567	0,927034	1,10953	1,053804
17	1,3283	1,396419	1,361696	1,383117
18	0,771	0,692342	0,857364	0,065863
19	1,514	1,631111	1,613862	1,383117
20	1,6998	2,100496	2,118194	1,712431

Open Office Calc та інші можуть бути ефективно застосовані у навчанні математики.

Можна виділити кілька аспектів використання ППЗ у освітньому процесі:

1. Змістовий (виклад навчального матеріалу, побудова таблиць, створення плакатів, тренажерів для самостійної роботи студентів тощо).

2. Мотиваційний (максимальне врахування індивідуальних особливостей студентів, усвідомлення студентами важливості ІКТ у навчанні та подальшій професійній діяльності тощо).

3. Методичний (можливість вибору різноманітних засобів навчання, форм, методів, прийомів, технологій, створення унікальних занять, організації самостійної роботи тощо).

4. Контрольно-оціночний (визначення рівня навчальних досягнень студентів, виставлення автоматично оцінок, проведення тестування як в режимі on-line та off-line, використовувати різноманітні програмні продукти, наприклад, My Test, Т-тест тощо).

Аналіз практичної діяльності засвідчи, що недостатнє практичне застосування інформаційно-комунікаційних технологій у практиці роботи викладачів математики обумовлене недостатньою кількістю або повною відсутністю відповідних навчально-методичних рекомендацій з навчання студентів застосуванню різноманітних програмних засобів.

Серед причин, які призводять до неефективного застосування програмних продуктів у процес навчання математики це ігнорування викладачем дидактичних принципів навчання, використання традиційних методів та принципів навчання без врахування особливостей навчання із інформаційно-комунікаційними технологіями, а також недоцільний добір навчального матеріалу для наповнення ним програмного продукту.

Отже, реформа сучасної освіти, математичної зокрема, може бути здійснена лише за умови інтеграції ІКТ у навчальний процес всіх дисциплін вузу. Адже застосування сучасних інформаційно-комунікаційних технологій на заняттях з математики призводить до створення умов для розвитку вмінь і навичок самоосвіти, саморозвитку, для формування дослідницьких якостей особистості студентів.

Характерною особливістю навчання математики із ІКТ є те, що у викладача з'являється можливість здійснювати навчання на якісно більш високому рівні. При цьому, на нашу думку, не завжди таке навчання вимагає створення принципово нової методики навчання. Нерідко викладачу достатньо органічно поєднати звичні форми та прийоми роботи із ІКТ.

1.5. Програмні засоби навчального призначення у навчанні математики

Перед викладачами математики завжди стоїть завдання не лише дати студентам міцні знання і навички з дисципліни, а і розвинути їх логічне мислення, зацікавити вивченням математики, активізувати їх пізнавальну

діяльність, привчити працювати самостійно, щоб, в подальшому вони могли підвищувати свій освітній та професійний рівні.

Крім перерахованих вище та основних завдань, передбачених програмою з математики, перед викладачами, що викладають математичні дисципліни постають додаткові завдання. А саме:

- отримання студентами досвіду прийняття ефективних рішень,
- формування елементів комп'ютерної грамотності засобами математики,
- розвиток раціонального мислення.

Використання ІКТ у навчальному процесі вносить суттєві, якісні зміни в традиційний навчальний процес. Це відбувається тому, що інформаційно-комунікаційні технології не є простим додатком до існуючої системи навчання, вони вносять зміни в усі компоненти навчально-виховного процесу, зокрема, мету, зміст, методи, організаційні форми, роль викладача тощо.

Сьогодні кількість комп'ютерних класів у закладах вищої освіти постійно зростає. Проте не використовуються ті, практично не обмежені можливості, що відкриваються перед студентами при реальному використанні ІКТ у навчальному процесі. Недоліком ІКТ є їх недоцільне або невміле використання викладачем на занятті. Значна частина викладачів математики не вміють користуватись комп'ютером на належному рівні, не мають відповідного програмного забезпечення або просто не бажають відмовлятися від традиційних методів навчання.

В основу технології створення програмних засобів навчання, покладено ідеї, взяті з різних галузей науки [69]:

- теорії управління (алгоритмізація дій студента, формалізація і передача ряду функцій викладача ЕОМ, безперервний контроль і реалізація зворотного зв'язку);
- психології (особистісно-орієнтований підхід до організації процесу навчання, формування розумової діяльності через зовнішні впливи, облік індивідуальних особливостей, що навчається і т.д.);
- дидактики (основні принципи традиційної дидактики і особливі принципи комп'ютерно-орієнтованих технологій навчання, управління пізнавальної діяльністю учня, підготовка і подача матеріалу, облік сучасних дидактичних можливостей комп'ютерної техніки та засобів телекомунікаційного зв'язку, зміна ролі та функцій педагога);
- методики (організації занять на основі пошуку оптимального поєднання індивідуальних, групових (малих груп) і колективних форм організації навчання; видозміна характеру спілкування між педагогом і студентом, використання особистісно-орієнтованого підходу до навчання).

Розробка технології виробництва програмних засобів навчання має на меті:

- прискорення розробки, підвищення якості та надійності засобів навчання і контролю.

- між фондоддачею та продуктивністю праці існує тісний зв'язок, якщо не враховувати вплив питомих інвестицій;

- між фондоддачею та питомими інвестиціями існує помірний зв'язок, якщо не враховувати вплив продуктивності праці;

- між продуктивністю праці та питомими інвестиціями існує слабкий зв'язок, якщо не враховувати вплив фондоддачі.

Перевірка на колінеарність кожної пари змінних за допомогою *t*-критерія

Критерії розраховуються за формулою:

$$t_{ij} = \frac{r_{ij,k} \sqrt{n-m}}{\sqrt{1-r_{ij,k}^2}}$$

t-критерій для фактору x_1 (фондоддача):

$$t_{12} = \frac{0,737 \times \sqrt{20-3}}{\sqrt{1-0,737^2}} = 4,496$$

t-критерій для фактору x_2 (продуктивність праці):

$$t_{13} = \frac{0,411 \times \sqrt{20-3}}{\sqrt{1-0,411^2}} = 1,859$$

t-критерій для фактору x_3 (питомі інвестиції):

$$t_{23} = \frac{0,237 \times \sqrt{20-3}}{\sqrt{1-0,237^2}} = 1,006$$

Розраховане значення *t*-критерію порівнюємо з табличним для $\nu = n - m = 20 - 3 = 17$ ступенів вільності та рівня значущості $\alpha = 0,05$ (див. табл. «Процентилі *t*-розподілу»): $t_{табл} = 1,740$.

Оскільки $t_{12} > t_{табл}$, то фондоддача та продуктивність праці колінеарні між собою; $t_{13} > t_{табл}$, то фондоддача та питомі інвестиції колінеарні між собою; $t_{23} < t_{табл}$, то продуктивність праці та питомі інвестиції не колінеарні між собою.

Розрахунки показали, що в масиві незалежних змінних існує колінеарність між факторами x_1 та x_2 , а також між x_1 та x_3 . Тому для визначення незміщених, ефективних та обґрунтованих оцінок параметрів (істинних оцінок) необхідно позбутися наслідків колінеарності.

Визначення істинних оцінок параметрів моделі $Y = f(X_1 X_2 X_3)$

Нормалізація фактору Y – прибуток фірми

За допомогою функцій СРЗНАЧ, СТАНДОТКЛОН та НОРМАЛІЗАЦІЯ здійснимо допоміжні обчислення (таблиця 2.21, 2.22).

Таблиця 2.21

Допоміжні обчислення	
	<i>y</i>
«СРЗНАЧ»	46,85
«СТАНДОТКЛОН»	5,38

$$F = (C_{ii} - 1) \left(\frac{n-m}{m-1} \right),$$

де C_{ii} - діагональний елемент матриці C (матриці, оберненої до кореляційної матриці r)

Знайдемо обернену матрицю C за допомогою функції МОБР категорії «Математичні»:

$$C = \begin{vmatrix} 10,671 & -7,376 & -3,055 \\ -7,376 & 9,393 & -1,650 \\ -3,055 & -1,650 & 5,169 \end{vmatrix} \quad \text{структура матриці } C = \begin{vmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{vmatrix}$$

F -критерій для фактору x_1 (фондовіддача):

$$F_1 = (10,671 - 1) \left(\frac{20-3}{3-1} \right) = 82,205$$

F -критерій для фактору x_2 (продуктивність праці):

$$F_2 = (9,393 - 1) \left(\frac{20-3}{3-1} \right) = 71,34$$

F -критерій для фактору x_3 (питомі інвестиції):

$$F_3 = (5,169 - 1) \left(\frac{20-3}{3-1} \right) = 35,436$$

Розраховане значення F -критерію порівняємо з табличним для $\nu_1 = n - m = 20 - 3 = 17$ та $\nu_2 = m - 1 = 3 - 1 = 2$ ступенів вільності та рівня значущості $\alpha = 0.05$ (див. табл. « F -розподіл»): $F_{табл} = 3,59$

Оскільки $F_1 > F_{табл}$, $F_2 > F_{табл}$, $F_3 > F_{табл}$ то кожна із незалежних змінних (фондовіддача, продуктивність праці, питомі інвестиції) колінеарна з іншими.

Визначення частинних коефіцієнтів кореляції $r_{ij,k}$

Коефіцієнти розраховуються за формулою:

$$r_{ij,k} = \frac{-C_{ij}}{\sqrt{C_{ii} C_{jj}}}$$

Коефіцієнт кореляції $r_{ij,k}$ для факторів x_1 (фондовіддача) та x_2 (продуктивність праці):

$$r_{12,3} = \frac{-(-7,376)}{\sqrt{10,671 \times 9,393}} = 0,737$$

Коефіцієнт кореляції $r_{ij,k}$ для факторів x_1 (фондовіддача) та x_3 (питомі інвестиції):

$$r_{13,2} = \frac{-(-3,055)}{\sqrt{10,671 \times 5,169}} = 0,411$$

Коефіцієнт кореляції $r_{ij,k}$ для факторів x_2 (продуктивність праці) та x_3 (питомі інвестиції):

$$r_{23,1} = \frac{-(-1,650)}{\sqrt{9,393 \times 5,169}} = 0,237$$

На основі частинних коефіцієнтів кореляції можна стверджувати:

- уніфікацію підготовки програмних засобів, що дозволить викладачу, що не є фахівцем в галузі програмування, готувати авторські автоматизовані курси навчання та контролю.

- забезпечення можливості безперервного уточнення та оновлення матеріалу, що виноситься на розгляд.

Виділимо основні стадії створення програмних засобів навчання:

- стадія формування (розробка концептуальної моделі навчального процесу в комп'ютерному середовищі, виявлення принципів і вимог реалізації моделі у вигляді програмного комплексу);

- стадія концептуального проектування: детальний аналіз початкового рівня підготовленості, типів розумової діяльності студентів; аналіз необхідності, можливості та доцільності використання різних алгоритмів управління пізнавальною діяльністю; розгляд вимог і особливостей досліджуваної предметної галузі; обґрунтування необхідності супроводу навчального матеріалу статичними і динамічними додатками; розробка загальної архітектури системи навчання;

- стадія реалізації моделі освітнього процесу в комп'ютерному середовищі: створення системи формування готовності педагогів до спільної роботі з розробниками комп'ютерної середовища навчання; розробка системи навчання з урахуванням різноманітного і багаторівневого навчання; створення або вибір готових інструментальних засобів розробки програмного комплексу навчальної системи; реалізація гнучкої системи оцінки дій студентів; можливість і необхідність (за запитом) включення педагога в роботу навчальної системи; забезпечення різноманітних видів взаємодії всіх суб'єктів освітнього процесу; реалізація мережевого варіанту навчання; різні варіанти переривання роботи системи навчання; збереження результатів і маршруту навчання; розробка всіх додатків, що моделюють вимоги предметної галузі; реалізація можливості звернення студентів до інформаційних джерел і зовнішніх програм;

- стадія впровадження та супроводу полягає в апробації і налагодженні розробленого програмного комплексу в реальних умовах експлуатації.

До основних вимог розробки програмних засобів слід віднести:

- облік індивідуальних особливостей студентів: типу розумової діяльності та рівня розвитку пам'яті; початкового рівня підготовки; індивідуального темпу навчання; інших особливостей;

- реалізація гнучкого алгоритму управління процесом пізнавальної діяльності на основі аналізу успіхів навчання; надання студенту можливості самостійного вибору траєкторії навчання і подальшого її коригування залежно від результатів роботи останнього; забезпечення різноманітного навчання і контролю (покроковий, поетапний, підсумковий) з наданням інформації про результати контролю викладачеві в різному ступені деталізації і формі; збереження історії навчання кожного студента; адаптивність комп'ютерного засобу навчання до індивідуальних особливостей студента; надання можливості виходу з програмних засобів в інші середовища роботи з подальшим поверненням в точку виходу;

- подання навчальної інформації в різних форматах з обґрунтованим використанням мультимедійних технологій, що використовуються при підготовці матеріалу для вивчення;

- застосування методу моделювання як при постановці навчальних завдань, так і при виконанні творчих робіт в комп'ютерному середовищі навчання;

- забезпечення гнучкого, персоналізованого, різнопланового діалогу в навчальних системах;

- надання студенту можливості введення відповідей у різній формі (вільної фразою; складної відповіді з перестановкою слів; відповіді суворої конструкції; альтернативної відповіді; в інших формах);

- висока інтерактивність роботи в комп'ютерно-орієнтованій системі навчання, надання студенту можливості створювати запити системі у випадках нерозуміння або неоднозначного сприйняття поставленого навчального завдання;

- забезпечення дизайн-ергономічних вимог до представленого навчального матеріалу, надання можливості індивідуального налаштування роботи в системі, забезпечення зручного інтуїтивно зрозумілого інтерфейсу;

- прагнення до уніфікації програмних засобів навчання та контролю з використанням інструментальних середовищ створення програмних засобів і єдиних вимог підготовки предметного вивчення матеріалу;

- створення розвиненої пошукової системи за матеріалами середовища навчання: режими «лупа»; «автопоказ»; навідні запитання, інше;

- реалізація в програмних засобах навчання різних видів допомоги за допомогою різних форм діалогу, навчальних впливів;

- наявність інтуїтивно зрозумілого дружнього інтерфейсу;

- використання всіх можливих способів представлення інформації: у вигляді тексту, графіки, анімації, гіпертексту, мультимедіа; студент повинен мати можливість гортання інформаційного матеріалу в двох напрямках «вперед-назад» з можливістю встановлення типу і розміру шрифту, а також повторення будь-якого фрагмента бажану кількість разів;

- забезпечення контролю стомлення користувача, блоку релаксації: останній повинен містити тематично однорідні невеликі «банки» жартів, афоризмів, музичні фрагменти і т. д.;

- забезпечення реєстрації студента, створення протоколу і ведення історії навчання, нагромадження результатів навчання і інформування про успіхи навчальної діяльності з метою подальшого коректування;

- забезпечення надійної роботи системи: технічна коректність; захист від випадкового або неправильного натискання; забезпечення адекватної реакції на будь-які, навіть самі несподівані відповіді; програма не повинна «зависати» через непередбачену послідовність спрацювання окремих її модулів або інших причин.

Необхідна оцінка ефективності роботи створеної системи навчання не тільки з технологічної точки зору, але і з точки зору дидактичної ефективності навчального середовища.

Розрахунок визначника кореляційної матриці $\det(r)$

Для цього використаємо функцію МОПРЕД, категорія «Математичні»:

$$\det(r) = 0.0218$$

Оскільки визначник кореляційної матриці *наближається до 0*, то в масиві пояснюючих змінних може існувати мультиколінеарність.

Перевірка всього масиву змінних на наявність колінеарності за критерієм χ^2

Критерій розраховується за формулою:

$$\chi^2 = -\left\{n-1-\frac{2m+5}{6}\right\} \ln(\det r),$$

де m – кількість незалежних змінних.

Визначимо логарифм визначника кореляційної матриці (функція LN) :

$$\ln \det(r) = -3.825$$

Тоді,

$$\chi^2 = -\left\{20-1-\frac{2 \times 3+5}{6}\right\} \times (-3.825) = 65.663$$

Розраховане значення χ^2 порівняємо з табличним для $\nu = \frac{m(m-1)}{2} = \frac{3(3-1)}{2} = 3$ ступенів вільності та рівня значущості $\alpha = 0.05$ (див. табл. «Перцентилі χ^2 -розподілу»): $\chi^2_{\text{табл}} = 7,81$.

Табличне значення χ^2 можна також знайти в категорії Статистические Мастера функций Excel, функція ХИ2ОБР (рис. 2.168).

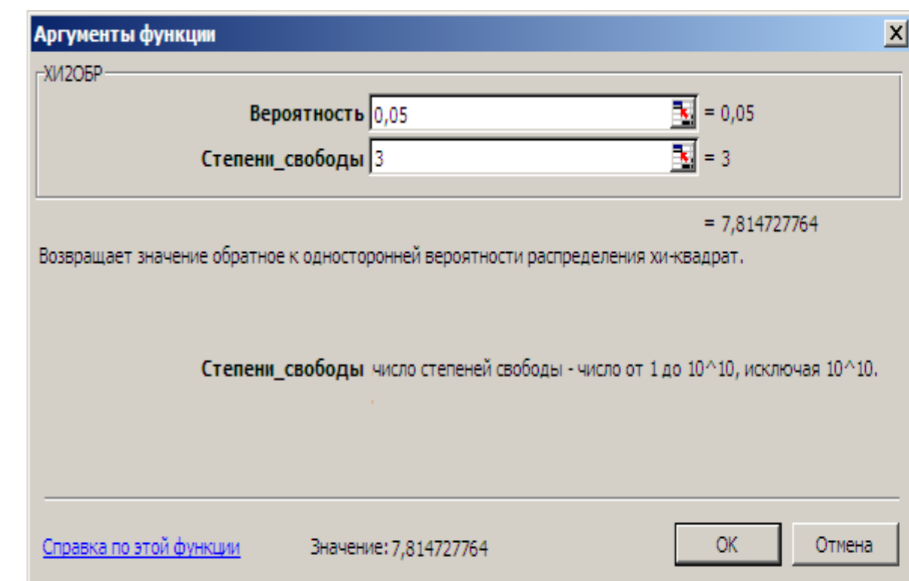


Рис. 2.168 – Визначення табличного значення χ^2

Оскільки $\chi^2 > \chi^2_{\text{табл}}$, то в масиві незалежних змінних (фондовіддача, продуктивність праці, питомі інвестиції) існує мультиколінеарність.

Перевірка на колінеарність окремих змінних з іншими за F-критеріями.

Критерій розраховується за формулою:

Транспонування нормалізованої матриці X^*

Для цього використаємо функцію ТРАНСП, категорія «Ссылки и массивы» або «Полный алфавитный перечень» (рис. 2.165):

29																				
30																				
31	-1.41989	-0.24643	-1.1852	-0.9505	-0.48112	-0.71581	-0.9505	-0.24643	-1.41989	-0.01173	0.45765	-0.24643	-0.01173	0.222958	0.45765	0.927034	1.396419	0.692342	1.631111	2.100496
32	-1.15996	-0.65563	-0.9078	-0.65563	-0.9078	-1.15996	-0.9078	-0.1513	-1.15996	0.100866	0.353032	0.100866	0.605198	-0.40347	-0.1513	1.10953	1.361696	0.857364	1.613862	2.118194
33	-1.58071	-0.59276	-1.25139	-0.92208	0.065863	-1.58071	-1.25139	-0.59276	-0.26345	0.065863	0.72449	0.395176	0.395176	0.065863	0.72449	1.053804	1.383117	0.065863	1.383117	1.712431
34																				

Рис. 2.165 – Результат використання функції ТРАНСП

Обчислення добутку матриць X^{*T} та X^*

Для цього використаємо функцію МУМНОЖ, категорія «Математичні» (рис. 2.166).

39				
40				
41		19.00001	17.89645	16.9414
42	$X^{*T} \times X^*$	17.89645	18.99998	16.64155
43		16.9414	16.64155	19.00001
44				
45				

Рис. 2.166 – Результат використання функції МУМНОЖ

Визначення кореляційної матриці r

Для цього необхідно кожний елемент матриці $X^{*T} \times X^*$ домножити на $\frac{1}{n-1}$, де n - кількість спостережень (у нашому випадку $n=20$)

$$r = \begin{vmatrix} 1 & 0,942 & 0,892 \\ 0,942 & 1 & 0,876 \\ 0,892 & 0,876 & 1 \end{vmatrix}$$

Визначити кореляційну матрицю також можна за допомогою програми Сервіс/Анализ данных/Корреляция. У результаті виведуться нижньо трикутна кореляційна матриця (рис. 2.167), яка є симетричною. Відтворити елементи, яких не вистачає, можна шляхом копіювання.

	A	B	C	D	E
1		Столбец 1	Столбец 2	Столбец 3	
2	Столбец 1	1			
3	Столбец 2	0.9419186	1		
4	Столбец 3	0.891652	0.87587116	1	
5					
6					

Рис. 2.167 – Кореляційна матриця

Таким чином, під час добору програмних засобів викладач, математики зокрема, має дотримуватись таких вимог:

- дидактичних,
- методичних,
- психологічних,
- ергономічних,
- технічних.

До дидактичних вимог програмного засобу для навчання математики ми відносимо:

- науковість змісту, що міститься у програмному продукті,
- достовірність даних, що представляються програмним продуктом,
- повнота наповнення, тобто охоплення всього курсу математики,
- доступність навчання,
- систематичність та послідовність, яка передбачає необхідність засвоєння системи знань,
- наступність та інтегрованість у оволодінні знаннями,
- адаптивність до конкретної дисципліни,
- інтерактивність (взаємодія студент-комп'ютер, викладач-студент, викладач-група студентів-комп'ютер тощо),
- доцільність застосування під час вивчення конкретної теми.

Методичні вимоги до програмного засобу полягають у модернізації цілей та завдань математичних дисциплін, врахуванні особливостей математики та методів її дослідження.

Психологічні вимоги до засвоєння математики передбачають врахування вікових та індивідуальних особливостей студентів.

Ергономічність, тобто пристосованість до взаємодії зі студентами. У тому числі дотримання санітарно-гігієнічних норм.

Технічні вимоги: простота у користуванні, надійність, мобільність (за необхідності), стійкість до можливих некоректних дій користувача.

Серед причин, які призводять до неефективного застосування програмних засобів у процес навчання математики це:

- ігнорування викладачем дидактичних принципів навчання,
- використання традиційних методів та принципів навчання без врахування особливостей навчання із інформаційно-комунікаційними технологіями,
- недоцільний добір навчального матеріалу для наповнення ним програмного засобу.

Разом з тим, однією з проблем, яка виникає при впровадженні інформаційно-комунікаційних технологій у навчання, є вибір програмного забезпечення. І, насамперед, це питання вибору виду програмного забезпечення, наприклад, комерційного або вільного.

Різновиди програмного забезпечення можна узагальними та подати у вигляді схеми (рис. 1.2):

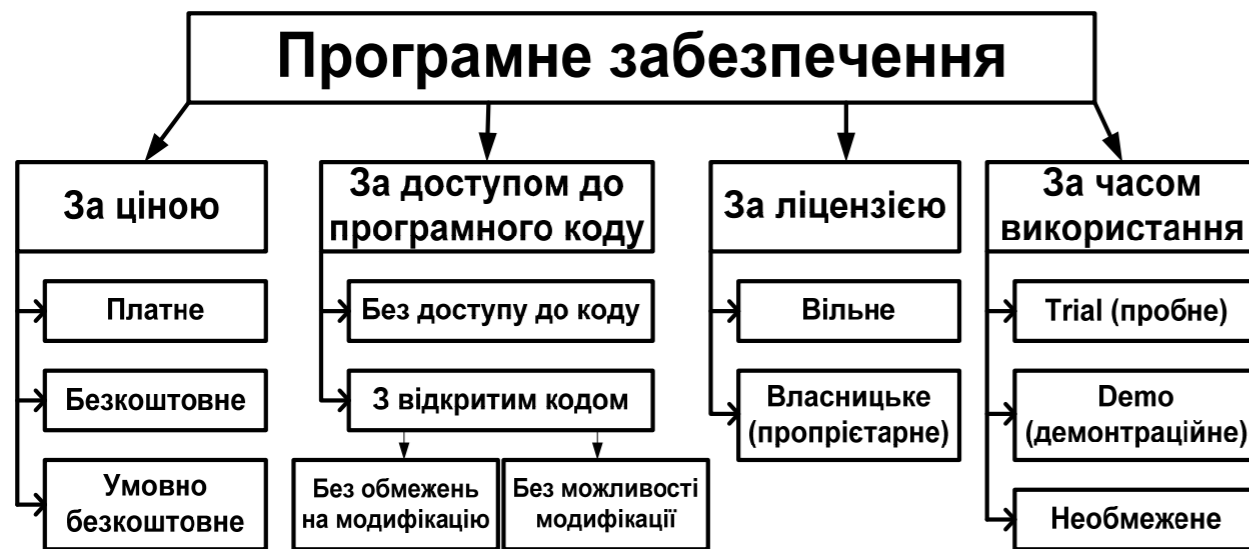


Рис. 1.2 – Класифікація програмного забезпечення

Платне – програмне забезпечення, обов’язковою умовою використання якого є його оплата. Зазвичай є власницьким (пропрієтарним) та охороняється авторським правом. Існує декілька способів оплати:

- початкова купівля програмного забезпечення з усіма функціональними можливостями – в даному випадку користувач отримує повністю функціональний програмний комплекс без обмежень в його використанні (операційні системи, пропрієтарні текстові, графічні редактори, тощо);
- оплата за періоди використання програмного забезпечення – користувач отримує безкоштовно демо- або trial-версію програмного забезпечення з обмеженими функціями, а після купівлі ліцензії або ключа на певний період використання (місяць, рік) отримує повний функціонал на вказаний період часу (антивіруси, службові програми).

Безкоштовне (Freeware) – програми без обмежень на (некомерційне) використання. Для користування таким програмним забезпеченням не потрібно його купувати. Охороняється законодавством. Важливо зазначити, що безкоштовність не означає, що вказане програмне забезпечення можна вільно поширювати. Ці програми можуть розповсюджуватися без програмного коду та обмежувати комерційне використання чи модифікацію програми, а при використанні їх в організаціях потрібно заплатити. В цей клас входить вільне програмне забезпечення, відкриті програми з вихідними текстами, також безкоштовним може бути і власницьке програмне забезпечення (ОС Linux, Браузери, GeoGebra, та ін.) Але в основному воно має обмежень по функціональності та періоду роботи.

Умовно безкоштовне (Shareware) – група платних програм з умовними безкоштовними періодами використання. Зазвичай потребують оплати або виконання інших дій для повнофункціонального використання. В даному випадку спочатку користувачеві пропонується безкоштовно протестувати та випробувати програму в дії, а потім, за його бажанням, оплатити її і користуватися повноцінно. Проте поняття «умовно безкоштовне» досить

1. Стандартизація (нормалізація) змінних.
2. Визначення кореляційної матриці.
3. Визначення критерію χ^2 (хі-квадрат).
4. Визначення оберненої матриці.
5. Обчислення F-критеріїв.
6. Обчислення часткових коефіцієнтів кореляції.
7. Обчислення t-критеріїв.

Здійснимо всі ці кроки із використанням можливостей табличного процесора Excel.

Нормалізація змінних передбачає додаткові обчислення. А саме, треба визначити середні значення та стандартні відхилення змінних x_1, x_2, x_3 .

Для цього використаємо функції СРЗНАЧ та СТАНДОТКЛОН Мастер функцій в категорії «Статистические» (таблиця 2.19).

Таблиця 2.19

Допоміжні розрахунки

	x_1	x_2	x_3
«СРЗНАЧ»	18,05	9,60	19,80
«СТАНДОТКЛОН»	4,26	3,97	3,04

Нормалізація (стандартизація) незалежних змінних

Для цього використаємо функцію НОРМАЛИЗАЦИЯ, категорія «Статистические» (таблиця 2.20).

Таблиця 2.20

Нормалізовані незалежні змінні

	X_1^*	X_2^*	X_3^*
	-1,41989	-1,15996	-1,58071
	-0,24643	-0,65563	-0,59276
	-1,1852	-0,9078	-1,25139
	-0,9505	-0,65563	-0,92208
	-0,48112	-0,9078	0,065863
	-0,71581	-1,15996	-1,58071
	-0,9505	-0,9078	-1,25139
	-0,24643	-0,1513	-0,59276
	-1,41989	-1,15996	-0,26345
X^*	-0,01173	0,100866	0,065863
	0,45765	0,353032	0,72449
	-0,24643	0,100866	0,395176
	-0,01173	0,605198	0,395176
	0,222958	-0,40347	0,065863
	0,45765	-0,1513	0,72449
	0,927034	1,10953	1,053804
	1,396419	1,361696	1,383117
	0,692342	0,857364	0,065863
	1,631111	1,613862	1,383117
	2,100496	2,118194	1,712431

Таким образом, отримали економетричну модель:

$$\hat{y} = 15,46 - 1,442x_1 + 2,791x_2.$$

За рис. 2.164 також можна визначити, що коефіцієнт детермінації дорівнює $R^2 = 0,673621$; F-статистика = 22,7, ступенів вільності – 22 і т.д.

2.7.5. Дослідження мультиколінеарності (за алгоритмом Фаррара-Глобера)

Приклад. У таблиці 2.18 задані дані з діяльності фірми. Прибуток фірми (y , у.г.о.) та фактори, від яких він залежить - фондовіддача (x_1 , у.г.о.), продуктивність праці (x_2 , у.г.о.), питомі інвестиції (x_3 , у.г.о.).

Таблиця 2.18

Вихідні дані

№ місяця	Прибуток фірми y , у.г.о.	Фондовіддача x_1 , у.г.о.	Продуктивність праці x_2 , у.г.о.	Питомі інвестиції x_3 , у.г.о.
1	40	12	5	15
2	45	17	7	18
3	40	13	6	16
4	43	14	7	17
5	48	16	6	20
6	39	15	5	15
7	42	14	6	16
8	45	17	9	18
9	38	12	5	19
10	48	18	10	20
11	50	20	11	22
12	48	17	10	21
13	49	18	12	21
14	45	19	8	20
15	49	20	9	22
16	52	22	14	23
17	54	24	15	24
18	51	21	13	20
19	55	25	16	24
20	56	27	18	25

Завдання: 1) перевірити масив незалежних змінних на наявність мультиколінеарності за алгоритмом Фаррара-Глобера;

2) при наявності мультиколінеарності змінних знайти істинні оцінки параметрів моделі;

3) побудувати модель, позбавлену від колінеарності.

Розв'язання.

Перевірка масиву даних на мультиколінеарність за алгоритмом Фаррара-Глобера включає в себе наступні кроки:

широке, і може варіюватись від звичайного нагадування про необхідність оплати за програмне забезпечення під час кожного запуску до обмеження періоду роботи і відключення в безкоштовній, пробній версії деяких функцій, що робить дуже незручним використання програми за прямим її призначенням. Сьогодні під shareware розуміють саме метод поширення пропрієтарного програмного забезпечення на ринку, при якому користувачеві пропонується версія комерційного продукту обмежена за можливостями, терміну дії або заборону на використання в цілях, відмінних від ознайомлювальних.

До цього класу програм входить:

- Demo, demoware - демонстраційне програмне забезпечення. Програми, які служать для демонстрації можливостей цього програмного забезпечення. Це "скорочені" версії комерційних програм, які дозволяється завантажувати для того, щоб протестувати програму в роботі. В цьому режимі, зазвичай, не працює деяка частина функцій програми. Наприклад, у текстовому редакторі, може надаватися тільки частина інструментів обробки тексту, або антивірусна програма в демонстраційній версії зможе тільки просканувати комп'ютер на наявність вірусів а знешкодити їх можна буде тільки у платній повнофункціональній версії. Також, іноді програмісти відключають функцію збереження результатів роботи, тобто користувач може лише ознайомитись з функціями програми, а зберегти результати роботи в ній не зможе.

Варіанти обмеження функціональності в Demoware:

- надання частини функцій програми;
- в демо-режимі, надаються тільки ті функції, які вже давно присутні на ринку, а разом з покупкою ліцензії додаються раніше відключені інноваційні функції (наприклад, запис даних на CD/DVD - є в демо режимі, а робота з Blu Ray дисками - тільки при покупці ліцензії);
- відключені функції збереження результатів роботи;
- додавання в збережений файл позначень того, що програма використовувалася в демо-режимі (наприклад, в фото або у відео, може додаватися копірайт авторів програми).

- Trial, trialware - вид ліцензії на ПЗ, яка також, як і Demoware використовується для демонстрації роботи платних програм. Відмінність полягає в тому, що в таких програмах обмежується не функціонал, а час використання. Розробник програми дає користувачеві можливість користуватися всіма функціями програми обмежений час. Пробний період може бути самим різним, і обчислюватися по різному. Дуже поширеним терміном пробного користування - є період в 30 днів. Також часто зустрічаються програми, що працюють в trial - режимі 14 днів. Однак зустрічаються й інші строки використання програм: 7 днів, 60 днів, 90 днів... Крім того, іноді розробники відраховують пробний період не за кількістю днів, а за кількістю запусків. Тобто ви можете запустити програму тільки, наприклад, 20 разів, далі вона працювати не буде.

Програмне забезпечення без доступу до програмного коду – користувач має можливість повного використання функціональних можливостей програми,

але при цьому не має доступу до початкового коду. Зазвичай це комерційне (власницьке, пропріетарне) програмне забезпечення.

Програмне забезпечення з відкритим кодом – користувач має доступ до програмного коду. Проте може накладатися обмеження на модифікацію та використання в комерційних цілях. В основному це безкоштовне, вільне програмне забезпечення. Цей клас програм поділяється на:

– програми з відкритим кодом для ознайомлення, з заборонаю його модифікації;

– програми з відкритим кодом без обмежень на зміну (модифікацію).

Вільне програмне забезпечення (англ. free software, software libre, чи libre software) — програмне забезпечення, яке надає користувачу ряд свобод:

– запускати програму («свобода 1»);

– вивчати й змінювати її початковий код відповідно до власних потреб («свобода 2»);

– вільно розповсюджувати копії програми («свобода 3»);

– розповсюджувати модифіковані версії програми («свобода 4»).

Надає максимальну кількість прав користувачам. Для цього типу ПЗ створюються спеціальні ліцензії для врегулювання прав та обов'язків авторів та користувачів.

Найпоширеніші:

GNU GPL (General Public License)- універсальна загальна ліцензія, що використовується для більшості програм "вільного ПЗ". Згідно умов, всі програмні продукти, отримані в результаті переробки чи модернізації проширюваного програмного коду, можуть передаватися далі лише на правах так само GNU GPL.

Free BSD - програмні продукти, отримані в результаті модифікацій початкового коду можуть поширюватися на будь-яких (в т.ч. платних) умовах.

Mozilla Public License (MPL) — одна з ліцензій на вільне програмне забезпечення. MPL містить в собі риси модифікованої ліцензії BSD і GNU General Public License. MPL використовується як ліцензії для Mozilla Suite, Mozilla Firefox, Mozilla Thunderbird та інших програм, розроблених Mozilla.

Apache License – ліцензія на вільне програмне забезпечення Apache Software Foundation. Подібно до всякої іншій ліцензії на вільне програмне забезпечення, ліцензія Апах дає користувачеві право використовувати програмне забезпечення для будь-яких цілей, вільно поширювати, змінювати, і поширювати змінені копії.

Різниця між безкоштовним та вільним ПЗ полягає в тому, що по ліцензії Freeware користувачі не мають права поширювати програми, дарувати, модифікувати і т.д., в той час як у Free software можливо. Часто (хоча не завжди) вони відрізняються ще й тим, що "вільне ПЗ" надається разом із сирцевими кодами.

Власницьке – пропріетарне, програмне забезпечення, яке продається за гроші й захищено різними законами. Це вид ліцензування, при якому головне метою поширення програми є отримання прибутку. Програми з такою

- *Известные значения X* – діапазон, який містить дані, що описують всі незалежні змінні;
- *Константа* – логічне значення, яке вказує на наявність або відсутність вільного члена у рівнянні; якщо *Константа* = 1, то вільний член розраховується звичайним чином, якщо *Константа* = 0, то вільний член = 0;
- *Статистика* – логічне значення, яке вказує, чи виводити додаткову інформацію регресійного аналізу (так чи ні). Якщо *Статистика* = 1, то додаткова інформація виводиться, якщо *Статистика* = 0, то виводяться тільки оцінки параметрів рівняння. Клацніть ОК;
- у лівій верхній комірці виділеної області з'явиться перший елемент результуючого масиву даних; для того, щоб вивести всі дані, клацніть <F2>, а потім – комбінацію клавиш <Ctrl>+<Shift>+<Enter>.

Додаткова статистика буде виводитись у такому вигляді (таблиця 2.17).

Таблиця 2.17

Додаткова статистика

Значення b_p	Значення b_{p-1}	...	Значення b_2	Значення b_1	Значення b_0
Стандартна помилка оцінки $b_p (s_{b_p})$	Стандартна помилка оцінки $b_{p-1} (s_{b_{p-1}})$...	Стандартна помилка оцінки $b_2 (s_{b_2})$	Стандартна помилка оцінки $b_1 (s_{b_1})$	Стандартна помилка оцінки $b_0 (s_{b_0})$
Коефіцієнт детермінації R^2	Стандартна помилка моделі (залишків) S_e	#Н/Д	#Н/Д	#Н/Д	#Н/Д
F-статистика	Кількість ступенів вільності $n - (p + 1)$	#Н/Д	#Н/Д	#Н/Д	#Н/Д
Регресійна сума квадратів	Залишкова сума квадратів	#Н/Д	#Н/Д	#Н/Д	#Н/Д

Функція ЛИНЕЙН також може бути використана для розрахунку оцінок параметрів моделей, які за допомогою перетворень зводяться до лінійного вигляду.

Для нашого прикладу результат застосування функції ЛИНЕЙН представлений на рис. 2.164:

	D	E	F	G	H	I
1						
2			2.791441	-1.44174	15.46076	
3			1.313674	0.25651	3.658697	
4			0.673621	4.462271	#Н/Д	
5			22.70316	22	#Н/Д	
6			904.1246	438.061	#Н/Д	
7						
8						


Рис. 2.164 – Результат застосування функції ЛИНЕЙН

У діалоговому вікні цієї функції необхідно ввести ймовірність на рівні значущості (0,05 або 0,01) та кількість ступенів вільності.

2.7.4. Використання стандартної програми «линейн»

Вбудована статистична функція ЛИНЕЙН визначає коефіцієнти лінійної регресії.

Порядок виконання дій:

- 1) введіть вихідні дані;
- 2) виділіть область порожніх комірок, яка складається з 5 рядків та (p + 1) стовпців (де p – кількість незалежних змінних) для виведення результатів регресійної статистики;
- 3) активізуйте Мастер функций будь-яким способом:
 - a) у головному меню виберіть Вставка/Функция;
 - b) на панелі інструментів Стандартная клацніть на Вставка функции 
- 4) у вікні, що відкрилось, виберіть Категорію Статистические, Функцію – ЛИНЕЙН. Клацніть ОК;
- 5) заповніть аргументи функції (рис. 2.163):

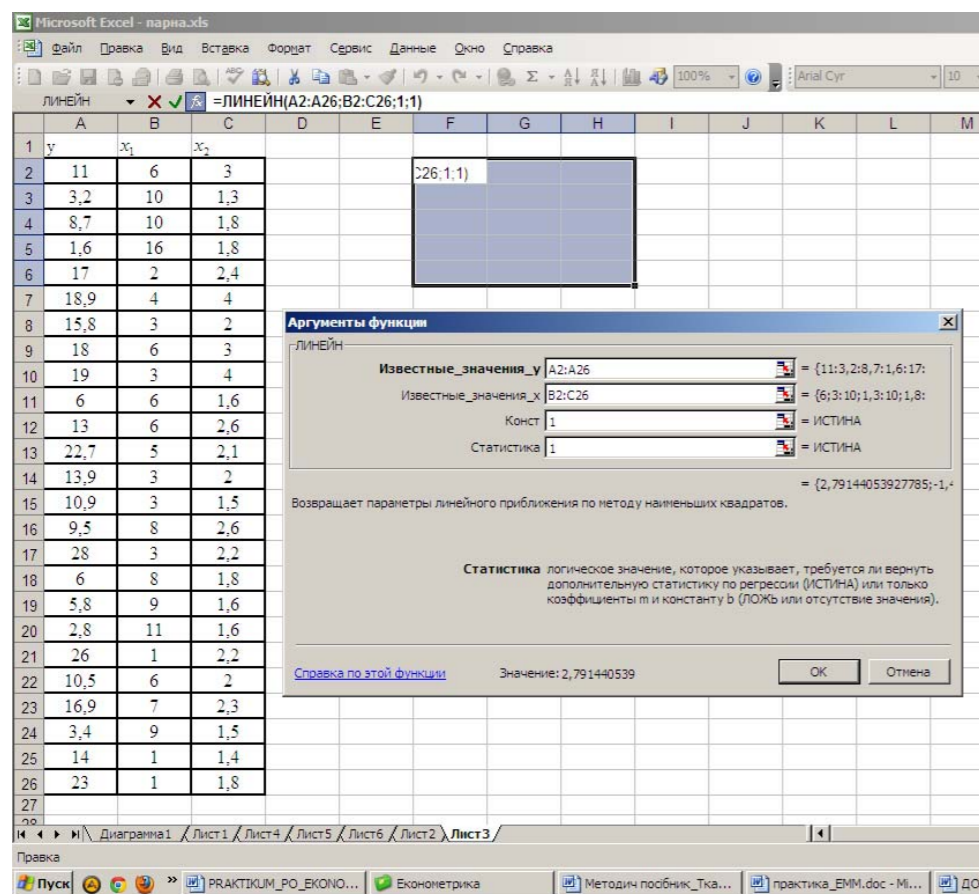


Рис. 2.163 – Введення даних

- *Известные значения Y* – діапазон, який містить дані, що характеризують результуючу ознаку;

ліцензією часто відрізняються тим, що розробник не випускає навіть пробні версії. Тобто користуватися такою програмою можна тільки після покупки ліцензії. Розробники комерційного програмного забезпечення часто займаються написанням програм під замовлення.

Це програмне забезпечення, на яке зберігаються як немайнові, так і майнові авторські права. Отримавши або придбавши таке програмне забезпечення, користувач отримує обмежені права користування ним: може бути заборонено або закрито доступ до коду (вивчення), внесення змін, тиражування, розповсюдження та перепродаж. Програмне забезпечення вважається власницьким, якщо наявне хоча б одне з перелічених обмежень. Власницьке програмне забезпечення та комерційне програмне забезпечення не є синонімами — власницьким може бути й безплатне (тобто, некомерційне) програмне забезпечення.

Відповідно до запропонованої класифікації визначимо типи програмного забезпечення, що висвітлені в монографії:

SciLab - Ліцензія Licence CeCILL (сумісна з GPL). Безкоштовне, власницьке програмне забезпечення, з підтримкою мов програмування (C, C++, Fortran) та можливістю за допомогою них додавання нових функцій. Заборонено подальше комерційне використання (продаж). Scilab був спроектований так, щоб бути відкритою системою, де користувачі можуть додавати свої типи даних і операції над цими даними шляхом перевантаження.

MatLab – платне, власницьке програмне забезпечення, без доступу до програмного коду. Комерційне використання заборонено.

MathCad – платне, власницьке програмне забезпечення, без доступу до програмного коду. Комерційне використання заборонено.

GeoGebra – Ліцензія GeoGebra Non-Commercial License Agreement і частково GPL. Безкоштовне, власницьке, без доступу до програмного коду. Без можливості комерційного використання.

Open Office Calc - Ліцензія Apache License. Безкоштовне, вільне програмне забезпечення, без доступу до програмного коду.

Microsoft Excel - Ліцензія Microsoft EULA. власницьке, платне, без доступу до коду та можливості комерційного використання.

Gran 1, Gran 2D, 3D – безкоштовне, власницьке, з доступом до програмного коду та можливістю його модифікації. Без можливості комерційного використання.

R-програмування - розповсюджується безкоштовно за ліцензією GNU General Public License у вигляді вільнодоступного вихідного коду або відкомпільованих бінарних версій більшості операційних систем: Linux, FreeBSD, Microsoft Windows, Mac OS X, Solaris.

Розглянемо більш детально перераховані вище програмні засоби.

Одним з найперших вітчизняних засобів візуалізації математичної задачі та її розв'язку, що робить діалог студента та викладача більш доступним та евристичним, є педагогічний програмний засіб GRAN, розробка якого розпочалася у 1989 році авторським колективом під керівництвом М.І. Жалдака.

До складу ППЗ GRAN входять педагогічні програмні засоби:

- GRAN1, що призначена для графічного аналізу функцій, звідки походить і її назва GRAphic ANalysis;
- GRAN-2D – для комп'ютерної підтримки навчання планіметрії;
- GRAN-3D – для комп'ютерної підтримки навчання стереометрії.

За допомогою GRAN1 можна розв'язувати наступні класи задач [27]:

- побудова графіків функцій, обчислення значення виразів;
- графічне розв'язування рівнянь та систем рівнянь;
- графічне розв'язування нерівностей та систем нерівностей;
- відшукування найбільших та найменших значень функції на заданій множині точок;
- побудова січних та дотичних до графіків;
- обчислення визначених інтегралів, площ довільних фігур, довжини дуги кривої, об'ємів та площ поверхонь обертання;
- елементи статистичного аналізу експериментальних даних.

GRAN 2D відноситься до розряду програм динамічної геометрії та призначений для графічного аналізу геометричних об'єктів на площині, звідки і походить назва (GRAphic ANalysis 2-Dimension).

Використання пакету GRAN-2D надає можливість:

- створювати динамічні моделі геометричних фігур та їхніх комбінацій аналогічно класичним побудовам за допомогою циркуля та лінійки, а також використовуючи елементи аналітичної геометрії (систему координат, рівняння прямих і кіл, алгебраїчні залежності між частинами побудови, графіки функцій тощо);
- проводити вимірювання геометричних величин, досліджувати геометричні місця точок;
- аналізувати динамічні вирази, висувати припущення, встановлювати закономірності;
- будувати графічні зображення, використовуючи коментарі, кнопки, підказки та гіперпосилання;
- експортувати рисунки у графічні формати для вбудовування їх у інші додатки і для створення геометричних ілюстрацій тощо.

Для графічного аналізу тривимірних об'єктів призначений пакет GRAN-3D (GRAphic ANalysis 3-Dimension). Використання пакету GRAN-3D надає можливість:

- створювати та перетворювати моделі базових просторових об'єктів;
- виконувати перерізи многогранників площинами;
- обчислювати об'єми та площі поверхонь многогранників і тіл обертання;
- вимірювати відстані та кути [29].

R – потужна мова і безкоштовне середовище програмування, що найчастіше застосовується для статистичних обчислень, аналізу даних та візуалізації (представлення даних у графічному вигляді). Створена Росом Іхаком (Ross Ihaka) та Робертом Джентлменом (Robert Gentleman), працівниками Оклендського Університету в Новій Зеландії. Це безкоштовний

Аналогічно знайдемо обернену матрицю $(X'X)^{-1}$ та добутки матриць $(X'Y)$, $(X'X)^{-1}(X'Y)$ (за допомогою функції МОБР та функції МУМНОЖ), рис. 2.161:

	A	B	C	D	E	F	G
28							
29							
30							
31			25	147	54,1		
32		XX=	147	1189	302		
33			54,1	301,6	129		
34							
35			0,672	-0,03	-0,21		
36		(XX)обр	-0,029	0,003	0		
37			-0,213	0,004	0,09		
38							
39			325,6				
40		X'Y	1400				
41			763				
42							
43			15,46				
44		A=	-1,442				
45			2,791				
46							
47							

Рис. 2.161 – Результат добутку відповідних матриць

Таким чином, отримали економетричну модель:

$$\hat{y} = 15,46 - 1,442x_1 + 2,791x_2.$$

Далі, використовуючи відомі вже нам можливості табличного процесора Excel, необхідно обчислити відповідні показники, за допомогою яких здійснюється аналіз ступеня адекватності побудованої моделі.

Табличні значення критеріїв можна знайти у таблицях критичних значень (див. додатки) або можна знайти в Excel. Наприклад, знайти критичні значення для t-критерія Стьюдента можна звернувшись до Мастера функцій, категорія Статистические, функція СТЬЮДРАСПОБР (рис. 2.162).

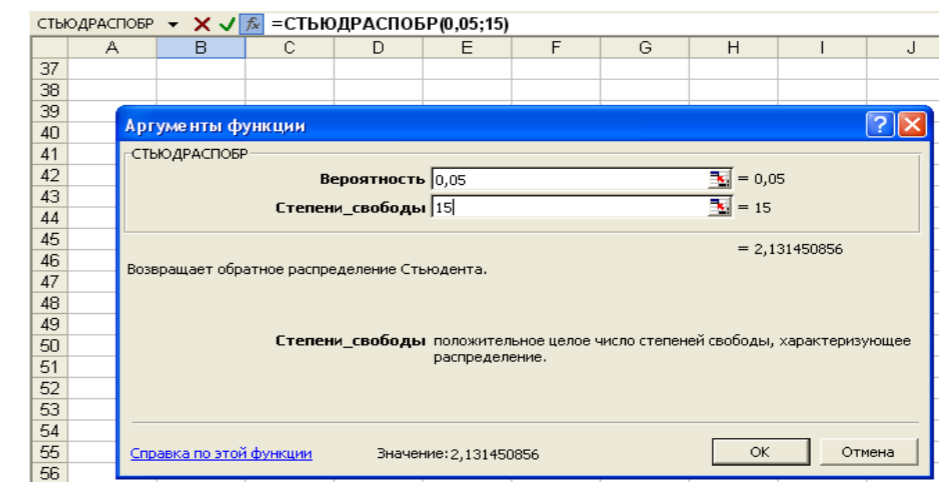


Рис. 2.162 – Вікно знаходження табличного значення t-критерія Стьюдента

- 4) у рядку Массив вкажіть діапазон комірок, у яких міститься матриця X. Клацніть ОК.
 - 5) у лівій верхній комірці виділеної області з'явиться перший елемент результуючого масиву даних; для того, щоб вивести всі дані, клацніть <F2>, а потім – комбінацію клавиш <Ctrl>+<Shift>+<Enter>.
- Результат представлено на рис. 2.159.

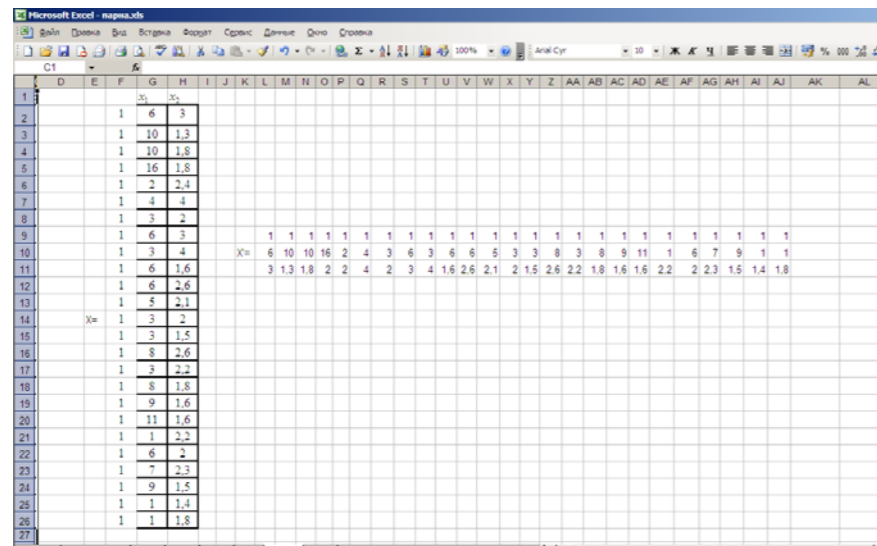


Рис. 2.159 – Результат транспонування масиву даних

Добуток матриць ($X'X$) знаходимо за допомогою Мастера функцій, використовуючи Категорію Статистические, функцію МУМНОЖ:

- 1) виділіть область порожніх комірок, яка складається з (p+1) рядків та (p+1) стовпців для виведення результату;
- 2) у вікні МУМНОЖ у рядку Массив 1 вкажіть діапазон комірок, у яких міститься матриця X' (перший співмножник), а у рядку Массив 2 – матриця X (другий співмножник). Клацніть ОК;
- 3) у лівій верхній комірці виділеної області з'явиться перший елемент результуючого масиву даних; для того, щоб вивести всі дані, клацніть <F2>, а потім – комбінацію клавиш <Ctrl>+<Shift>+<Enter>.

Результат множення матриць представлений на рис. 2.160.

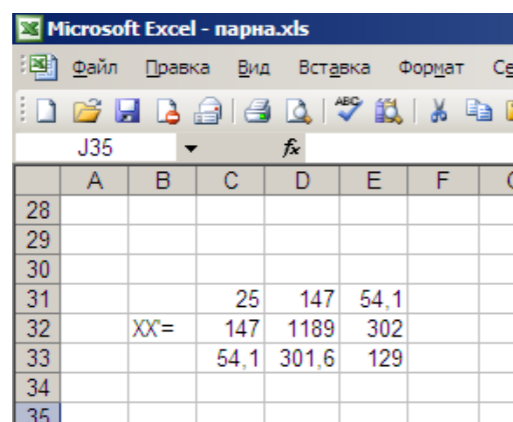


Рис. 2.160 – Результат множення матриць

проект, що розповсюджується за ліцензією GNU General Public License у вигляді вільнодоступного вихідного коду або відкомпільованих бінарних версій більшості операційних систем: Linux, FreeBSD, Microsoft Windows, Mac OS X, Solaris. R схожий на комерційні мову і середовище S, яка була розроблена в Bell Laboratories (раніше AT & T, зараз Lucent Technologies) Джоном Чемберсом (John Chambers) і колегами. R можна розглядати як «діалект» мови S, що широко використовується в якості навчальної мови та дослідницького інструменту. Основними перевагами проекту R є той факт, що R є безкоштовним і що є багато допоміжної інформації, доступної в Інтернеті. Цей проект дуже схожий на інші програмні пакети, такі як MatLab (не безкоштовний), але він більш зручний для користувача, ніж мови програмування, такі як C++ або Fortran. Ви можете використовувати R таким, яким він є, але в освітніх цілях ми надаємо перевагу використовувати R в поєднанні з інтерфейсом RStudio (також безкоштовним), який має зручні шари і кілька додаткових опцій.

Особливості мови R:

- синтаксис дуже схожий на S, що робить його легким для переходу користувачів S-PLUS,
- семантика зовні схожа на S, але насправді зовсім різна,
- працює практично на будь-якій стандартній обчислювальній платформі/ОС (навіть на PlayStation 3),
- часті релізи (річний + релізи з виправленням помилок); активний розвиток,
- досить невелике за розмірами програмне забезпечення,
- функціональність ділиться на модульні пакети,
- графічні можливості дуже розвинуті і краще, ніж у більшості статистичних пакетів,
- зручний для інтерактивної роботи, але також містить і потужну мову програмування для розробки нових інструментів (користувач -> програміст),
- дуже активне і енергійне співтовариство користувачів; R-help і R-Devel списки розсилки і Stack Overflow.

Недоліки мови R:

- по суті базується на основі 40-річної технології.
- невелика вбудована підтримка динамічної або 3-D графіки (але все значно покращилося з так званих «старих днів»).
- функціональність заснована на споживчому попиті і власних внесків користувачів. Якщо ніхто не відчуває потреби в реалізації вашого улюбленого методу, то це ваша робота!
- об'єкти мають бути збережені, як правило, у фізичній пам'яті.
- не ідеально підходить для всіх можливих ситуацій (але це недолік всіх програмних пакетів).

Scilab - система комп'ютерної математики, яка призначена для виконання інженерних і наукових обчислень, таких як: рішення нелінійних рівнянь і систем, рішення задач лінійної алгебри, рішення задач оптимізації, диференціювання та інтегрування, обробка експериментальних даних

(інтерполяція та апроксимація, метод найменших квадратів), рішення звичайних диференціальних рівнянь і систем. Крім того, Scilab надає широкі можливості по створенню і редагуванню різних видів графіків і поверхонь. По інтерфейсу пакет Scilab близький до Matlab. Для вирішення нестандартних завдань в Scilab є досить потужна мова програмування високого рівня.

GeoGebra — безкоштовна програма, що надає можливість створення динамічних («живих») креслень для використання на різних рівнях навчання геометрії, алгебри, планіметрії та інших суміжних дисциплін. Програма володіє багатьма можливостями роботи з функціями (побудова графіків, обчислення коренів, екстремумів, інтегралів і т. д.).

На відміну від інших програм для динамічного маніпулювання геометричними об'єктами, ідея GeoGebra полягає в інтерактивному поєднанні геометричного, алгебраїчного і числового подання. Ви можете створювати конструкції з точками, векторами, лініями, кінчними перерізами, а також математичними функціями, а потім динамічно змінювати їх.

Mathcad — система комп'ютерної алгебри з класу систем автоматизованого проектування, орієнтована на підготовку інтерактивних документів з обчисленнями і візуальним супроводженням, відрізняється легкістю використання і застосування для колективної роботи.

Mathcad має простий і інтуїтивний для використання інтерфейс користувача. Для введення формул і даних можна використовувати як клавіатуру, так і спеціальні панелі інструментів. [вікіпедія].

Незважаючи на те, що ця програма в основному орієнтована на користувачів-непрограмістів, Mathcad також використовується в складних проектах, щоб візуалізувати результати математичного моделювання, шляхом використання розподілених обчислень і традиційних мов програмування. Також Mathcad часто використовується у великих інженерних проектах, де велике значення має трасируемого і відповідність стандартам.

Mathcad досить зручно використовувати для навчання, обчислень і інженерних розрахунків. Відкрита архітектура програми в поєднанні з підтримкою технологій .NET і XML дозволяють легко інтегрувати Mathcad практично в будь-яку ІТ-структуру і інженерні програми. Є можливість створення електронних книг (e-Book).

Серед можливостей Mathcad можна виділити:

- рішення диференціальних рівнянь, в тому числі і чисельними методами,
- побудова двовимірних і тривимірних графіків функцій (в різних системах координат, контурні, векторні і т. д.),
- використання грецького алфавіту як в рівняннях, так і в тексті,
- виконання обчислень в символному режимі,
- виконання операцій з векторами і матрицями,
- символний розв'язок систем рівнянь,
- апроксимація кривих,
- виконання підпрограм,
- пошук коренів многочленів і функцій,

2.7.3. Побудова багатofакторної лінійної економетричної моделі на основі матричного оператора Імнк

Матричний оператор ІМНК має вигляд:

$$A = (X' X)^{-1} X' Y,$$

де

$$X = \begin{pmatrix} 1 & 6 & 3 \\ 1 & 10 & 1,3 \\ 1 & 10 & 1,8 \\ 1 & 16 & 1,8 \\ 1 & 2 & 2,4 \\ 1 & 4 & 4 \\ 1 & 3 & 2 \\ 1 & 6 & 3 \\ 1 & 3 & 4 \\ 1 & 6 & 1,6 \\ 1 & 6 & 2,6 \\ 1 & 5 & 2,1 \\ 1 & 3 & 2 \\ 1 & 3 & 1,5 \\ 1 & 8 & 2,6 \\ 1 & 3 & 2,2 \\ 1 & 8 & 1,8 \\ 1 & 9 & 1,6 \\ 1 & 11 & 2,2 \\ 1 & 1 & 2 \\ 1 & 6 & 2,3 \\ 1 & 7 & 2 \\ 1 & 9 & 1,5 \\ 1 & 1 & 1,4 \\ 1 & 1 & 1,8 \end{pmatrix} \quad Y = \begin{pmatrix} 11 \\ 3,2 \\ 8,7 \\ 1,6 \\ 17 \\ 18,9 \\ 15,8 \\ 18 \\ 19 \\ 6 \\ 13 \\ 22,7 \\ 13,9 \\ 10,9 \\ 9,5 \\ 28 \\ 6 \\ 5,8 \\ 2,8 \\ 26 \\ 10,5 \\ 16,9 \\ 3,4 \\ 14 \\ 23 \end{pmatrix},$$

X' - транспонована матриця X .

Для транспонування матриці X виконаємо наступні дії:

- 1) виділіть область порожніх комірок, яка складається з (p+1) рядків та n стовпців, для виведення результату (p – кількість незалежних змінних), n – кількість спостережень;
- 2) активізуйте Мастер функцій одним із можливих способів:
 - а) у головному меню виберіть Вставка/Функція;
 - б) на панелі інструментів Стандартная клацніть на Вставка функції;
- 3) у вікні виберіть Категорію Ссылки и массивы, Функцію – ТРАНСП (рис. 2.158) клацніть на кнопці ОК;

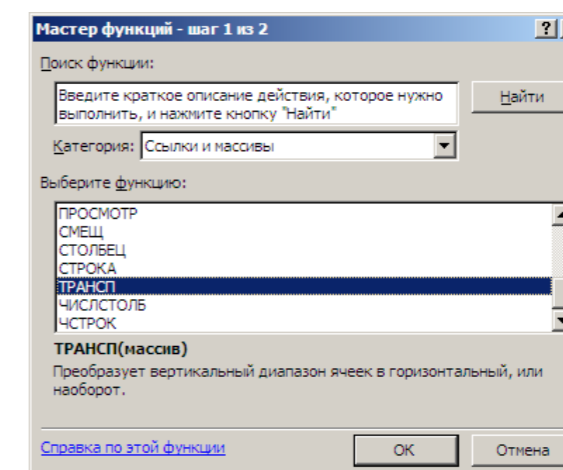


Рис. 2.158 – Вікно Мастер функции при виборі функції ТРАНСП

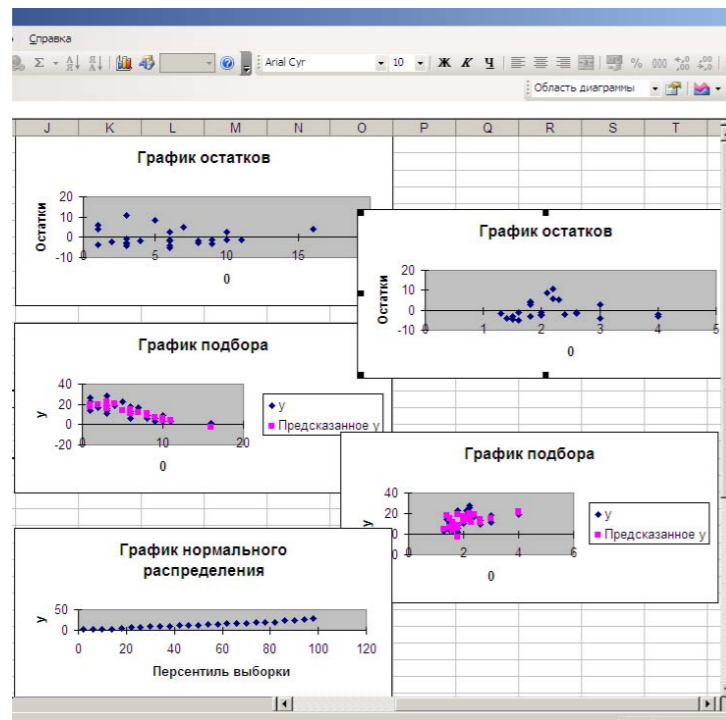


Рис. 2.156 - Приклад зображення виводу регресійного аналізу

На рис. 2.156 представлені графіки «остатков», «підбора» і «нормального распределения», тобто функція Анализ данных/Регрессия дозволяє вивести графіки залежності результативної ознаки від кожного з факторів, а також графіки залишків для парних залежностей. Для виводу цих графіків необхідно у вікні Регрессия встановити відповідні позначки (дивись рис. 2.154).

У головному меню виберіть Файл/Сохранить как. У діалоговому вікні Сохранение документа, що з'явилося (рис. 2.157), вкажіть ім'я файлу (zadanie№.....), вкажіть в якій папці буде зберігатись файл (приміром, STATISTICA), після чого натисніть Сохранить.

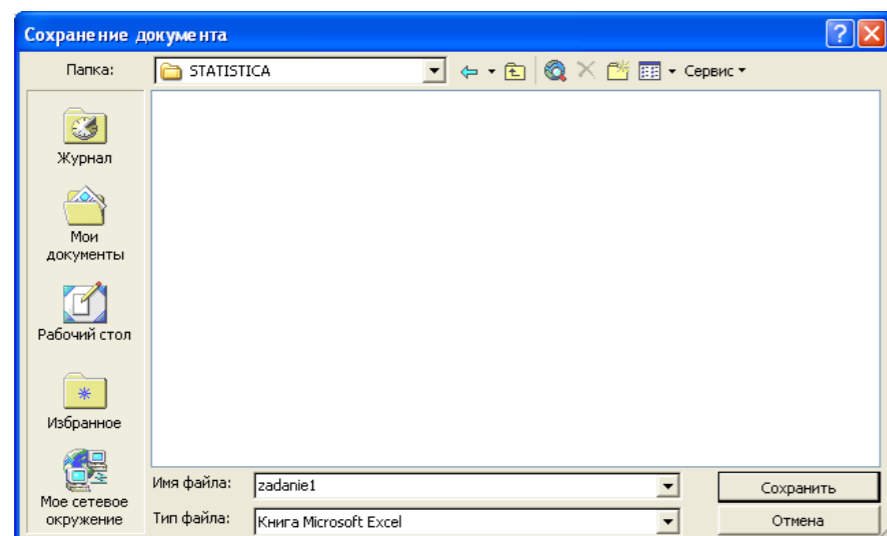


Рис. 2.157 – Вікно Сохранение документа

- проведення статистичних розрахунків і робота з розподілом ймовірностей,
- пошук власних чисел і векторів,
- обчислення з одиницями виміру,
- інтеграція з САПР системами, використання результатів обчислень в якості керуючих параметрів,
- за допомогою Mathcad інженери можуть документувати всі обчислення в процесі їх проведення.

MATLAB – програмний засіб багатоцільового призначення, що постійно удосконалюється. Серед *MATLAB* містить інтерпретатор команд мовою високого рівня, графічну систему, пакети розширень та реалізована на мові C. Користувач вводить команди або запускає на виконання файли з текстами мовою *MATLAB*. Програма опрацьовує введені дані та виводить результат.

Microsoft Excel – це програмний засіб для роботи з електронними таблицями, який містить зручний апарат у вигляді великого набору функцій для обробки великих масивів даних, для опрацювання тексту, створення ділової графіки, роботи з базами даних тощо.

Ключовими перевагами табличного редактора MS Excel є можливість швидкої обробки масивів даних і виведення результату у зручному вигляді, автокореляція формул та перегляд макету сторінки. Крім того, за допомогою табличного редактора MS Excel можна відкривати з URL, працювати з документами HTML, графічно обробляти інформацію, інтегрувати отримані результати у пакети MS Office а також програмування в Excel.

OpenOffice Calc - електронні таблиці. За своїми можливостями OpenOffice Calc майже відповідає Microsoft Excel. Крім всіх можливостей Microsoft Excel зазначена програма містить додаткові функції, зокрема система, яка автоматично визначає послідовності графіків, побудованих на основі даних користувача, а також за у програмі OpenOffice Calc можна писати таблиці прямо в форматі PDF.

Одним з рішень, які знімають різноманітні проблеми у застосуванні ІКТ у навчанні математики у закладах вищої освіти, є комбіноване застосування програмного забезпечення. Будь-якому фахівцеві в ході практичної діяльності доводиться здійснювати операції над кількісними даними. Для більшості фахівців – «не математиків» найбільш важливим є вміння провести необхідні обчислення в ряді типових задач. Використання комп'ютера при проведенні розрахунків зміщує акценти в математичній підготовці фахівця. З появою спеціалізованих математичних програм виникає необхідність вводити у навчання математики та вміння користуватися цими програмами. Розгляд основних розділів курсу не в традиційному викладі, а з перспективою подальшого розгляду на комп'ютері.

Таким чином, сучасним фахівцям та викладачам необхідно володіти не тільки математичними методами обробки даних, а й відповідними програмними засобами.

Отже, реформа сучасної освіти, математичної зокрема, може бути здійснена у напрямку інтеграції ІКТ у навчальний процес всіх дисциплін вузу.

1.6. Інтеграція математики, інформаційно-комунікаційних технологій та фахових дисциплін

Глобалізаційні процеси у світі та в Україні, а також сучасні напрями розвитку вищої школи обумовили особливу значущість фундаменталізації в освіті. Це призводить до того, що заклади вищої освіти України повинні трансформуватись. У першу чергу треба змінити вимоги до підготовки студента. Майбутній фахівець повинен під час навчання у вищому навчальному закладі набути таких фундаментальних знань (компетентностей), які нададуть йому можливість професійно керувати підприємством (або фірмою), створювати та ефективно вести власний бізнес, приймати рішення в швидкоплинних умовах на основі цілісного сприйняття економічних явищ та процесів, бути мобільними тощо. Разом з тим, сучасний період розвитку суспільства характеризується чисельними інтеграційними процесами в економічній, інформаційній, культурній та інших сферах життя людини. Пожвавлення цих процесів повинно відбуватися і в освітній сфері.

Фундаменталізація є процесом, що спрямований на становлення цілісної наукової картини світ і тим самим сприяє істотному підвищенню якості освіти і освітнього рівня студентів шляхом відповідної зміни змісту дисциплін, що вивчаються, і методології навчального процесу [923].

В.П. Бех писав, що проблема фундаменталізації освіти взагалі і, в першу чергу, її вищої ланки, є дійсно проблемою, великою за обсягом, терміновою за часом, перспективною за наслідками, яка ще тільки формується [20].

У процесі навчання у вищому навчальному закладі майбутні фахівці вивчають досить широкий перелік навчальних дисциплін. Розширення масштабів та поглиблення наукового пізнання, що спостерігаються у навчальних програмах, супроводжується посиленням роз'єднаності і ослабленням зв'язків між дисциплінами. Це в свою чергу може призвести до зниження ефективності пізнавального процесу та якості підготовки студентів.

У той же час відбувається зміна парадигми освіти у бік фундаменталізації, підвищуються вимоги до рівня підготовки майбутніх фахівців, які висуваються державою.

Таким чином, у навчальному процесі повинен відбуватись процес фундаменталізації професійної підготовки майбутніх фахівців.

Одним із шляхів розв'язання даної проблеми є інтеграція фундаментальних (математика, інформатика) та фахових дисциплін. Мають бути визначені спільності у методичному та методологічному плані підготовки з кожної окремої дисципліни та розкриті цілісність знань та вмінь на основі інтегративного підходу. Не виняток і математична підготовка студентів різних спеціальностей.

Нині економіці України особливо необхідні висококваліфіковані фахівці різних галузі здатні працювати самостійно, творчо, активно, зацікавлено, креативно. Важливим у цьому плані є формування у студентів логічного, алгоритмічного та економічного мислення, а також прагнення до самоаналізу та аналізу діяльності інших.

Так, наприклад, b_0 :

для $\alpha = 0,05$ та $25-(2+1)=22$ ступені вільності табличне значення $t_\alpha = 2,0739$, тоді $S_{b_0} \cdot t_\alpha = 3,658697153 \cdot 2,0739 = 7,5877720256$;

$$15,46076159 - 7,5877720256 \leq b_0 \leq 15,46076159 + 7,5877720256;$$

$$7,873088135 \leq b_0 \leq 23,04843504.$$

Аналогічно:

$$-1,973711081 \leq b_1 \leq -0,909771543$$

$$0,067047603 \leq b_2 \leq 5,515833476.$$

Таким чином, з ймовірністю 95%, збільшення віку автомобіля на одну одиницю забезпечить зменшення ціни автомобіля в межах від 1,973711081 до 0,909771543. Аналогічно інтерпретуються решта довірчих інтервалів.

Якщо довірчий інтервал деякої оцінки параметру регресії включає нуль, то це свідчить про недостовірність цього параметру.

Наблюдение	Предсказанное y	Остатки	стандартные остатки	Перцентиль	y
1	15,18463533	-4,184635333	-0,979480836	2	1,6
2	4,672221168	-1,472221168	-0,344596913	6	2,8
3	6,067941438	2,632058562	0,616075408	10	3,2
4	-2,582506435	4,182506435	0,978982533	14	3,4
5	19,27673626	-2,27673626	-0,532906539	18	5,8
6	20,8595585	-1,959558497	-0,458666011	22	6
7	16,71841873	-0,918418731	-0,214970595	26	6
8	15,18463533	2,815364667	0,658981134	30	8,7
9	22,30129981	-3,301299809	-0,772722027	34	9,5
10	11,27661858	-5,276618578	-1,235076981	38	10,5
11	14,06805912	-1,068059118	-0,249996321	42	10,9
12	14,11408016	8,58591984	2,009671875	46	11
13	16,71841873	-2,818418731	-0,659695986	50	13
14	15,32269846	-4,422698461	-1,035203319	54	13,9
15	11,18457649	-1,684576493	-0,394302074	58	14
16	17,27670684	10,72329316	2,50995829	62	15,8
17	8,951424062	-2,951424062	-0,690828012	66	16,9
18	6,951394642	-1,151394642	-0,269502333	70	17
19	4,067912018	-1,267912018	-0,296775089	74	18
20	20,16018946	5,839810537	1,366901068	78	18,9
21	12,39319479	-1,893194794	-0,443132524	82	19
22	11,78888564	5,111114356	1,19633807	86	22,7
23	6,672250588	-3,272250588	-0,765922592	90	23
24	17,92703703	-3,927037031	-0,919185833	94	26
25	19,04361325	3,956386753	0,926055604	98	28

Рис. 2.155 - Приклад зображення виводу регресійного аналізу

На рис. 2.155 представлена інформація про залишки, а саме Вивід залишків («Вывод остатков»)

У цьому блоці результатів наводяться розрахункові значення залежної змінної та залишків, які визначаються як різниця між фактичними значеннями та розрахунковими.

- збільшення x_2 (об'єм двигуна автомобіля) на 1 дм^3 , а інші фактори незмінні, буде сприяти збільшенню y (ціни автомобіля) на 2,89 (тис. у.о.).

Порівнювати силу впливу окремих факторів на результуючу співставляючі величини коефіцієнтів не варто, оскільки ці коефіцієнти залежать від власних одиниць вимірювання. З метою виявлення найбільш впливових показників (факторів) треба перейти до стандартизованого (нормалізованого) рівняння, у якому одиницею вимірювання впливу всіх факторів є середнє квадратичне відхилення.

Третій стовпчик містить стандартні похибки оцінок параметрів моделі:

$$S_{b_0} = 3,658697153; \quad S_{b_1} = 0,256510285; \quad S_{b_2} = 1,313673913.$$

Ці показники показують, яка частка значення даної характеристики сформувалася під впливом випадкових факторів. Стандартні похибки використовуються для розрахунку t-критерію Стюдента, значення якого для різних оцінок представлені в четвертому стовпці.

Четвертий стовпчик – t-критерій:

$$t_0 = 4,225756039; \quad t_1 = -5,620593461; \quad t_2 = 2,124911297.$$

Якщо значення t-критерія більше 2–3, то можна зробити висновок про значущість даного параметра, який формується під впливом не випадкових величин. Для більш ґрунтовних висновків використаємо результати, які знаходяться у п'ятому стовпчику.

П'ятий стовпчик – рівень значущості – показник ймовірності випадкових значень параметрів регресії:

$$\alpha_0 = 0,000347624; \quad \alpha_1 = 0,000011894; \quad \alpha_2 = 0,045067814.$$

Якщо α_j менше за прийнятий нами рівень значущості (зазвичай 0,1; 0,05 або 0,01), то роблять висновок про не випадковість даного значення оцінки, тобто про те, що оцінки параметрів достовірні (статистично значущі). В іншому випадку гіпотеза про випадковість значень коефіцієнтів регресії приймається.

Для нашої задачі α_j менші за прийнятий нами рівень значущості $\alpha = 0,05$, то ми робимо висновок про те, що оцінки параметрів моделі достовірні.

У випадку, якщо виникає протиріччя, а саме F-критерій стверджує про достовірність моделі у цілому, а t-критерій – недостовірність окремих оцінок, зазвичай дослідження оцінок параметрів моделі продовжується на наявність мультиколінеарності.

Решта чотирьох стовпчиків з ймовірністю 95% визначають верхні та нижні границі оцінок параметрів моделі, тобто дозволяють здійснити інтервальне оцінювання параметрів (довірчі інтервали).

Інтервальне оцінювання параметрів b_j ($j=1,2$) виконується наступним чином:

$$b_j \pm S_{b_j} \cdot t_\alpha, \quad \text{або} \quad b_j - S_{b_j} \cdot t_\alpha \leq \beta_j \leq b_j + S_{b_j} \cdot t_\alpha.$$

Оскільки однією із фундаментальних дисциплін для студентів економічних та технічних спеціальностей вищого навчального закладу є вища математика, то передусім проблема інтеграції має розв'язуватись у процесі навчання вищої математики.

Про значущість проблеми інтеграції свідчить і те, що їй завжди приділялась увага видатними психологами, дидактами та методистами.

Проблемі інтеграції приділяли значну увагу Платон, Аристотель, Кант, Гегель, А. Ейнштейн, І.Г.Песталоцці, І.Ф.Герbart, А.Дістервег, Дж.Дьюї, В.Кіппатрик та інші. К. Ушинський вперше (у ХІХ ст.) теоретично обґрунтував окремі аспекти інтеграції навчального змісту на основі встановлення міжпредметних зв'язків. Пізніше (на початку ХХ ст.) видатні вітчизняні та закордонні науковці, так як М.Гейне, П.Янковський, О.Александров, О.Кулінич, Д.Скуратівський та інші пропагували ідею інтеграції змісту освіти та пропонували її реалізовувати у вигляді міжпредметних комплексів. З середини ХХ ст. І.Зверев, В.Федорова, Н.Борисенко, А.Усова, В.Максимової, В.Ільченко, М.Іванчук та інші розробляли концепції навчання на інтегративній основі. Їх дослідження засвідчили, що інтеграція відображає міжпредметні зв'язки у змісті й методах навчання як гуманітарних так і природничо-математичних дисциплін.

Сучасні погляди на застосування інтеграційного підходу у навчанні природничих, технічних та математичних дисциплін викладені у роботах В. Андрущенко, П.Атутова, В.Бевз, В. Безрукової, М.Берулави, О.Білик, О.Булейко, Л.Васіної, С.Гончаренка, К.Гуза, Р.Гуревича, О.Дятлової, І.Зязюна, І.Козловської, Д.Коломийця, С.Клепка, О.Левчук, Ю.Мальованого, Р.Пастушенко, В.Разумовського, Ж.Сарани, Я.Собка, Н.Талалуєвої, С.Тищенко, В.Фоменка, Т.Якимович та інших. Науковцями доведено, що здійснення інтеграційного підходу у навчанні забезпечить формування цілісних, комплексних знань та вмінь у студентів, що сприятиме усвідомленню майбутнім фахівцем, у галузі економіки зокрема, глибинних взаємозв'язків структурних компонентів аналізованого явища.

Дослідниками було запропоновано підходи до виділення об'єктивних факторів та умов інтеграції знань, дано опис її характеристик, наприклад, рівнів, типів, видів, форм тощо.

Разом з тим, всі вони називають інтеграцію провідною тенденцією розвитку суспільства, науки та освіти. Таким чином, важливим є виявлення умов успішного здійснення інтеграційних процесів у освіті, зокрема у математиці.

Відомі педагогіки тлумачили інтеграцію наступним чином.

К.Д. Ушинський підкреслював, що дуже важливо зводити знання у систему по мірі їх накопичення: «Голова, наповнена отрывочными знаниями, похожа на кладовую, в которой все в беспорядке и где сам хозяин ничего не отыщет; голова, где только система без знания, похожа на лавку, в которой на всех ящиках есть подписи, а в ящиках пусто» [77]. Відомий педагог обґрунтував зв'язки між навчальними предметами, він вважав, що тільки система, яка виходить із суті самих предметів, дає нам владу над нашими знаннями. Рівень

інтеграції знань у науці стає ознакою її зрілості, результатом внутрішніх закономірностей її розвитку.

Я.А. Коменський стверджував, що все що знаходиться у взаємозв'язку має передаватись у такому ж самому зв'язку. Також він висунув припущення про те, що найважливішою умовою формування у учнів (студентів) системи знань є встановлення зв'язку між предметами. [46]. Ідеї Я.А.Коменського були підтримані та розвинуті у педагогічних теоріях Дж. Локка, І.Г. Песталоцці, А. Дістервега, І.Ф. Гербарта.

Наприклад, у Дж. Локка об'єднуючим інтегративним стрижнем виступала загальна абстрактна ідея. І.Г. Песталоцці розглядав освіту як гармонійний розвиток у процесі навчання всіх інтегруючих сил людини. Він (І.Г. Песталоцці) дотримувався вимоги, що у своїй свідомості треба встановити такі взаємозв'язки між предметами, якими вони поєднуються у природі. При цьому «дело обучения должно состоять в том, чтобы с одной стороны разграничивать между собой предметы, а с другой – объединить в нашем сознании сходные и родственные [58].

Ж.Ж. Руссо писав: «когда имеешь настоящую склонность к наукам первое, что чувствуешь, предаваясь им, это их связь между собой, в силу которой они притягиваются, помогают друг другу и объясняют друг друга так, что ни одна из них не может обойтись без другой» [67].

А.С. Макаренко зазначав: «человек не воспитывается по частям» [1].

А.Дістерверг вважав, що кожен предмет вимагає для того, щоб його зрозуміли, розуміння всіх наук, або хоча б розумно використаних їх елементів [17].

У науково-педагогічній літературі існують різні тлумачення поняття «інтеграція».

У енциклопедичному словнику [79] інтеграція (лат. integratio – відтворення, заповнення, від integer – цілий) – це стан пов'язаності окремих диференційованих частин та функцій системи у ціле, а також процес, який веде до такого стану; процес або дія, який має своїм результатом цілісність, об'єднання, з'єднання, відтворення єдності.

Німецький дослідник Павельциг Г. тлумачив інтеграцію як процес руху та розвитку визначеної системи, у якої кількість та інтенсивність її елементів зростає, посилюється їх взаємозв'язок та зменшується відносна самостійність [57].

В.С.Безрукова зазначала, що педагогічна інтеграція являє собою одну із форм взаємозв'язку та взаємодії предметів та явищ в теорії та практиці, відтворення їх природної цілісності [4].

М.М.Берулава окреслює інтеграцію як засіб досягнення єдності знань у всіх формах та типах його представлення (змістовому, структурному, логіко-гносеологічному, науково-організаційному, лінгвістично-семіотичному та ін.) [5].

І.П.Яковлев стверджував, що інтеграція – це процес об'єднання у ціле будь-яких елементів, у результаті якого виникають нові властивості [80].

$$F = \frac{S_p^2}{S_e^2} = \frac{\sum(\hat{y}_i - \bar{y})^2}{p} = \frac{\sum(\hat{y}_i - \bar{y})^2}{p \sum(y_i - \bar{y})^2} = \frac{R^2}{1 - R^2} = \frac{R^2[n - (p + 1)]}{(1 - R^2)p} = 22,7;$$

5) (Значущість F) – рівень значущості F-критерія $\alpha = 4,47654E-06 = 0,00000447$.

Коефіцієнт детермінації R^2 дорівнює 0,6736, що свідчить про досить сильну залежність між незалежними змінними та результуючою. Можна використати F-статистику, для того щоб визначити чи є цей результат випадковим (саме значення R^2). Величина α застосовується для позначення ймовірності помилкового висновку про те, що має місце досить сильний зв'язок.

Припустимо, що насправді взаємозв'язок між змінними відсутній. Ми просто обрали дуже рідкісні спостереження, для яких статистичний аналіз показав сильний взаємозв'язок.

Оцінку надійності рівняння регресії у цілому і коефіцієнта детермінації дає F-критерій Фішера. Його розрахункове (фактичне) значення $F = 22,7$ порівнюється із табличним, яке при рівні значущості $\alpha = 0,05$ та при кількості ступенів вільності $v_1 = p = 2$, $v_2 = n - (p + 1) = 22$ становить 3,44. Якщо фактичне значення перевищує табличне, то з ймовірністю 0,95 гіпотеза про ненадійність рівняння відхиляється і стверджується статистична значущість моделі та коефіцієнта детермінації. Або, що те саме, з таблиці дисперсійного аналізу слідує, що ймовірність випадково отримати таке значення F-критерія ($F = 22,7$) становить 0,0000447, що не перевищує гранично допустимого значення значущості $\alpha = 0,05$. Отже, отримане значення є не випадковим, воно сформувалось під впливом суттєвих факторів, тобто, статистична значущість всієї моделі та коефіцієнта детермінації підтверджується.

Оцінка параметрів моделі та їх значущості («Оценка параметров модели и их значимость»)

Цей блок результатів містить 9 стовпців.

Перший та другий – назва та величина оцінок параметрів моделі:

Y – пересечение – $b_0 = 15,46076159$; змінна x_1 – $b_1 = -1,441741312$; змінна x_2 – $b_2 = 2,791440539$.

Таким чином, отримали модель: $\hat{y} = 15,4608 - 1,4417x_1 + 2,7914x_2$.

Величини b_j ($j = 1,2$) показують, наскільки зміниться результуюча, якщо збільшити значення деякого фактора на одиницю (при незмінній величині решти факторів). Наприклад,

- збільшення x_1 (вік автомобіля) на 1 рік, при незмінних інших факторах, буде сприяти зменшенню y (ціни автомобіля) на 1,44 (тис. у.о.);

а) $R = 0,821$ – коефіцієнт кореляції;

б) $R^2 = 0,674$ – не скоригований коефіцієнт детермінації (без урахування кількості ступенів вільності) він оцінює долю варіації результуючої Y за рахунок введення в модель факторів. У нашому прикладі ці доля складає 67,4%. Це свідчить про досить високий ступінь обумовленості варіації результуючої від варіації факторів (тобто, це свідчить про досить тісний зв'язок фактору та результуючої);

в) Нормований $\overline{R^2} = 0,64$ – скоректований коефіцієнт детермінації, що визначає тісноту зв'язку з урахуванням кількості ступенів вільності загальної та залишкової дисперсій. Він дає оцінку, яка не залежить від кількості незалежних змінних (факторів) моделі, і тому може порівнюватись з різними моделями з різною кількістю факторів.

Зв'язок між не скоригованим та скоригованим коефіцієнтами детермінації визначається за формулою:

$$\overline{R^2} = 1 - (1 - R^2) \frac{(n-1)}{(n-p-1)},$$

д) де n – кількість спостережень, p – кількість незалежних змінних.

Обидва коефіцієнта (R^2 , $\overline{R^2}$) вказують на досить високу детермінованість результуючої Y у моделі із заданими факторами x_1 , x_2 .

Стандартна похибка залишків $s_e = 4,46$.

Дисперсійний аналіз («Дисперсионный аналіз») включає п'ять стовпців:

1) (**df**) – ступені вільності, тобто кількість свободи незалежної варійованої ознаки. Кількість ступенів вільності пов'язана із кількістю одиниць сукупності n і кількістю оцінок, які за нею визначаються:

для регресії $df = p =$; для залишку $df = n - (p + 1) = 22$; загальна $df = n - 1 = 24$.

2) (**SS**) – суми квадратів відхилень:

$$\sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2 = 904,125 \text{ - регресія (пояснена);}$$

$$\sum_{i=1}^n (Y_i - \hat{Y}_i)^2 = 438,061 \text{ - залишкова (не пояснена);}$$

$$\sum_{i=1}^n (Y - \bar{Y})^2 = 1342,1856 \text{ - загальна (залежної змінної).}$$

3) (**MS**) – дисперсія на одну ступінь вільності:

$$S_p^2 = \frac{\sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2}{p} = 452,062 \text{ - регресій (пояснена);}$$

$$S_e^2 = \frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{n - (p + 1)} = 19,912 \text{ - залишків (не пояснена);}$$

4) (**F**) – фактичне значення F-критерія:

М.Г.Чепіков у своїх дослідженнях писав про інтеграцію наук як про форму руху матерії. Цей рух відбувається як у синтезуючих процесах «внутрішніх» (взаємопроникнення напрямів, яке відбувається у кожній галузі) та «зовнішніх» (взаємозв'язок, єдність між галузям знань, що створює цілісність системи науки) так і в процесах «вертикальних» (що інтегрують науки від більш загальних, теоретичних (математика, кібернетика) до зв'язуючих (природничим та суспільним), потім до прикладних, технічних, що безпосередньо пов'язані із виробництвом, та «горизонтальних» (зв'язок наукових галузей всередині великих здавна сформованих наук – природничих, суспільних, технічних) [78].

Н.С.Антонов трактував інтеграцію наступним чином: «В цілях наглядності свяжи между различными предметами, возникающими на этой основе, можно представить в виде своеобразной «силовой линии», пересекающей радиальные потоки знаний, стекающихся к студенту. В точке пересечения находятся своего рода «стыковые узлы», позволяющие студентам увидеть один и тот же факт с разных точек зрения, обнатужить закономерные свяжи между разнородными знаниями, установит их внутренне (сущностное) родство» [3].

Б.М.Кедров сформулював таке означення: інтеграція – це конкретне вираження синтезу наук як міждисциплінарного процесу, їх злиття в єдине ціле та взаємозв'язок [44].

М.М. Левіна зазначає: «Интеграция – не эмпирическое объединение произвольного множества элементов процесса обучения, связанных лишь ситуативно, а переход количества в качество. Это взаимосвязанная внутренняя и взаимообусловленная целостность процесса обучения, обладающая свойствами, отсутствующими у составляющих ее компонентов (целей, содержания, методов, форм и т.д.). Это открытие новых связей и отношений между компонентами путем включения в новые системы связей» [52].

А.В.Ястребов [81] у своїй статті «Міждисциплінарний підхід у викладанні математики» обґрунтував необхідність розкриття полідисциплінарної суті математики. Він влучно зауважив, що дуалізм є справжнім обличчям не тільки математики, а і науки в цілому. Наприклад, відомий математик Г.Харді ввів у біологію поняття рівноважної (ідеальної популяції), також математик Н.Хомський ввів у сучасну лінгвістику поняття формальної граматики та контексно-вільної мови, фізик Маріот створив капілярну теорію (перша у світі теорія підняття розчинів уверх по рослині) тощо. Цей перелік не є вичерпним, але він свідчить, що саме дуалізм, як зазначав А.В.Ястребов, або навіть багатогранність науки може слугувати однією з основ міждисциплінарності у навчанні різних дисциплін, математики зокрема.

У даний час увага до проблеми інтеграції посилюється.

Стратегічні напрями розвитку вищої освіти в Україні визначаються Конституцією України, законом України «Про вищу освіту», Національною доктриною розвитку освіти України у XXI ст., Державною національною програмою «Освіта» («Україна XXI століття»), указами президента України,

постановами Кабінету Міністрів України тощо. У всіх цих документах прямо або опосередковано підкреслюється важливість інтеграційних процесів.

Так, у Національній доктрині розвитку освіти України у XXI ст. [56] одним із шляхів реалізації неперервності освіти є створення інтегрованих навчальних планів та програм, а згідно з Державною національною програмою «Освіта» («Україна ХНІ століття») реформувати освіту можна за допомогою інтеграції освіти та науки [15].

Узагальнюючи різні тлумачення поняття «інтеграція» Васяк Л.В. [6] запропонувала більш загальне висвітлення даного явища: інтеграція – це процес взаємодії на єдиній світоглядній та логіко-методологічній основі структурних елементів тих чи інших наук, що супроводжується зростанням їх уніфікації та комплексності. Таким чином, інтеграцію треба розглядати як систему, яка має свою структуру, або як процес, що має певні рівні розвитку.

Існують наступні ступені інтеграції: предметна (інтеграторами є складні об'єкти дослідження чи комплексні проблеми), проблемна (інтеграторами є загальні методи дослідження); а також горизонтальна (у природничих науках або всередині дисципліни) і вертикальна (між групами наук) [38].

Як засвідчив педагогічний експеримент, студенти, які отримали підготовку із фундаментальних дисциплін, мають труднощі із застосуванням отриманих знань при вивченні фахових дисциплін. Зокрема, студентам бракує досвіду використання отриманих під час вивчення математики знань, умінь та навичок до розв'язування нетипових або прикладних задач. Крім того, практично відсутня наступність у навчанні дисциплін у вищому навчальному закладі, як наслідок відбувається:

- утворення прогалів у знаннях студентів з окремих тем,
- дублювання навчального матеріалу в ході вивчення різних дисциплін,
- порушення логічна послідовність вивчення суміжних розділів різних наук (дисциплін),
- неузгодженість термінів, символів тощо.

Інтеграція може здійснюватись на таких рівнях:

- 1) змістовому (внутрішньопредметна, міжпредметна, рівнева (за освітніми рівнями «школа-вуз»)),
- 2) організаційному (форми навчальних занять, методів, прийомів, засобів тощо).

Разом з тим, інтеграція дозволяє уникати дискретності у оволодінні знаннями та вміннями, та надає можливості розробки умов, що сприяють формуванню системних, цілісних наукових знань та практичних навичок.

Інтеграція у навчанні – це процес об'єднання знань з різних дисциплін в єдину наукову картину світу. Встановлення взаємозв'язків між окремими елементами знань з різних дисциплін у процесі наукового пізнання сприяє формуванню у студентів системного мислення, що є основною умовою формування наукового світогляду

Враховуючи висловлювання загальновідомих педагогів та науковців й історичний досвід можна стверджувати, що виникла необхідність відобразити реальні взаємозв'язки оточуючого середовища і у навчальному процесі. При

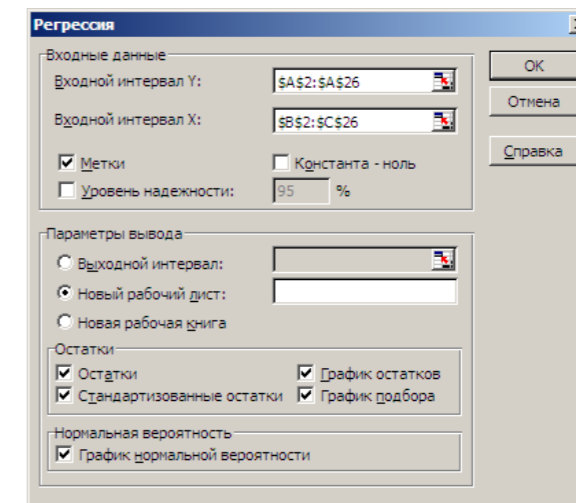


Рис. 2.153 – Вікно налагодження функції Регрессия

Входной интервал Y – це діапазон, який містить дані результативної ознаки, що аналізуються;

Входной интервал X – це діапазон, який містить дані факторів незалежної ознаки;

Метки – флажок, який вказує, чи містить перший рядок назву стовпців або ні;

Константа – нуль – флажок, який вказує наявність або відсутність вільного члена у рівнянні;

Выходной интервал – тут досить вказати ліву верхню комірку майбутнього діапазону;

Новый рабочий лист – можна задати довільне ім'я нового листка.

Якщо необхідно отримати додаткову інформацію за залишками і графіками залишки, то встановіть відповідні флажки в діалоговому вікні, після чого клацніть на кнопці ОК.

Результати регресійного аналізу представлені на рис. 2.154, 2.155, 2.156.

Microsoft Excel - парна.xls									
Файл Правка Вид Вставка Формат Сервис Данные Окно Справка									
123									
A	B	C	D	E	F	G	H	I	
1	Вывод итогов								
2									
3	Регрессионная статистика								
4	Множественный R	0,820744262							
5	R-квадрат	0,673621144							
6	Нормированный R-квадрат	0,643950339							
7	Стандартная ошибка	4,462271134							
8	Наблюдения	25							
9									
10	Дисперсионный анализ								
11		df	SS	MS	F	Значимость F			
12	Регрессия	2	904,1245991	452,0622996	22,70316365	4,47654E-06			
13	Остаток	22	438,0610009	19,91186368					
14	Итого	24	1342,1856						
15									
16		Коэффициенты	Стандартная ошибка	t-статистика	P-значение	Нижние 95%	Верхние 95%	Нижние 95,0%	Верхние 95,0%
17	Y-пересечение	15,46076159	3,658697153	4,225756039	0,000347624	7,873088135	23,04843504	7,873088135	23,04843504
18		-1,441741312	0,256510285	-5,620598461	1,1894E-05	-1,973711081	-0,909771543	-1,973711081	-0,909771543
19		2,791440539	1,313673913	2,124911297	0,045067814	0,067047603	5,515833476	0,067047603	5,515833476
20									

Рис. 2.154 – Приклад зображення виводу регресійного аналізу

За рис. 2.154 робимо такі висновки:

Регресійна статистика («Регрессионная статистика»)

Новий робочий лист – можна задати довільне ім'я нового листка.

Якщо необхідно отримати додаткову інформацію *Итоговой статистики, Уровня надежности*, k - го наибольшего и k - го наименьшего значений, встановить відповідні позначки у діалоговому вікні, після чого клацніть ОК.

Результати обчислень відповідних показників представлені на рис. 2.151.

	A	B	C	D	E	F
1	Столбец1		Столбец2		Столбец3	
2						
3	Среднее	13,024	Среднее	5,88	Среднее	2,164
4	Стандартная ошибка	1,495652366	Стандартная ошибка	0,73557234	Стандартная ошибка	0,14362915
5	Медиана	13	Медиана	6	Медиана	2
6	Мода	6	Мода	6	Мода	1,8
7	Стандартное отклонение	7,47826183	Стандартное отклонение	3,6778617	Стандартное отклонение	0,71814576
8	Дисперсия выборки	55,9244	Дисперсия выборки	13,5266667	Дисперсия выборки	0,51573333
9	Эксцесс	-0,762971938	Эксцесс	0,75901231	Эксцесс	1,70461477
10	Асимметричность	0,270538648	Асимметричность	0,78101307	Асимметричность	1,38263801
11	Интервал	26,4	Интервал	15	Интервал	2,7
12	Минимум	1,6	Минимум	1	Минимум	1,3
13	Максимум	28	Максимум	16	Максимум	4
14	Сумма	325,6	Сумма	147	Сумма	54,1
15	Счет	25	Счет	25	Счет	25
16	Наибольший(1)	28	Наибольший(1)	16	Наибольший(1)	4
17	Наименьший(1)	1,6	Наименьший(1)	1	Наименьший(1)	1,3
18	Уровень надежности(95,0%)	3,086874746	Уровень надежности(95,0)	1,51814668	Уровень надежности(95,0)	0,296436

Рис. 2.151 – Приклад зображення обчислень

За даними рис. 2.152 знаходимо, що:

- 1) вибіркоче середнє значення у дорівнює 13,024, x_1 - 5,88, x_2 - 2,164;
- 2) середнє квадратичне відхилення у дорівнює 7,48, x_1 - 3,68, x_2 - 0,72;
- 3) дисперсія у дорівнює 55,9, x_1 - 13,5, x_2 - 0,52;
- 4) і т.д.

У головному меню виберіть Сервис/Анализ данных/Регрессия (рис. 4.14), після цього клацніть ОК.

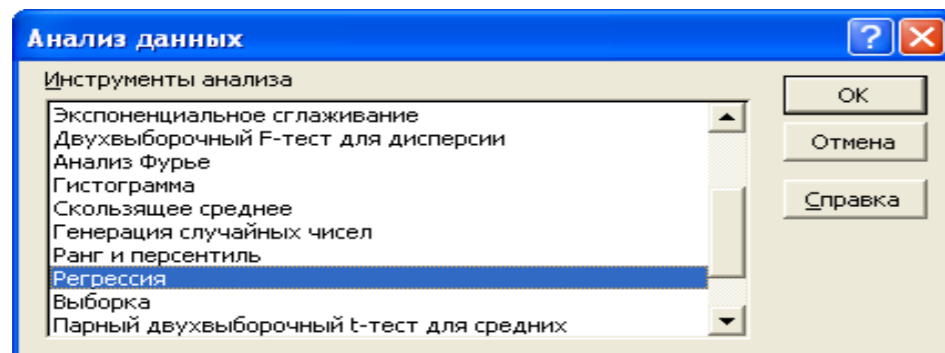


Рис. 2.152 - Вікно Анализ данных при виборі функції Регресия

Заповніть діалогове вікно вводу параметрів інструменту «Регрессия» (рис. 2.153).

цьому на перший план у навчальному процесі вищого навчального закладу має вийти інтеграція між фундаментальними та фаховими дисциплінами.

Таким чином, основною тенденцією розвитку сучасної вищої освіти стає інтеграція різних способів пізнання оточуючого середовища, зокрема, під час навчання студентів економічних та технічних спеціальностей важливим є вивчення глибинних зв'язків між різноманітними процесами. Математика є потужним засобом пізнання у цих галузях науки. Тому вона має значні можливості для підвищення рівня підготовки студентів економічних та технічних спеціальностей університетів.

З метою забезпечення високого освітнього рівня майбутніх фахівців у галузі економіки та техніки треба модернізувати математичну освіти. Одним із шляхів має стати інтеграція математики та фахових дисциплін із застосуванням інформаційно-комунікаційних технологій.

Наведемо приклад, інтеграції математики, ІКТ та однієї зі фахових дисциплін, майбутніх економістів «Гроші та кредит».

У ході дослідження питання діяльності фондової біржі у цілому та України зокрема, студенти мають зосередитись на індексах фондового ринку, які на думку провідних економістів є основними показниками стану фінансового ринку країни.

Фондові індекси були придумані для того, щоб інвестори бачили тенденції розвитку фондового ринку. На їх підставі експерти прогнозують подальшу поведінку ринку. Якщо економіка певної країни демонструє хороші темпи зростання, то інвестори всього світу намагаються купувати акції її компаній. Це викликає не тільки зростання цін цих акцій, а й зростання курсу національної валюти держави.

Фондовий індекс UX є вимірником доходу, який може бути отриманий власником конкретного набору акцій. Це числове представлення руху цін набору акцій щодо їх базового значення на початкову дату в минулому.

Таким чином, в Україні біржові індекси є важливим інструментарієм для вітчизняних та іноземних інвесторів у складі загальної фінансової інформації для прийняття ними інвестиційних рішень.

Одним із цих показників є Індекс Української біржі (надалі Індекс UX). На сьогоднішній день Індекс UX розраховується на основі цін 10 акцій "блакитних фішок" України – акцій найбільш великих українських компаній, лідерів у своїх галузях (табл. 1.1):

Таблиця 1.1

Перелік акцій «блакитних фішок»

Код	Назва	Кількість випущених акцій	Коефіцієнт, який враховує free-float (Wi)	Коефіцієнт, який обмежує вагу акцій (Ci)	Вага акції станом на 30.08.2013, %
ALMK	Алчевський металургійний комбінат, аз	25 775 254 803	4%	1	5,05%

AVDK	Авдіївський коксохімічний завод, аз	195 062 500	7,50%	1	6,04%
AZST	Азовсталь, аз	4 204 000 000	4%	1	13,53%
BAVL	Райффайзен Банк Аваль, аз	29 977 749 080	3,5%	1	12,97%
CEEN	Центренерго, аз	369 407 108	22%	0,3783798	20,00%
DOEN	Донбасенерго, аз	23 644 301	14%	1	9,87%
ENMZ	Єнакієвський металургійний завод, аз	10 550 688	9%	1	4,09%
MSICH	Мотор Січ, аз	2 077 990	24%	0,1819597	20,00%
UNAF	Укрнафта, аз	54 228 510	0,5%	1	3,90%
USCB	Укрсоцбанк, аз	18 140 137 719	1,5%	1	4,53%

Список цінних паперів, що використовуються для розрахунку індексу, визначається Індексним комітетом і складається не менше ніж з 10 найбільш ліквідних акцій українських компаній. Вибір акцій здійснюється на основі експертної оцінки серед цінних паперів, допущених до торгів на Біржі.

Експертна оцінка заснована на використанні такої інформації як: обсяг торгів, частота укладання угод, наявність попиту та пропозиції, величина спреда, капіталізація з урахуванням кількості акцій, що перебувають у вільному обігу, галузева приналежність емітентів цінних паперів. Також можуть враховуватися й інші фактори, що впливають на ліквідність акцій.

При прийнятті рішення про склад Списку цінних паперів досліджується статистична інформація про торги за три попередні місяці.

Індекс UX розраховується як відношення сумарної ринкової капіталізації цінних паперів (MC_n), включених до списку для розрахунку індексу (далі - Список цінних паперів), до сумарної ринкової капіталізації цінних паперів на початкову дату (MC_1), помножене на значення індексу на початкову дату (I_1) і на поправочний коефіцієнт (Z_n):

$$I(UX) = Z_n * I_1 * \frac{MC_n}{MC_1} \quad (1.1)$$

У свою чергу сума ринкової капіталізації акцій на поточний час в українських гривнях (MC_n) обчислюється за формулою:

$$MC_n = \sum_{i=1}^N W_i * P_i * Q_i * C_i, \quad (1.2)$$

де W_i - поправочний коефіцієнт, що враховує кількість і-их акцій у вільному обігу (коефіцієнт free-float),

C_i - коефіцієнт, який обмежує частку капіталізації і-ой акції (ваговий коефіцієнт);

Q_i - загальна кількість і-их акцій,

У головному меню послідовно виберіть Сервіс/Надстройки. Встановіть позначку (галочку) навпроти Пакет анализа (рис. 2.148), після чого клацніть на ОК.

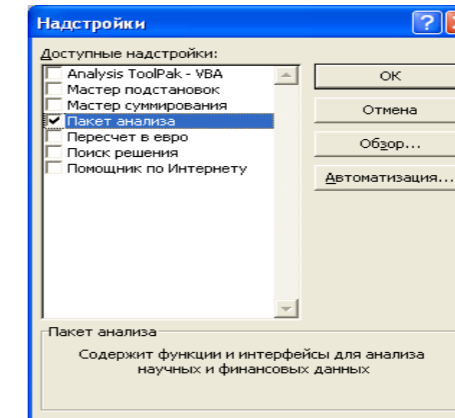


Рис. 2.148 – Вікно Надстройки

У головному меню виберіть Сервіс/Анализ данных/Описательная статистика (рис. 2.149), після чого клацніть ОК.

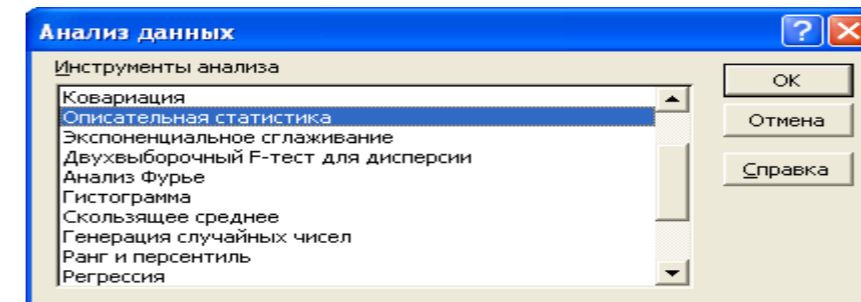


Рис. 2.149 – Вікно Анализ данных при виборі функції Описательная статистика

Заповніть діалогове вікно вводу даних та параметрів вводу (рис. 2.150).

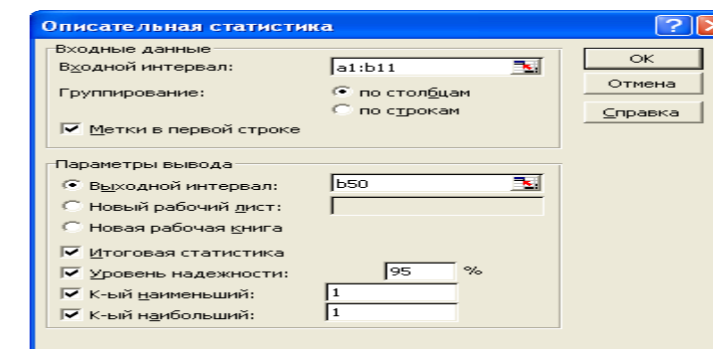


Рис. 2.150 – Вікно меню Описательная статистика

Входной интервал – це діапазон, який містить дані, що аналізуються (це може бути один або декілька стовпців, рядків);

Группирование – за стовпцями або за рядками (треба вказати додатково);

Метки – флажок, який вказує, чи містить перший рядок назву стовпців чи ні;

Выходной интервал – достатньо вказати ліву верхню комірку майбутнього діапазону;

Продовження таблиці 2.16

5.	17	2	2,4
6.	18,9	4	4
7.	15,8	3	2
8.	18	6	3
9.	19	3	4
10.	6	6	1,6
11.	13	6	2,6
12.	22,7	5	2,1
13.	13,9	3	2
14.	10,9	3	1,5
15.	9,5	8	2,6
16.	28	3	2,2
17.	6	8	1,8
18.	5,8	9	1,6
19.	2,8	11	1,6
20.	26	1	2,2
21.	10,5	6	2
22.	16,9	7	2,3
23.	3,4	9	1,5
24.	14	1	1,4
25.	23	1	1,8
Середнє	13,024	5,88	2,164

Припустимо, що між ціною та означеними технічними характеристиками існує лінійна залежність:

$$Y = b_0 + b_1 X_1 + b_2 X_2 + \varepsilon$$

Необхідно:

- 1) визначити b_i для залежності між досліджуваним фактором y (ціною автомобіля) та пояснюючими змінними x_1 (вік автомобіля) і x_2 (об'єм двигуна);
- 2) проаналізувати ступінь адекватності побудованої моделі та вибіркових даних.

Розв'язання. Введіть вихідні дані так як це показано на рис. 2.147.

	A	B	C	D	E
1	y	x ₁	x ₂		
2	11	6	3		
3	3,2	10	1,3		
4	8,7	10	1,8		
5	1,6	16	1,8		
6	17	2	2,4		
7	18,9	4	4		
8	15,8	3	2		
9	18	6	3		
10	19	3	4		
11	6	6	1,6		
12	13	6	2,6		
13	22,7	5	2,1		
14	13,9	3	2		
15	10,9	3	1,5		
16	9,5	8	2,6		
17	28	3	2,2		
18	6	8	1,8		
19	5,8	9	1,6		
20	2,8	11	1,6		
21	26	1	2,2		
22	10,5	6	2		
23	16,9	7	2,3		
24	3,4	9	1,5		
25	14	1	1,4		
26	23	1	1,8		
27					

Рис. 2.147 – Приклад введення даних

P_i - ціна i -тої акції в гривнях на момент розрахунку n ,

N - число акцій в Списку цінних паперів.

Для запропонованих вище формул початкові значення наступні:

$I_1 = 500$ за підсумками торгів 26 березня 2009, $MC_1 = 1243418850$,

$Z_n = 2,8480384$ (дата останнього оновлення 18.03.2013).

Розкриття інформації про індекс і методика його розрахунку здійснюється на веб-сайті Біржі [7].

Для спрощення обчислення та економії часу студентам можна запропонувати використати ІКТ, наприклад табличний процесор Microsoft Excel. При цьому використовується значна кількість функцій, наприклад: =СУММ; =СРЗНАЧ; =СТЕПЕНЬ та ін. Для прогнозування та формулювання висновків, використовують графіки, побудовані на основі отриманих даних за допомогою функції – Мастер Диаграм.

Значення Індексу UX публікуються щоп'ятнадцять секунд між 10:30 й 17:30 та траншуються усім учасникам торгів через біржовий термінал – прикладом якого є система QUIK (рис. 1.3). QUIK - це програмний комплекс для організації доступу до біржових і позабіржових торговельних систем в режимі реального часу [7]. На основі технічного аналізу, теорії Доу, яка описує поведінку цін акцій в час (аналіз акцій «блакитних фішок» України тощо), можна визначити перспективу руху значення Індексу Української біржі, що дасть змогу отримати певну картину, на базі якої робити певні економічні дії.

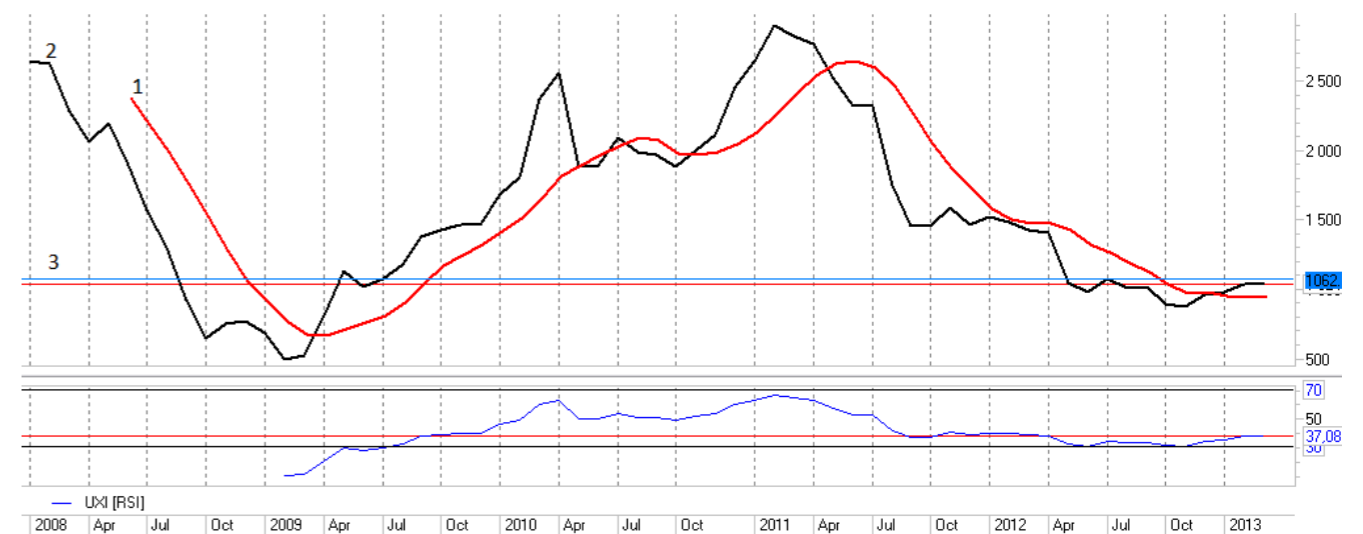


Рис.1.3 – Графік динаміки Індексу (UX)

На графіку (рис.1.3) зображено динаміку Індексу UX, де вісь абсцис – проміжок часу, вісь ординат – ціна індексу. За графіком можна стверджувати, що починаючи з 2011 року і до жовтня 2012 року індекс рухався у спадному напрямку, не даючи значних імпульсів вгору. Зараз, як бачимо, рівень спаду призупинився, ціна Індексу на даний момент перевищує середні піврічні ціни (рис.1.3, лінія1). Може відбутися так званий «розворот ринку», якщо ціна (рис.1.3, лінія2) «проб'є» лінію опору із значенням 1062 (рис. 1.3, лінія 3).

Таким чином, з того, що було зазначено нами вище, студенти мають зробити висновок, що математика та інформаційно-комунікаційні технології досить активно використовуються в роботі фондової біржі. Вони є потужним засобом, без якого, на нашу думку, функціонування фондової біржі на сьогоднішній день не представляється можливим.

У подальшому у ході вивчення теми «Емпіричні формули» дисципліни «Вища математика» студентам варто запропонувати аналогічно дослідити питання доцільності включення до переліку «блакитних фішок» тих чи тих українських компаній. Подібні операції із використанням математичного апарату студентам треба буде здійснювати під час вивчення у подальшому таких фахових дисциплін як «Статистика» та «Економетрика».

Студентами були зібрані дані про значення цін акцій «блакитних фішок» України та значення ф'ючерса на індекс UХ по закінченню торговельного дня на певну дату, які подані в таблиці 1.2:

Таблиця 1.2

Ціни акцій «блакитних фішок»

Дата (2013 рік)	Значення індексу UХ	ALM К	AVD К	AZS Т	BAVL	CEE N	DOE N	ENM Z	MSI CH	UNA F	USCB
04.01	1476,74	0,1073	6,236	1,378	0,134	8,624	28,91	71,04	2227	364,7	0,173
03.02	1555,36	0,109	6,161	1,518	0,138	9,254	31,91	72,38	2390	377,3	0,1906
06.03	1406,58	0,0969	5,408	1,395	0,1192	8,11	28,1	62,55	2344	326	0,1736
02.04	1428,25	0,0921	5,172	1,291	0,1227	8,048	28,82	67,85	2807	290,9	0,2169
15.05	1119,3	0,0636	3,595	0,955	0,1096	6,213	20,51	43,56	2471	213	0,1922
07.06	828,57	0,0475	2,19	0,659	0,0854	4,348	16,7	30,23	1861	164,8	0,1522
03.07	1144,19	0,0652	3,716	1,11	0,1036	6,8	22,73	43,77	2370	178	0,195
07.08	1095,02	0,0564	3,77	0,956	0,0988	7,1	24,15	37,88	2315	169,5	0,1707
05.09	952,27	0,0527	3,247	0,814	0,0878	6,157	18,14	31,69	2079	157,6	0,1564
04.10	949,72	0,054	3,36	0,799	0,0789	6,097	18,52	34,52	2140	149	0,1646
12.11	850,21	0,0471	2,865	0,642	0,0676	5,6	18,2	38,7	1920	138,8	0,143
11.12	930,73	0,0541	3,297	0,76	0,857	5,615	20,92	37,28	1995	137,4	0,1362

Таким чином, треба з'ясувати: який існує зв'язок між ціною кожної окремо взятої акції та ціною ф'ючерса на індекс UХ.

Ф'ючерс – це різновид так званого форвардного контракту, який укладається в майбутньому. Тобто це документ, який засвідчує зобов'язання придбати (продати) цінні папери, товари або кошти у визначений час та на визначених умовах у майбутньому. Покупець ф'ючерсного контракту має право придбати такий контракт протягом строку його дії іншим особам без погодження умов такого продажу з продавцем контракту [7].

Побудуємо лінійну залежність ціни ф'ючерса від індексу UХ кожної зазначених у таблиці 1 компаній.

Обчислимо за допомогою MS Excel коефіцієнт лінійної кореляції, та запишемо результати в таблицю 1.3:

- коефіцієнт детермінації ($0 \leq R^2 \leq 1$):

$$R^2 = r^2 \text{ або } R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2} = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

- середню похибку апроксимації:

$$\bar{A} = \frac{1}{n} \sum \left| \frac{y_i - \hat{y}_i}{y_i} \right| \cdot 100\%$$

- F-критерій Фішера:

$$F_{\text{дод}} = \frac{\sum \frac{(\hat{y}_i - \bar{y})^2}{m}}{\sum \frac{(y_i - \hat{y}_i)^2}{n-m-1}} = \frac{r_{xy}^2}{1-r_{xy}^2} \cdot \frac{n-m-1}{m}$$

де n - кількість одиниць сукупності, m - кількість параметрів при змінних (для лінійної регресії $m=1$).

- t-критерій Стьюдента:

$$t_{b_0} = \frac{b_0}{m_{b_0}}; t_{b_1} = \frac{b_1}{m_{b_1}}; t_r = \frac{r_{xy}}{m_{r_{xy}}}$$

$$\text{де } m_{b_1} = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{n-2}}; m_{b_0} = \sqrt{\frac{\sum (y_i - \hat{y}_i)^2}{n-2} \cdot \frac{\sum x_i^2}{n \sum (x_i - \bar{x})^2}}; m_{r_{xy}} = \sqrt{\frac{1-r_{xy}^2}{n-2}}$$

- довірчі інтервали:

$$\gamma_{b_0} = b_0 \pm \Delta_{b_0}; \gamma_{b_1} = b_1 \pm \Delta_{b_1}$$

$$\text{де } \Delta_{b_0} = t_{\text{табл}} \cdot m_{b_0}; \Delta_{b_1} = t_{\text{табл}} \cdot m_{b_1}$$

Примітка. Обчислити всі ці показники можна аналогічно використовуючи можливості табличного процесору Excel. Після обчислення необхідно зробити висновки.

2.7.2. Побудова багатofакторної економетричної моделі за допомогою можливостей табличного процесора excel

Приклад. Необхідно провести дослідження залежності ціни автомобіля (Y) від таких характеристик як вік авто (X_1) та його об'єм двигуна (X_2) на основі вибіркового даних, наведених в таблиці 2.16

Таблиця 2.16

№	Вихідні дані		
	Ціна автомобіля (у, тис. у.о.)	Вік автомобіля (x_1 , років)	Об'єм двигуна автомобіля (x_2 , дм^3)
1.	11	6	3
2.	3,2	10	1,3
3.	8,7	10	1,8
4.	1,6	16	1,8

$$\begin{cases} b_0 \cdot n + b_1 \cdot \sum_{i=1}^n \frac{1}{x_i} = \sum_{i=1}^n y_i; \\ b_0 \cdot \sum_{i=1}^n \frac{1}{x_i} + b_1 \cdot \sum_{i=1}^n \frac{1}{x_i^2} = \sum_{i=1}^n \frac{y_i}{x_i} \end{cases}$$

3. Рівняння регресії степенної функції

$$y = b_0 \cdot x^{b_1}$$

Система нормальних рівнянь у загальному вигляді:

$$\begin{cases} n \cdot \lg b_0 + b_1 \cdot \sum_{i=1}^n \lg x_i = \sum_{i=1}^n \lg y_i; \\ \lg b_0 \cdot \sum_{i=1}^n \lg x_i + b_1 \cdot \sum_{i=1}^n (\lg x_i)^2 = \sum_{i=1}^n \lg x_i \cdot \lg y_i \end{cases}$$

4. Показникове рівняння регресії

$$y = b_0 \cdot b_1^x$$

Система нормальних рівнянь у загальному вигляді::

$$\begin{cases} n \cdot \lg b_0 + \lg b_1 \cdot \sum_{i=1}^n x_i = \sum_{i=1}^n \lg y_i; \\ \lg b_0 \cdot \sum_{i=1}^n x_i + \lg b_1 \cdot \sum_{i=1}^n x_i^2 = \sum_{i=1}^n x_i \cdot \lg y_i \end{cases}$$

5. Параболічне рівняння регресії

$$y = b_0 + b_1 \cdot x + b_2 \cdot x^2$$

Система нормальних рівнянь у загальному вигляді:

$$\begin{cases} b_0 \cdot n + b_1 \cdot \sum_{i=1}^n x_i + b_2 \cdot \sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i; \\ b_0 \cdot \sum_{i=1}^n x_i + b_1 \cdot \sum_{i=1}^n x_i^2 + b_2 \cdot \sum_{i=1}^n x_i^3 = \sum_{i=1}^n x_i \cdot y_i \\ b_0 \cdot \sum_{i=1}^n x_i^2 + b_1 \cdot \sum_{i=1}^n x_i^3 + b_2 \cdot \sum_{i=1}^n x_i^4 = \sum_{i=1}^n x_i^2 \cdot y_i \end{cases}$$

Перевірити значущість рівняння регресії необхідно за допомогою обчислення відповідних показників. Зокрема, треба обчислити - лінійний коефіцієнт парної кореляції r_{xy} ($-1 \leq r_{xy} \leq 1$):

$$r_{xy} = \frac{\frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sigma_x \sigma_y} = \frac{\overline{yx} - \bar{x} \cdot \bar{y}}{\sqrt{(\overline{x^2} - \bar{x}^2) \cdot (\overline{y^2} - \bar{y}^2)}}$$

Коефіцієнти кореляції

	Ціна ф'ючерса індексу UX
ALMK	0,978511102
AVDK	0,975278828
AZST	0,989981672
BAVL	0,165801945
CEEN	0,963199621
DOEN	0,971387544
ENMZ	0,964739821
MSICH	0,731502493
UNAF	0,95368613
USCB	0,707696162

Отже, як видно з результатів розрахунків, між ціною акцій «блакитних фішок» та ціною ф'ючерса індексу UX існує дуже міцний лінійний зв'язок, та важко не помітити, що ціна акцій BAVL здійснює досить малий вплив на ціну ф'ючерса на індекс UX.

Результативний показник може реагувати на зміну іншого з запізненням, тому треба визначити лаг такої залежності. У результаті відповідних обчислень студентами було встановлено, що модель залежності ціни ф'ючерса індексу UX від цін акцій «блакитних фішок» України потрібно будувати без урахування лагу.

Після цього за допомогою можливостей табличного процесора MS Excel студенти побудували лінійне рівняння регресії, що показує залежність ціни індексу UX від зміни цін акцій "блакитних фішок" України.

Модель має вигляд:

$$y = 93,217 - 3897,54x_1 + 88,255x_2 + 304,44x_3 + 29,984x_4 - 11,66x_5 + 6,198x_6 + 0,797x_7 + 0,052x_8 + 0,994x_9 + 1225,103x_{10}$$

де x_1 – ALMK; x_2 – AVDK; x_3 – AZST; x_4 – BAVL; x_5 – CEEN; x_6 – DOEN; x_7 – ENMZ; x_8 – MSICH; x_9 – UNAF; x_{10} – USCB.

Коефіцієнт детермінації:

$$R^2 = 0,9998.$$

Фактичне значення F-критерію Фішера:

$$F_{\text{факт}} = 679,2.$$

Знаходимо табличне значення критерію Фішера за допомогою функції FРАСПРОБР, при рівні значущості $\alpha = 0,05$, ступенях вільності $k_1 = 10$, $k_2 = 1$
 $F_{табл} = 241,9$.

Оскільки $F_{факт} > F_{табл}$, то побудоване рівняння регресії є достовірним.

У результаті аналізу побудовано рівняння студентами був зафіксований цікавий факт, котрий дозволив зробити певні припущення: з поміж всіх інших, коефіцієнт кореляції між значенням ціни ф'ючерса індексу UX та ціни акцій BAVL =0,165801945, що свідчить про дуже слабкий зв'язок між вище зазначеними факторами. Тому, на основі отриманих даних, студенти зробили висновок про необхідність перегляду Індексним комітетом Української біржі доцільність розміщення акцій BAVL в Списку цінних паперів для розрахунку індексу UX.

Наведені вище приклади не вичерпують всіх можливих варіантів навчання математики на інтеграційній основі, вони лише ілюструють один із можливих варіантів інтеграції математики, ІКТ та фахових дисциплін. У результаті застосування запропонованих нами прийомів у навчанні математики у студентів формується стійкий інтерес до вивчення дисципліни, з'являється внутрішня мотивація до навчання в цілому.

1.7. Моніторинг навчальних досягнень студентів за допомогою інформаційно-комунікаційних технологій

Одним із завдань повсякденної викладацької роботи є необхідність здійснювати контроль знань студентів. Форми контролю, що застосовуються викладачами, дуже різноманітні, але найбільш часто використовуються письмове або усне опитування. На жаль, ці форми не позбавлені недоліків. Під час проведення усного опитування - це відносно велика витрата часу при невеликій кількості виставлених оцінок, при проведенні письмових робіт кількість оцінок зростає, але багато часу йде на перевірку.

Тестування як ефективний спосіб перевірки знань знаходить все більше застосування. Одним з основних і безперечних його переваг є мінімум витрат часу на отримання надійних підсумків контролю. При тестуванні використовують як паперові, так і електронні варіанти. Останні особливо привабливі, тому що дозволяють отримати результати практично відразу після закінчення тесту.

Тестування в педагогіці виконує три основні взаємопов'язані функції: діагностичну, навчальну і виховну:

- діагностична функція полягає у виявленні рівня знань, умінь, навичок учня. Це основна, і найочевидніша функція тестування. За об'єктивності, широті і швидкості діагностування, тестування перевершує всі інші форми педагогічного контролю;

- навчальна функція тестування полягає в мотивуванні студентів до активізації роботи по засвоєнню навчального матеріалу. Для посилення навчальної функції тестування, можуть бути використані додаткові заходи

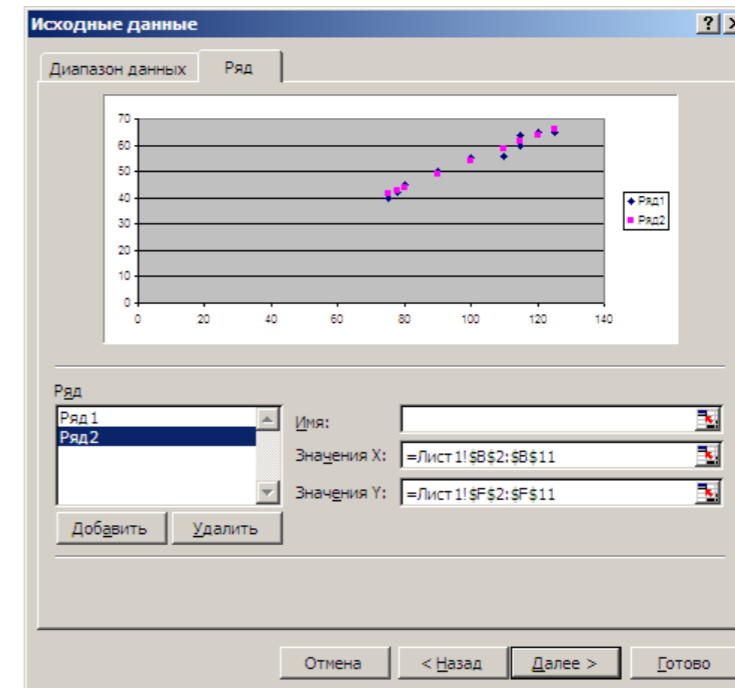


Рис. 2.146 - Вікно Мастер диаграмм при введенні декількох рядів

Значення фактора беруться ті ж самі, що і при побудові поля кореляції, а значення результативного ознаки беруться не вихідні, а розрахункові \hat{y} . Далі натискається кнопка Готово.

Для форматування графіка або його окремих компонентів необхідно зробити подвійне клацання мишею по елементу, що цікавить. Наприклад, можна лінію регресії зробити дійсно лінією, а не крапками. Для цього робиться подвійне клацання мишею на рожевих точках і у вікні вибирається тип, колір і товщина лінії, прибираються маркери. Аналогічно можна відформатувати осі графіка, наприклад, нанести назви, змінити діапазон зміни і так далі.

Різні види рівнянь регресії

1. Лінійне рівняння регресії

$$y = b_0 + b_1 \cdot x$$

Система нормальних рівнянь у загальному вигляді:

$$\begin{cases} b_0 \cdot n + b_1 \cdot \sum_{i=1}^n x_i = \sum_{i=1}^n y_i; \\ b_0 \cdot \sum_{i=1}^n x_i + b_1 \cdot \sum_{i=1}^n x_i^2 = \sum_{i=1}^n x_i \cdot y_i \end{cases}$$

2. Гіперболічне рівняння регресії

$$y = b_0 + \frac{b_1}{x}$$

Система нормальних рівнянь у загальному вигляді:

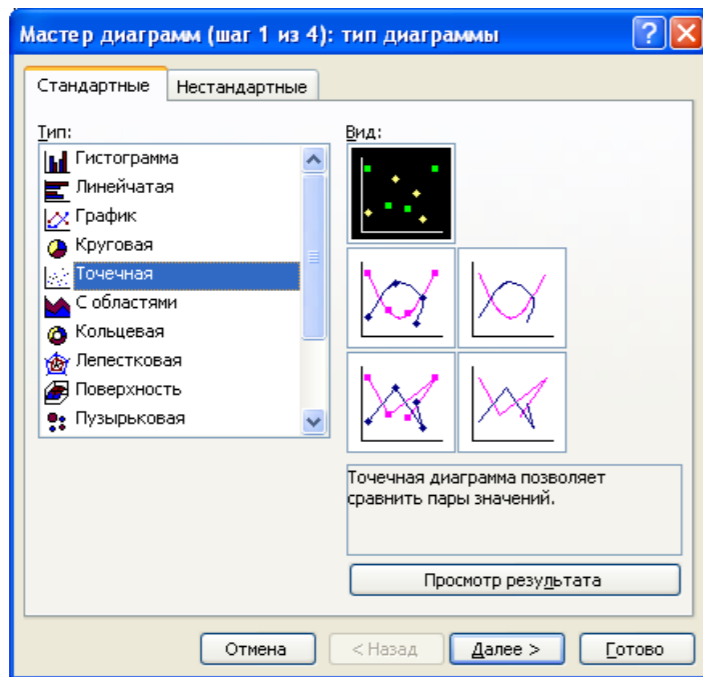


Рис. 2.144 – Вікно Мастер диаграмм при виборі типу діаграми

Натискається кнопка Далее. У вікні вибирається вкладка Ряд. Натискається кнопка Добавить. У полі, що з'явилося, вводяться посилання на значення фактора x і результативної ознаки y (рис. 2.145).

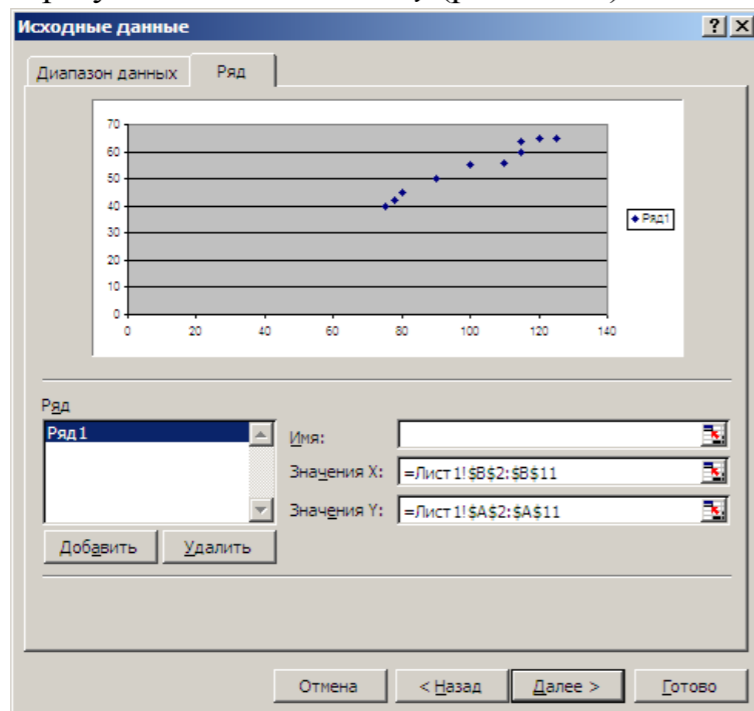


Рис. 2.145 - Вікно Мастер диаграмм при введенні вихідних даних

У вікні перегляду з'являється точковий графік, що відображає вихідні дані - Поле кореляції. Щоб отримати на одному графіку і поле кореляції і лінію регресії необхідно натиснути кнопку Добавить та у відповідні поля занести посилання на клітинки з даними для побудови лінії регресії (рис. 2.146). Натиснути Далее.

стимулювання студентів, такі, як роздача викладачем наближеного переліку питань для самостійної підготовки, наявність в самому тесті навідних запитань і підказок, спільний розбір результатів тесту;

- виховна функція проявляється в періодичності та неминучості тестового контролю. Це дисциплінує, організовує і направляє діяльність студентів, допомагає виявити і усунути прогалини в знаннях, формує прагнення розвинути свої здібності.

Тестування - більш справедливий метод, воно ставить всіх студентів в рівні умови, як в процесі контролю, так і в процесі оцінки, практично виключаючи суб'єктивізм викладача.

Сьогодні існує значна кількість тестових оболонок, за допомогою яких можна здійснювати моніторинг навчальних досягнень студентів, таких як Test-W, Editor, MyTestXPro тощо. Також можна проводити тестовий контроль знань студентів у системі Moodle.

Наприклад, за допомогою програми MyTestXPro можлива організація і проведення тестування, іспитів в будь-яких освітніх установах (вузи, коледжі, школи) як з метою виявити рівень знань по будь-яким навчальним дисциплінам, так і з навчальними цілями. Підприємства та організації можуть здійснювати атестацію та сертифікацію своїх співробітників.

Програма пройшла довгий шлях розвитку від простої оболонки для проведення простих тестів до потужного і зручного комплексу програм. Програма складається з трьох модулів: Модуль тестування (MyTestStudent), Редактор тестів (MyTestEditor) і Журнал тестування (MyTestServer).

Програма MyTestXPro працює з десятьма типами завдань:

- одиночний вибір;
- множинний вибір;
- встановлення порядку проходження;
- встановлення відповідності;
- вказівка істинності чи хибності тверджень;
- ручне введення числа (чисел);
- ручне введення тексту;
- вибір місця на зображенні;
- перестановка букв;
- заповнення пропусків.

Параметри тестування, завдання, звуки і зображення до завдань для кожного окремого тесту - все зберігається в одному файлі тесту. Ніяких баз даних, ніяких зайвих файлів - один тест - один файл. Файл з тестом зашифрований і стиснутий.

Програма MyTestXPro працює з десятьма різними типами завдань. У тесті можна використовувати як завдання одного типу, так і завдання різних типів. Кількість груп і завдань у тесті не обмежена. Питання з варіантами відповіді можуть включати до десяти варіантів. Для кожного завдання можна задати до п'яти формулювань питання.

Текст питання і варіантів відповіді (там, де вони можливі) підтримують можливості форматування тексту, вставки малюнків, таблиць, символів. У програмі є зручний вбудований текстовий редактор. Форматувати текст, вставляти таблиці, малюнки і символи можна не тільки в питання, але і в варіанти відповідей.

Програма підтримує декілька незалежних один від одного режимів тестування. Використовуючи різні режими і параметри тестування, можливо ефективно вирішувати різноманітні завдання, як навчання, так і перевірки знань.

Як і які саме завдання з тесту будуть задані студенту, скільки часу буде йому на обдумування, чи буде показаний вірну відповідь, чи будуть збережені або відправлені результати тестування і багато інших опцій легко налаштовуються за допомогою редактора тестів.

За допомогою MyTestXPro ви можете організувати як локальне, так і мережеве тестування. Можна проводити тестування і не маючи підключення до будь-якої мережі. При мережевому тестуванні результати тестування можуть бути автоматично передані по мережі в модуль Журнал, а можуть бути відправлені по електронній пошті або на веб-сервер в Інтернет методом POST.

При неможливості провести комп'ютерне тестування з електронного тесту можна швидко сформулювати і роздрукувати «паперовий тест». Для зручності поширення тестів серед учнів можна створювати «автономні тести» - програми, що містять один тест і настройки модуля тестування в одному виконуваному exe-файлі.

наступні додаткові дії. Виділяємо діапазон комірок A18: B19, на клавіатурі натискаємо клавішу F2 (при цьому в комірці A18 з'явиться введена раніше формула). Далі натискається комбінація трьох клавіш: Ctrl + Shift + Enter (утримуючи клавіші Ctrl і Shift, останньою натискається клавіша Enter). У результаті в комірках A18: B19 з'явиться матриця обернена даній.

Для отримання остаточного розв'язку необхідно отриману матрицю помножити на вектор-стовпець правих частин рівнянь системи. Для цього скористаємося функцією МУМНОЖ (Категорія: Математичні). Введемо її в клітинку D18, задамо аргументи (при цьому першим масивом повинна бути обов'язково зворотна матриця) і натиснемо ОК (рис. 2.143).

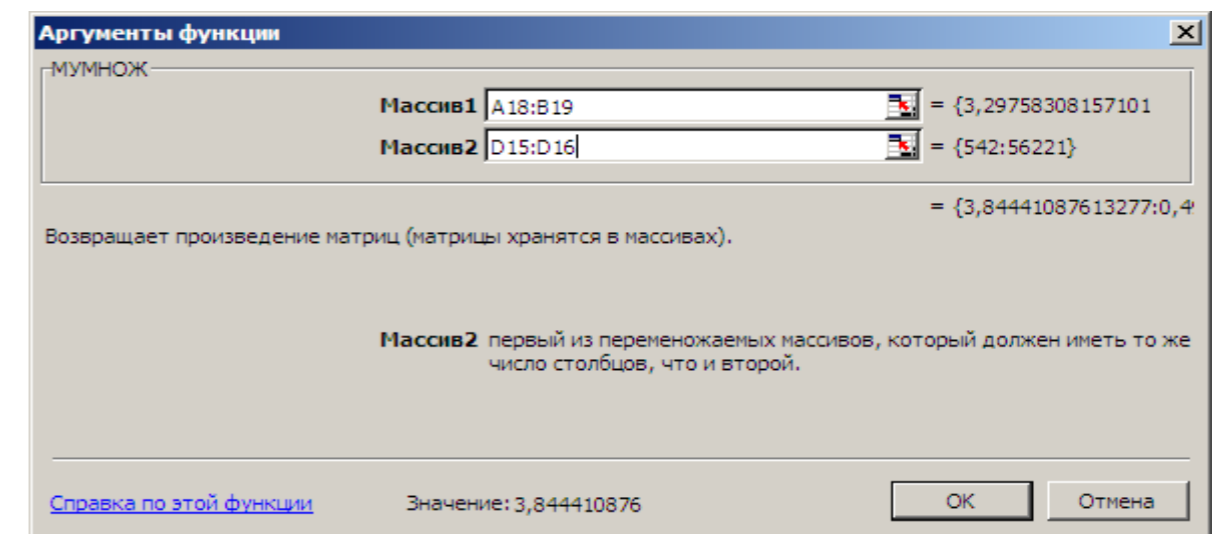


Рис. 2.143 – Відповідне вікно функції Аргументы функции

У результаті в комірці D8 буде отримано значення 3,844411.

Оскільки рішення системи має складатися з двох значень (значення b_0 і b_1), то необхідно здійснити наступні додаткові дії. Виділяємо діапазон комірок D18: D19, натискаємо клавішу F2. Далі натискається комбінація трьох клавіш: Ctrl + Shift + Enter. У результаті в комірках D18: D19 виходить остаточне рішення: $\hat{b}_0 = 3,8$; $\hat{b}_1 = 0,5$.

Економетрична модель має вигляд

$$y = 3,8 + 0,5x + \varepsilon$$

Для побудови графіка і оцінки якості побудованого рівняння регресії необхідно за побудованим рівнянням розрахувати теоретичні значення результативної змінної \hat{y} (Стовпець F). У комірці F2 вводимо формулу: = \$ D \$ 18 + \$ D \$ 19 * B2 (знаки \$ відображають, те що дана клітинка є зафіксованою; фіксування здійснюється за допомогою клавіші F4).

Дана формула поширюється на нижче лежачі комірки шляхом протягування за допомогою миші.

Для побудови графіка необхідно викликати Майстер діаграм, наприклад через головне меню: Вставка / Діаграма ... Тип і вид діаграми вибирається відповідно до рис. 2.144.

Реалізуємо даний підхід в Excel. При цьому розв'язувати будемо в два етапи:

1. Визначимо матрицю обернену до матриці коефіцієнтів лівих частин рівнянь системи, за допомогою функції МОБР.

1. Визначимо розв'язок системи множенням оберненої матриці на матрицю правих частин рівнянь системи за допомогою МУМНОЖ.

Занесемо параметри системи в комірки, при цьому в осередках A15: B16 буде знаходитися матриця лівих частин рівнянь, а в осередках D15: D16 - матриця-стовпець правих частин (рис. 2.141).

	A	B	C	D	E	F	G	H
1	Y	x	x ²	xy				
2	40	75	5625	3000				
3	42	78	6084	3276				
4	45	80	6400	3600				
5	50	90	8100	4500				
6	55	100	10000	5500				
7	56	110	12100	6160				
8	60	115	13225	6900				
9	64	115	13225	7360				
10	65	120	14400	7800				
11	65	125	15625	8125				
12	542	1008	104784	56221				
13								
14								
15	10	1008		542				
16	1008	104784		56221				
17								
18	=МОБР(A15:B16)							
19	=МУМНОЖ(A18:B18;D15:D16)							
20								
21	y=3,8+0,5x							
22								
23								
24								
25								
26								
27								

Рис. 2.141 - Приклад зображення відповідних обчислень

У комірку A18 введемо функцію МОБР (Категорія: Математичні). Ця функція обчислює матрицю обернену до даної. Як аргумент функції задамо масив комірок A15: B16 (рис. 2.142) і натиснемо ОК.

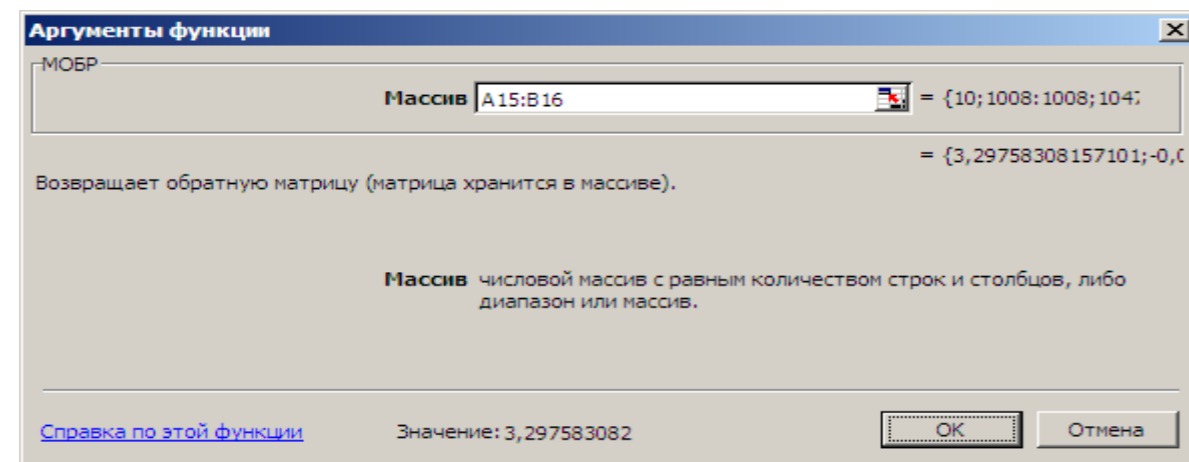


Рис. 2.142 – Відповідне вікно функції Аргументы функции

У результаті в комірці A18 буде отримано значення 3,297583. Оскільки результатом повинна бути матриця розмірністю 2x2, то необхідно здійснити

РОЗДІЛ 2. ВИКОРИСТАННЯ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ У ПРОЦЕСІ НАВЧАННЯ МАТЕМАТИЧНИХ ДИСЦИПЛІН У ВНЗ

2.1. R-програмування

R – потужна мова і безкоштовне середовище програмування, що найчастіше застосовується для статистичних обчислень, аналізу даних та візуалізації (представлення даних у графічному вигляді). Створена Росом Іхаком (Ross Ihaka) та Робертом Джентлменом (Robert Gentleman), працівниками Оклендського Університету в Новій Зеландії. Це безкоштовний проект, що розповсюджується за ліцензією GNU General Public License у вигляді вільнодоступного вихідного коду або відкомпільованих бінарних версій більшості операційних систем: Linux, FreeBSD, Microsoft Windows, Mac OS X, Solaris. R схожий на комерційні мову і середовище S, яка була розроблена в Bell Laboratories (раніше AT & T, зараз Lucent Technologies) Джоном Чемберсом (John Chambers) і колегами. R можна розглядати як «діалект» мови S, що широко використовується в якості навчальної мови та дослідницького інструменту. Основними перевагами проекту R є той факт, що R є безкоштовним і що є багато допоміжної інформації, доступної в Інтернеті. Цей проект дуже схожий на інші програмні пакети, такі як MatLab (не безкоштовний), але він більш зручний для користувача, ніж мови програмування, такі як C++ або Fortran. Ви можете використовувати R таким, яким він є, але в освітніх цілях ми надаємо перевагу використовувати R в поєднанні з інтерфейсом RStudio (також безкоштовним), який має зручні шари і кілька додаткових опцій.

Що таке S?

- ✓ S це мова програмування, яка була розроблена Джоном Чемберсом та іншими в Bell Labs.
- ✓ S була розпочата в 1976 році в якості внутрішнього середовища для статистичного аналізу - спочатку реалізована у вигляді бібліотеки Fortran.
- ✓ Ранні версії мови не містили функцій для статистичного моделювання.
- ✓ У 1988 році система була переписана на C і стала нагадувати систему, яку ми маємо сьогодні (це була 3 версія мови). Книга «Статистичні моделі в S» Чемберса і Хесті (так звана «біла книга») документує функціональність статистичного аналізу.
- ✓ Версія 4 мови S була випущена в 1998 році і є версією яку ми використовуємо сьогодні. Книга «Програмування з даними» Джона Чемберса («зелена книга») документує цю версію мови.

2.1.1. Історичні факти

- ✓ У 1993 році Bell Labs дав StatSci (зараз Insightful Corp.) ексклюзивну ліцензію на розробку і продаж мови S.

✓ В 2004 Insightful придбали мову S у Lucent за \$ 2 млн і є поточним власником.

✓ У 2006 році Alcatel придбала Lucent Technologies і тепер називається Alcatel-Lucent.

✓ Insightful продає свою реалізацію мови S під назвою продукту S-PLUS з великою кількістю різних можливостей, головним чином графічний користувацький інтерфейс (GUI).

✓ В 2008 Insightful купується TIBCO за \$ 25 млн.

✓ Основа самої мови S суттєво не змінилася з 1998 року.

✓ У 1998 році мова S виграла премію в області систем програмного забезпечення асоціації Обчислювальної техніки.

2.1.2. Розвиток мови R

✓ 1991: Створена в Новій Зеландії Россом Іхаком і Робертом Джентлменом.

✓ 1993: Перший анонс R громадськості.

✓ 1995: Мартін Маклер переконує Росса і Роберта використовувати GNU General Public License, щоб зробити R безкоштовним програмним забезпеченням.

✓ 1996: Створюється список розсилки (R- help і R-Devel)

✓ 1997: Формується основна група R (що містить деяких людей, що пов'язані з S-PLUS). Основна група контролює вихідний код для R.

✓ 2000: Випущена версія 1.0.0 R.

✓ 2014: Випущена версія 3.1.1 R на липень 2014.

2.1.3. Особливості мови R

✓ Синтаксис дуже схожий на S, що робить його легким для переходу користувачів S-PLUS.

✓ Семантика зовні схожа на S, але насправді зовсім різна.

✓ Працює практично на будь-якій стандартній обчислювальній платформі/ОС (навіть на PlayStation 3)

✓ Часті релізи (річний + релізи з виправленням помилок); активний розвиток.

✓ Досить невелике за розмірами програмне забезпечення; Функціональність ділиться на модульні пакети

✓ Графічні можливості дуже розвинуті і краще, ніж у більшості статистичних пакетів.

✓ Зручний для інтерактивної роботи, але також містить і потужну мову програмування для розробки нових інструментів (користувач -> програміст).

✓ Дуже активне і енергійне співтовариство користувачів; R- help і R-Devel списки розсилки і Stack Overflow.

✓ Це безкоштовно!

	A	B	C	D	E
1	Y	x	x ²	xy	
2	40	75	5625	3000	
3	42	78	6084	3276	
4	45	80	6400	3600	
5	50	90	8100	4500	
6	55	100	10000	5500	
7	56	110	12100	6160	
8	60	115	13225	6900	
9	64	115	13225	7360	
10	65	120	14400	7800	
11	65	125	15625	8125	
12	542	1008	104784	56221	
13					
14					
15					
16					

Рис. 2.140 – Приклад зображення відповідних обчислень

Суми $\sum_{i=1}^n x_i$, $\sum_{i=1}^n y_i$ обчислюються відповідно у комірках з використанням

кнопки [Автосумма] (Σ) на панелі інструментів.

Квадрати значень фактора x^2 обчислюються у стовпчику C. Для цього у комірку C2 заноситься формула: = A2^2. Дана формула розповсюджується на комірки, що лежать нижче, шляхом протягування за допомогою миші. У останній комірці стовпчику C обчислюється сума значень цього стовбця, тобто

$$\sum_{i=1}^n x_i^2.$$

Добуток $x \cdot y$ обчислюється у стовбці D. Для цього у комірку D2 вводиться формула: =A2*B2. Дана формула розповсюджується на комірки, що лежать нижче, шляхом протягування за допомогою миші. У останній комірці стовпчику D обчислюється сума значень цього стовбця, тобто $\sum_{i=1}^n x_i \cdot y_i$.

На основі проведених обчислень складається система нормальних рівнянь:

$$\begin{cases} 10\hat{b}_0 + 1008\hat{b}_1 = 542 \\ 1008\hat{b}_0 + 104784\hat{b}_1 = 56221. \end{cases}$$

Цю систему необхідно розв'язати з метою визначення параметрів b_0 і b_1 .

Розв'язком системи є параметри $\hat{b}_0 = 3,8$; $\hat{b}_1 = 0,5$.

Отже, економетрична модель має вигляд

$$y = 3,8 + 0,5x.$$

Розглянемо інший можливий варіант побудови економетричної моделі за допомогою можливостей Excel.

З математики відомо, що розв'язати систему лінійних рівнянь можна матричним способом. Для цього необхідно знайти обернену матрицю та помножити її на вектор-стовпець правих частин рівнянь системи.

Розв'яжемо задачу, запропоновану у практичній, за допомогою табличного процесору Excel.

Приклад. Побудувати економетричну модель, яка характеризує залежність витрат на одиницю продукції від рівня фондомісткості продукції. Зробити економічні висновки. Вихідні дані наведені в таблиці 2.15.

Таблиця 2.15

Вихідні дані

№ п/п	1	2	3	4	5	6	7	8	9	10
Y_i	50	40	65	55	45	42	56	60	64	65
X_i	90	75	120	100	80	78	110	115	115	125

Розв'язування. Вихідні дані необхідно розмістити в стовпцях А і В. При цьому перший рядок відводиться для назв стовпців. Останній рядок призначений для обчислення автосум значень відповідних стовпців.

Для отримання якісних графіків вихідні дані необхідно відсортувати. Для чого виділяються значення фактора x (без назви !!!), що знаходяться у колонці А. На панелі інструментів натискається кнопка сортування за зростанням ($\begin{matrix} \uparrow \\ \downarrow \end{matrix}$).

У діалоговому вікні (рис. 2.139) необхідно натиснути кнопку [Сортировка].

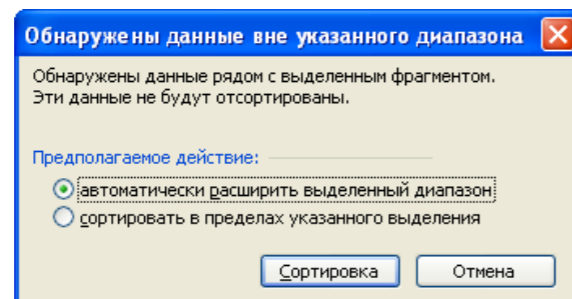


Рис. 2.139 – Діалогове вікно

В результаті значення фактора будуть відсортовані за зростанням. При цьому відповідність між значеннями фактора x і результату y зберігаються.

Розглянемо процес обчислення параметрів лінійного рівняння регресії $\hat{y} = b_0 + b_1x$, а також оцінку його точності.

Для знаходження параметрів b_0 і b_1 необхідно скласти і вирішити наступну систему:

$$\begin{cases} b_0n + b_1 \sum_{i=1}^n x_i = \sum_{i=1}^n y_i; \\ b_0 \sum_{i=1}^n x_i + b_1 \sum_{i=1}^n x_i^2 = \sum_{i=1}^n x_i y_i. \end{cases}$$

де n – кількість спостережень ($n=10$).

Таким чином, для складання системи необхідно обчислити наступні

значення сум: $\sum_{i=1}^n x_i$, $\sum_{i=1}^n y_i$, $\sum_{i=1}^n x_i^2$ и $\sum_{i=1}^n x_i \cdot y_i$ (рис. 2.140).

2.1.4. Недоліки мови R

- ✓ По суті базується на основі 40-річної технології.
- ✓ Невелика вбудована підтримка динамічної або 3-D графіки (але все значно покращилося з так званих «старих днів»).
- ✓ Функціональність заснована на споживчому попиті і власних внесків користувачів. Якщо ніхто не відчуває потреби в реалізації вашого улюбленого методу, то це ваша робота!
- ✓ Об'єкти мають бути збережені, як правило, у фізичній пам'яті.
- ✓ Не ідеально підходить для всіх можливих ситуацій (але це недолік всіх програмних пакетів).

2.1.5. Встановлення R

Для встановлення R на ваш комп'ютер (безкоштовно), відвідайте домашню сторінку R:

<http://www.r-project.org/>

і виконайте наступні кроки (за умови, що ви працюєте на комп'ютері з операційною системою сімейства Windows):

- Натисніть скачати CRAN на лівій панелі (дивись рис.2.1).



Рис. 2.1

- Виберіть сайт для скачування, наприклад <http://cran.rstudio.com/>.
- Виберіть Windows як цільову операційну систему, для цього натисніть Download R for Windows.
- Натисніть base.

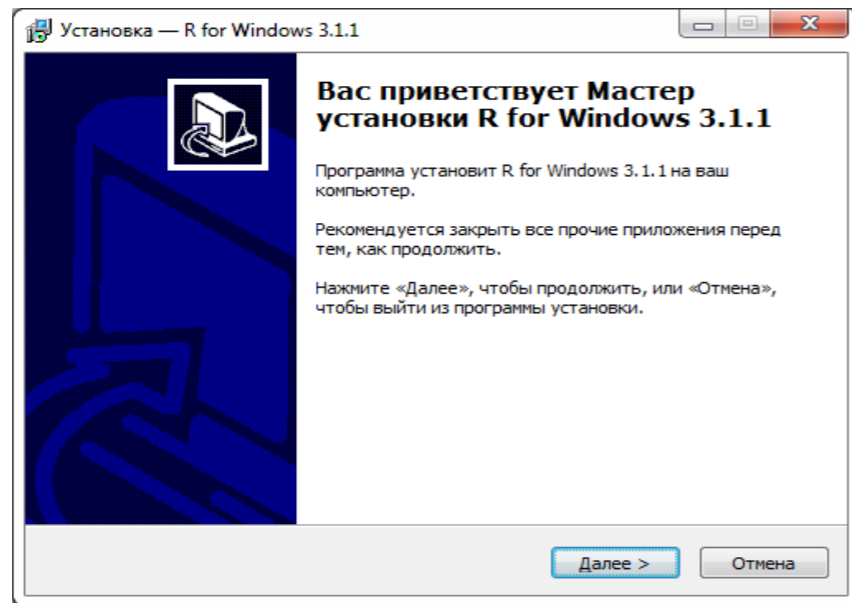


Рис. 2.2

- Виберіть « Download R 3.1.1 for Windows » (або іншу новішу версію, якщо вона доступна) і натисніть зберегти.
 - Запустіть файл що завантажився (R-3.1.1-win.exe)
 - Виберіть мову для встановлення (дивись рис. 2.2) і натисніть кнопку ОК.
 - Натисніть кнопку «Далі» (дивись рис. 2.3).
 - Знову натисніть кнопку «Далі» (дивись рис. 2.4).
- Виберіть місце на комп'ютері куди встановити R та натисніть «Далі» (дивись рис. 2.5).

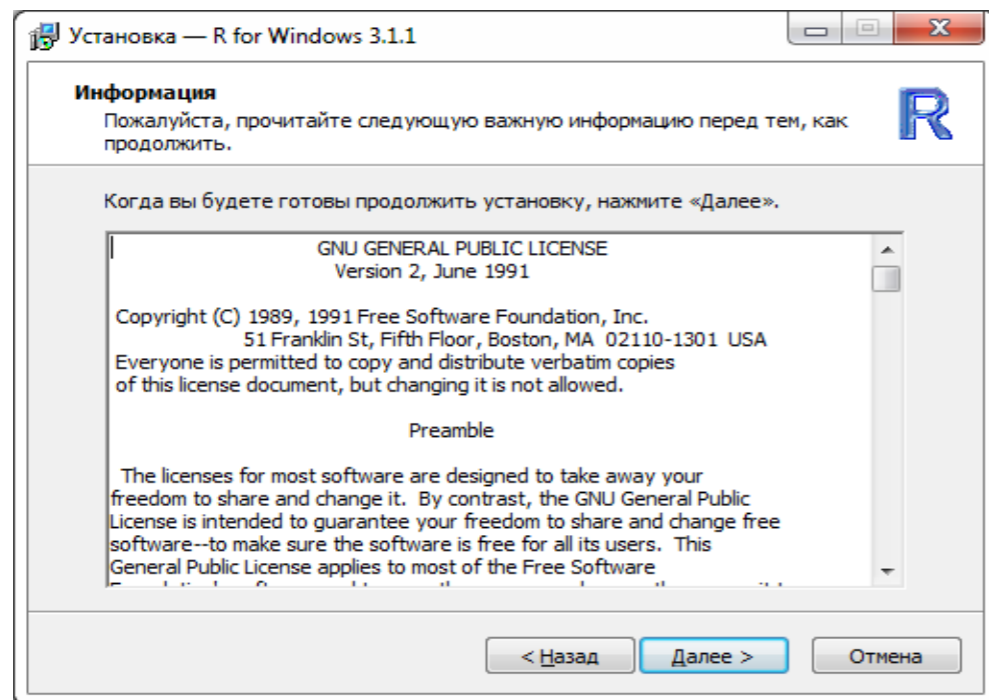
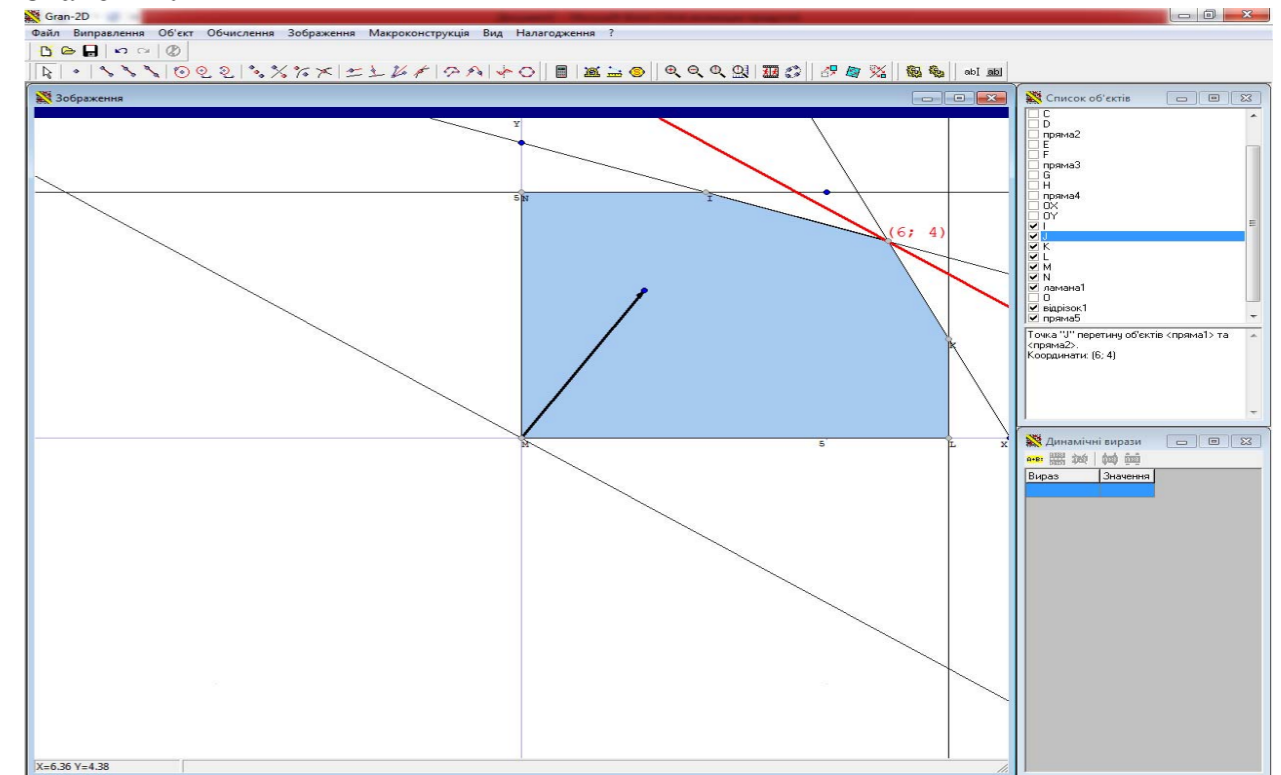


Рис. 2.3

Пересуваючи перпендикуляр в напрямку градієнта (для задачі на максимум) або в протилежному напрямі (для задачі на мінімум), знаходимо вершину многокутника розв'язків, де цільова функція досягає екстремального значення.



Визначаємо координати точки, в якій цільова функція набуває максимального значення. За допомогою GRAN – 2D їх можна дізнатися у вікні списку об'єктів: (6; 4).

Обчислюємо значення цільової функції в цій точці:

$$F(6, 4) = 2 * 6 + 3 * 4 = 24$$

Відповідь. $F_{\max} = 24$

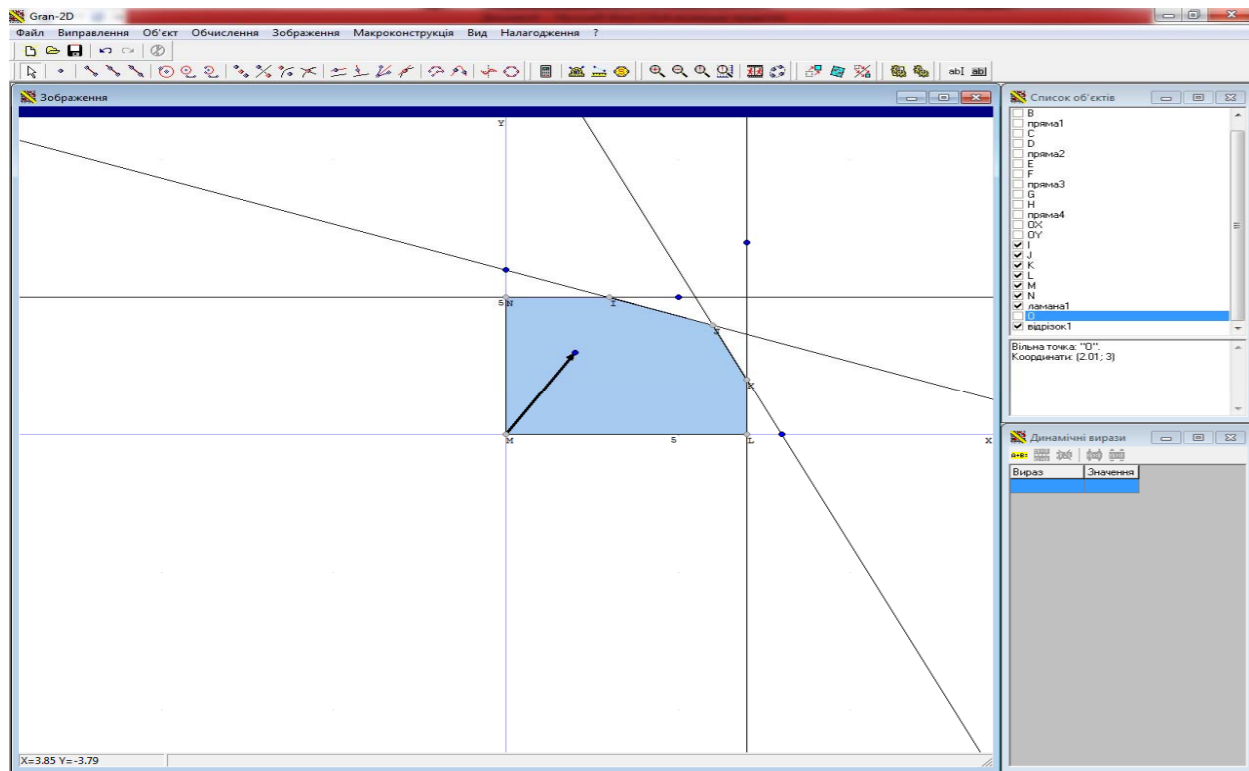
2.7. MS Excel

2.7.1. Побудова парної економетричної моделі із використанням можливостей табличного процесору EXCEL

Запустити табличний процесор Excel. Це можна зробити кількома способами:

- натиснути кнопку Пуск та в переліку Все програми вибрати назву Microsoft Excel;
- якщо на робочому столі є ярлик Microsoft Excel, то двічі клацніть на ньому лівою кнопкою миші.

При запуску Excel автоматично виводить на екран нову робочу книгу з умовним ім'ям Книга 1. Це ім'я з'являється праворуч від імені Microsoft Excel, його можна змінити на те ім'я, яке Вам потрібно.



До градієнта у точці (0,0) будемо перпендикулярну пряму використовуючи послугу «Об'єкт» → «Створення» → «Перпендикуляр до прямої».

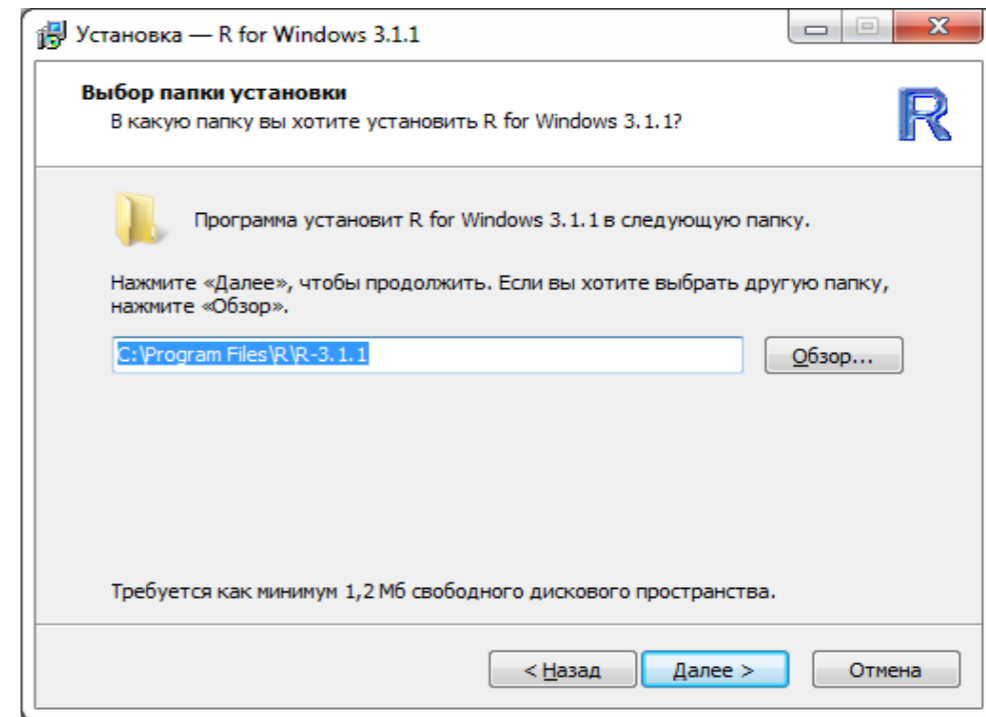
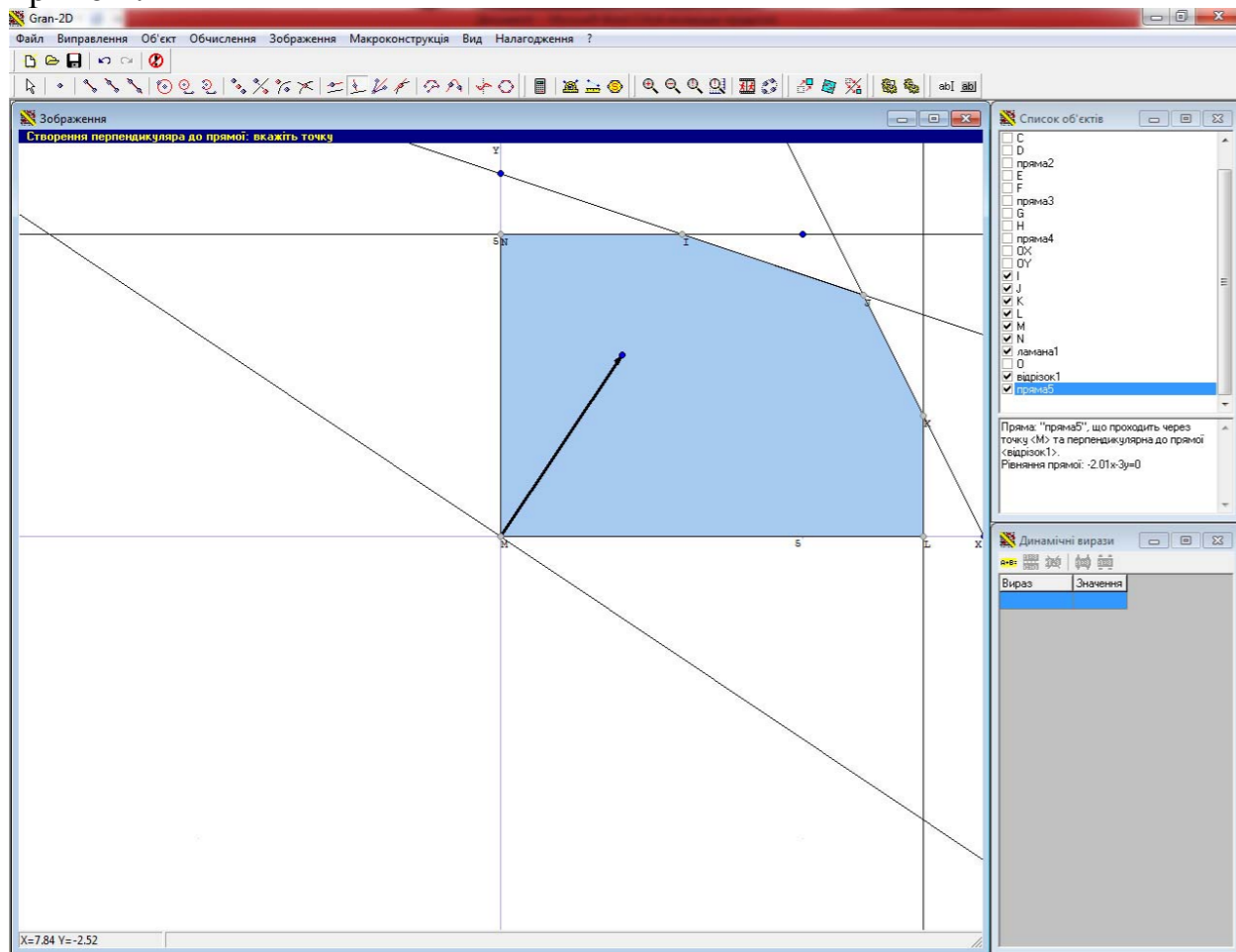


Рис. 2.4

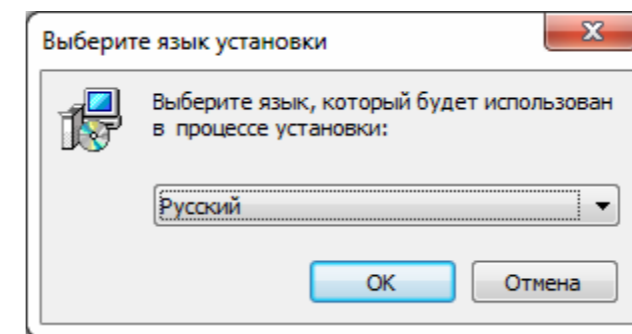


Рис. 2.5

- Виберіть «Користувачке встановлення» і натисніть «Далі» (дивись рис. 2.6).

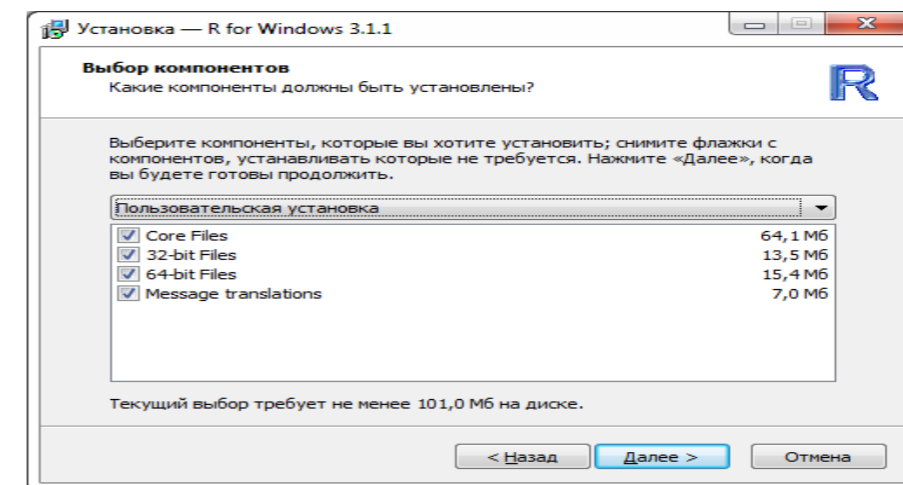


Рис. 2.6

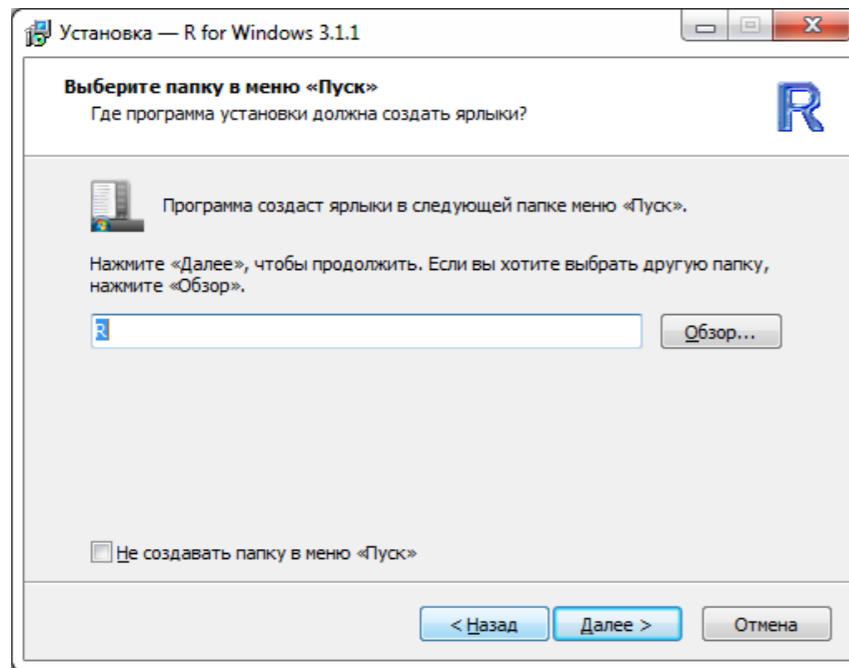


Рис.2.7

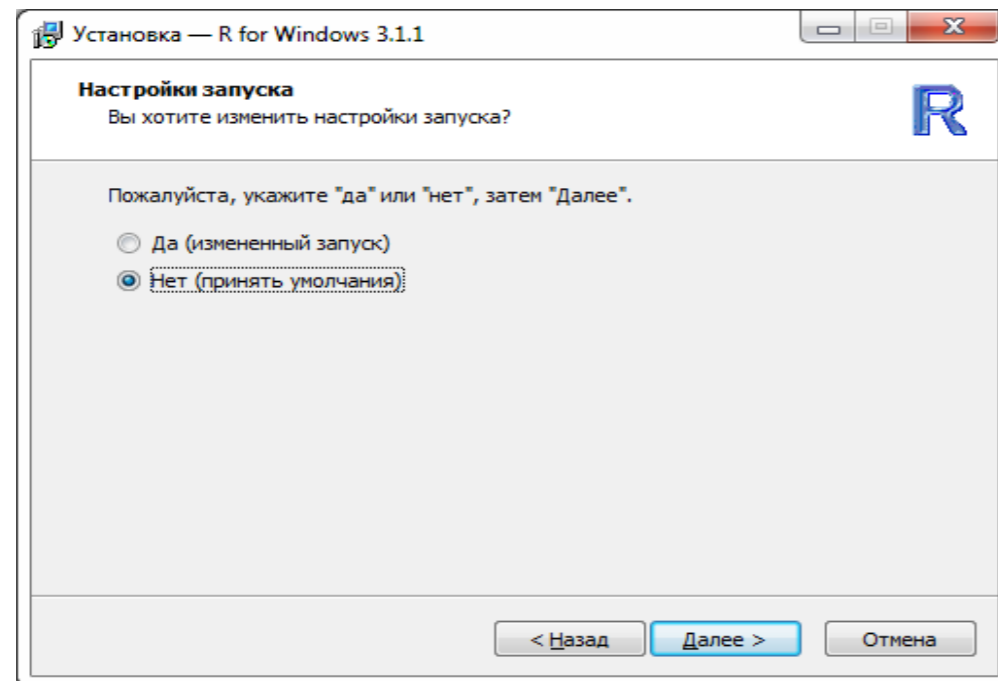
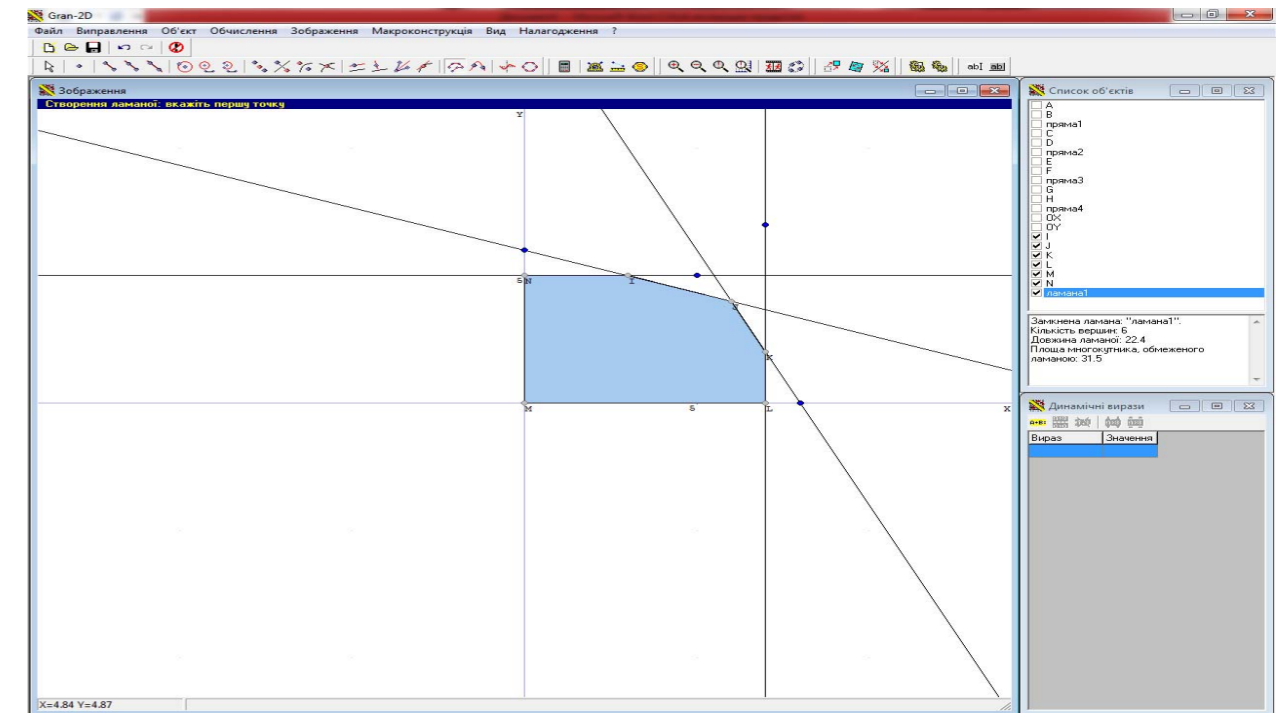


Рис.2.8

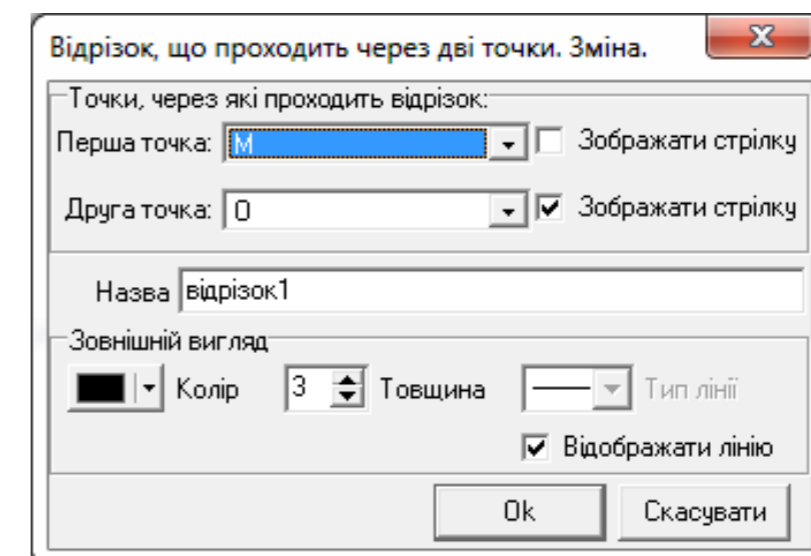
- Знову натисніть кнопку «Далі» (дивись рис. 2.7) .
- Знову натисніть кнопку «Далі» (дивись рис. 2.8) .
- Знову натисніть кнопку «Далі» (дивись рис. 2.9) .

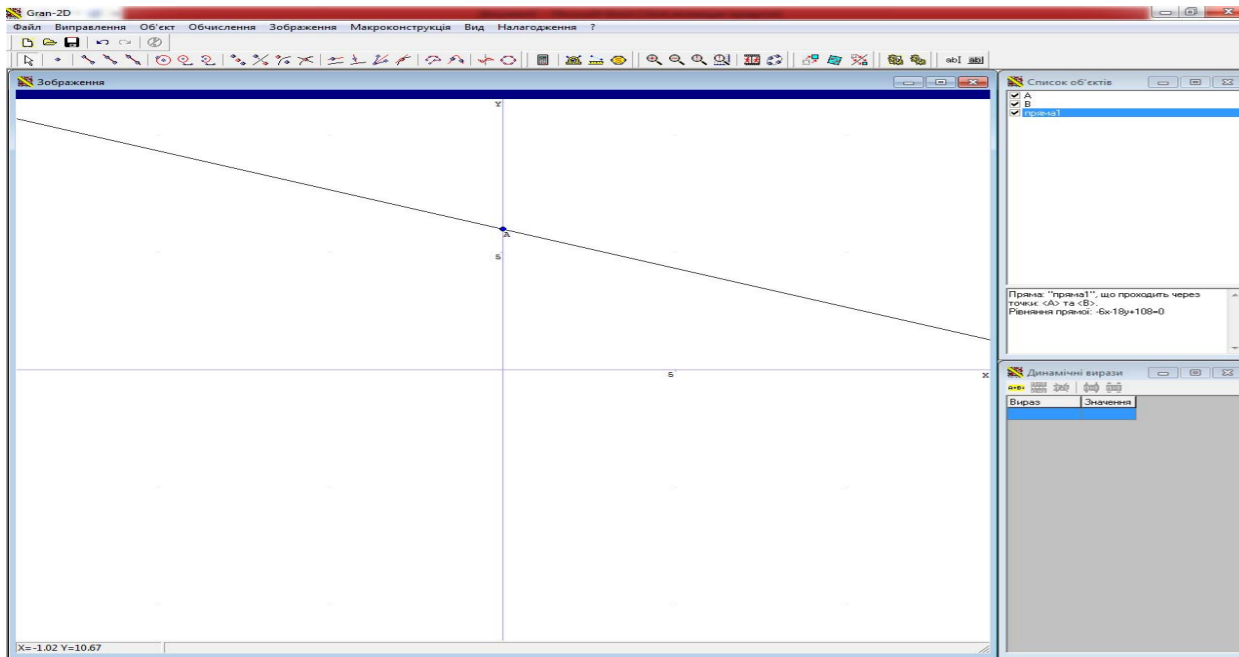
За допомогою GRAN – 2D можна легко дізнатися координати точок перетину прямих. Для цього необхідно використовуючи послугу «Об’єкт» → «Створення з екрану» → «Точка перетину об’єктів» обрати необхідні прямі і шукана точка додається до списку об’єктів.

Після побудови граничних прямих визначаємо область розв’язків системи нерівностей. Для виділення цієї області створимо замкнену ламану через потрібні точки. Для цього необхідно використовуючи послугу «Об’єкт» → «Створення з екрану» → «Ламана» обрати необхідні точки.



Тепер побудуємо вектор градієнта цільової функції $grad(F) = (2,3)$, який виходить з початку координат використовуючи послугу «Об’єкт» → «Створення» → «Відрізок».





Аналогічно побудуємо інші граничні прямі системи обмежень. І додаємо активні осі координат, через пункт меню «Об'єкт». Після цього отримуємо наступне зображення:

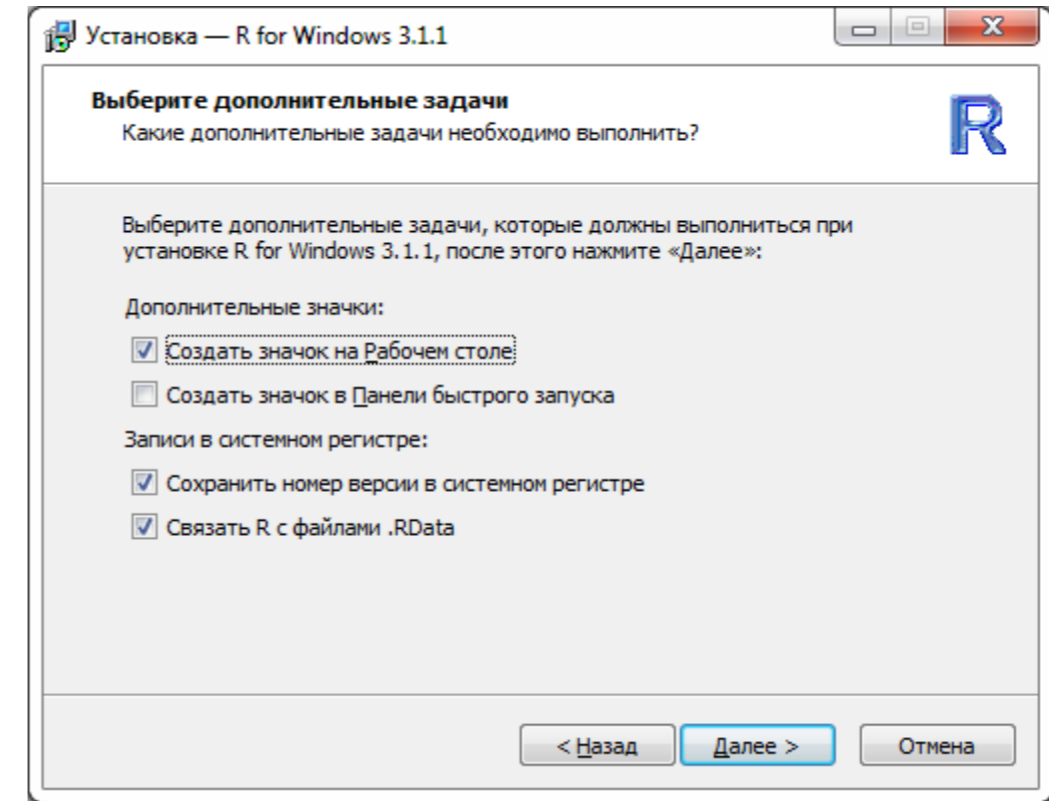
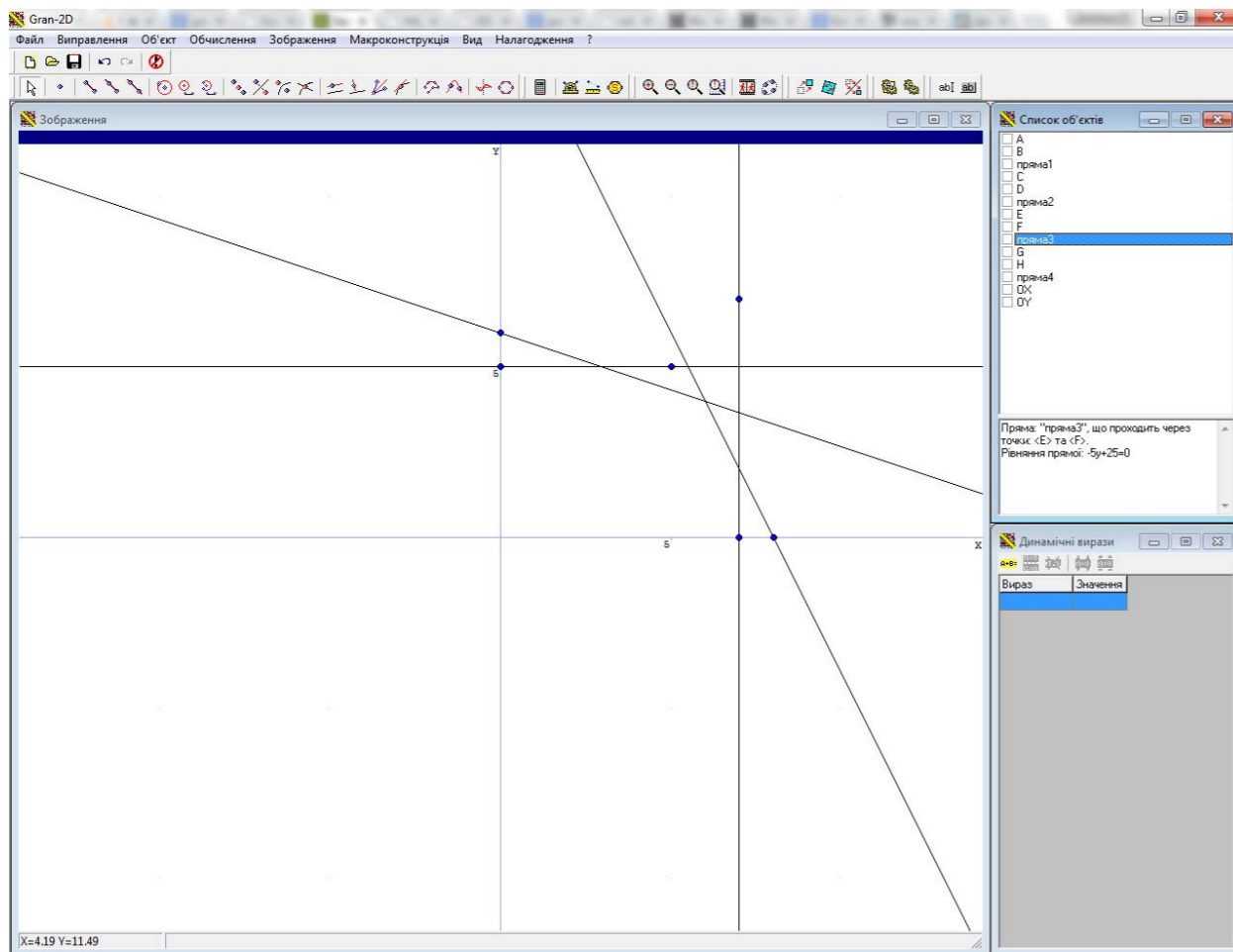


Рис. 2.9

- Дочекайтеся поки R встановиться на ваш комп'ютер (дивись рис. 2.10) .

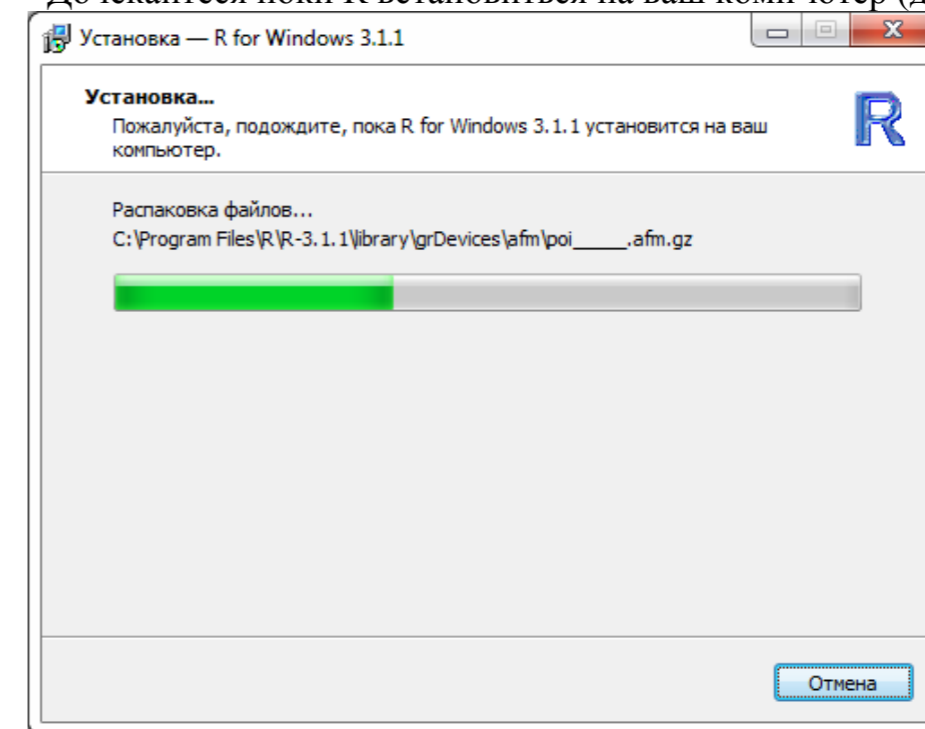


Рис. 2.10

- Натисніть кнопку «Завершити». R вже встановився на ваш комп'ютер (дивись рис. 2.11) .

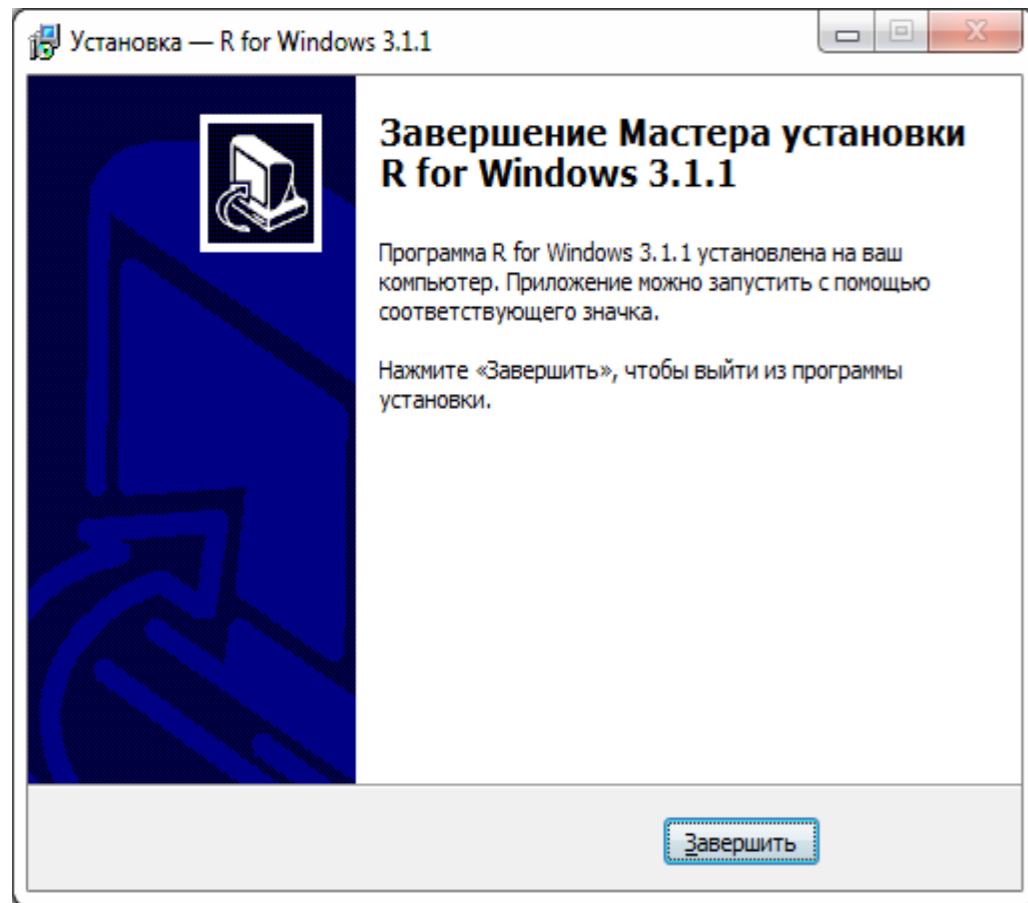


Рис. 2.11

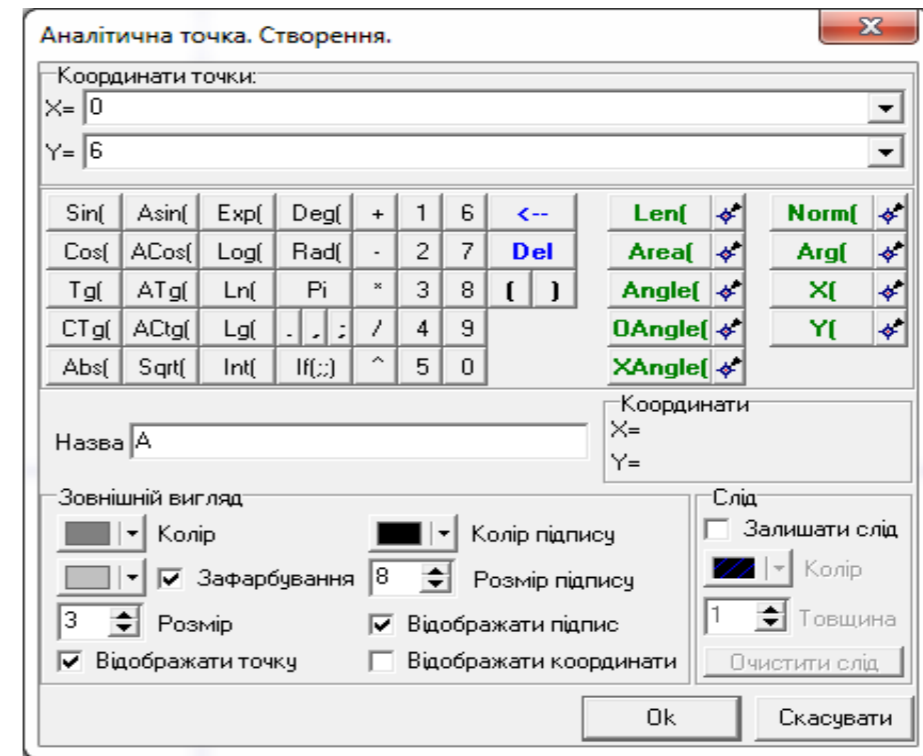
2.1.6. Встановлення R-Studio

Після завершення попереднього встановлення, ви повинні побачити іконку «R» на вашому робочому столі. Натиснувши на неї запуститься стандартний інтерфейс. Для більшої зручності, ми рекомендуємо використовувати інтерфейс RStudio. Крім того існує багато інших безкоштовних інтерфейсів, якими теж можна користуватися. Для встановлення RStudio на ваш комп'ютер, перейдіть по наступному посиланню:

<http://www.rstudio.org/>

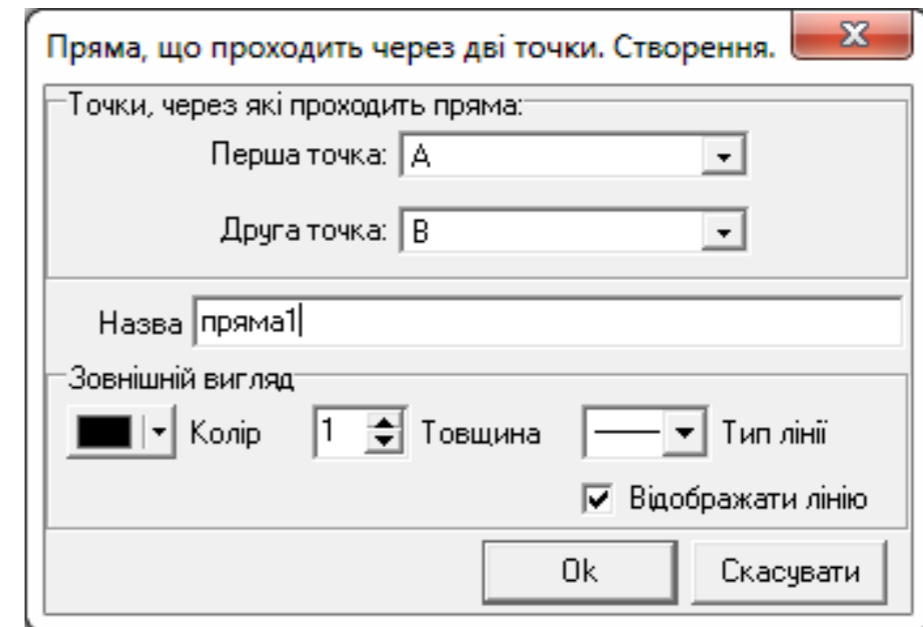
і виконайте наступні кроки (за умови, що ви працюєте на комп'ютері з операційною системою сімейства Windows):

- Натисніть «Download RStudio» (дивись рис. 2.12) .



Аналогічно додати точку з координатами (18, 0).

Далі створюємо пряму, що проходить через ці точки, використовуючи послугу «Об'єкт» → «Створення» → «Пряма»



У вікні що відкрилося, вибрати першу(A) і другу (B) точки через які проходить пряма, вибрати колір, тип та товщину лінії, і натиснути кнопку «ОК».

Після цього повинно з'явитися таке зображення графіка прямої.

$$\begin{cases} x_1 + 3x_2 \leq 18, (I) \\ 2x_1 + x_2 \leq 16, (II) \\ x_2 \leq 5, (III) \\ 3x_1 \leq 21, (IV) \\ x_1 \geq 0, x_2 \geq 0, (V, VI) \end{cases}$$

Розв'язок.

Зобразимо многокутник розв'язків за допомогою програми GRAN – 2D.

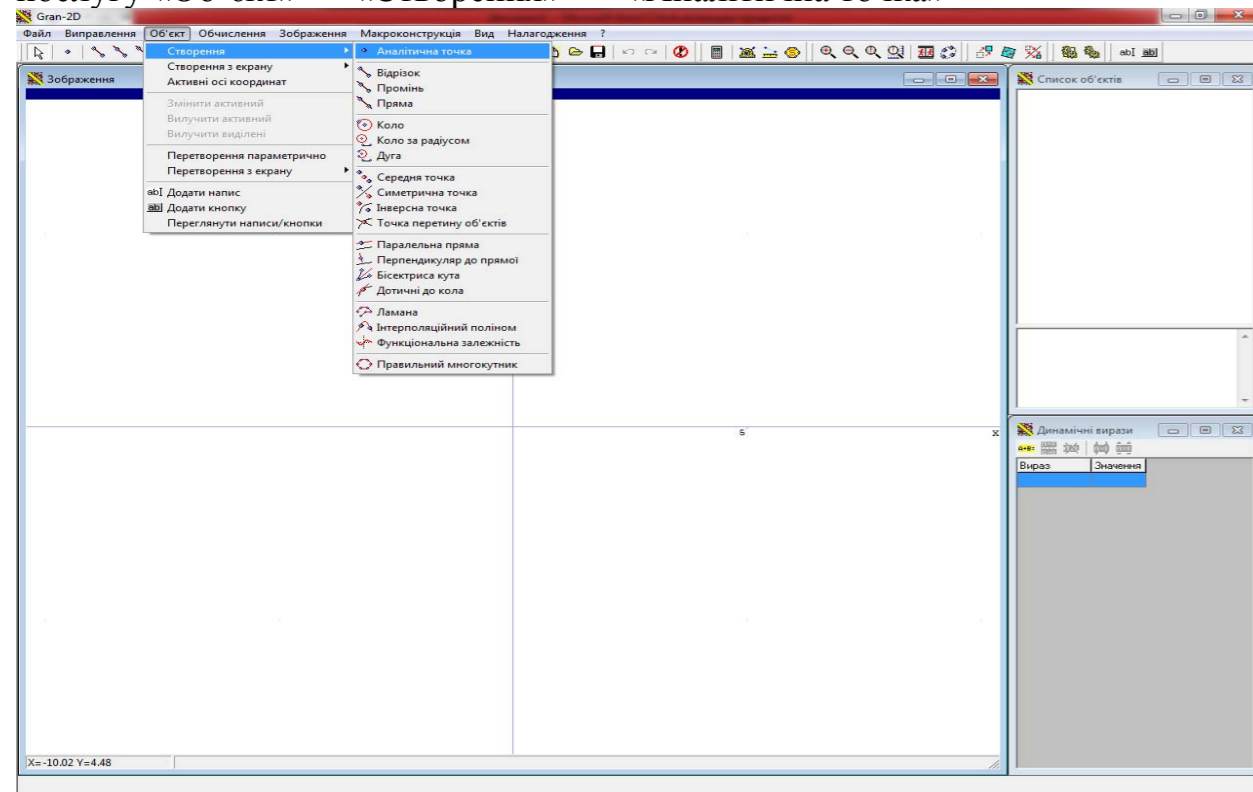
Щоб розв'язати систему нерівностей, спочатку побудуємо граничні прямі до кожного обмеження.

(I): $x_1 + 3x_2 = 18$

Для побудови прямої достатньо задати дві точки через які вона проходить.

x_1	0	18
x_2	6	0

За допомогою програми GRAN-2D створюємо точку, використовуючи послугу «Об'єкт» → «Створення» → «Аналітична точка»



У вікні що відкрилося, ввести координати точки (0, 6), вибрати необхідний колір, тип та розмір точки, і натиснути кнопку «ОК».

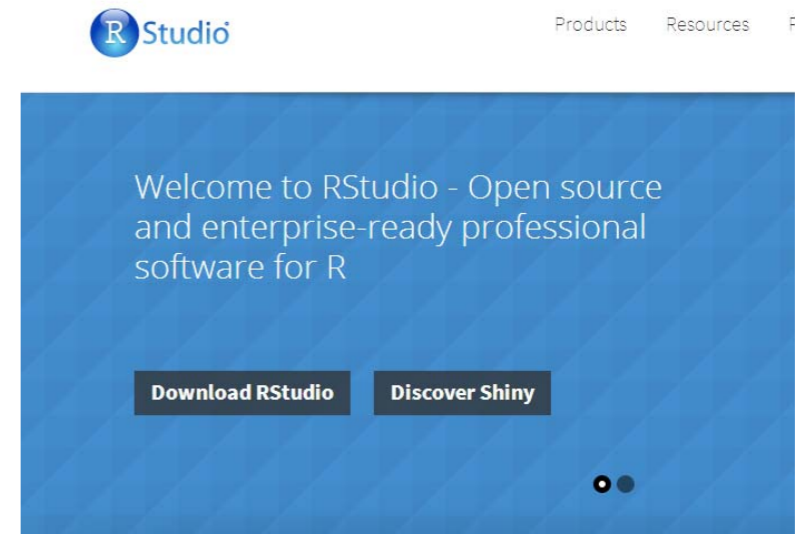


Рис. 2.12



Рис. 2.13

• Натисніть «Download RStudio Desktop» (дивись рис. 2.13) .

• Виберіть версію RStudio для операційної системи Windows (дивись рис. 2.14) .

• Завантажте .exe файл і запустіть його. Далі виберіть стандартні відповіді на всі запитання, аналогічно тим крокам, які виконувалися при встановленні R.

2.1.7. Огляд RStudio

Інтерфейс RStudio складається з декількох вікон (дивись рис. 2.15).

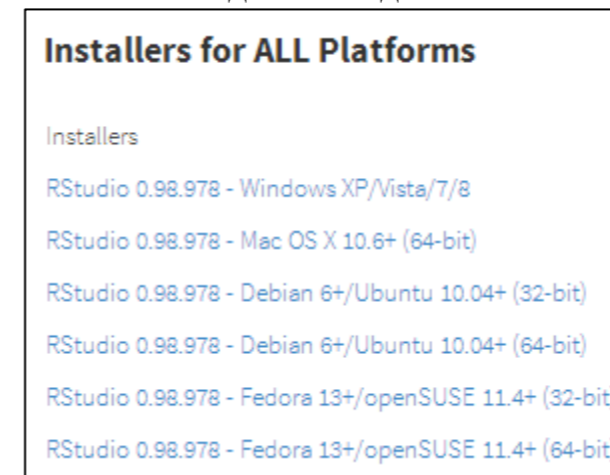


Рис. 2.13

З лівої сторони розташоване вікно консолі (яке ще можуть називати командним вікном). Тут ви можете писати різні команди після символу «>»

підказки і R виконає вашу команду. Це одне з найважливіших вікон, тому що це вікно де R насправді виконує всі розрахунки.

В правому лівому куті розташоване вікно робочого простору / історії. У вікні робочого простору ви можете побачити які дані та значення R тримає в пам'яті. Ви можете продивитись та редагувати значення натиснувши на них. Вікно історії показує що було введено раніше.

В правому нижньому куті розташоване вікно файлів / графіків / пакетів / довідки. Тут ви можете відкривати файли, продивляти графіки (також і попередні графіки), встановлювати та завантажувати пакети або використовувати функцію довідки.

Ви можете змінювати розмір вікон перетягуючи сірі смуги між вікнами.

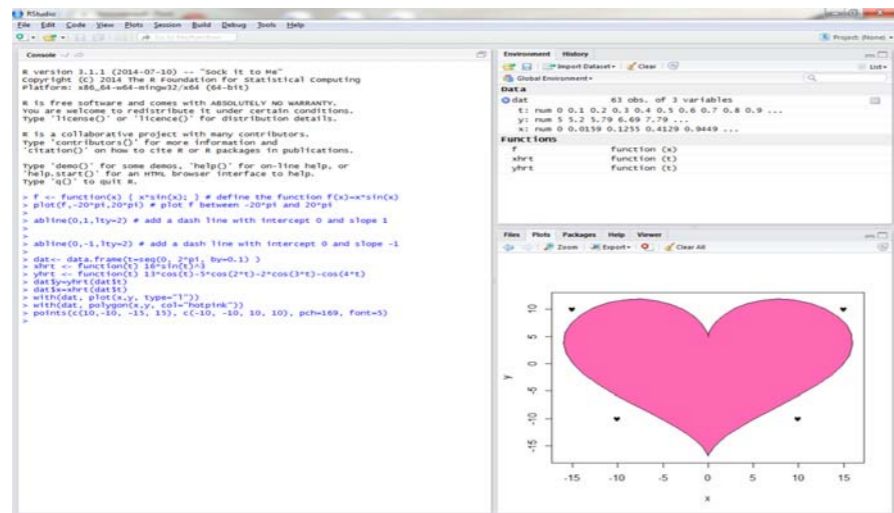


Рис. 2.14

2.1.8. Бібліотеки

R має значні можливості для здійснення статистичних аналізів, включаючи лінійну і нелінійну регресію, класичні статистичні тести, аналіз часових рядів (серій), кластерний аналіз і багато іншого. R легко розбудовується завдяки використанню додаткових так званих пакетів або бібліотек. Разом із стандартним встановленням, більшість найбільш вживаних пакетів вже встановлено.

Щоб отримати список всіх встановлених пакетів, перейдіть у вікно пакетів, або наберіть в консолі `library()`. Якщо навпроти імені пакету стоїть прапорець, це означає що даний пакет завантажений (активований) і може використовуватися.

Є ще дуже багато пакетів доступних на сайті R. Якщо ви хочете встановити і використовувати пакет (наприклад, пакет називається «geometry») ви повинні:

- Встановити пакет: натиснути «install» у вікні пакетів і ввести `geometry` або набрати `install.packages(«geometry»)` у командному вікні.
- Завантажити пакет: поставити прапорець навпроти `geometry` або набрати `library(«geometry»)` у командному вікні.

одержить $P_{старе} - P_{нове} = 140 - 132,5 = 7,5$ (гр.од.), тоді споживач – $10,7 - 7,5 = 2,5$ (гр.од.).

Розв'язуючи цю задачу аналітично, ми одержали такі ж результати, але часу пішло на це значно більше. Крім того, що на занятті економиться час, студент одержує можливість зосередити увагу саме на економічних аспектах задачі (так як навички складати, перетворювати та розв'язувати рівняння студент вже набув на попередніх уроках математики).

Тобто це ще раз підтверджує педагогічну доцільність використання інформаційно-комунікаційних технологій у процесі навчання математики.

2.6.2. GRAN – 2D

На сьогодні розроблена значна кількість програмних засобів, що дають змогу розв'язувати за допомогою комп'ютера досить широке коло математичних задач різних рівнів складності. Слід зазначити, що вища математика вивчається на молодших курсах. І тут доцільне використання пакета програм GRAN, з яким студенти працювали ще під час вивчення курсу математики в школі.

Приклад 1.

Задача про використання ресурсів (планування виробництва).

Для виготовлення двох видів продукції P_1 і P_2 використовують чотири види ресурсів S_1, S_2, S_3, S_4 . Запаси ресурсів, число одиниць ресурсів, які використовують на виготовлення одиниці продукції наведені в таблиці

Вид ресурса	Запас ресурса	Число одиниць ресурсів, які використовують на виготовлення одиниці продукції	
		P_1	P_2
S_1	18	1	3
S_2	16	2	1
S_3	5	–	1
S_4	21	3	–

Прибуток, який отриманий від реалізації одиниці продукції P_1 і P_2 , відповідно складає 2 і 3 умовні одиниці. Необхідно скласти такий план виробництва продукції, при якому прибуток від її реалізації буде максимальним.

Розв'язати графічним методом.

$$F = 2x_1 + 3x_2 \rightarrow \max$$

За умов

Яка ситуація складеться на ринку в даному випадку? На ринку виникає ситуація обернена до попередньої. Тобто в результаті таких дій держави на ринку виникне надлишок товару в розмірі $290-250=40$ (одиниць товару).

За умовою задачі державою вводиться податок в розмірі 10 гр.од. на одиницю товару. Як та на яку з функцій вплине дана дія держави? Так як податок є неціновим фактором пропозиції (це відомо учням з уроків основ економіки), то її нове рівняння буде наступним: $Q_s = 3(P-10)-160$.

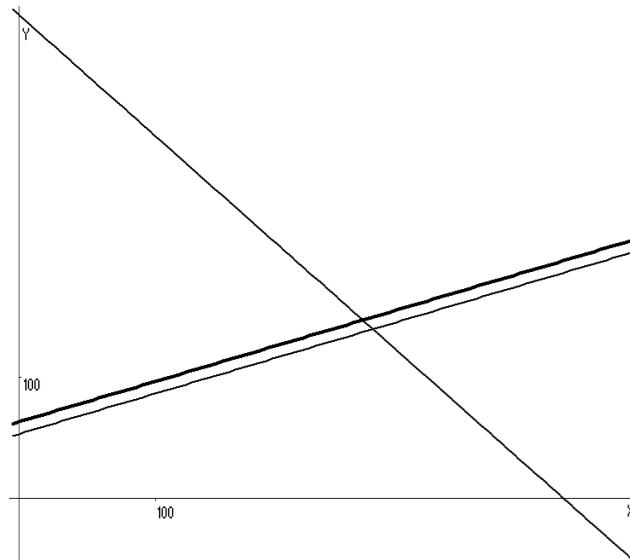


Рис. 2.137

Перетворимо рівняння відносно Q : $P = \frac{Q_s + 190}{3}$. Побудуємо попереднє та нове рівняння пропозиції та визначимо нові параметри рівноваги (рис.2.137). Скориставшись послугою «M_» знайдемо нові параметри рівноваги. Вони будуть дорівнювати: $Q=252,5$, $P = 147,5$.

Податок між споживачем та виробником розподілиться так: споживач буде сплачувати $P_{нова} - P_{старе} = 147,5 - 140 = 7,5$ (гр.од.), тоді виробник – $10 - 7,5 = 2,5$ (гр.од.).

Якщо держава надає дотацію виробнику у розмірі 10 гр.од. на одиницю товару, то що відбудеться на ринку. З економіки студенти знають, що дотація є неціновим фактором пропозиції. Тому зміниться тільки функція пропозиції. Нове рівняння пропозиції: $Q_s = 3 \cdot (P+10) - 160$.

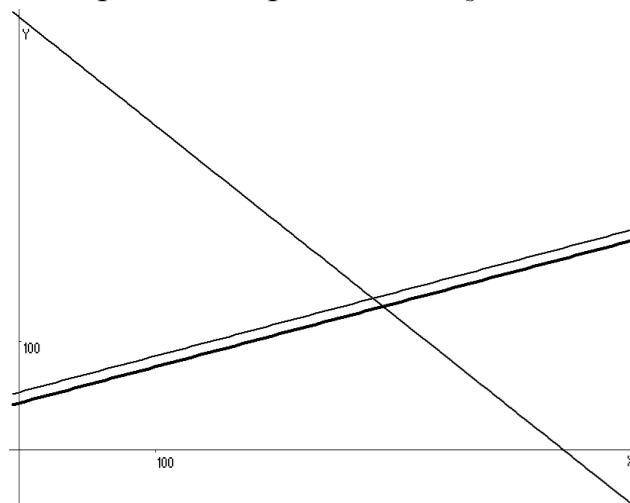


Рис. 2.138

Перетворимо рівняння відносно Q : $P = \frac{Q_s + 130}{3}$. Побудуємо попереднє та нове рівняння пропозиції (рис.2.138). Визначимо нові параметри рівноваги за допомогою послуги «M_»: $Q=267,5$, $P = 132,5$.

Дотація між споживачем та виробником розподілиться так: виробник

R-код дуже легкий для сприйняття, як тільки ви розумієте, що:

- «<->» це оператор присвоєння.
- Все, починаючи з «#», є коментарем.
- Мова R враховує регістр: «a» і «A» це два різні об'єкти.

Консоль

Команди можна вводити у вікні консолі R в командному рядку (>) і використовувати R в якості потужного наукового калькулятора.

Наприклад, ввівши у вікні консолі:

```
> pi*2.234^5; 43*5/sqrt(56); log(36)
```

Ми отримаємо відповідь

```
[1] 174.8098
```

```
[1] 28.73058
```

```
[1] 3.583519
```

Тут sqrt і log – вбудовані функції в R; pi – вбудована стала; крапка з комою (;) використовується для відокремлення R-команд.

У вікні консолі, <ВГОРУ> і <ВНИЗ> клавіші зі стрілками використовуються для переміщення між раніше введеними пропозиціями.

Отримання довідки, приклади, демонстрації

R має великий центр довідки. Крім вікна довідки, довідка також доступна з командного рядка консолі. Щоб отримати довідку про певну функцію необхідно лише ввести символ «?» перед її назвою.

Наприклад, набравши

```
> ?log
```

```
> ?sin
```

```
> ?sqrt
```

Отримаємо пояснення про функцію логарифмів і експонент, тригонометричних функцій, і інші функції.

А набравши

```
> ?Arithmetic
```

Отримаємо перераховані арифметичні операції в R.

Більшість файлів з довідкою також включають приклади. Ви можете запустити всі з них за допомогою R-команди «example».

Наприклад, набравши у вікні консолі:

> example(matrix)

виконуються всі приклади з файлу допомоги «matrix».

> example(pairs)

виконуються всі приклади з файлу допомоги «pairs». (Спробуйте це – «pairs» є дуже потужним засобом візуалізації парних відносин).

Крім того, ви можете вибрати один приклад з довідки, скопіювати його в буфер обміну (Ctrl-C для користувачів Windows), а потім вставити його (Ctrl-V) у вікно консолі.

Крім того, основне програмне забезпечення R та багато R-пакетів містять демонстраційний матеріал.

Введіть

> demo()

це дасть список доступних демонстрацій в головному програмному забезпеченні.

> demo(graphics)

продемонструє кілька простих графічних засобів.

Потрібно пам'ятати

1. Імена шляхів в R записуються з косою рисою «/», хоча в ОС Windows(®), використовуються зворотні косі риси, «\». Таким чином, щоб встановити робочий каталог в R:

> setwd(«C:/R_Code/»)

2. Якщо вираз на одній лінії синтаксично правильний, R буде виконувати його, навіть якщо цей вираз продовжується на наступному рядку. Наприклад, якщо ми пишемо:

```
A <- 5 + cos(pi)
-sqrt(7)
[1] -2.645751
```

То A буде мати значення (5 + cos (pi)) і R виведе значення (-sqrt(7)).

На відміну від цього, в наступних рядках:

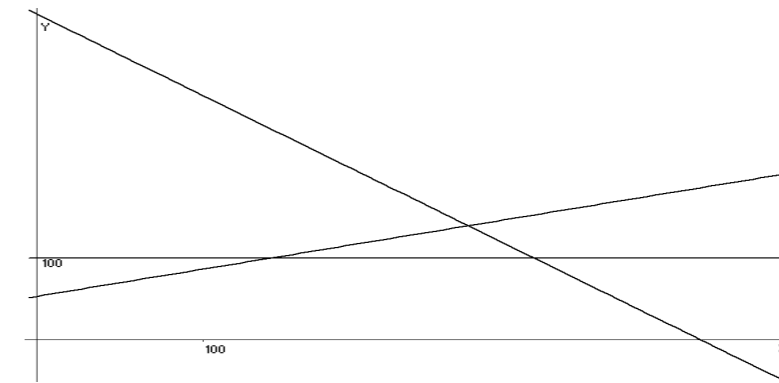


Рис. 2.132

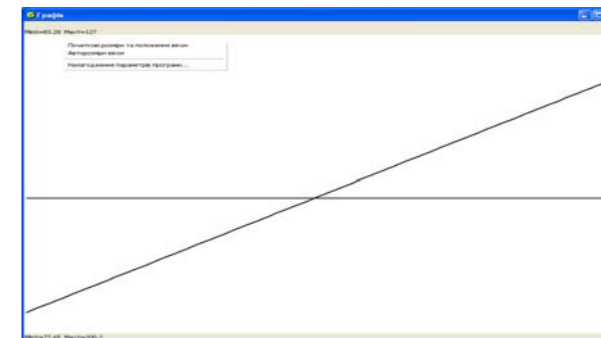


Рис. 2.133

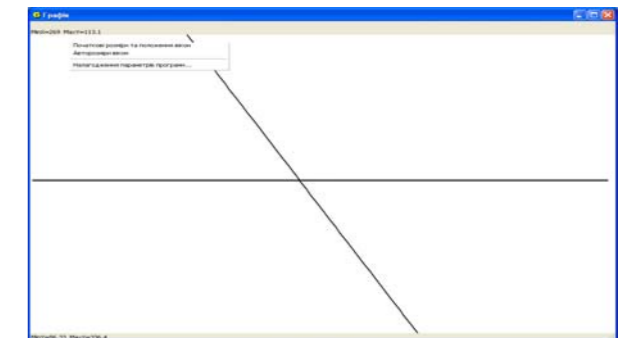


Рис. 2.134

З рис.2.133 $Q_s = 140$, а з рис.2.134 $Q_d = 300$. Який економічний зміст отриманих даних?

Кількість товарів, яку готові придбати споживачі за ціною 100 гр.од., перевищує кількість товарів, яку готові продати за цією ж ціною продавці.

Який висновок щодо дій держави та наслідків державного втручання ми можемо зробити? Отже, в результаті таких дій держави на ринку виникне дефіцит товару в розмірі: $300 - 140 = 160$ (одиниць товару).

Розглянемо випадок, коли держава ввела фіксовану ціну на рівні 150 гр. од. В одній системі координат побудуємо пряму $P = 150$ і графіки функцій попиту та пропозиції, знайдемо координати точок перетину (рис.2.135, 2.136). За рис.2.135 $Q_s = 290$, а рис.2.12.136 $Q_d = 250$.

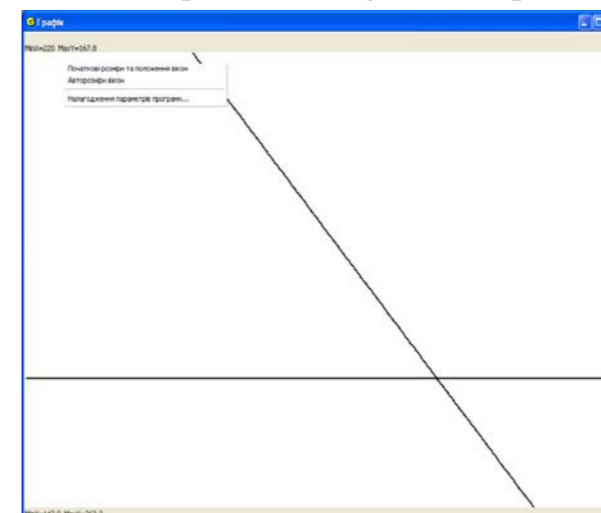


Рис. 2.135

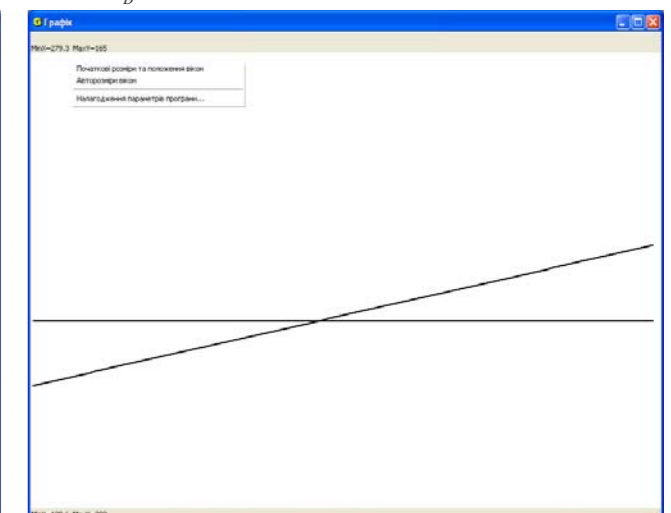


Рис. 2.136

Задача 2. Попит на товар задано рівнянням $Q_D = 400 - P$, а пропозиція – $Q_S = 3 \cdot P - 160$. Виконайте такі завдання:

1) побудуйте графіки попиту та пропозиції та визначте рівноважну ціну і обсяг продажу;

2) опишіть ситуацію, яка виникне на ринку, якщо держава:

а) введе фіксовану ціну на товар розміром 100 гр.од., а потім 150 гр.од., проілюструйте графічно;

б) як зміняться параметри рівноваги, якщо держава введе податок на виробництво товару у розмірі 10 гр.од. на одиницю товару (як розподілиться цей податок між виробником та споживачем);

в) як зміняться параметри рівноваги, якщо держава надає виробнику дотацію у розмірі 10 гр. од. на одиницю товару (як розподілиться цей податок між виробником та споживачем).

Розв'язання. Перед тим, як почати будувати графіки, доцільно нагадати студентам про те, що в економіці прийнято відкладати ціну P (незалежна змінна) по вертикальній осі координат, а обсяг попиту Q або інші параметри, що залежні від ціни – по горизонтальній осі. З точки зору математики, це необхідно було б зробити навпаки: незалежну змінну відкласти по горизонталі, а залежну – по вертикалі. Тому під час роботи в GRAN 1 ми припускаємо, що $x = Q$ (обсяг товару чи послуги), а $y = P$ (ціна).

Побудуємо в GRAN 1 графіки функцій попиту та пропозиції та знайдемо координати точки їх перетину (точку рівноваги). Для цього спочатку перетворимо рівняння попиту та пропозиції: $P = 400 - Q_D$, $P = \frac{Q_S + 160}{3}$ (рис.2.131).

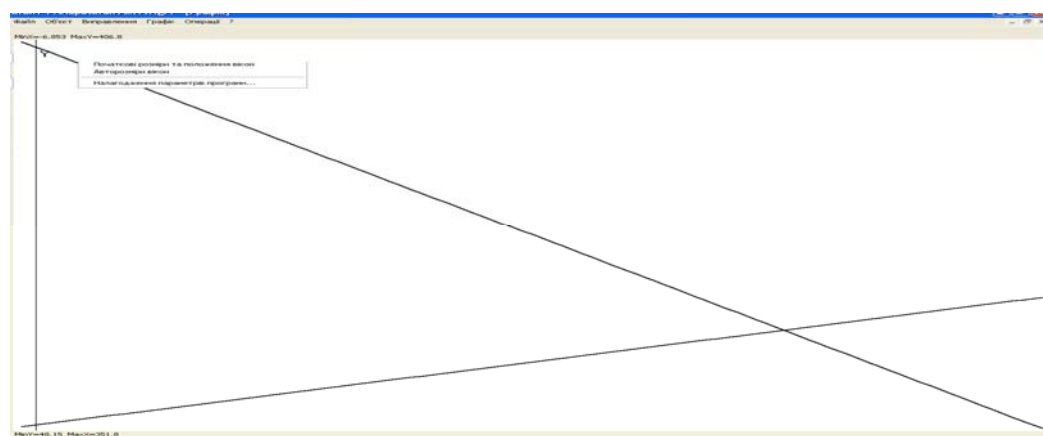


Рис. 2.131

1) Точка рівноваги: $P = 140$ гр.од., $Q = 260$

2) За умовою задачі держава встановила ціну на рівні 100 гр.од.

Побудуємо пряму $P = 100$ грн. (рис.2.132), знайдемо координати точок перетину (рис. 2.133, 2.134).

$5 + \cos(\pi) -$

$\sqrt{7}$

[1] 1.354249

R буде виводити значення $5 + \cos(\pi) - \sqrt{7}$: оскільки вираз на першій лінії не є синтаксично завершеним, R (правильно) припускає, що він буде продовжений на наступному рядку.

Будьте обережні, якщо ви хочете розділити складний вираз на кілька рядків! Ці помилки дуже важко простежити, так що краще, уникати їх.

2.1.9. Використання R як калькулятора

Дуже зручно використовувати R в якості потужного калькулятора. Найкраще це можна зробити використовуючи R-консоль.

Використовуючи консоль, обчисліть значення виразів:

Приклад 1. $\left(\frac{4}{6} \cdot 8 - 1\right)^{\frac{2}{3}}$

> $(4/6*8-1)^(2/3)$

[1] 2.657958

Приклад 2. $\ln(20)$

> $\log(20)$

[1] 2.995732

Приклад 3. $\log_2(4096)$

> $\log2(4096)$

[1] 12

Приклад 4. $2 \cdot \pi \cdot 3$

> $2 * \pi * 3$

[1] 18.84956

Приклад 5. $e^{2+\cos(0.5\pi)}$

> $\exp(2+\cos(0.5*\pi))$

[1] 7.389056

Приклад 6. $\sqrt{2.3^2 + 5.4^2 - 2 \cdot 2.3 \cdot 5.4 \cdot \cos\left(\frac{\pi}{8}\right)}$

> # length of 3rd side of a triangle with size 2.3 and 5.4 and angle pi/8

> $\sqrt{2.3^2+5.4^2-2*2.3*5.4*\cos(\pi/8)}$

[1] 3.391288

Порада: вам може знадобиться подивитися на файли довідки для деяких з цих функцій. Щоб це зробити, наберіть ? «+» . Це відкриє файл довідки із загальними арифметичними операторами.

2.1.10. Числа, вектори, матриці та масиви

Присвоєння

Коли використовуються змінні, вони повинні бути ініціалізовані числами.

```
> A <- 1
> B <- 2
> A + B
[1] 3
```

R може приймати в якості аргументів функцій числа, вектори, матриці або масиви.

```
> V <- factorial(10)
```

Виконується обчислення $10!$ ($= 1 * 2 * 3 * 4 * 5 * \dots * 10$). Потім оператор «<-» присвоює результат цього обчислення змінній V. V може потім бути використана в подальших розрахунках:

```
> V/10
[1] 362880
```

Зверніть увагу, що присвоєння значення змінній V не відображає його у вікні. Для відображення значення V ми просто повинні написати:

```
> V
[1] 3628800
```

Крім того, ми можемо присвоїти результат розрахунків змінної і переглянути результат, взявши вираз в дужки:

```
> (X <- sin(3/2*pi))
[1] -1
```

Окрім цілих, дійсних і комплексних чисел, R також розпізнає нескінченність (Inf) і значення, які є невизначеними (NaN). Спробуйте:

```
> 1/0
[1] Inf
```

```
> 0/0
[1] NaN
```

```
> 1e-8 * 1000
[1] 1e-05
```

(Де «e-8» позначення позначає 10^{-8}).

Вектори

Вектори складаються з упорядкованого набору чисел і можуть бути створені різними способами:

- Використовуючи R-функцію «vector»

Здійснене нами дослідження показало, що при рівні виробництва 10 або 50 дохід дорівнює витратам, а прибуток дорівнює нулю. Це означає, що компанія знаходиться в точці беззбитковості. Для рівня виробництва $10 < Q_D < 50$ дохід перевищує витрати, і відповідно компанія має прибуток. Для всіх інших значень Q_D , тобто для $Q_D < 10$ та $Q_D > 50$ компанія несе збитки.

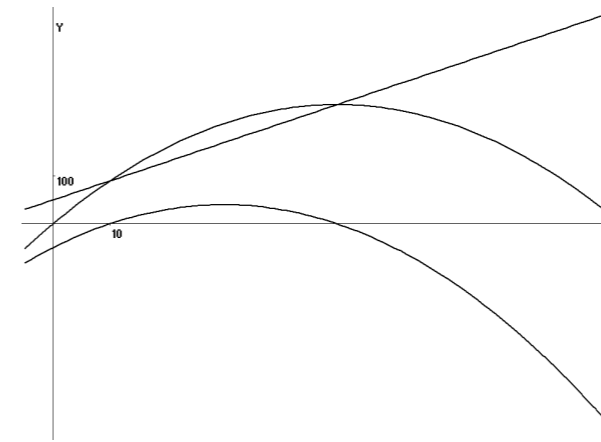


Рис. 2.129

Проведемо дослідження функції прибутку. Для цього нам необхідно повернутись до попереднього зображення, виконавши операцію «Побудувати». Відомо, що прибуток (Z) дорівнює: $Z = TR - TC$. На рис.2.129 зображена функції загального доходу, витрат та прибутку.

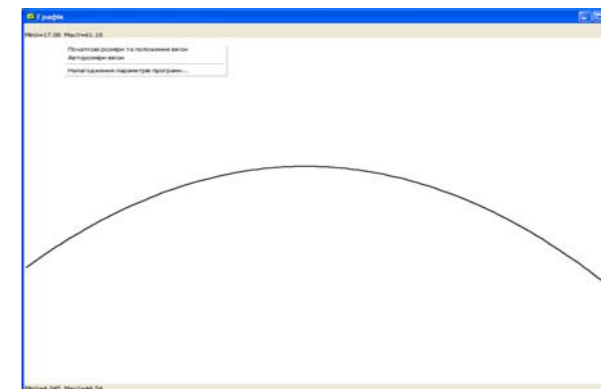


Рис. 2.130

Визначимо координати точки максимуму функції Z (тобто знайдемо максимальний прибуток). Це можна зробити шляхом збільшення графіку функції Z (див. рис. 2.130). Граничним прибутком буде точка $Q_D = 30$.

Це говорить про те, що прибуток збільшуватиметься при зростанні виробництва, якщо рівень виробництва менше 30, і прибуток зменшуватиметься при зростанні виробництва, якщо рівень виробництва більше 30. При $Q_D = 30$ одиниць продукції прибуток не змінюватиметься при «невеликій» зміні обсягу виробництва.

Отриманий результат з використанням GRAN 1 також можна одержати і аналітичним шляхом, але на розв'язування аналітичним шляхом у студентів піде значно більше часу. Також більша ймовірність того, що студенти „заплутаються”, роблячи висновок про збільшення або зменшення прибутку при зміні обсягу виробництва.

Отже, після проведених досліджень можна зробити висновок, що найкращим рівнем виробництва, який дає максимальний прибуток, є 30 одиниць товару, а максимальний обсяг попиту складає 50 одиниць товару, після чого настає перенасичення ринку цим товаром і дохід компанії починає зменшуватись.

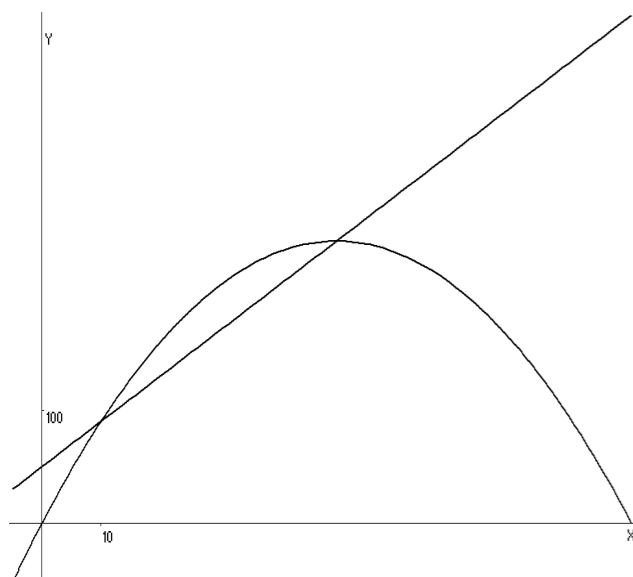


Рис. 2.126

Для того, щоб визначити точку (або точки) безбитковості для наших даних, необхідно знайти координати точок перетину графіків функцій. На рис.2.126 графіки функції загального доходу TR та загальних витрат TC мають дві точки перетину, тобто існує дві точки безбитковості.

Однак, для того, щоб більш точно визначити координати точок перетину, необхідно скористатись послугою «М». Зміна масштабу, в якому будуються графіки, призводить до збільшення точності обчислень в околі точки, яку досліджують. Вгорі на екрані можна побачити координати точки перетину.

На рис. 2.127 за допомогою цих операцій було визначено координати першої точки перетину, а на рис. 2.128 – другої.

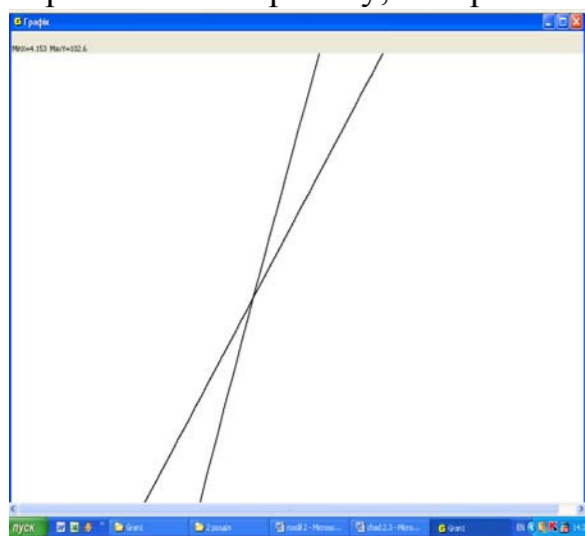


Рис. 2.127

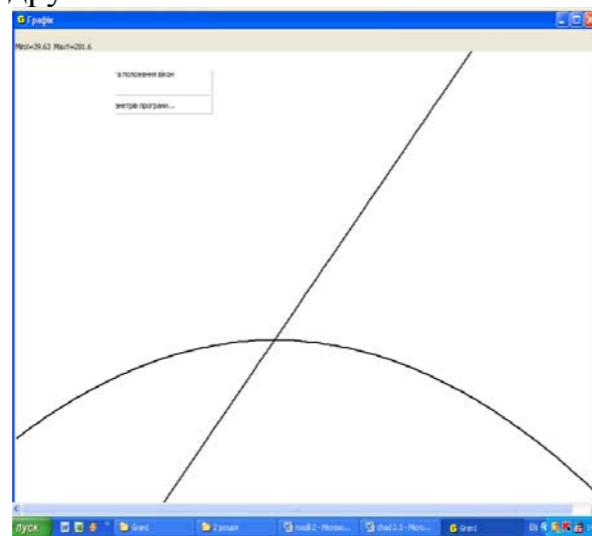


Рис.2.128

Ці точки перетину звичайно можна знайти і аналітичним шляхом:

$$50 + 4 \cdot Q_D = 10 \cdot Q_D - \frac{Q_D^2}{10}$$

Звідси після перетворень одержимо, що

$$Q_{D1}=10, Q_{D2}=50. \text{ Тоді } TR(10) = 90, TC(10) = 90.$$

$$\text{Аналогічно } TR(50) = TC(50) = 250.$$

Порівнюючи результати одержані за допомогою GRAN1 та аналітичним способом, можна зробити висновок, що за допомогою GRAN 1 студент може визначити координати точки з необхідною точністю.

•Функція «c()» поєднує числа у вектор (це, напевно, найважливіша функція в R)

•Оператор «:» створює послідовність значень, кожне наступне більше (або менше), ніж попереднє

•Більш загальна послідовність може бути створена за допомогою R-функції «seq»

Наприклад:

```
> c(0, pi/2, pi, 3*pi/2, 2*pi)
[1] 0.000000 1.570796 3.141593 4.712389 6.283185
```

```
> seq(from = 0, to = 2*pi, by = pi/2 )
[1] 0.000000 1.570796 3.141593 4.712389 6.283185
```

```
> seq(0, 2*pi, pi/2)
[1] 0.000000 1.570796 3.141593 4.712389 6.283185
```

Всі ці команди створять вектор, що складається з: 0, π , ... $2 * \pi$.

Зверніть увагу, що R-функція «seq» приймає в якості вхідних параметрів «from» (початкове значення), «to» (кінцеве значення) і «by» (крок) (другий приклад). Якщо порядок зберігається, можна не вказувати ім'я параметрів (третій приклад).

Наступна команда обчислює синус цього вектора і виводить результат:

```
> sin(seq(0, 2*pi, pi/2))
[1] 0.000000e+00 1.000000e+00 1.224606e-16 -1.000000e+00 -2.449213e-16
```

Наступний вираз:

```
> V <- 1:100
> sqrt(V)
[1] 1.000000 1.414214 1.732051 2.000000 2.236068 2.449490 2.645751
[8] 2.828427 3.000000 3.162278 3.316625 3.464102 3.605551 3.741657
[15] 3.872983 4.000000 4.123106 4.242641 4.358899 4.472136 4.582576
[22] 4.690416 4.795832 4.898979 5.000000 5.099020 5.196152 5.291503
[29] 5.385165 5.477226 5.567764 5.656854 5.744563 5.830952 5.916080
[36] 6.000000 6.082763 6.164414 6.244998 6.324555 6.403124 6.480741
[43] 6.557439 6.633250 6.708204 6.782330 6.855655 6.928203 7.000000
[50] 7.071068 7.141428 7.211103 7.280110 7.348469 7.416198 7.483315
[57] 7.549834 7.615773 7.681146 7.745967 7.810250 7.874008 7.937254
[64] 8.000000 8.062258 8.124038 8.185353 8.246211 8.306624 8.366600
[71] 8.426150 8.485281 8.544004 8.602325 8.660254 8.717798 8.774964
[78] 8.831761 8.888194 8.944272 9.000000 9.055385 9.110434 9.165151
[85] 9.219544 9.273618 9.327379 9.380832 9.433981 9.486833 9.539392
[92] 9.591663 9.643651 9.695360 9.746794 9.797959 9.848858 9.899495
```

[99] 9.949874 10.000000

створює послідовність цілих чисел від 1 до 100 і обчислює квадратний корінь з усіх з них, і відображає результат на екран.

Оператор «<-» присвоює послідовність в V.

Деякі інші приклади з використанням оператора «<»:

```
> (X <- 0.5:10.5)
```

```
[1] 0.5 1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5 9.5 10.5
```

```
> 6:1
```

```
[1] 6 5 4 3 2 1
```

Примітка: Особливістю R є те, що елементам вектора можна також дати власні назви:

```
> (fruit <- c(banana = 1, apple = 2, orange = 3))
```

```
banana apple orange  
 1     2     3
```

```
> names(fruit)
```

```
[1] "banana" "apple" "orange"
```

Матриці

Матриці також можуть бути створені декількома способами:

- За допомогою R-функції «matrix»
- За допомогою R-функції «diag», яка будує діагональну матрицю
- Функції «cbind» і «rbind» додають стовпці і рядки до існуючої матриці

або в інший вектор

Вираз:

```
> A <- matrix(nrow = 2, data = c(1, 2, 3, 4))
```

створює матрицю A, з двома рядками, і, оскільки є чотири елементи, двома стовпчиками. Відзначимо, що дані вводяться як вектор (за допомогою функції c()).

Наступні два вирази відображають матрицю A та матрицю квадратних коренів з її елементів:

```
> A
```

```
  [,1] [,2]  
[1,]  1  3  
[2,]  2  4
```

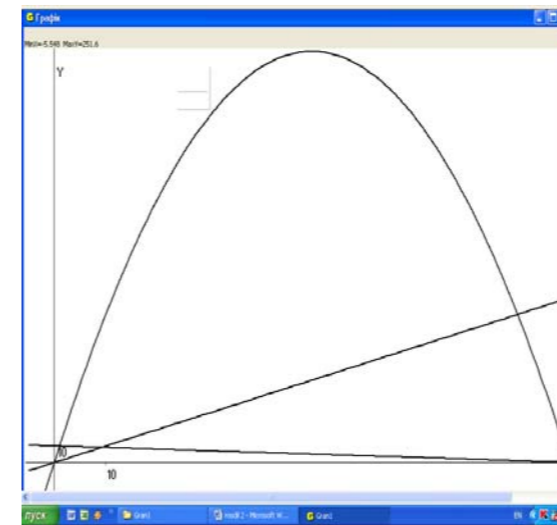


Рис. 2.124

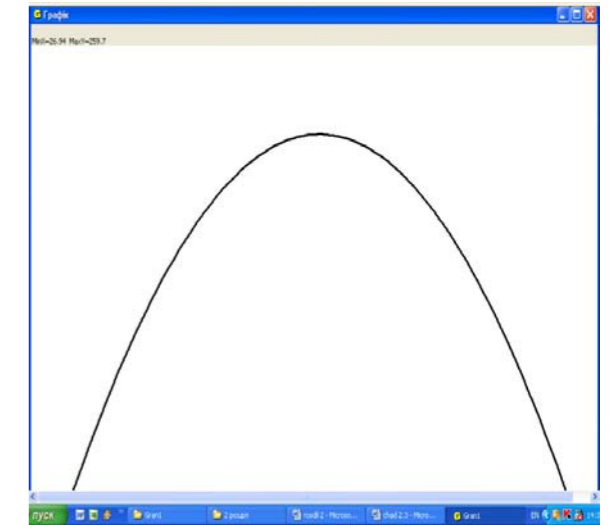


Рис. 2.125

Правильність розв'язання задачі за допомогою GRAN 1 можна перевірити аналітичним способом. Наприклад.

Враховуючи те, що за умовою задачі нам відомий вигляд функції попиту, то $TR = P \cdot Q_D = Q_D(10 - \frac{Q_D}{10}) = 10 \cdot Q_D - \frac{Q_D^2}{10}$.

Тепер ми можемо знайти граничний дохід: $MR = (TR)' = 10 - \frac{Q_D}{5}$.

Розв'яжемо рівняння $(TR)' = 0$, $10 - \frac{Q_D}{5} = 0$, звідки $Q_D = 50$. Це означає, що для $Q_D < 50$, наприклад при $Q_D = 20$, маємо $(TR)' > 0$, а для $Q_D > 50$, наприклад, при $Q_D = 70$, $(TR)' < 0$. Це означає, що при рівні виробництва $Q_D = 20$ дохід зростає при збільшенні виробництва в "невеликій" кількості (на декілька одиниць), при рівні $Q_D = 50$ дохід не зміниться при "невеликій" зміні виробництва, та при рівні $Q_D = 70$ дохід зменшуватиметься в той час, коли виробництво зростатиме. Таким чином, поняття граничного доходу дає можливість дослідити функцію доходу. За допомогою граничного аналізу ми можемо визначити зони попиту, при яких дохід компанії зростає; величину попиту, при якому доходи компанії найбільші, і зони попиту, при яких подальше збільшення виробництва призведе до падіння доходів компанії.

Результати, одержані за допомогою комп'ютера, та результати, знайдені аналітичним шляхом, однакові. Це говорить про те, що студент на уроці може зосередити увагу на самій задачі, на створенні моделі, що розглядається, може здійснити швидкий пошук необхідної навчальної інформації, вичленити необхідні йому дані за допомогою комп'ютера, не витрачаючи час на математичні підрахунки тощо.

Отже, в нашому прикладі максимальний обсяг попиту складає 50 одиниць товару, після чого настає перенасичення ринку цим товаром, і дохід компанії починає зменшуватись.

Побудуємо графіки функцій TR і TC в одній системі координат.

2.6.1. GRAN 1

Під час підготовки до проведення заняття викладачу доцільно розв'язати задачу як за допомогою GRAN 1, так і аналітично. Тому паралельно з розв'язуванням задачі з використанням ІКТ будемо проводити аналітичну перевірку одержаних результатів.

Задача 1. Компанія займається виробництвом та продажем відеомагнітофонів. Відділ маркетингових досліджень цієї компанії здійснив дослідження ринку та склав рівняння попиту: $Q_D = 100 - 10 \cdot P$, де Q_D – обсяг попиту на цей товар, або максимальна кількість одиниць товару, що його готові придбати покупці за ціну P ; P – роздрібна ціна в грн. за один відеомагнітофон. Також фінансовий відділ компанії здійснив відповідне дослідження і запропонував функцію загальних витрат у вигляді $TC = 50 + 4 \cdot Q_D$, де 50 – постійні витрати виробництва (не залежать від обсягів виробництва, наприклад, витрати на оплату приміщення, засобів зв'язку, на придбання устаткування тощо), а 4 грн. – змінні витрати виробництва одиниці продукції, які зростають разом зі зростанням обсягів виробництва (наприклад, витрати на сировину, транспортування, заробітну плату працівників тощо). Знайти рівень виробництва, при якому прибуток буде максимальним та точку насичення ринку товаром.

Розв'язання. Розв'яжемо рівняння попиту відносно P . Звідси, $P = 10 - \frac{Q_D}{10}$, де Q_D – це кількість відеомагнітофонів, які можливо продадуть у роздріб за ціною P грн. за один відеомагнітофон. Спираючись на власний досвід або пригадуючи уроки з основ економіки, учні формулюють закон попиту. Тобто, чим вища ціна на товар, тим менший попит, і навпаки, чим менша ціна, тим більший попит. Отже, функція $P(Q_D)$ є спадною.

Знайдемо граничні витрати: $MC = \frac{\Delta TC}{\Delta Q} = (TC)' = 4$, де MC – граничні витрати, TC – загальні витрати, Q – обсяг продажу. Це означає, що витрати на виробництво кожної додаткової одиниці продукції складають 4 грн.

Дати означення доходу та записати формулу для його обчислення.

Дохід (TR) – сума грошей, які компанія отримає від реалізації Q_D одиниць товару за ціною P грн. за одиницю. Він дорівнює: $TR = P \cdot Q_D$, де P – ціна проданого товару, а Q – обсяг продажу.

Побудуємо в GRAN1 графіки функцій $P(Q_D)$, Q_D , знайдемо їх добуток (рис.2.124) та за допомогою операції « M_{-} » знайдемо координати точки максимуму (рис. 2.125).

Одержані координати точки максимуму автоматично з'являються в лівому кутку вікна. Отже, $Q_D = 50$, $TR = 250$. З графіку результуючої функції TR видно, що при значеннях $Q_D < 50$ загальний дохід зростає із зростанням обсягів виробництва, при $Q_D > 50$ загальний дохід зменшується із зростанням обсягів виробництва, а при $Q_D = 50$ дохід не зміниться при «невеликій» зміні обсягів виробництва.

```
> sqrt(A)
      [,1] [,2]
[1,] 1.000000 1.732051
[2,] 1.414214 2.000000
```

За замовчуванням, R заповнює матрицю по стовпцях (див. приклад вище). Тим не менш, це можна легко скасувати, за допомогою параметра «byrow»:

```
> (M <- matrix(nrow = 4, ncol = 3, byrow = TRUE, data = 1:12))
```

```
      [,1] [,2] [,3]
[1,]  1  2  3
[2,]  4  5  6
[3,]  7  8  9
[4,] 10 11 12
```

Одинична матриця (I) створюється за допомогою R-функції «diag»:

```
> diag(1, nrow = 2)
```

```
      [,1] [,2]
[1,]  1  0
[2,]  0  1
```

Назви стовпців і рядків встановлюються таким чином:

```
> rownames(A) <- c("x","y")
> colnames(A) <- c("a","b")
> A
```

```
  a b
x 1 3
y 2 4
```

зверніть увагу, що ми використовуємо «c()» функцію тут! Назви рядків і стовпців насправді є векторами, що містять рядки.

Матриці також можуть бути створені шляхом об'єднання (зв'язування) векторів, наприклад по рядкам:

```
> V <- 0.5:5.5
> rbind(V, sqrt(V))
```

```
      [,1] [,2] [,3] [,4] [,5] [,6]
```

```
V 0.5000000 1.500000 2.500000 3.500000 4.50000 5.500000
0.7071068 1.224745 1.581139 1.870829 2.12132 2.345208
```

t(A) виконає транспонування матриці A (перестановка рядків і стовпців).

```
> t(A)
```

```
  x y
a 1 2
b 3 4
```

Масиви

Масиви – це багатовимірні узагальнення матриць; матриці і масиви в R насправді є векторами з атрибутом розмірності.

Багатовимірний масив створюється наступним чином:

```
> AR <- array(dim = c(2, 3, 2), data = 1)
```

В цьому випадку AR – масив $2 * 3 * 2$, і всі його елементи 1.

```
> AR
```

```
,, 1
     [,1] [,2] [,3]
[1,]  1   1   1
[2,]  1   1   1
```

```
,, 2
```

```
     [,1] [,2] [,3]
[1,]  1   1   1
[2,]  1   1   1
```

Розмірності

Команди:

```
> length(A)
```

```
[1] 4
```

```
> dim(A)
```

```
[1] 2 2
```

```
> ncol(M)
```

2.6. Gran

Одним з найперших вітчизняних засобів візуалізації математичної задачі та її розв'язку, що робить діалог учня (студента) та викладача більш доступним та евристичним, є педагогічний програмний засіб GRAN, розробка якого розпочалася у 1989 році авторським колективом під керівництвом М. І. Жалдака. До складу ППЗ GRAN входять педагогічні програмні засоби: GRAN1, що призначена для графічного аналізу функцій, звідки походить і її назва G^Raphic ANalysis; GRAN-2D – для комп'ютерної підтримки навчання планіметрії; GRAN-3D – для комп'ютерної підтримки навчання стереометрії. За допомогою GRAN1 можна розв'язувати наступні класи задач [25]:

– побудова графіків функцій, обчислення значення виразів; – графічне розв'язування рівнянь та систем рівнянь; – графічне розв'язування нерівностей та систем нерівностей; – відшукування найбільших та найменших значень функції на заданій множині точок; – побудова січних та дотичних до графіків; – обчислення визначених інтегралів, обчислення площ довільних фігур, обчислення довжини дуги кривої, обчислення об'ємів та площ поверхонь обертання; – елементи статистичного аналізу експериментальних даних. GRAN-2D відноситься до розряду програм динамічної геометрії та призначений для графічного аналізу геометричних об'єктів на площині, звідки і походить назва (G^Raphic ANalysis 2-Dimension). Використання пакету GRAN-2D надає можливість: – створювати динамічні моделі геометричних фігур та їхніх комбінацій аналогічно класичним побудовам за допомогою циркуля та лінійки, а також використовуючи елементи аналітичної геометрії (систему координат, рівняння прямих і кіл, алгебраїчні залежності між частинами побудови, графіки функцій тощо); – проводити вимірювання геометричних величин, досліджувати геометричні місця точок; – аналізувати динамічні вирази, висувати припущення, встановлювати закономірності; – будувати графічні зображення, використовуючи коментарі, кнопки, підказки та гіперпосилання; Педагогічний дискурс, випуск 14, 2013 385 – експортувати рисунки у графічні формати для вбудовування їх у інші додатки і для створення геометричних ілюстрацій тощо. Для графічного аналізу тривимірних об'єктів призначений пакет GRAN-3D (G^Raphic ANalysis 3-Dimension). Використання пакету GRAN-3D надає можливість: – створювати та перетворювати моделі базових просторових об'єктів; – виконувати перерізи многогранників площинами; – обчислювати об'єми та площі поверхонь многогранників і тіл обертання; – вимірювати відстані та кути [6]. Складовою програмно-методичного комплексу GRAN є також посібник для вчителів у якому наведена значна кількість математичних прикладів, що унаочнюють графічні зображення задач і вправ для самостійного виконання, питання для самоконтролю. Ці завдання можна використовувати також для вивчення деяких розділів вищої математики.

3. Збережіть побудови у папці, вказаній викладачем, під ім'ям "Завдання 4(в)".

$$\begin{cases} (x-5)^2 + (y-2)^2 = 5 \\ y^2 - x = 0 \end{cases}$$

4. Для розв'язання системи рівнянь 4(б), необхідно виконати наступні дії:
- побудуйте графік рівняння $(x-5)^2 + (y-2)^2 = 5$;
 - побудуйте графік рівняння $y^2 - x = 0$;
 - маємо дві точки перетину графіків рівнянь (в залежності від налаштувань програми «за замовчанням» ви отримаєте координати точок з різною точністю, наприклад, такі, як на рис. 2.123 ($\approx 2,79$; $1,67$) і ($\approx 7,13$; $2,67$)), отже, система має два розв'язки;

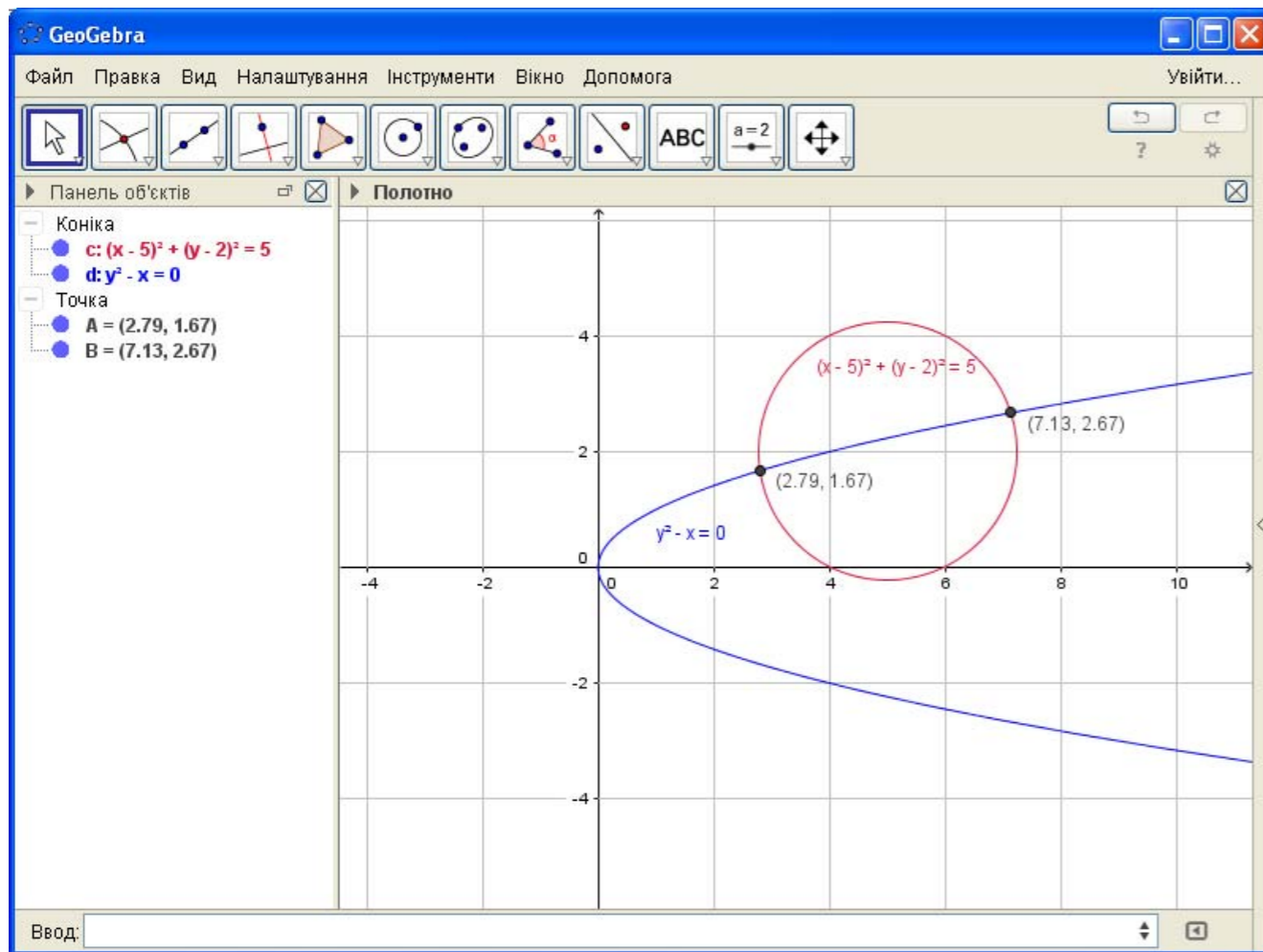


Рис. 2.123

– збільшити точність відповіді можна, скориставшись пунктом меню Опції / Округлення.

5. Збережіть побудови у папці, вказаній викладачем, під ім'ям "Завдання 4(б)".

6. Розв'яжіть графічно систему рівнянь 4(а) самостійно, скориставшись досвідом, набутим під час виконання завдань 4(б) і 4(в).

7. Закрийте програму.

[1] 3

> nrow(M)

[1] 4

Поверне довжину (загальне число елементів) V (вектора чи матриці), розмірність матриці або масиву A, число стовпців і число рядків матриці M відповідно.

Вибір і вилучення елементів

Для вибору підмножини векторів або матриць, ми можемо або

- вказати номери елементів, які нам потрібно (проста індексація)
- вказати вектор логічних значень (TRUE / FALSE), щоб вказати, які елементи включити (TRUE) і які не включити (FALSE). Для цього використовуються логічні вирази

Проста індексація

Елементи векторів, матриць і масивів індексуються використовуючи оператор «[]» :

> M[1, 1]

> M[1, 1:2]

> M[1:3, c(2,4)]

Результатом буде елемент з першого рядка і першого стовпця матриці M (1-ша лінія), потім вибірка елементів з першого рядка і перших двох стовпців (2-га лінія), а потім елементи з перших трьох рядків, 2-го і 4-го стовпця матриці M (3-а лінія).

Якщо індекс пропущений, то всі рядки (перший індекс пропущений) або стовпці (2-й індекс пропущений) вибираються. В наступному виразі:

> M[, 2] <- 0

> M[1:3,] <- M[1:3,] * 2

спочатку всі елементи в другому стовпці (1-ша лінія) M обнуляються, а потім елементи на перших трьох рядках M множаться на 2 (2-га лінія).

Подібні методи вибору застосовуються до векторів:

> V[1:10]

[1] 0.5 1.5 2.5 3.5 4.5 5.5 NA NA NA NA

> V[seq(from = 1, to = 5, by = 2)]

[1] 0.5 2.5 4.5

У виразі на 1-ій лінії вибираються перші 10 елементів вектора V, а у виразі на 2-ій лінії, вибираються 1-й, 3-й і 5-й елементи вектора V.

2.1.11. Логічні вирази

Логічні вирази часто використовуються, щоб вибрати елементи векторів і матриць, які підпорядковуються певним критеріям. R розрізняє логічні змінні TRUE (істина) і FALSE (хиба), які представлені числами 1 і 0.

Наступні операції

```
> ?Comparison
> ?Logic
```

перераховують реляційні і логічні оператори, доступні в R.

Наступна дія поверне TRUE для значень послідовності V, які є додатними:

```
> (V <- seq(-2, 2, 0.5))
[1] -2.0 -1.5 -1.0 -0.5 0.0 0.5 1.0 1.5 2.0

> V > 0
[1] FALSE FALSE FALSE FALSE FALSE TRUE TRUE TRUE TRUE
```

в той час як

```
> V[V > 0]
[1] 0.5 1.0 1.5 2.0
```

вибере додатні значення з V,

```
> V[V > 0] <- 0
```

обнулить всі додатні елементи в V,

```
> sum(V < 0)
[1] 4
```

поверне кількість від'ємних елементів: просумує значення TRUE (= 1), і

```
> V[V != 0]
[1] -2.0 -1.5 -1.0 -0.5
```

покаже всі ненульові елементи з V («!» це оператор логічного заперечення («не»)).


Логічні вирази також можна об'єднувати, за допомогою операторів «|» (логічний оператор «або») і «&» (логічний оператор «і»).

```
> V[V < (-1) | V > 1]
[1] -2.0 -1.5
```

Технологія виконання.

1. Запустіть програму GeoGebra.
2. Щоб розв'язати систему рівнянь 4(в), виконайте наступні дії:

- побудуйте графік рівняння $y^2 - xy = -6$;
- побудуйте графік рівняння $x^2 - xy = 20$;

- для визначення координат точок перетину оберіть інструмент  (Перетин двох об'єктів);
- клацніть лівою кнопкою мишки на графіку кожного з рівнянь;
- на полотні будуть побудовані точки перетину графіків, представлення яких ми отримаємо на панелі об'єктів;
- якщо на полотні не відображуються координати точок перетину графіків, то для того, щоб це зробити, виконайте наступні дії:
 - клацніть на точці правою кнопкою мишки і у контекстному меню оберіть пункт "Властивості";
 - у діалоговому вікні, що з'явиться, на вкладці <Основні> перевірте, чи стоїть прапорець біля пункту "Показувати значення", якщо він відсутній, то поставте його (клацніть на ньому лівою кнопкою мишки);
 - у списку поряд оберіть пункт <Значення>;
- скориставшись вкладками <Колір> і <Стиль>, змініть колір графіків рівнянь та товщину лінії, що їх зображує, на власний розсуд.

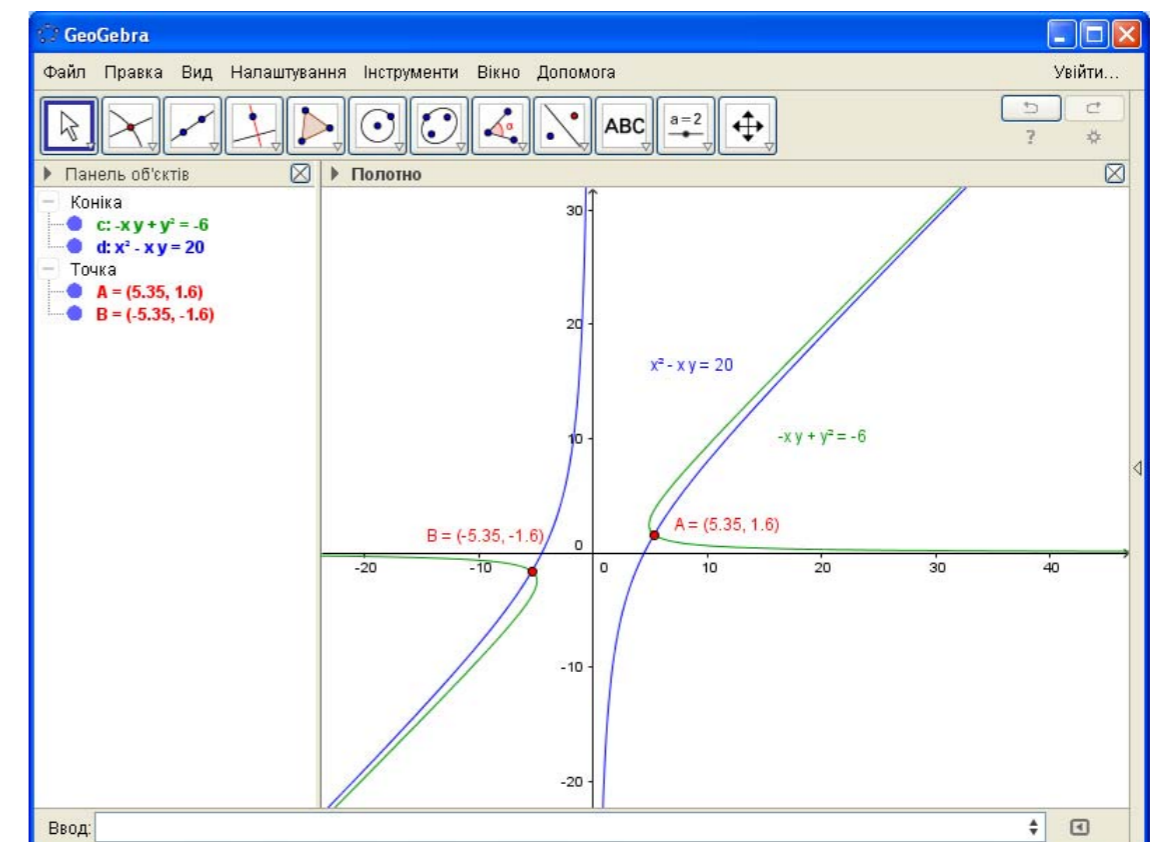


Рис.2.122

- знайдіть точки перетину графіка функції з віссю абсцис;
- ви отримаєте чотири точки;
- збільшить масштаб (за допомогою коліщатка миші)
- з рисунку видно, що умові відповідає всі чотири точки;
- отже, відповідь: $x = -2; -1,43; -0,23; 0,67$.

3. Проаналізуйте отриману відповідь з точки зору існування інших розв'язків.

4. Збережіть побудови у папці, вказаній викладачем, під ім'ям "Завдання 3(б)".

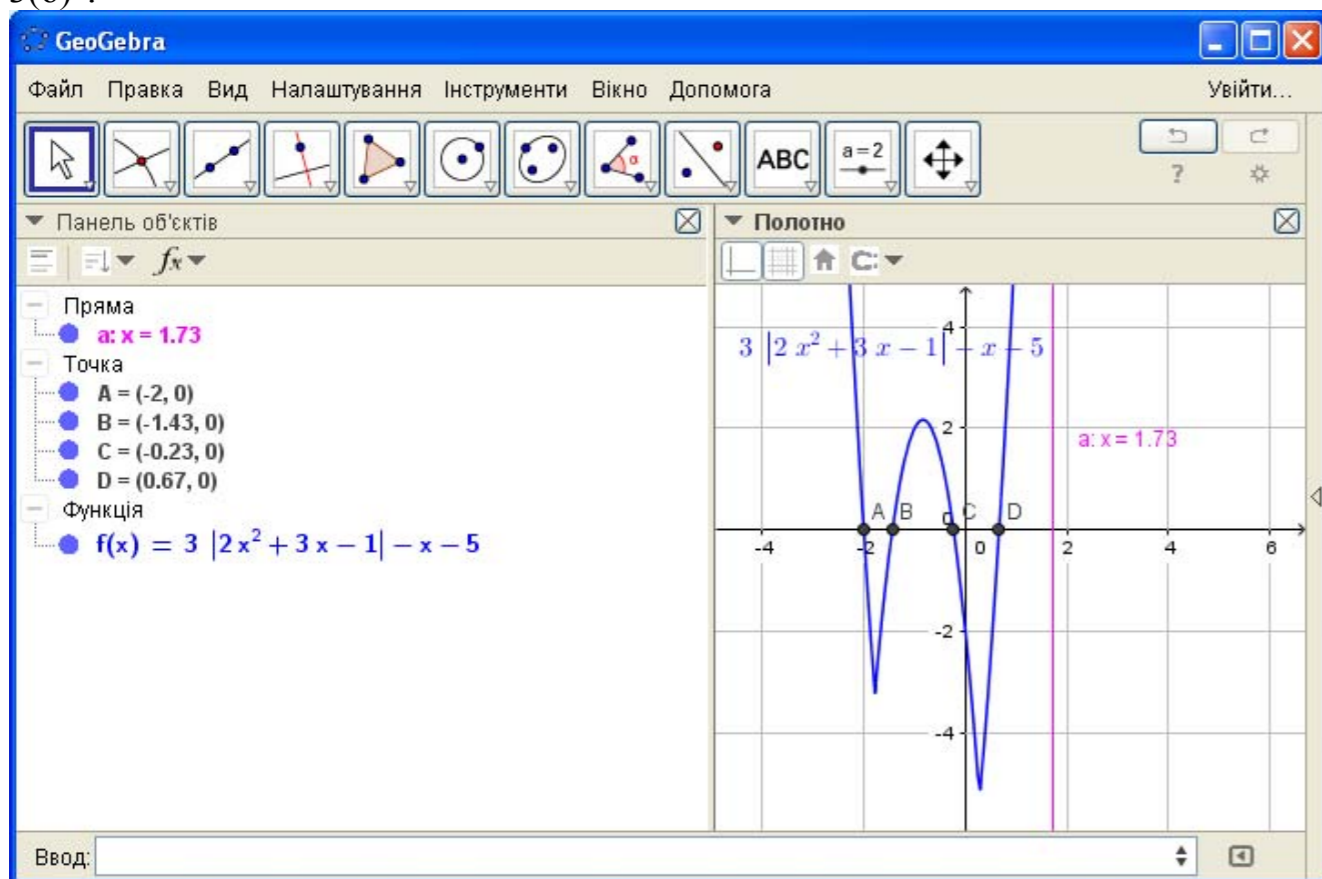


Рис.2.121

5. Вийдіть з програми.

Задачі 3(в), 3(г) і 3(д) розв'яжіть самостійно, скориставшись досвідом, набутим під час виконання завдань 3(а) і 3(б). Збережіть побудови у папці, вказаній викладачем, під іменами "Завдання 3(в)", "Завдання 3(г)", "Завдання 3(д)".

Завдання №4

За допомогою GeoGebra розв'яжіть графічно системи рівнянь:

а) $\begin{cases} xy = -8 \\ 5x + 2y = 20 \end{cases}$	б) $\begin{cases} (x-5)^2 + (y-2)^2 = 5 \\ y^2 - x = 0 \end{cases}$
в) $\begin{cases} y^2 - xy = -6 \\ x^2 + xy = 20 \end{cases}$	г) $\begin{cases} x + 3y = 8 \\ 2x - y = 4 \end{cases}$

Це виведе на екран всі значення з V, які є <-1 або >1 . Зверніть увагу, що ми вклали «-1» в дужки (ви можете побачити, чому це необхідно?)

I, нарешті,

```
> which(V == 0)
[1] 5 6 7 8 9
```

```
> which.min(V)
[1] 1
```

поверне індекс елемента із значенням 0 і мінімальним значенням.

Видалення елементів

Якщо перед індексом елемента поставити «-», елемент видаляється.

```
> M[,-1]
```

На екран виведеться вміст матриці M, за винятком першого стовпчика.

```
> x<-x[-1]
> M<-M[-1,]
> V<-V[-V=0]
```

Видалиться перший елемент X (перша лінія), 1-й рядок M (друга лінія), і всі додатні елементи V (третя лінія).

Для отримання додаткової інформації, наберіть

```
> ?Extract
```

Оголошення функцій

Однією із сильних сторін R є те, що користувач може оголошувати свої власні функції, які додаються до вбудованих функцій R.

Як правило, складні функції, пишуться у **R-script** файлах, які потім підключаються до середовища R. Наприклад,

```
> Circlesurface <- function (radius) pi*radius^2
```

визначає функцію (яка має назву «Circlesurface»), яка приймає в якості вхідного аргументу змінну «radius» і яка повертає значення $\pi \cdot \text{radius}^2$ (що є значенням площі круга з радіусом «radius»).

Після підключення цієї функції до середовища R, ми можемо використовувати її для обчислення площі круга із заданим радіусом:

```
> Circlesurface(10)
```

[1] 314.1593

або

> Circlesurface(1:20)

```
[1] 3.141593 12.566371 28.274334
[4] 50.265482 78.539816 113.097336
[7] 153.938040 201.061930 254.469005
[10] 314.159265 380.132711 452.389342
[13] 530.929158 615.752160 706.858347
[16] 804.247719 907.920277 1017.876020
[19] 1134.114948 1256.637061
```

яка буде обчислювати площі кругів з радіусами 1, 2, ..., 20.

Більш складні функції можуть повертати більше одного елемента:

```
> Sphere <- function(radius)
+ {
+   volume <- 4/3*pi*radius^3
+   surface <- 4 *pi*radius^2
+   return(list(volume=volume,surface=surface))
+ }
```

Тут ми визначаємо:

- заголовок функції (1-я лінія), вказавши ім'я функції (Sphere) і вхідний параметр (radius)
- тіло функції. Оскільки функція включає декілька операторів, то тіло функції пишеться всередині фігурних дужок {...}.
- значення, що повертаються (остання стрічка тіла функції). Функція «Sphere» поверне об'єм і площу поверхні сфери, у вигляді списку.

Земля має приблизно радіус 6371 км, так що її об'єм (км³) і площа поверхні (км²) є:

```
> Sphere(6371)
$volume
[1] 1.083207e+12

$surface
[1] 510064472
```

В наступному виразі буде відображатися тільки об'єм сфер з радіусами 1, 2, ... 5

– інших точок перетину не існує, оскільки графік функції $y = x^2 - 6|x| + 8$ отримуємо з графіка функції $y = x^2 - 6x + 8$, побудованого для $x \geq 0$ в результаті його симетричного відображення відносно осі ординат;



- для визначення координат точок перетину оберіть інструмент (Перетин двох об'єктів);
- клацніть лівою кнопкою мишки на осі абсцис лівіше від крайньої (зліва) точки перетину, а потім на — графіку функції (теж лівіше);
- повторіть вказану послідовність дій для інших трьох точок перетину;
- на полотні та панелі об'єктів з'являться відповідні точки (рис. 2.119);
- отже, рівняння має чотири розв'язки: -3; -2; 2; 3.

3. Проаналізуйте отриману відповідь.

4. Збережіть побудови у папці, вказаній викладачем, під ім'ям "Завдання 3(A)".

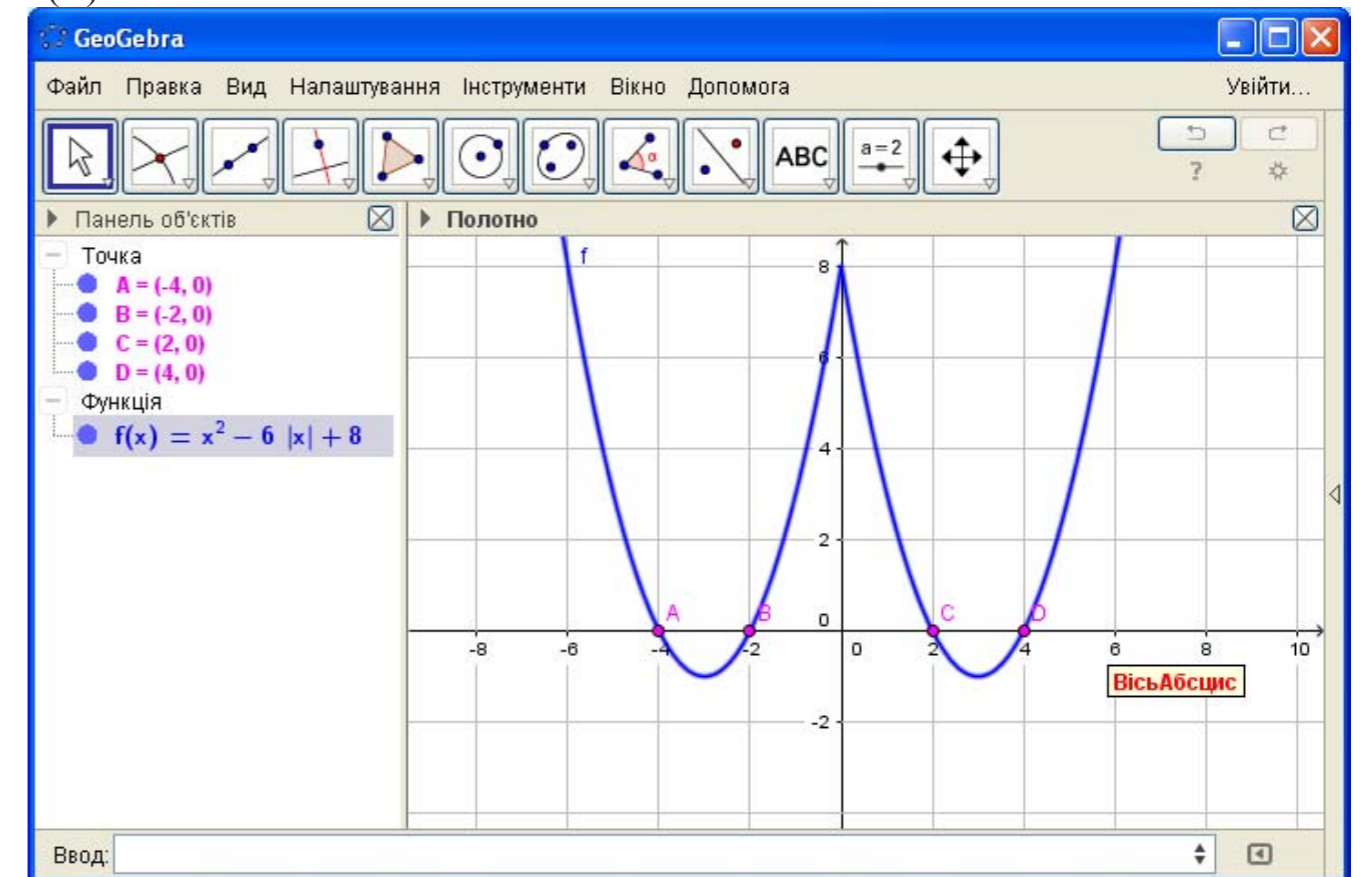


Рис. 2.120

5. Закрийте програму, клацнувши на кнопці . Завдання №3(б)

1. Запустіть програму GeoGebra.

2. Щоб розв'язати рівняння, виконайте наступні дії:

- побудуйте графік функції $y = 3|2x^2 + 3x - 1| - x - 5$;
- побудуйте пряму $x = \sqrt{3}$;

Завдання №3

а) З допомогою GeoGebra розв'яжіть графічним способом рівняння $x^2 - 6|x| + 8 = 0$

б) За допомогою GeoGebra знайдіть усі корені рівняння $3|2x^2 + 3x - 1| = x + 5$, що задовольняють нерівності $x < \sqrt{3}$.

в) Розв'яжіть графічним способом рівняння $5\sqrt{x+5} - \sqrt{x-3} = 8$ за допомогою GeoGebra.

г) Розв'яжіть графічним способом за допомогою GeoGebra рівняння $\sqrt[3]{4x-2} + \sqrt[3]{x-2} = 1$.

д) Розв'яжіть графічним способом за допомогою GeoGebra рівняння $4^x \cdot 7^{\frac{x}{x-2}} = 30$.

Технологія виконання

Завдання №3(а)

1. Запустіть програму GeoGebra.

2. Щоб розв'язати рівняння, виконайте наступні дії:

- побудуйте графік функції $y = x^2 - 6|x| + 8$;
- графік функції $y = x^2 - 6|x| + 8$ перетинає вісь абсцис у чотирьох точках (рис. 2.118);

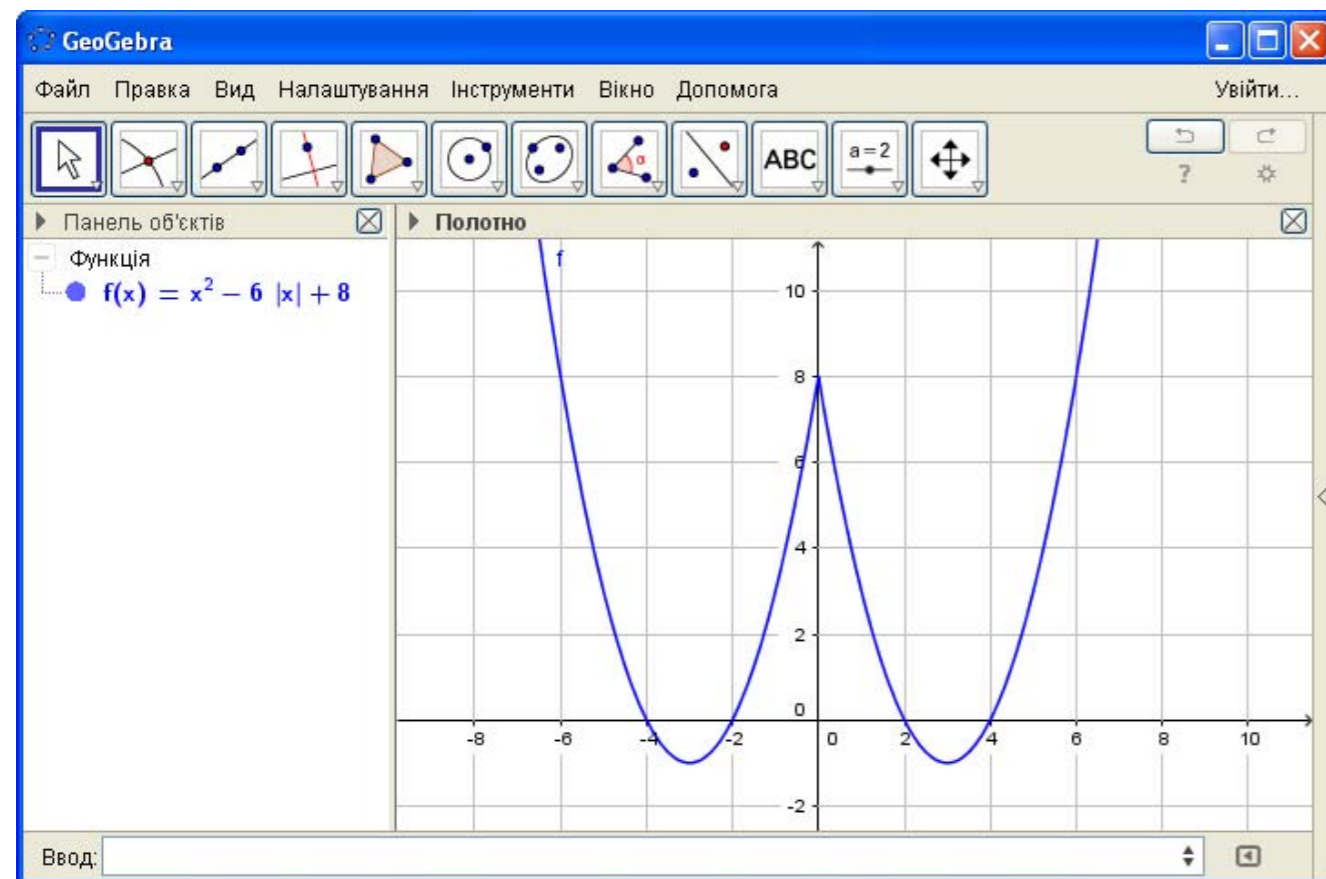


Рис. 2.118

```
> Sphere(1:5)$volume
[1] 4.18879 33.51032 113.09734 268.08257
[5] 523.59878
```

Іноді зручно, щоб вхідні параметри мали значення за замовчуванням.

Наприклад, наступна функція оцінює щільність «стандартної океанської води» (в кг/м^3), залежно від температури T , (і для солоності salinity = 0, тиску pressure = 1 атм) (Міллер і Пуассон, 1981); вхідний параметр T за замовчуванням, рівний 20°C :

```
> Rho_W <- function(T=20)
+ {
+   999.842594 + 0.06793952 * T - 0.00909529 * T^2 +
+   0.0001001685 * T^3 - 1.120083e-06 * T^4 + 6.536332e-09 * T^5
+ }
```

В кінці першого рядка стоїть «+», цим ми вказали, що вираз не закінчився і має продовження в наступному рядку. Було б неправильно ставити «+» на початку другого рядка. (Дивись главу 3)

Виклик функції без вказування температури, використовує значення за замовчуванням:

```
> Rho_W()
[1] 998.2063
> Rho_W(20)
[1] 998.2063
> Rho_W(0)
[1] 999.8426
```

Програмування

R має всі можливості мови програмування високого рівня:

Конструкції if, else, ifelse

Постарайтеся зрозуміти наступне:

```
> Dummy <- function (x)
+ {
+   if ( x<0 ) string <- "x<0" else
+   if ( x<2 ) string <- "0>=x<2" else
+   string <- "x>=2"
+   print(string)
+ }
> Dummy(-1)
```

```
[1] "x<0"
> Dummy(1)
[1] "0>=x<2"
> Dummy(2)
[1] "x>=2"
```

Зверніть увагу, що ми вказали вираз «else» на тому ж рядку, що і частина «if» так, щоб R зрозумів, що вираз триває на наступному рядку!

«if» і «else» конструкції, що включають тільки один оператор можуть поєднуватися:

```
> x<-2
> ifelse (x>0, "positive", "negative,0")
[1] "positive"
```

Цикли

Цикли дозволяють виконувати певний набір інструкцій кілька разів.

Цикл «for» перебирає певний набір значень. У наведеному нижче прикладі змінна «i» приймає значення (1,2,3):

```
> for (i in 1:3) print(c(i,2*i,3*i))
[1] 1 2 3
[1] 2 4 6
[1] 3 6 9
```

Цикли «while» і «repeat» виконуватимуться, поки зазначена умова не буде виконана.

```
> i<-1 ; while(i<3) {print(i); i<-i+1}
[1] 1
[1] 2
```

«break» виходить із циклу.

«next» зупиняє поточну ітерацію і переходить до наступного ітерації.

```
> i<-1
> {print(i)
+ i <-i+1
+ if(i>2) break
+ }
[1] 1
```

Фігурні дужки «{...}» охоплюють кілька операторів, які виконуються в кожній ітерації.

– зверніть увагу на відмінності в позначенні рівнянь (рис. 2.117);

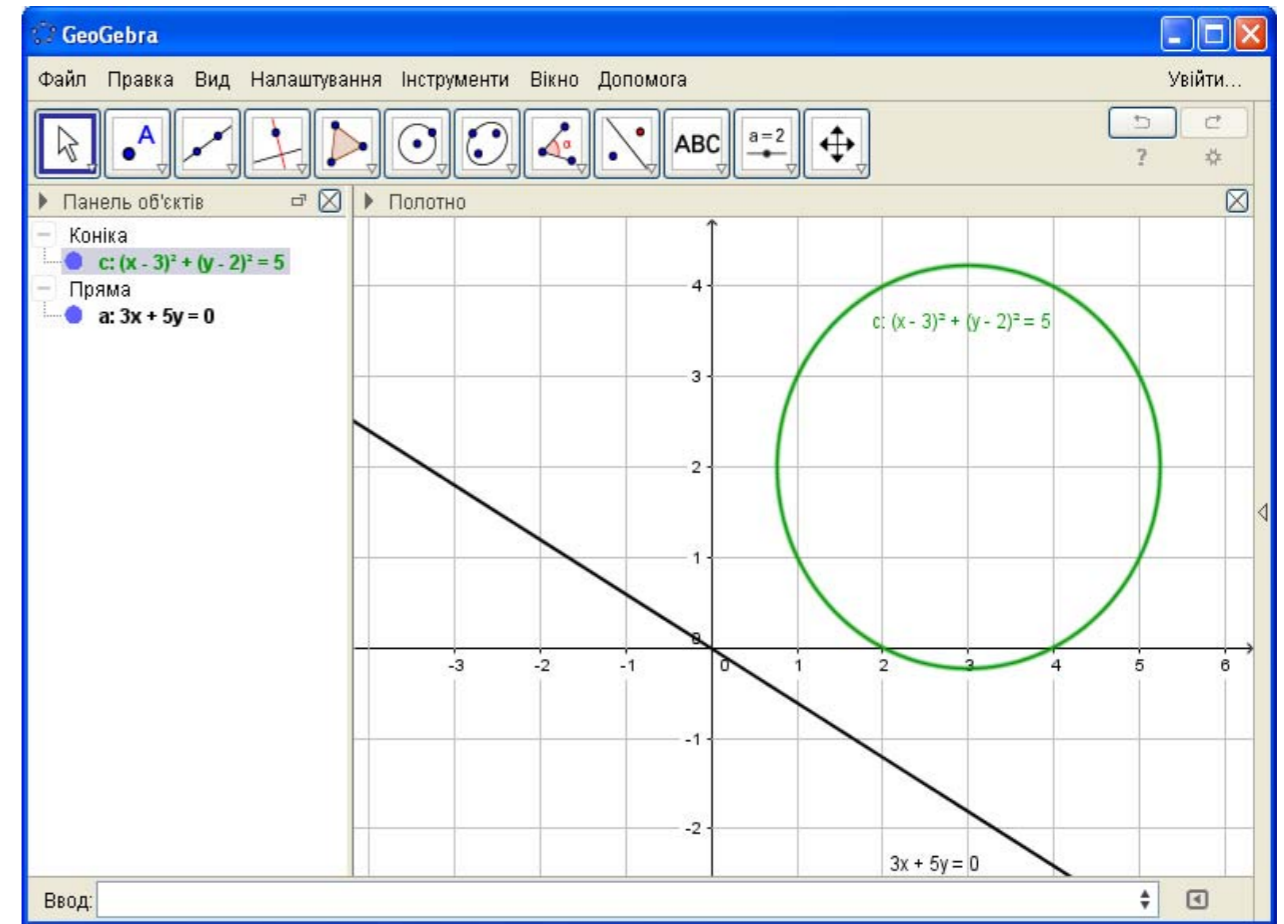


Рис 2.117

- скориставшись вкладками «Колір» і «Стиль», змініть колір графіків рівнянь та товщину лінії, що їх зображує, на власний розсуд.
- Скориставшись набутим досвідом, побудуйте графіки інших рівнянь.
- Збережіть побудови у папці, вказаній викладачем під ім'ям "Завдання 2". Закрийте програму, натиснувши кнопку

Зауваження. У своєму житті, професійній діяльності людина найчастіше має справу з наближеними значеннями чисел у вигляді десяткових дробів з тією чи іншою точністю. Дуже часто при розв'язуванні прикладних математичних задач, що виникають у процесі життєдіяльності людини, потрібно отримати число (як правило, округлене з певною наперед заданою точністю), а не вираз (наприклад: $1 - \sqrt{5}$, $3 + \lg 7$, $\cos \frac{\pi}{5}$).

За допомогою GeoGebra у багатьох випадках можна швидше розв'язати рівняння чи систему рівнянь графічним способом, ніж це зробити аналітично. Якщо розв'язком є нескінченний десятковий періодичний дріб або число ірраціональне, то можна знайти його наближене значення.

- у рядку «Ввод:» введіть рівняння $3x + 5y = 0$ і натисніть клавішу <Enter>;
- на полотні буде побудовано графік шуканого рівняння (рис. 2.120), а на панелі об'єктів, відповідно, з'явиться його аналітичне представлення;
- навівши покажчик мишки на графік функції, клацніть правою кнопкою мишки і в контекстному меню оберіть пункт "Властивості";
- у діалоговому вікні «Позначення», що з'явиться, на вкладці «Основні» перевірте, чи стоїть прапорець біля пункту «Показувати позначення», якщо він відсутній, то поставте його (клацніть на ньому лівою кнопкою миші);
- у списку поряд оберіть пункт «Значення»;
- на полотні біля графіка з'явиться рівняння, якому він відповідає.

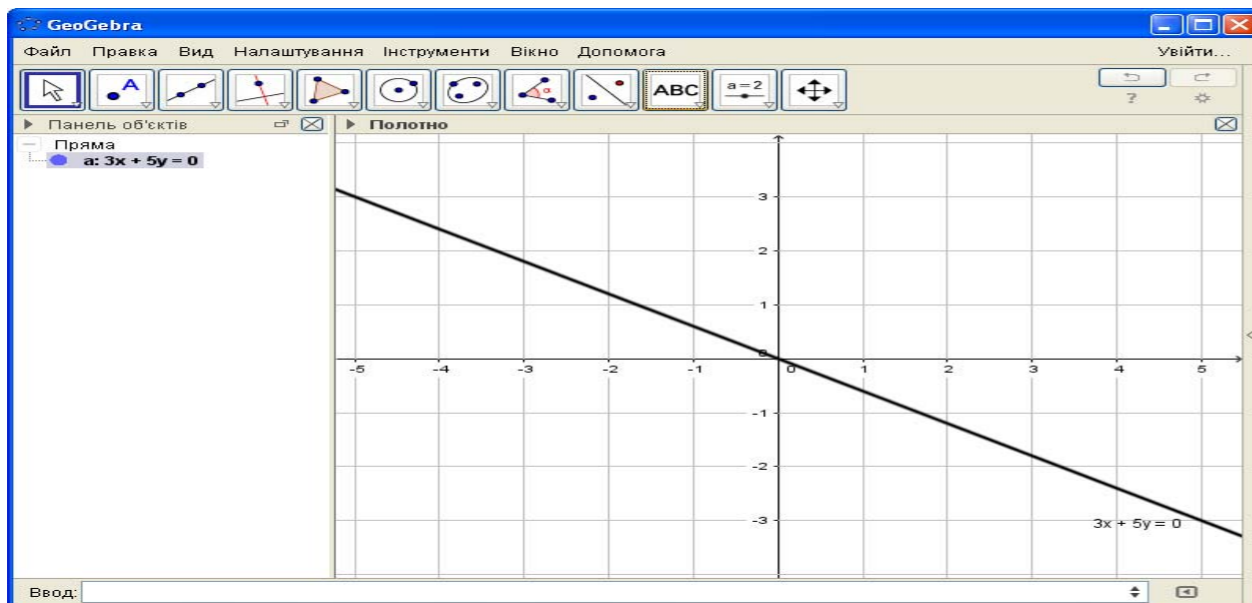


Рис. 2.116

3. Побудуйте графік рівняння $(x - 3)^2 + (y - 2)^2 - 5 = 0$. Для цього:

- у рядку «Ввод:» введіть рівняння $(x - 3)^2 + (y - 2)^2 - 5 = 0$ і натисніть клавішу <Enter>;
- на полотні буде побудовано графік шуканого рівняння, а на панелі об'єктів, відповідно, буде відображено його аналітичне представлення;
- навівши покажчик мишки на графік функції, клацніть правою кнопкою мишки і в контекстному меню оберіть пункт "Властивості";
- у діалоговому вікні, що з'явиться, на вкладці «Основні» перевірте, чи стоїть прапорець біля пункту «Показувати позначення», якщо він відсутній, то поставте його (клацніть на ньому лівою кнопкою мишки);
- у списку поряд оберіть пункт «Ім'я та значення»;

Примітка: цикли реалізуються вкрай неефективно в R і їх слід уникати якомога частіше. На щастя, R пропонує безліч команд високого рівня, які працюють з векторами і матрицями. Їх треба використовувати якомога більше!

Для отримання додаткової інформації про, конструкцію «if» і цикли, наберіть

```
> ?Control
```

2.1.12. R- пакети

Пакет в R являє собою файл, що містить безліч функцій, які виконують певні завдання.

Пакети можна завантажити з офіційного веб-сайту R.

Після установки, для того щоб згенерувати список всіх доступних пакетів, завантажити пакет і отримати список з його вмістом, треба виконати наступні командами:

```
> library()
> library(deSolve)
> library(help=deSolve)
> help(package=deSolve)
```

2.1.13. Використання R за чотири кроки

R виник як статистичний пакет, і він як і раніше переважно використовується для цієї мети. Ви можете зробити практично будь-який статистичний аналіз в R.

Припустимо, ви хочете виконати ієрархічну кластеризацію і побудувати дендрограму багатовимірного набору даних. Під дендрограмою зазвичай розуміється дерево, тобто граф без циклів, який побудований з матриці подібності. Дендрограма дозволяє зобразити взаємні зв'язки між об'єктами з заданого переліку. Для створення дендрограми потрібно матриця подібності (відмінності), яка визначає рівень подібності між парами об'єктів. Частіше використовуються агломеративні методи. Якщо ви ніколи цього не робили в R, то ось необхідні кроки:

1. Знайти функцію, яка виконує поставлене завдання. Наприклад, використовуючи для допомоги `help.search(«cluster»)`. Залежно від кількості пакетів, які ви встановили, R перерахує ряд можливих функцій, файл довідки яких містить слово «cluster». Використовуйте функції з пакету статистики (частина ядра R).

2. Відкрийте файл довідки (`? <ім'я-функції>`) і подивитися синтаксис цієї функції. Якщо у вас немає часу, щоб прочитати його повністю, принаймні, прочитайте (частину) з розділів «Опис», «Використання» і «Приклади».

3. Застосуйте приклади з файлу довідки. Ви можете:
- Спробувати їх усі відразу (`example(<ім'я-функції>)`).

•Крім того, ви можете вибрати окремі приклади, які виглядають застосовними до вашої задачі. Для цього скопіюйте і вставте їх у ваш файл сценарію (Ctrl-C / Ctrl-V) і виконайте їх.

4. Перетворіть багатообіцяючий приклад так, щоб він вирішував вашу задачу.

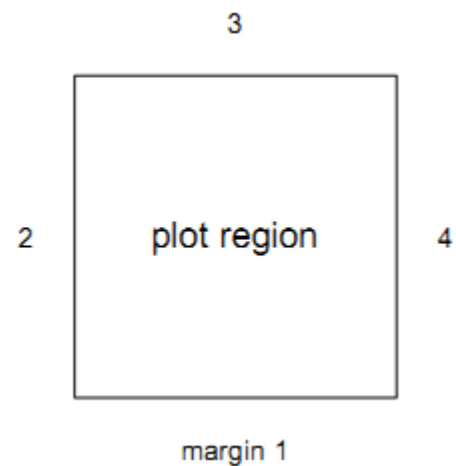
R має широкі графічні можливості. Спробуйте:

- > demo(graphics)
- > demo(image)
- > demo(persp)

для отримання відображення можливостей R в 1-D, 2-D і 3-D зображеннях.

2.1.14. Основи

Графіка відображається в окремому вікні. Вся область графіка складається з області відображення самого графіка, та 4 граней, що його оточують. Грані нумеруються за годинниковою стрілкою, від 1 до 4, починаючи від нижньої.



R розрізняє:

1. команди високого рівня. За замовчуванням, вони створюють нову фігуру, наприклад,

- hist, barplot, pie, boxplot, ... (1-D графіки)
- plot, curve, matplot, pairs, ... (x-y графіки)
- image, contour, filled.contour, ... (2-D графіки поверхонь)
- persp, scatterplot3d (3-D графіки)

2. команди низького рівня, які додають нові об'єкти в існуючі фігури, наприклад,

- lines, points, segments, polygon, rect, text, arrows, legend, abline, locator, rug, ... які додають об'єкти в область графіка

– за допомогою діалогового вікна, що з'явиться, скориставшись вкладками «Колір» і «Стиль», змініть колір графіка та товщину лінії, що його зображує (колір графіка оберіть на власний розсуд, а товщину – 5 пунктів).

4. Скориставшись набутим досвідом, побудуйте графіки інших функцій.

5. Збережіть побудови, у папці, вказаній викладачем, під ім'ям Завдання 1. Для цього:

- 1. Для цього:
 - оберіть пункт меню Файл / Зберегти як...;
 - у полі «Папка» оберіть потрібну папку;
 - у полі «Ім'я файлу» введіть напис "Завдання 1";
 - натисніть кнопку «Зберегти».

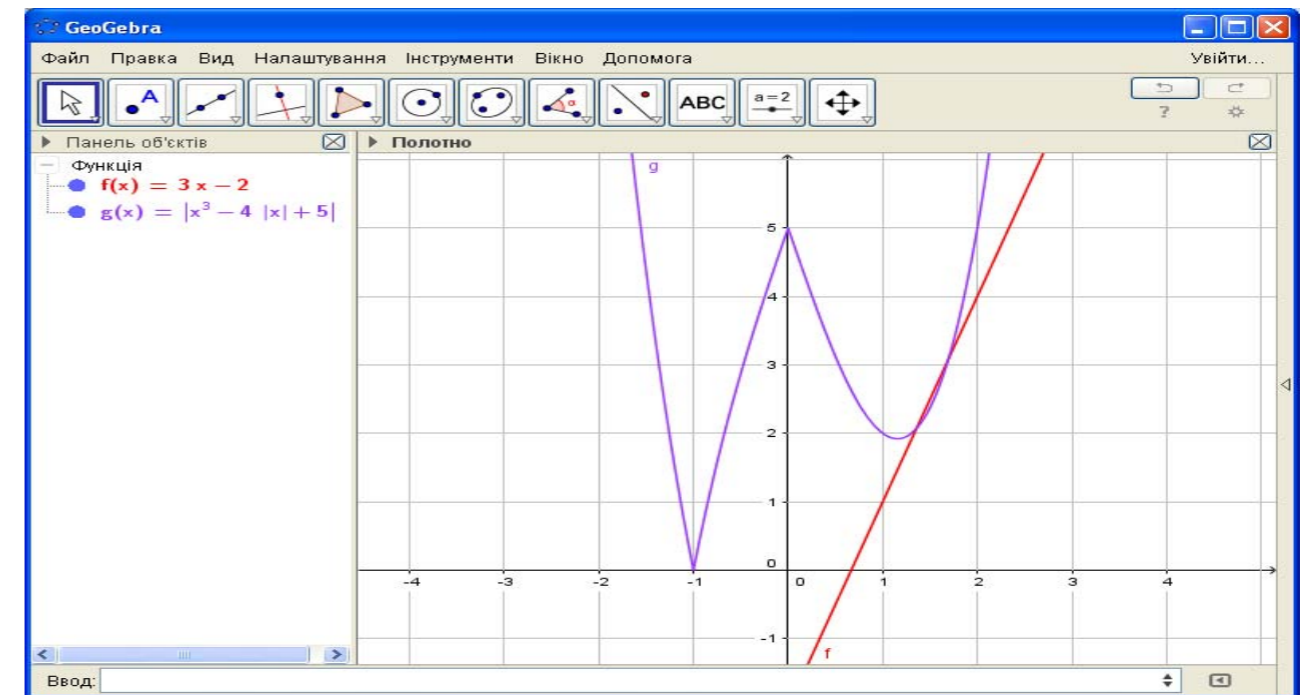


Рис. 2.115

6. Закрийте програму, клацнувши на кнопці

Завдання №2 Скориставшись середовищем GeoGebra, побудуйте графіки наступних рівнянь

а) $3x + 5y = 0$	б) $x^2y + 6x - 5 = 0$
в) $(x - 3)^2 + (y - 2)^2 - 5 = 0$	г) $\frac{x^2}{8} + \frac{y^2}{6} = 1$
д) $\frac{x^3}{27} - \frac{y^3}{18} = 1$	е) $(y - 3)^2 = 5(x + 2)$

Технологія виконання

1. Запустіть програму GeoGebra.
2. Побудуйте графік рівняння $3x + 5y = 0$, виконавши наступні дії:

– на полотні графічних побудов ми отримуємо графік шуканої функції (рис. 2.114), а на панелі об'єктів, відповідно, його аналітичне представлення;

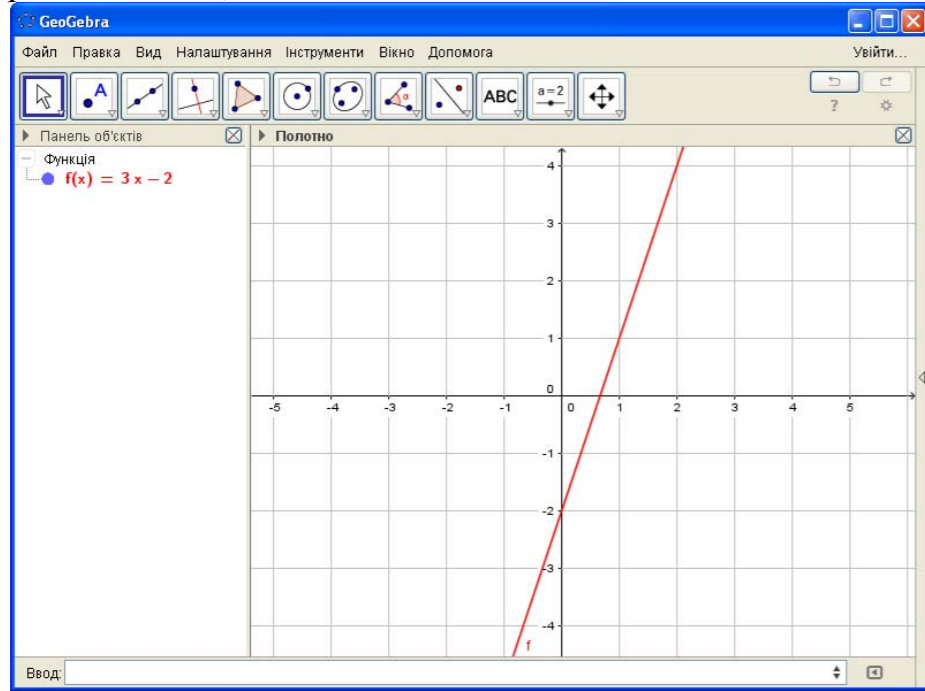


Рис. 2.114

– навівши покажчик мишки на графік функції, клацніть правою кнопкою мишки і в контекстному меню оберіть пункт **Властивості ...**;

– за допомогою діалогового вікна, що з'явиться, скориставшись вкладками <Колір> і <Стиль>, змініть колір графіка та товщину лінії, що його зображує, на власний розсуд.

3. Побудуйте графік функції $y = |x^3 - 4|x| + 5|$, виконавши наступні дії:

– у рядку <Введення> введіть формулу $\text{abs}(x^3 - 4\text{abs}(x) + 5)$ і натисніть клавішу <Enter> (для автоматизації введення формул можна використовувати функції, щоб отримати до них доступ, потрібно обрати **Математичні функції**);

Примітка. Для полегшення процесу введення формули можна скористатись таблицею символів **☐**. Щоб це зробити, потрібно клацнути на кнопці **☐** і в таблиці, що відкриється, обрати потрібний символ (клацнути на ньому лівою кнопкою мишки). Також введення формул зручно робити за допомогою віртуальної клавіатури.

– на полотні ми отримуємо графік шуканої функції, а на панелі об'єктів, відповідно, його аналітичне представлення (рис. 2.119);

– навівши покажчик мишки на графік функції, клацніть правою кнопкою мишки і в контекстному меню оберіть пункт **Властивості ...**;

• **box**, **axis**, **mtext** (текст на гранях), **title**, ... які додають об'єкти на грані графіка

3. **графічні параметри**, які контролюють зовнішній вигляд.

• Графічні об'єкти: **sex** (розмір тексту і символів), **col** (колір), **font**, **las** (орієнтація надпису осей), **lty** (тип ліній), **lwd** (ширина ліній), **pch** (тип точок), ...

• Графічне вікно: **mar** (розмір граней), **mfgrow** (кількість фігур на рядку), **mfcow** (кількість фігур у стовпці), ...

> ?plot.default
> ?plot.window
> ?par
> ?points

відкриє файли довідки, в той час як

> example(plot.default)
> example(points)

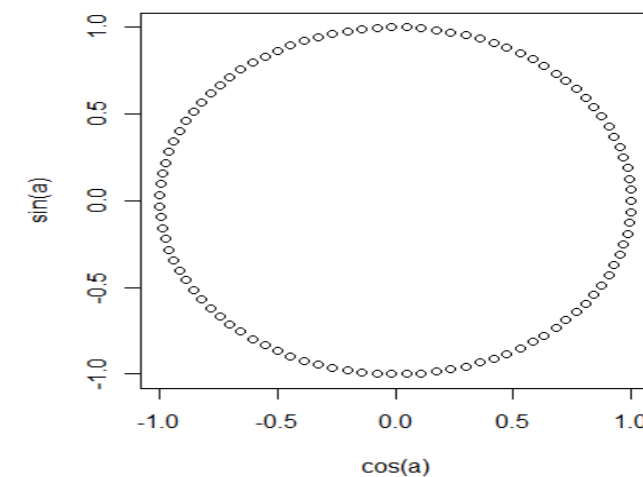
буде запускати приклади, показуючи кожен новий графік, після натискання клавіші <ENTER> (спробуйте!)

2.1.15. X-Y графіки

Круг можна побудувати по набору (x, y) точок, де $x = r \cdot \cos(\alpha)$ і $y = r \cdot \sin(\alpha)$, з кутом α , від 0 до 2π , та радіусом r.

У наступному прикладі ми спочатку створили послідовність значень кута α , від 0 до 2π , що включає 100 значень (length.out), а потім побудували коло з одиничним радіусом:

```
> a <- seq(0, 2*pi, length.out=100)
> plot(x=cos(a), y=sin(a))
```



«plot» – це команда високого рівня, вона створює нову фігуру.

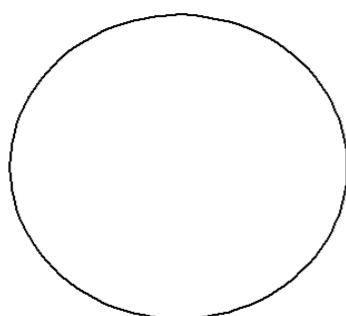
За замовчуванням, R додає осі і надписи осей, і представляє (x, y) дані у вигляді маленьких крапок. Зауважимо, що графік не є симетричним.

Тепер ми зробимо більш складну фігуру, яка нагадує мішень, наприклад, для практикуючих стрілянина з лука або метання дротиками.

Спочатку використаємо ту ж команду («plot»), що описано вище, але ми додамо ряд графічних параметрів, які визначають, що:

- замість крапок, точки повинні бути з'єднані лініями (type);
- лінія повинна бути в два рази ширше ніж за замовчуванням (lwd) ;
- надписи X і Y-осей (xlab, ylab) повинні бути порожніми ;
- осі і анотації осей (axes) повинні бути видалені ;
- графік має бути симетричним, тобто x / y співвідношення сторін = 1(asp).

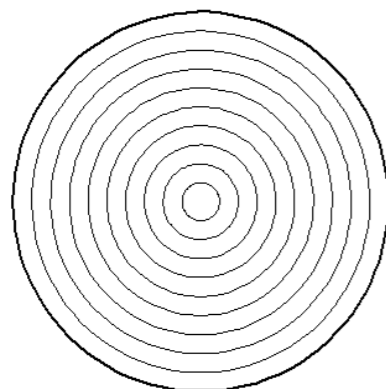
```
>plot(cos(a), sin(a), type="l", lwd=2, xlab="", ylab="", axes=FALSE, asp=1)
```



До цієї фігури, ми можемо тепер додати кілька об'єктів низького рівня:

- серія ліній, що представляють все менші й менші кола (lines).

```
> for (i in seq( 0.1,0.9,by=0.1)) lines(i*sin(a), i*cos(a))
```



- внутрішній червоний багатокутник (polygon).

```
> polygon(sin(a)*0.1,cos(a)*0.1,col="red")
```

Створити новий інструмент...	Застосовується для створення інструментів користувача.
Керування інструментами...	Визначається назвою.
<i>Розділ «Вікно»</i>	
Нове вікно <Ctrl+F>	Використовується для відкриття нового вікна програми.
<i>Розділ «Допомога»</i>	
Допомога <F1>	Обирання пункту спричиняє до відкриття он-лайн-довідки у вікні браузера. Натискання клавіші <F1> призводить до виклику діалогового вікна "Опис", яке містить коротку довідку з використання обраного інструменту, а також кнопку, за допомогою якої можна перейти до он-лайн-довідки.
Уроки	Використовується для отримання он-лайн-довідки.
GeoGebraTube	Застосовується для відкриття GeoGebraTube у вікні браузера.
Повідомити про помилку	Використовується для відкриття веб-сторінки, на якій Ви маєте можливість повідомити про помилки у роботі програми.
Про програму / Ліцензія	При обиранні цього пункту відкривається вікно «Про програму / Ліцензія». За його допомогою можна ознайомитись із ліцензійною угодою, складом розробників та перекладачів.

2.5.12. Практикум з розв'язування математичних задач та створення комп'ютерних моделей

Завдання №1. У середовищі GeoGebra побудуйте графіки наступних функцій:

а) $y = 3x - 2$

б) $y = x^2 + 6x - 2$

в) $y = x^2 - 4x + 4$

г) $y = |x^3 - 4|x| + 5|$

д) $y = 5^x \cdot 5^{|x|}$

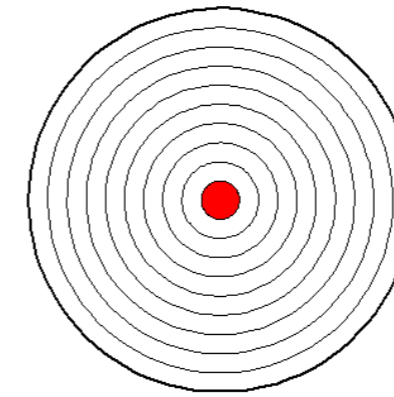
е) $y = \frac{1}{\sqrt{2x^2 + 4x + 9}}$

Технологія виконання

1. Запустіть програму GeoGebra.
2. Побудуйте графік функції $y = 3x - 2$, виконавши наступні дії:
 - у рядку (полі) «Ввод:» введіть формулу $3x - 2$ і натисніть клавішу <Enter>;

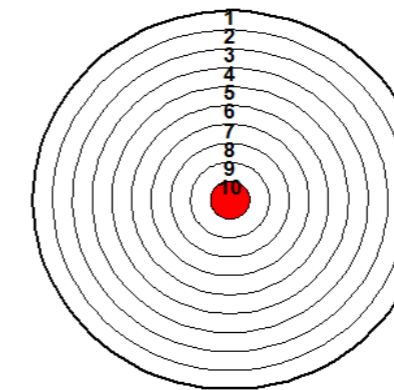
Продовження таблиці 2.14

Полотно <Ctrl+Shift+1>	Відображає (або ховає) вікно Полотно.
Полотно 2 <Ctrl+Shift+2>	Відображає (або ховає) вікно Полотно 2.
Протокол побудови <Ctrl+Shift+L>	Відображає (або ховає) вікно Протокол побудови.
Клавіатура	Виводить на екран (або ховає) віртуальну клавіатуру.
Рядок вводу	Додає до вікна програми (або ховає) рядок вводу.
Розмітка ...	Виводить на екран діалогове вікно Налаштування з активною вкладкою (Розмітка).
Оновити <Ctrl+F>	Визначається назвою.
Обчислити заново <Ctrl+R>	Визначається назвою.
<i>Розділ «Налаштування»</i>	
Опис об'єктів	Містить три пункти підменю (<i>Значення</i> , <i>Визначення</i> , <i>Команда</i>), які дозволяють обрати тип опису об'єктів.
Прив'язка точки	Містить пункти підменю ("Автоматично", "Прив'язати до сітки", "Фіксація у вузлах сітки", "Вимкнути"), що дозволяють обрати тип прив'язки точки до сітки або вимкнути її.
Округлення	Дозволяє визначити точність округлення (кількість десяткових знаків або значущих цифр).
Позначення	Містить пункти підменю: "Автоматично", "Показувати", "Не показувати", "Тільки для точок". За їх допомогою користувач має можливість обрати чи показувати позначення об'єктів, що будуть створені (пункт "Показувати") чи ні (пункт "Не показувати"). Якщо користувач обере пункт "Тільки для точок", то на полотні будуть відображатись тільки позначення точок.
Розмір шрифту	Дозволяє обрати розмір шрифту інтерфейсу.
Мова	За допомогою даного пункту меню можна змінити мову інтерфейсу
Додатково ...	Використовується для виклику діалогового вікна "Налаштування", за допомогою якого можна змінити налаштування програми.
Зберегти налаштування	Визначається назвою.
Відновити налаштування	Визначається назвою.
<i>Розділ «Інструменти»</i>	
Налаштування...	Використовується для налаштування панелі інструментів.



- точки позначити текстовими мітками, починаючи від 10 до 1 (text).

Чим ближче до центру, тим більше значення
 > for (i in 1:10) text(x=0,y=i/10-0.025,labels=11-i,font=2)



Тепер два лучника зробили по 10 пострілів у мішень. Ми імітуємо їхні постріли, генеруючи нормальний розподіл (x, y) чисел, з математичним очікуванням = 0 (центр!) і де досвід лучника імітується за допомогою стандартного відхилення. Лучник більш досвідчений, чим ближче стріли будуть до центру, тобто чим нижче стандартне відхилення.

- R-вираз `rnorm` генерує нормально розподілені числа; нам потрібні 20 з них, які розташовані у вигляді матриці з двома стовпчиками.

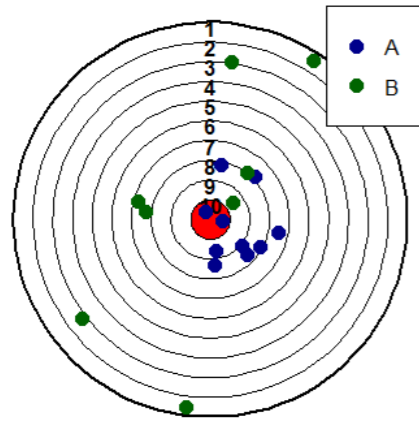
```
> shots1 <- matrix(ncol=2, data=rnorm(n=20,sd=0.2))
> shots2 <- matrix(ncol=2, data=rnorm(n=20,sd=0.5))
```

- Постріли додаються до фігури як точки (points), розфарбовуються синім `darkblue` (досвідчений стрілець) і темно-зеленим `darkgreen` (початкового рівня стрілець). Зверніть увагу, що ми вибираємо на 50% збільшений розмір точок (сех), і ми вибираємо точки круглої форми (pch = 16)

```
> points(shots1,col="darkblue",pch=16,cex=1.5)
> points(shots2,col="darkgreen",pch=16,cex=1.5)
```

• Нарешті, ми додаємо легенду, пояснюючи, хто зробив постріли:

```
> legend("topright", legend=c("A", "B"), pch=16,
+       col=c("darkblue", "darkgreen"), pt.cex=1.5)
```



Зверніть увагу, що текст і колір легенди вводяться у вигляді вектора рядків, використовуючи функцію «с ()» (наприклад, С («А», «В»)).

2.1.16. X-Y графіки – умовна побудова

У якості більш витонченої демонстрації використання символів в R-графіках, ми будемо працювати на біологічному прикладі, із набору даних R під назвою «Orange».


Цей набір даних містить окружності (у мм, на висоті грудей), виміряних в різному віці на п'яти апельсинових деревах. Ми почнемо з розгляду даних (відображається тільки частина).

```
> head(Orange)
  Tree age circumference
1  1 118          30
2  1 484          58
3  1 664          87
4  1 1004         115
5  1 1231         120
6  1 1372         142
```

```
> tail(Orange)
  Tree age circumference
30  5 484          49
31  5 664          81
32  5 1004         125
33  5 1231         142
```


Зберегти <Ctrl+S>	Відкриває діалогове вікно «Зберегти» при першому збереженні проекту та зберігає зміни у раніше створеному та збереженому документі.
Зберегти як ...	Відкриває діалогове вікно «Зберегти».
Поділитися ...	Застосовується для завантаження проекту на GeoGebraTube.
Експортувати	Містить перелік пунктів підменю для експорту файлу GeoGebra в інші формати.
Друк. Попередній перегляд <Ctrl+P>	Зрозуміло з назви пункту.
Закрити <Alt+F4>	Зрозуміло з назви пункту.
<i>Розділ «Правка»</i>	
Скасувати <Ctrl+Z>	Скасовує останню команду (дію) чи кілька команд (дій).
Повторити <Ctrl+Y>	Повертає скасовані команди (дії).
Копіювати <Ctrl+C>	Виконує команду копіювання виділеного об'єкта.
Вставити <Ctrl+V>	При обиранні пункту виконується команда вставки з буфера обміну.
Копіювати полотно до буфера обміну <Ctrl+Shift+C>	Обирання даного пункту призводить до копіювання вмісту полотна до буферу обміну.
Вставити зображення з	Містить пункти підменю «Файл» (для вставки зображення з файлу) та «Буфер обміну» (для вставки зображення з буфера обміну).
Властивості <Ctrl+E>	Обирання даного пункту призводить до виклику діалогового вікна «Налаштування» за допомогою якого можна змінити властивості різних об'єктів.
Вибрати все	При виборі цього пункту виділяються всі об'єкти, які побудовані на полотні
Вибрати поточний шар	Визначається назвою
Вибрати нащадків	Визначається назвою
Вибрати предків	Визначається назвою.
Обернути виділення	Використовується для виділення невиділених об'єктів і зняття виділення з виділених
Показати / сховати об'єкти	Визначається назвою.
Показати / сховати позначення	Визначається назвою.
Видалити	Використовується для видалення об'єктів.
<i>Розділ «Вид»</i>	
Панель об'єктів <Ctrl+Shift+A>	Додає вікно (або ховає) Панель об'єктів.
Таблиця <Ctrl+Shift+S>	Додає (або ховає) вікно Таблиця.
CAS <Ctrl+Shift+K>	Виводить на екран (або ховає) вікно системи комп'ютерної алгебри – CAS.

- оберіть інший інструмент;
- об'єкт (об'єкти) з полотна зникнуть;
- щоб показати приховані об'єкти, знов оберіть інструмент.

 - Показати / Сховати позначення.

Технологія використання:

- оберіть вказаний інструмент ;
- клацніть на об'єкті, позначення якого треба сховати (показати).

 - Копіювати стиль. Цей інструмент дозволяє копіювати властивості (колір, товщину ліній, ...) з одного об'єкта на інші.

Технологія використання:

- виділіть об'єкт стиль якого необхідно скопіювати;
- оберіть вказаний інструмент;
- натисніть на об'єкті до якого вказаний стиль потрібно застосувати.

 - Вилучити. Цей інструмент використовується для вилучення побудованих об'єктів.

Технологія використання:

- оберіть вказаний інструмент;
- виділіть об'єкт, який необхідно вилучити;

2.5.11. Меню GeoGebra

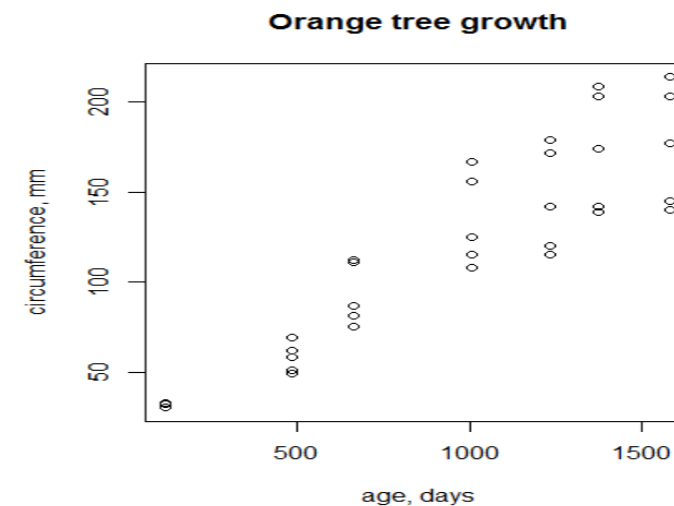
Рядок меню програми GeoGebra складається з семи розділів: «Файл», «Правка», «Вид», «Опції», «Інструменти», «Вікно», «Допомога». Розгля-немо пункти меню кожного з розділів, скориставшись таблицею 2.14.

Таблиця 2.14.

Назва пункту меню <комбінація клавіш>	Призначення
<i>Розділ «Файл»</i>	
Нове вікно <Ctrl+N>	Використовується для відкриття нового вікна GeoGebra.
Створити	Застосовується з метою створення нового проекту (файлу GeoGebra).
Відкрити ... <Ctrl+O>	Використовується для відкриття раніше створеного файлу GeoGebra.
Відкрити з GeoGebraTube	Дозволяє відкрити документ, що зберігається на GeoGebraTube.
Нещодавно відкриті	Містить перелік файлів, які відкривались останнім часом.

```
34 5 1372 174
35 5 1582 177
```

```
і зробимо грубий графік окружностей в залежності від віку:
> plot(Orange$age, Orange$circumference,xlab="age, days",
ylab="circumference, mm", main= "Orange tree growth")
```



(Оскільки Orange є data.frame, колонки можуть бути розглянуті за їх іменами, Orange\$age і Orange\$circumference).

Результат (рисунок) показує, що існує великий розкид даних, що пов'язано з тим, що п'ять дерев не ростуть з однією швидкістю.

Корисно побудувати залежність між окружністю і віком окремо для кожного дерева. В R, це зробити просто: ми можемо використати деякі графічні параметри (типи символів, кольору, розміру, ...) в залежності від певних «факторів». Фактори відіграють дуже важливу роль в статистичних використаннях R. Для нашого застосування, досить знати, що фактори є цілими числами, починаючи з 1.

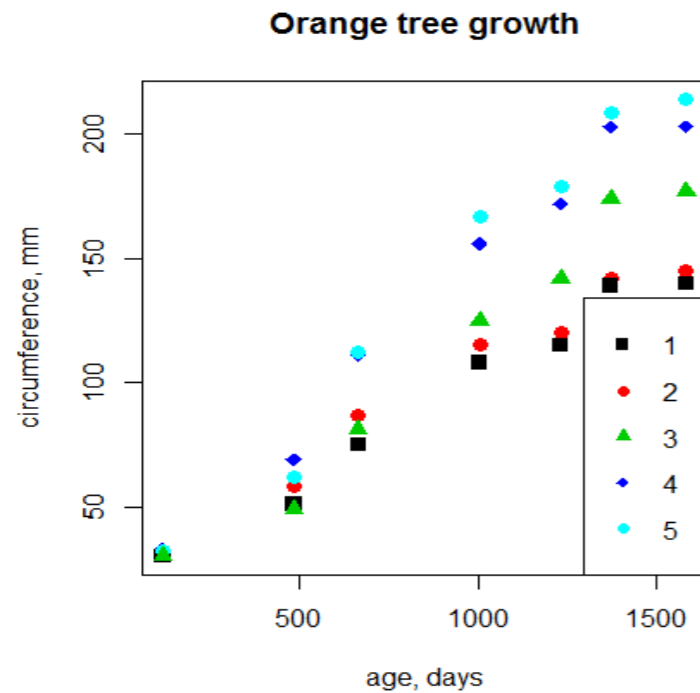
В R-виразі нижче, ми просто використовуємо різні символи (pch) і кольору (col) для кожного дерева: pch=(15:20)[Orange\$Tree] означає, що, в залежності від значення Orange\$Tree (тобто номера дерева), символ (pch) прийме значення 15 (дерево = 1), 16 (дерево = 2), ... 20 (дерево = 5). col=(1:5) [Orange\$Tree] робить те ж саме для кольору точок. В останньому виразі додається легенда, розташована у правому нижньому куті.

```
> plot(Orange$age, Orange$circumference, xlab="age, days",
ylab="circumference, mm", main= "Orange tree growth", pch=(15:20)[Orange$Tree],
col=(1:5) [Orange$Tree],cex=1.3)
```

```
> legend("bottomright",pch=15:20,col=1:5,legend=1:5)
```

Результат (рисунок) показує, що дерево число номер 5 росте швидко, а дерево номер 1 повільно зростає.

(Зверніть увагу: корисно виконати приклади з файлу довідки Orange.)



2.1.17. Темпи росту зоопланктону

«Zoogrowth» з пакету «marelac» представляє собою набір літературних даних, складений Хансена та ін. (1997) з результатами вимірювань максимальних швидкостей росту організмів зоопланктону в залежності від об'єму тіла. Запустіть приклад для цього набору даних (вам спочатку потрібно буде завантажити пакет «marelac» виконавши `install.packages(«marelac»)`):

```
> require(marelacTeaching)
> example(Zoogrowth)
```

2.1.18. Зображення та контурні графіки

R має деякі дуже потужні функції для створення зображень і додавання контурів. Наприклад, набір даних «Bathymetry» (сукупність даних про глибини водного об'єкта) з пакету `marelac` може використовуватися для генерації батиметрії (і гіпсометрії) Світового океану (і землі):

```
> require(marelac)
> image(Bathymetry$x, Bathymetry$y, Bathymetry$z, col=femmecol(100,
asp=TRUE, xlab="", ylab="")
> contour(Bathymetry$x, Bathymetry$y, Bathymetry$z, add=TRUE)
```

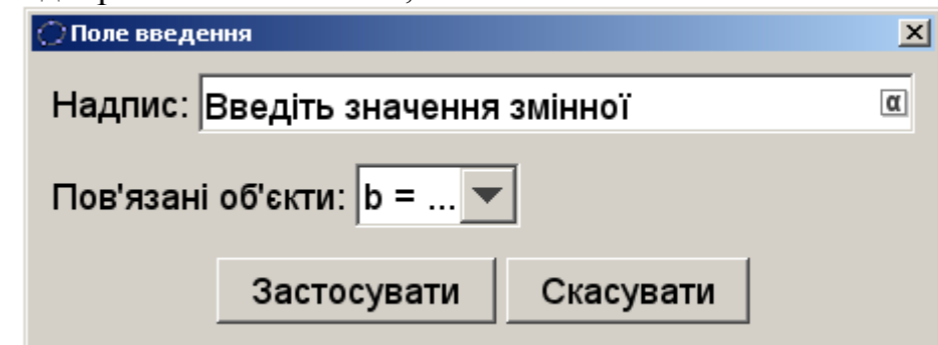
Зверніть увагу на використання «`asp=TRUE`», яка підтримує співвідношення сторін.

– натисніть на кнопці **ЗАСТОСУВАТИ**.

a=1 - Поле введення. Використовується для введення значень величин

Технологія використання:

- оберіть необхідний інструмент ;
- клацніть на полотні там, де повинно знаходитись поле;
- у діалоговому вікні введіть (якщо потрібно) напис, що буде відображатись біля поля;



- оберіть об'єкт;
- натисніть на кнопці **ЗАСТОСУВАТИ**.

↕ - Переміщення полотна. Застосовується для переміщення полотна (зміни масштабу осі координат).

Технологія використання:

- оберіть відповідний інструмент ;
- розташуйте курсор мишки у потрібному місці на полотні (розташуйте курсор на осі координат);
- натисніть ліву кнопку миші і, не відпускаючи її, перемістіть полотно (переміщуйте мишку при натиснутій лівій кнопці до встановлення потрібного масштабу на обраній осі координат);
- відпустіть кнопку.

🔍 - Збільшити. Використовується для збільшення масштабу.

Технологія використання:

- оберіть інструмент «Збільшити»;
- клацніть на полотні (щоб продовжити, клацніть ще).

🔍 - Зменшити. Використовується для зменшення масштабу.

Технологія використання:


- оберіть інструмент «Зменшити»;
- клацніть на полотні (щоб продовжити, клацніть ще).

🔍 - Показати / Сховати об'єкт

Технологія використання:

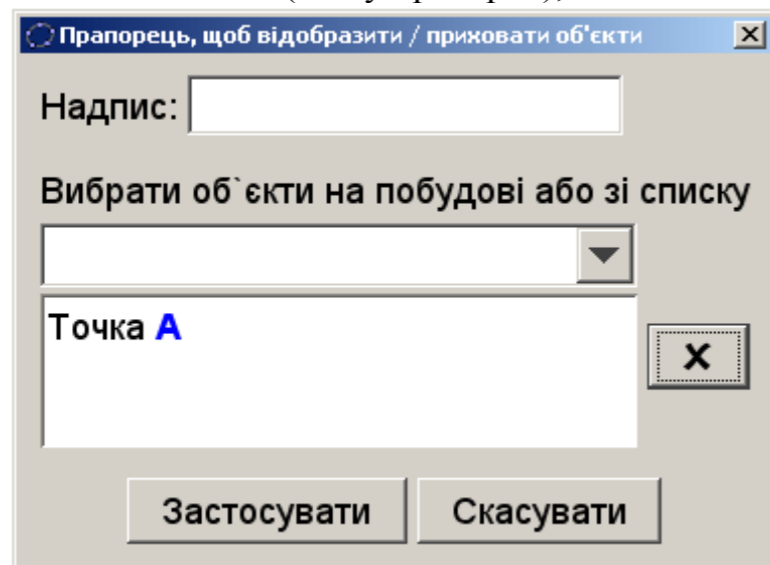
- оберіть даний інструмент;
- клацніть на потрібному об'єкті або кількох;

– клацніть на кнопці **ЗАСТОСУВАТИ**.

 - Прапорець. Застосовується для створення прапорців, які використовуються у випадках, коли тимчасово потрібно сховати або показати (тимчасово сховані) об'єкти, що побудовані на полотні.

Технологія використання:

- оберіть вказаний інструмент;
- клацніть на полотні там, де прапорець повинен знаходитись;
- за допомогою діалогового вікна введіть, якщо це потрібно, до текстового поля текст (назву прапорця);

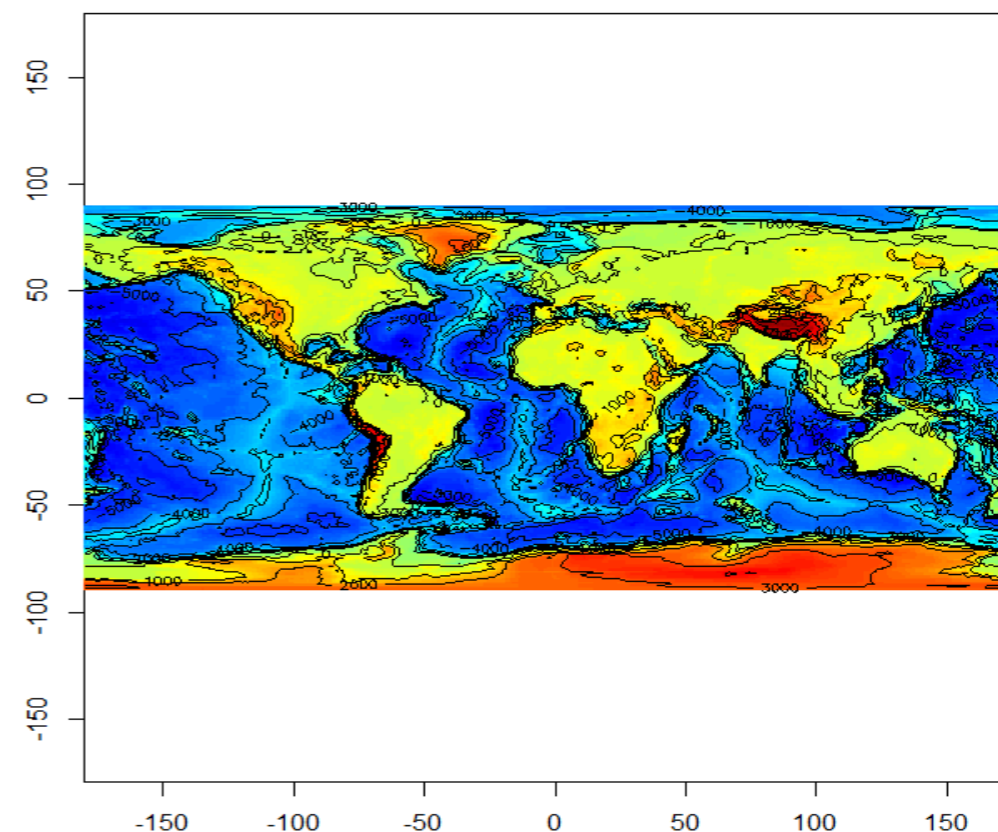
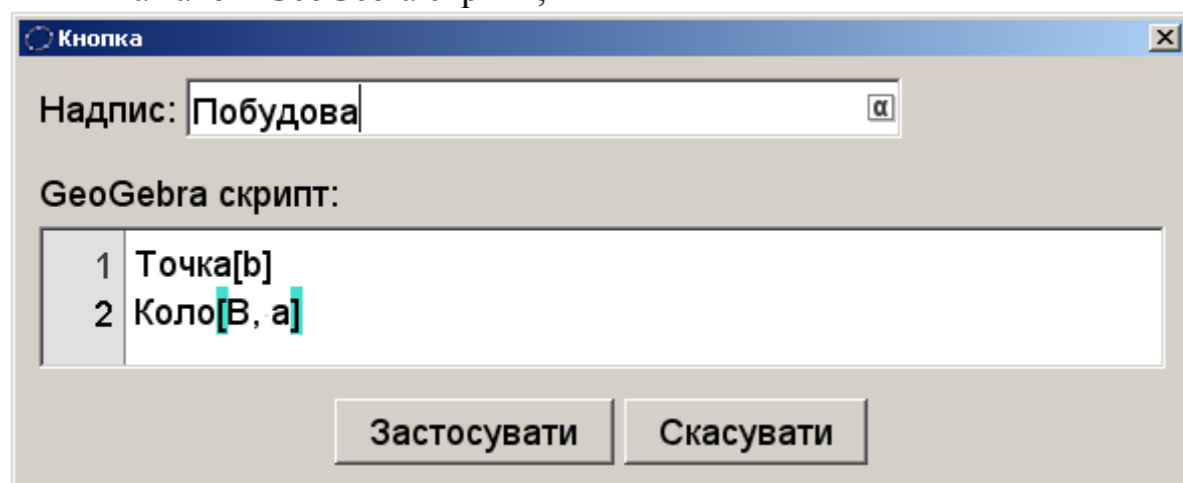


- оберіть об'єкти, які потрібно сховати або показати;
- клацніть на кнопці **ЗАСТОСУВАТИ**.

 - Кнопка. За допомогою інструменту створюються об'єкти «Кнопка», що використовуються для запуску GeoGebra скриптів.

Технологія використання:

- оберіть даний інструмент ;
- клацніть на полотні там, де повинна знаходитись кнопка;
- у діалоговому вікні введіть напис, що буде відображатись на кнопці, а також GeoGebra скрипт;



2.1.19. Побудова математичних функцій

Зображення графіків математичних функцій будуються швидко за допомогою R-команди «curve»:

```
> curve(sin(3*pi*x))
```

Намалює графік для $y=\sin(3\pi x)$, використовуючи налаштування за замовчуванням (рис. ліворуч), в той час як:

```
> curve(sin(3*pi*x),from=0,to=2,col="blue",
```

```
+ xlab="x",ylab="f(x)",main="curve")
```

```
> curve(cos(3*pi*x),add=TRUE,col="red",lty=2)
```

спочатку малює графік $y=\sin(3\pi x)$, синього кольору (col), і для значень x в діапазоні від 0 до 2 (from, to), додаючи назву графіка (main) та назви для X і Y осей (xlab, ylab) (перша команда).

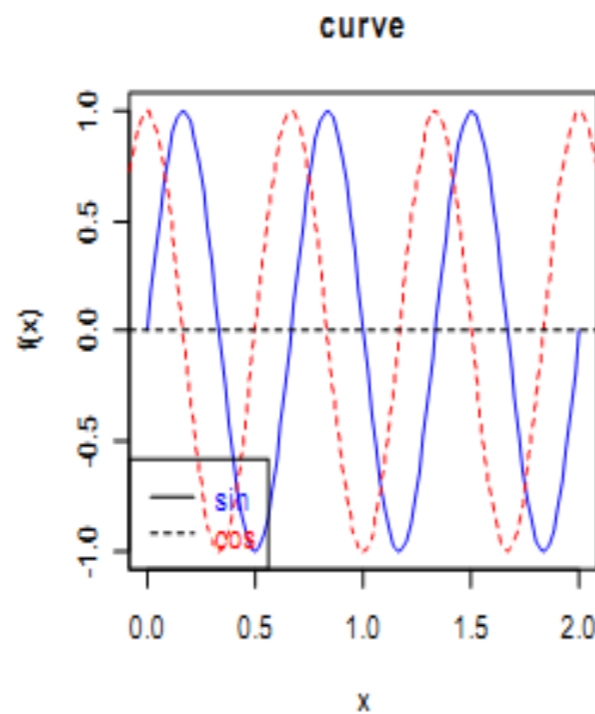
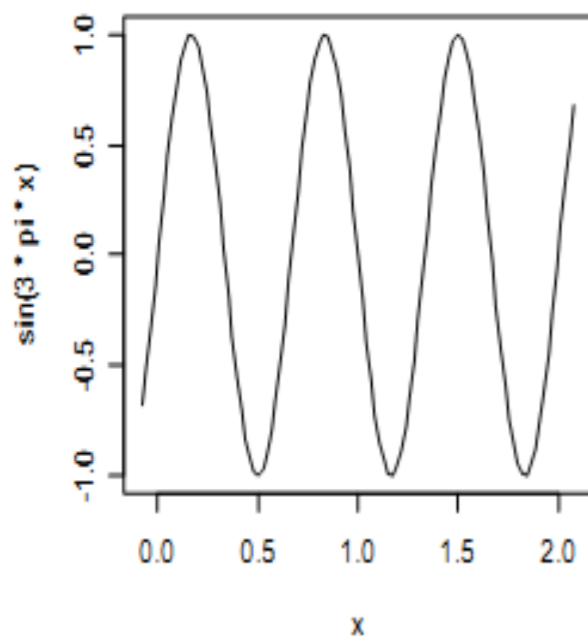
Друга R-команда додає графік функції $y=\cos(3\pi x)$, червоного кольору (col) пунктирною лінією (lty). Зверніть увагу на використання параметра «add=TRUE», оскільки команда «curve» за замовчуванням створює новий графік.

Остання команда додає вісь X, пунктирною (lty=2) лінією (abline) при $y = 0$ і легенду.

```
> abline(h=0,lty=2)
```

```
> legend("bottomleft",c("sin","cos"),
```

```
+ text.col=c("blue","red"),lty=1:2)
```



Декілька фігур в одному вікні

Є декілька способів, як можна відобразити декілька фігур на одному графіку.

1. Найпростіший, вказати кількість фігур у рядку (mfrow) і в стовпчику (mfcol):

```
> par(mfrow=c(3,2))
```

організовує наступні графіки в 3 рядки і 2 колонки. Графіки будуть побудовані по-рядково.

```
> par(mfcol=c(3,2))
```

організує наступні графіки в 3 колонки і 2 рядки по стовпцях.

Слід зазначити, що як mfrow так і mfcol повинні бути введені в якості вектора. Спробуйте:

```
> par(mfrow=c(2,2))
> for ( i in 1:4) curve(sin(i*pi*x),0,1,main=i)
```



- Фігура від руки. Застосовується для побудови ліній та відрізків від руки.

Технологія використання:

- оберіть вказаний інструмент ;
- виконайте операцію протягування мишки для побудови.

Примітка:

Точки побудованої фігури не повинні містити точок з однаковими абсцисами.



- Відношення. Використовується для отримання даних про взаємозв'язок між двома об'єктами.

Технологія використання:

- оберіть даний інструмент ;
- клацніть на одному об'єкті, а потім на другому;
- познайомтесь із даними у вікні, що з'явиться.



- Калькулятор ймовірності. Викликає діалогове вікно «Калькулятор ймовірностей», що містить набір інструментів для використання при розв'язуванні задач за допомогою теорії ймовірності та математичної статистики.



- Інспектор функцій. Використовується для виклику діалогового вікна «Інспектор функцій», що містить набір інструментів для роботи з функцією, графік якої побудовано на полотні.

Технологія використання:

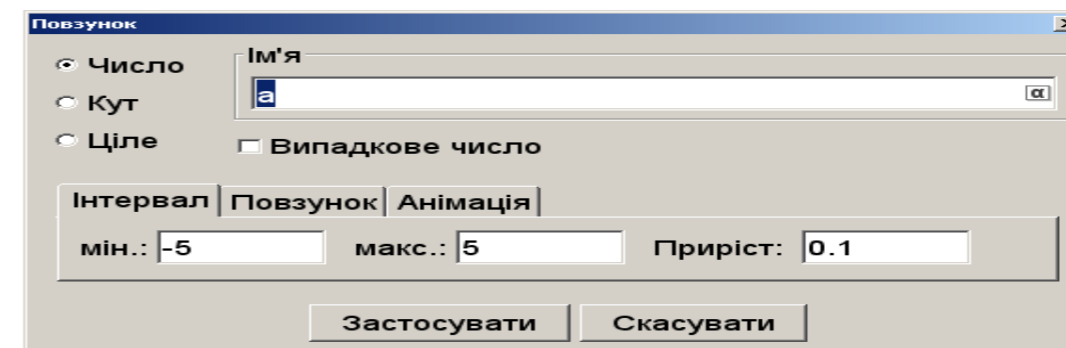
- оберіть відповідний інструмент ;
- клацніть на графіку потрібної функції;
- за допомогою діалогового вікна «Інспектор функцій», яке відкриється, скористайтеся інструментами, що потрібні для розв'язування вашої задачі;
- закрийте діалогове вікно.



- Повзунок. Дозволяє створити графічне представлення змінної., значенням якої є число або міра кута.

Технологія використання:

- оберіть інструмент «Повзунок»;
- клацніть на полотні там, де повзунок повинен знаходитись;
- за допомогою діалогового вікна введіть значення параметрів:





- Гомотетія відносно точки

Технологія використання:

- оберіть даний інструмент;
- оберіть фігуру;
- оберіть центр гомотетії;
- до текстового поля діалогового вікна, що з'явиться, введіть коефіцієнт гомотетії;
- клацніть на кнопці **ОК**.



- Вставити текст. За допомогою цього інструменту можна створити статичний або динамічний текст, а також формули системи LaTeX.

Технологія використання:

- оберіть вказаний інструмент ;
- клацніть на полотні там, де повинен знаходитись напис;
- за допомогою діалогового вікна «Текст», що з'явиться, введіть до текстового поля «Правка» напис;
- клацніть на кнопці **ОК**.

Примітка

Для введення тексту у вигляді формули потрібно у діалоговому вікні поставити прапорець «LaTeX-формула».

Примітка

Статичний текст не залежить від якихось математичних об'єктів і, як правило, на нього не впливають зміни побудованих об'єктів.

Динамічний текст містить значення певних властивостей об'єктів, які автоматично адаптуються до змін у цих об'єктах.

Змішаний текст являє собою поєднання статичного і динамічного тексту.

У змішаному тексті статична частина повинна бути в лапках. Різні частини тексту (наприклад, статичні і динамічні частини) повинні бути з'єднані знаком плюс «+».



- Зображення. Використовується для вставки рисунка з файлу.

Технологія використання:

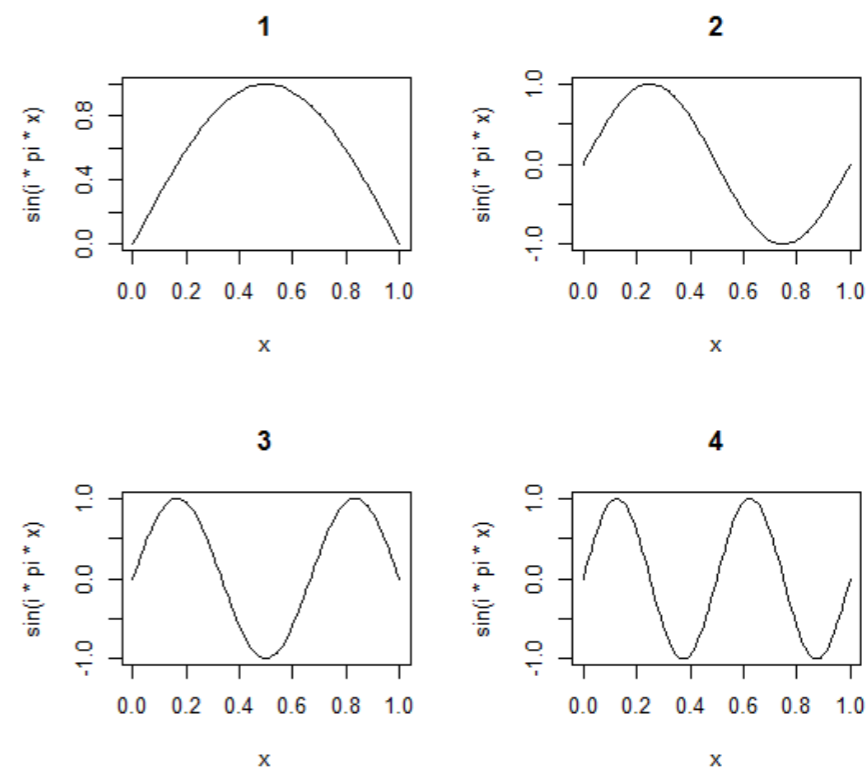
- оберіть відповідний інструмент ;
- клацніть на полотні там, де повинен знаходитись рисунок;
- за допомогою діалогового вікна, що з'явиться, оберіть потрібний рисунок;
- клацніть на кнопці < Відкрити >.



- Перо. Використовується для побудови довільної лінії.

Технологія використання:

- оберіть даний інструмент ;
- виконайте операцію протягування мишки для побудови лінії.



2. R-функція «layout» дозволяє набагато більш складні налаштування відображення декількох графіків.

2.1.20. Робота з матрицями

Працювати з матрицями в R – дуже просто. Практично все можливо! Ось найбільш важливі R-функції, які працюють на матрицях:

- `%*%` множення матриць
- `t(A)` транспонування матриці A
- `diag(A)` діагональна матриця до A
- `solve(A)` обернена матриця до A
- `solve(A,B)` розв'язок матричного рівняння Ax=B відносно x
- `eigen(A)` власні значення і власні вектори матриці A
- `det(A)` визначник матриці A

Наприклад:

```
> (A <-matrix(nrow=2,data=c(1,2,3,4)))
```

```

     [,1] [,2]
[1,]  1   3
[2,]  2   4

```

```
> solve(A) %*% A
```

спочатку обчислюється матриця обернена до A ($\text{solve}(A)$), і множиться на A ($\%*\%$), результатом є одинична матриця:

```
[,1] [,2]
[1,] 1 0
[2,] 0 1
```

в той час як $t(A)$ виконає транспонування матриці A (перестановка рядків і стовпців).

```
> t(A)
[,1] [,2]
[1,] 1 2
[2,] 3 4
```

Наступний набір операторів розв'яже лінійну систему $Ax = B$ для невідомого вектора x :

```
> B <- c(5,6)
> solve(A,B)
[1] -1 2
```

Нарешті, власні значення і власні вектори матриці A оцінюються з використанням R -функції «eigen». Ця функція повертає список, що містить як власні значення ($\$values$) так і власні вектори ($\$vectors$), (стовбчики).

```
> eigen(A)
$values
[1] 5.3722813 -0.3722813
```

```
$vectors
[,1] [,2]
[1,] -0.5657675 -0.9093767
[2,] -0.8245648 0.4159736
```

2.1.21. Корінь функції однієї змінної

Припустимо, ми хочемо розв'язати наступну задачу: $\cos(x) = 2*x$ відносно x . Математично, ми шукаємо корінь функції $y = \cos(x) - 2*x$, це значення x , при яких $y = 0$. Оскільки, функція є досить складною, то не можливо знайти точний розв'язок (явний вираз) для цього кореня. Це завжди гарна ідея, розв'язати дане рівняння графічним методом.

```
> curve(cos(x)-2*x,-10,10)
> abline(h=0,lty=2)
```


- Нахил прямої. Дозволяє визначити кутовий коефіцієнт прямої.


Технологія використання:

- оберіть даний інструмент;
- клацніть на прямій (промені, відрізьку).


- Створити список.


Технологія використання:

- оберіть відповідний інструмент;
- виділіть об'єкти (об'єкт).


- Симетрія відносно прямої. Використовується для побудови фігури, симетричної даній, відносно обраної прямої.


Технологія використання:

- оберіть вказаний інструмент;
- оберіть фігуру;
- клацніть на прямій (відрізьку, промені).


- Симетрія відносно точки. Використовується для побудови фігури, симетричної даній, відносно обраної точки.

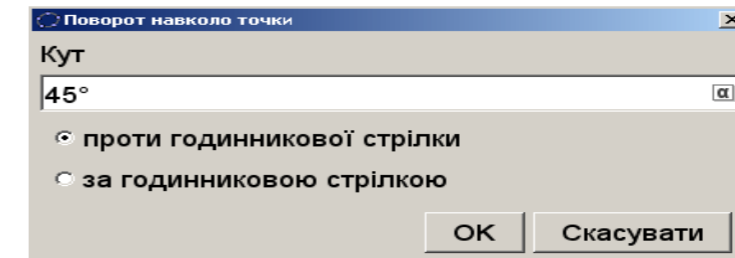
Технологія використання:

- оберіть вказаний інструмент;
- оберіть фігуру;
- клацніть на точці (центрі симетрії).


- Поворот навколо точки

Технологія використання:

- оберіть даний інструмент ;
- оберіть фігуру;
- оберіть центр повороту (клацніть на точці);
- до текстового поля діалогового вікна, що з'явиться, введіть значення кута:



- клацніть на кнопці ОК.


- Паралельне перенесення на вектор

Технологія використання:

- оберіть інструмент ;
- оберіть фігуру, а потім вектор.

- оберіть три точки, щоб створити кут між ними (друга обрана точка буде вершиною кута).


Кут між двома відрізками (прямими, променями, векторами):

- оберіть інструмент «Кут»;
- оберіть два відрізки (прямі, промені, вектори).

Примітки:

Якщо багатокутник був створений з вибором вершин «проти годинникової стрілки», інструмент «Кут» створить внутрішні кути багатокутника.


Кути створюються в напрямку «проти годинникової стрілки». Таким чином, порядок вибору об'єктів має значення для інструменту «Кут». Якщо ви хочете обмежити максимальний розмір кута до 180° , зніміть прапорець «Дозволити зовнішній кут» на вкладці «Основні» діалогового вікна «Властивості».

 - Кут заданої величини. Використовується для побудови кутів з визначеною користувачем градусною мірою.

Технологія використання:


- оберіть інструмент «Кут заданої величини»;
- оберіть дві точки А і В;
- введіть розмір кута до текстового поля вікна, що з'явиться, та оберіть напрямок для відкладання кута (проти годинникової стрілки чи за годинниковою стрілкою);
- клацніть на кнопці .

Примітка: Цей інструмент створює точку С і кут α , де α – кут АВС.

 - Кут заданої величини. Використовується для побудови кутів з визначеною користувачем градусною мірою.


Технологія використання:

- оберіть інструмент «Кут заданої величини»;
- оберіть дві точки А і В;
- введіть розмір кута до текстового поля вікна, що з'явиться, та оберіть напрямок для відкладання кута (проти годинникової стрілки чи за годинниковою стрілкою);
- клацніть на кнопці .

 - Відстань або довжика. Застосовується для визначення відстані між двома точками, довжини відрізка (кола, еліпса), периметра багатокутника.

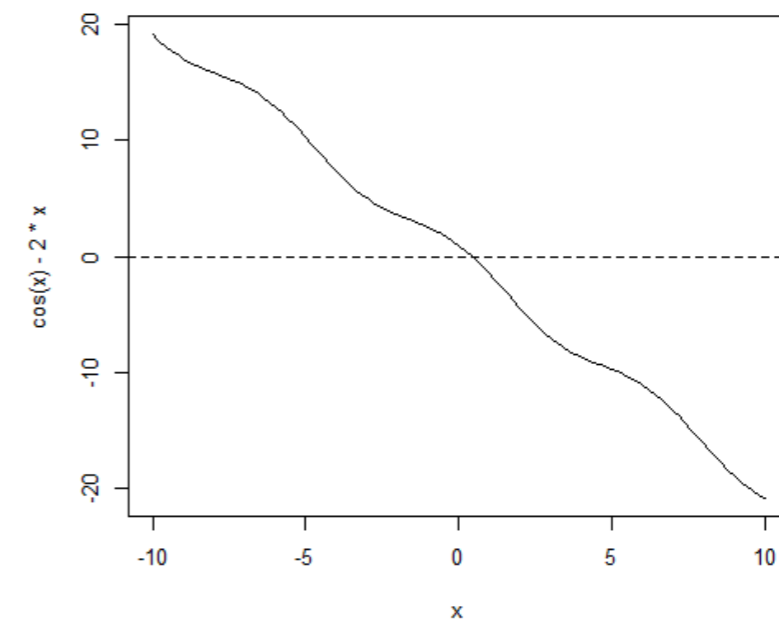
Технологія використання:

- оберіть відповідний інструмент;
- вкажіть дві точки (відрізок, коло, еліпс, багатокутник).

 - Площа. Використовується для визначення площі багатокутника, кола або еліпса.

Технологія використання:

- оберіть даний інструмент ;
- вкажіть багатокутник (коло, еліпс).



З графіка видно, що насправді існує таке значення x , для якого $Y = 0$.

Тепер можна використати R-функцію «uniroot», щоб знайти це значення.

Функції, які шукають корінь нелінійного рівняння як правило, працюють «ітераційно», тобто вони наближаються все ближче і ближче до кореня поетапно (ітераційно).

Ці методи, як правило, не мають можливості, щоб знайти цей корінь точно, а знаходять корінь з певною точністю (tol, дуже маленьке значення). Більш конкретно: коренями $y = \cos x - 2 * x$ є значення x , для яких $|\cos x - 2 * x| < tol$, або для яких послідовні зміни $x \in < tol$

Для того, щоб метод працював, повинен бути принаймні один корінь на інтервалі. Вираз нижче знаходить корені рівняння; він повертає кілька значень, у вигляді списку.

```
> uniroot(f = function(x) cos(x)-2*x, interval=c(-10,10))
```

```
$root
[1] 0.4501686
```

```
$f.root
[1] 3.655945e-05
```

```
$iter
[1] 5
```

```
$init.it
[1] NA
```

```
$estim.prec
[1] 6.103516e-05
```

Найважливіше значення сам корінь ($\$root$), який є 0.4501686; значення функції в корені було $3.66e-5$, метод виконав 5 ітерацій.

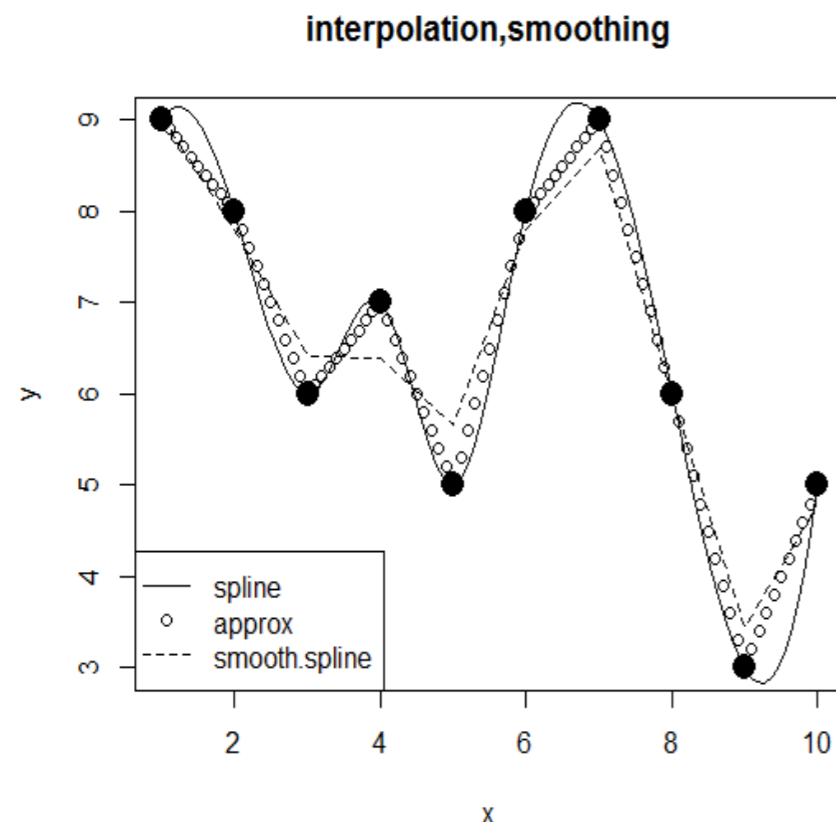
2.1.22. Інтерполяція, згладжування

Інтерполяції і згладжування в R можна зробити декількома способами:

- `approx` лінійно інтерполює через точки
- `spline` використовує інтерполяцію сплайн, яка є більш гладкою
- `smooth.spline` розгладжує набори даних; це означає, що цей метод не пов'язує оригінальні точки.

Використання цих функцій ілюструється наступним сценарієм і відповідним результатом:

```
> x <- 1:10
> y <- c(9,8,6,7,5,8,9,6,3,5)
> plot(x,y,pch=16,cex=2,main="interpolation,smoothing")
> lines(spline(x,y,n=100),lty=1)
> points(approx(x,y,xout=seq(1,10,0.1)),pch=1)
> lines(smooth.spline(x,y),lty=2)
> legend("bottomleft",lty=c(1,NA,2),pch=c(NA,1,NA),
+ legend=c("spline","approx","smooth.spline"))
```



- оберіть вказаний інструмент ;
- оберіть центр кола;
- оберіть початок дуги, що відповідає руху проти годинникової стрілки;
- оберіть точку для визначення довжини дуги.



- Дуга за трьома точками

Технологія використання:

- оберіть вказаний інструмент;
- клацніть у трьох точках, що належатимуть майбутній дузі.



- Сектор за центром та двома точками

Технологія використання:

- оберіть інструмент «Сектор за центром та двома точками»;
- оберіть центр кола;
- оберіть початок дуги, що відповідає сектору;
- оберіть точку для визначення довжини дуги сектора.



- Сектор за трьома точками

Технологія використання:

- оберіть інструмент «Сектор за трьома точками»;
- оберіть початок дуги, що відповідає сектору;
- оберіть точку кола, що належить сектору;
- оберіть кінець дуги, що відповідає сектору.



- Еліпс. Використовується для побудови еліпса за його фокусами та точкою.

Технологія використання:

- оберіть інструмент «Еліпс»;
- клацніть у точках, що є фокусами еліпса;
- клацніть у точці, що належатиме еліпсу.



- Коніка за п'ятьма точками. Використовується для побудови конічного перерізу за його п'ятьма точками.

Технологія використання:

- оберіть даний інструмент ;
- вкажіть п'ять точок, що належатимуть майбутньому конічному перерізу.




- Кут. Застосовується для створення кутів (позначення, стиль, колір ...).

Технологія використання:

Кут між трьома точками:


- оберіть інструмент «Кут»;

- оберіть необхідний інструмент ;
- клацніть по черзі на кожній вершині многокутника, а потім — ще раз на першій.

 - Векторний многокутник. Дозволяє побудувати многокутник всі вершини якого є кінцями векторів з початком у спільній точці.


Технологія використання:

- оберіть даний інструмент ;
- клацніть по черзі на кожній вершині многокутника, а потім — ще раз на першій.

 - Коло за центром та точкою на колі. Використовується для побудови кола за його центром і точкою, що належить колу.


Технологія використання:

- оберіть вказаний інструмент ;
- клацніть на точці, що буде центром кола;
- клацніть на точці, що належатиме колу.

 - Коло за центром та радіусом. Дозволяє побудувати коло за його центром та довжиною радіуса, яка вводиться з клавіатури.


Технологія використання:

- оберіть вказаний інструмент ;
- клацніть на точці, що буде центром кола;
- у вікні, що з'явиться, введіть довжину радіуса;
- натисніть кнопку .

 - Циркуль. Застосовується для побудови кола з радіусом, що дорівнює довжині обраного відрізка чи відстані між двома обраними точками та центром у даній точці.


Технологія використання:

- оберіть вказаний інструмент ;
- оберіть радіус кола (клацніть на відрізку або на двох точках, відстань між якими дорівнюватиме радіусу);
- оберіть центр кола.

 - Коло за трьома точками. Використовується для побудови кола що проходить через три задані точки на площині.


Технологія використання:

- оберіть вказаний інструмент ;
- клацніть у трьох точках, що належатимуть майбутньому колу.

 - Півколо за двома точками. Використовується для побудови півкола що проходить через дві задані точки на площині.

Технологія використання:

- оберіть вказаний інструмент ;
- клацніть у двох точках, що належатимуть майбутньому півколу.

 - Дуга за центром та двома точками.

Технологія використання:

2.1.23. Диференціальні рівняння

Диференціальні рівняння виражають швидкість зміни функції уздовж однієї або декількох розмірностей, зазвичай часу та/або простору.

Розглянемо наступну систему з двох диференціальних рівнянь:

$$\frac{dA}{dt} = r \cdot (x - A) - k \cdot A \cdot B$$

$$\frac{dB}{dt} = r \cdot (y - B) + k \cdot A \cdot B$$

- A і B називаються диференціальними змінними (або змінними стану),
- $\frac{dA}{dt}$ є похідною (або швидкістю зміни),
- r, x, y і k параметри (константи).

Для розв'язання систем диференціальних рівнянь в R ми визначимо функцію (тут називається model), яка має в якості вхідних параметрів час (t), значення змінних стану (state) і значення параметрів (parms). Ця функція просто обчислює швидкість зміни змінних стану (dA і dB) і повертає їх у вигляді списку. R-вираз «with (as.list (c(state,parms)),{ })» гарантує, що змінні стану і параметри можуть бути розглянуті за їх іменами.

```
> model <- function(t,state,parms)
+ {
+   with (as.list(c(state,parms)),
+   {
+     dA <- r*(x-A)-k*A*B
+     dB <- r*(y-B)+k*A*B
+     return (list(c(dA,dB)))
+   }
+ )
+ }
```

Перш ніж ми зможемо розв'язати цю модель, потрібно

- згенерувати послідовність значень часу, в якому ми хочемо отримати результат (times),
- задати початкові умови для змінних стану (state) та
- задати значення для параметрів (parms):

```
> times <- seq(0,300,1)
> state <- c(A=1,B=1)
> parms <- c(x =1, y = 0.1, k = 0.05, r = 0.05)
```

Модель тепер може бути розв'язана. Щоб зробити це, ми використовуємо функцію інтегрування `ode` з R, яку можна знайти в R-пакеті `deSolve`. Спочатку потрібно завантажити цей пакет.

```
> require(deSolve)
```

У кожен момент часу t , «`ode`» викликає функцію «`model`», з поточними значеннями змінних стану і значень параметрів. Результат зберігається в `data.frame`, що називається «`out`».

```
> out <- as.data.frame(ode(state,times,model,parms))
```

Все, що ми повинні зробити зараз, це побудувати графік моделі. Перш, ніж ми це зробимо, ми повинні переглянути отримані результати в `data.frame` «`out`»:

```
> head(out)
```

	time	A	B
1	0	1.0000000	1.0000000
2	1	0.9523189	1.0037869
3	2	0.9090687	1.0052854
4	3	0.8699226	1.0047151
5	4	0.8345728	1.0022854
6	5	0.8027203	0.9982009

Дані розташовані в три стовпчики: перший – час, наступні – значення A і B. Оскільки результат «`out`» це `data.frame`, то ми можемо мати доступ до даних, використовуючи їх імена (`out$time`, `out$A`, `out$B`).

Перед побудовою графіка моделі, діапазон значень A і B оцінюється; це виконується для того, щоб встановити межі осі Y (`ylim`). R-функція `plot` створює новий графік; `lines` додає лінію на цей графік; `lty` вибирає тип лінії; `lwd=2` робить лінії в два рази товще ніж за замовчуванням; `legend` додає легенду.

```
> ylim <- range(c(out$A,out$B))
> plot(out$time,out$A,xlab="time",ylab="concentration",
+      lwd=2,type="l",ylim=ylim,main="model")
> lines(out$time,out$B,lwd=2,lty=2)
> legend("topright",legend=c("A","B"),lwd=2,lty=c(1,2))
```

- якщо кут не побудовано, то клацання на трьох точках, наприклад A, B, C, навіть у випадку їх відсутності, призведе до побудови прямої, що містить бісектрису $\angle ABC$;

2-й спосіб

- клацніть на одній стороні кута, а потім — на другій;
- буде побудовано бісектриси кутів, утворених прямими, що містять сторони даного кута.



- Дотична. Застосовується для побудови дотичних до кінцевих перерізів та до графіків функцій.

Технологія використання:

Щоб побудувати дотичну до графіка функції у точці з абсцисою $x = x_0$, потрібно виконати наступні дії:

- обрати інструмент «Дотична»;
- клацнути на графіку функції;
- клацнути на точці з координатами $(x_0; y)$.

Для побудови дотичних до кінцевого перерізу, що проходять через дану точку, потрібно виконати наступну послідовність дій:

- обрати інструмент «Дотична»;
- клацнути на кінцевому перерізі;
- клацнути на даній точці (у випадку її відсутності вона буде створена).

Щоб побудувати дотичну (дотичні) до кінцевого перерізу, що буде паралельною (паралельними) до даної прямої, потрібно:

- обрати інструмент «Дотична»;
- клацнути на кінцевому перерізі, а потім — на даній прямій.



- Многокутник. Використовується для побудови довільного многокутника.

Технологія використання:

- оберіть інструмент «Многокутник»;
- клацніть по черзі на кожній вершині многокутника, а потім — ще раз на першій.



- Правильний многокутник. Дозволяє побудувати правильний многокутник.

Технологія використання:


- оберіть даний інструмент ;
- клацніть на двох точках, що будуть сусідніми вершинами майбутнього правильного многокутника;
- у діалоговому вікні, що відкрилося введіть кількість вершин многокутника;
- клацніть на кнопці **ОК**.



- Жорсткий многокутний. Використовується для побудови многокутника, розміри сторін та кутів якого змінити неможливо.


Технологія використання:

- клацніть спочатку на точці, що є крайньою вершиною ламаної, а потім — на кожній наступній вершині і знову — на першій точці.

 - Вектор. Використовується для побудови вектора за його початком і кінцем.


Технологія використання:

- оберіть даний інструмент;
- клацніть на початку вектора, а потім на його кінці.

 - Відкласти вектор. Застосовується для відкладання вектора, який дорівнює даному.


Технологія використання:

- оберіть інструмент «Відкласти вектор»;
- оберіть точку (клацніть на ній);
- клацніть на векторі.

 - Перпендикулярна пряма. Дозволяє побудувати пряму, яка є перпендикулярною даній і проходить через обрану точку.


Технологія використання:

- оберіть даний інструмент;
- оберіть пряму (вектор, відрізок, промінь);
- клацніть на точці (у випадку відсутності точки під курсором буде створена нова).

 - Паралельна пряма. Використовується для побудови прямої, яка є паралельною до даної прямої і проходить через обрану точку.


Технологія використання:

- оберіть вказаний інструмент;
- клацніть на точці (у випадку відсутності точки під курсором буде створена нова), а потім на — прямій (відрізьку, промені, векторі) або навпаки – спочатку на прямій(відрізьку, ...), а потім — на точці.

 - Серединний перпендикуляр. Дозволяє побудувати серединний перпендикуляр.

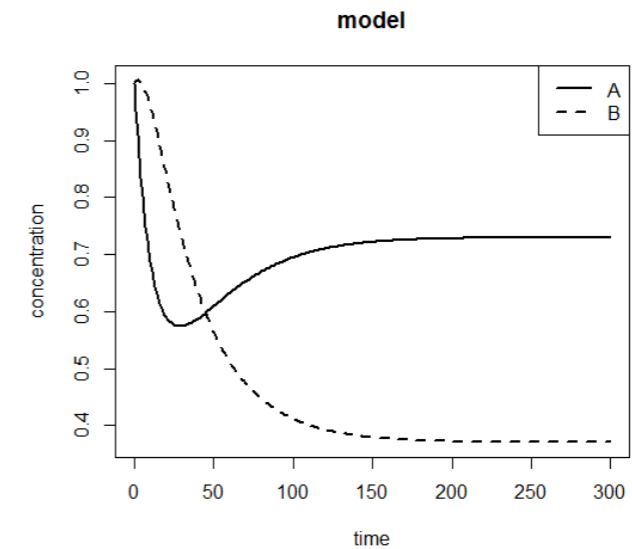
Технологія використання:

- оберіть даний інструмент;
- клацніть на двох точках (у випадку їх відсутності під курсором вони будуть створені) або на відрізьку.

 - Бісектриса кута. Використовується для побудови бісектрис кутів.

Технологія використання:

- оберіть інструмент «Бісектриса кута»;
- 1-й спосіб
- клацніть на точці, що належить до одної сторони кута, потім — на вершині кута і, нарешті, на точці, що належить до іншої сторони кута;
- буде побудовано пряму, що містить бісектрису даного кута;



2.1.24. Статистика

Ось деякі пакети, які часто використовуються:

- fastR (пакет з деякими хорошими утилітами; є на CRAN)
- lattice (пакет для графіки)
- mosaic (для вивчення статистики; є на CRAN)
- Hmisc (пакет з деякими хорошими утилітами; є на CRAN)

Обробка числових даних

R включає в себе функції, які обчислюють широкий спектр чисельних і графічних даних для статистичного аналізу.

Для прикладу будемо використовувати набір даних mtcars. Дані витягуються з журналу Motor Trend в США 1974 року, і включають в себе споживання палива і 10 аспектів автомобільного дизайну і продуктивності для 32 автомобілів (моделей 1973-1974 років).

Цей набір даних містить наступні дані:

Індекс	Назва змінної	Опис
[,1]	mpg	Міль/галон(США)
[,2]	cyl	Кількість циліндрів
[,3]	disp	Об'єм (в кубічних дюймах)
[,4]	hp	Механічна кінська сила
[,5]	drat	Передатне відношення головної передачі
[,6]	wt	Вага (фунт/1000)
[,7]	qsec	Час проходження ¼ милі
[,8]	vs	V/S
[,9]	am	Передача (0 = автоматична, 1 = механічна)
[,10]	gear	Кількість передач
[,11]	carb	Кількість карбюраторів

Декілька прикладів обчислення різних статистичних значень:
 Середнє значення набору даних mtcars по показнику mpg:

```
> mean(mtcars$mpg)
[1] 20.09062
```

Медіана набору даних mtcars по показнику mpg:

```
> median(mtcars$mpg)
[1] 19.2
```

Квантиль набору даних mtcars по показнику mpg:

```
> quantile(mtcars$mpg)
 0%  25%  50%  75% 100%
10.400 15.425 19.200 22.800 33.900
```

Інтерквантиль набору даних mtcars по показнику mpg:

```
> IQR(mtcars$mpg)
[1] 7.375
```

Функція favstats() з пакету mosaic обчислює декілька статистичних значень одразу.

```
> favstats(mtcars$mpg)
 min  Q1 median  Q3  max  mean  sd n missing
10.4 15.425 19.2 22.8 33.9 20.09062 6.026948 32 0
```

В пакеті Hmisc є функція summary() яка виводить данні в зручному вигляді.


```
> summary(mpg ~ am, data = mtcars, fun=favstats)
mpg  N=32

+-----+-----+-----+-----+-----+-----+-----+-----+-----+
|   | |N|min|Q1 |median|Q3 |max|mean |sd  |n|missing|
+-----+-----+-----+-----+-----+-----+-----+-----+
|am  |No |19|10.4|14.950|17.3 |19.2|24.4|17.14737|3.833966|19|0 |
|   |Yes|13|15.0|21.000|22.8 |30.4|33.9|24.39231|6.166504|13|0 |
+-----+-----+-----+-----+-----+-----+-----+-----+
|Overall| |32|10.4|15.425|19.2 |22.8|33.9|20.09062|6.026948|32|0 |
+-----+-----+-----+-----+-----+-----+-----+-----+

```

Граткова графіка

Є кілька способів, щоб побудувати графіки в R. Один з підходів є використання системи граткових графіків. Першим кроком для

 - Середня точка або центр. Використовується для побудови середини відрізка, центрів кола, кінчних перерізів.


Технологія використання:

- оберіть даний інструмент ;
- для отримання точки, що є серединою відрізка, який їх сполучає, клацніть спочатку на одній точці, а потім на другій;
- щоб побудувати середину відрізка, клацніть на відрізку;
- для побудови центру кола або кінчного перерізу клацніть на ньому.

 - Комплексне число. Використовується для створення комплексних чисел.


Технологія використання:

- оберіть відповідний інструмент;
- клацніть у потрібному місці на полотні.

 - Пряма. Використовується для побудови прямої за двома точками.

Технологія використання:

- оберіть вказаний інструмент;
- клацніть спочатку на одній точці, а потім на другій.

 - Відрізок. Застосовується для побудови відрізка за його кінцями.


Технологія використання:

- оберіть даний інструмент;
- клацніть по черзі на кожному з кінців відрізка, який ви хочете побудувати.

 - Відрізок заданої довжини. Використовується для побудови відрізка заданої довжини.


Технологія використання:

- оберіть даний інструмент;
- клацніть на точці, що буде одним з кінців відрізка;
- в діалоговому вікні, що з'явиться, введіть довжину відрізка;
- клацніть на кнопці **OK**.

 - Промінь. Дозволяє побудувати промінь за його початком і точкою на ньому.

Технологія використання:

- оберіть інструмент «Промінь»;
- клацніть спочатку на точці, що буде початком променя, а потім на іншій точці, яка йому буде належати.


 - Ламана. Застосовується для побудови ламаної.

Технологія використання:

- оберіть інструмент «Ламана»;


- виконайте операцію протягування по діагоналі уявного прямокутника (натисніть ліву кнопку мишки і, не відпускаючи її, переміщуйте мишку, після чого відпустіть кнопку).


4. Для переміщення об'єкта наведіть на нього курсор, натисніть ліву кнопку мишки, і, не відпускаючи її, перемістіть об'єкт у потрібне місце й відпустіть ліву кнопку. Переміщати об'єкти на полотні можна і при натиснутій правій кнопці миші.

 - Поворот навколо точки. Дозволяє обертати точку навколо обраної точки, яка є центром обертання.

Технологія використання:


- оберіть даний інструмент;
- оберіть центр обертання;
- перетягніть за допомогою мишки потрібну точку.

 - Записати у таблицю. Цей інструмент використовується для чисел, точок і векторів. За його допомогою можна перенести до таблиці число та координати точки або вектора.

 - Точка. Використовується для побудови нових точок.


Технологія використання:

- оберіть інструмент вказаний інструмент;
- клацніть на полотні (прямії, промені, відрізок, колі, графіку функції тощо) у потрібному місці.

 - Точка на об'єкті. Застосовується для побудови нових точок, що належать певному об'єкту.


Технологія використання:

- оберіть вказаний інструмент ;
- клацніть у необхідному місці на потрібному об'єкті.

 - Приєднати / від'єднати точку. Використовується для приєднання або від'єднання точок від побудованих об'єктів.

Технологія використання:

- оберіть вказаний інструмент ;
- клацніть на точці, яку необхідно приєднати, а потім — на об'єкті, до якого потрібно приєднати;
- для від'єднання приєднаної до об'єкта точки, клацніть на точці, яку необхідно приєднати.

 - Перетин. Застосовується для побудови точок перетину двох об'єктів.

Технологія використання:

- оберіть відповідний інструмент ;
- для отримання однієї точки перетину двох об'єктів клацніть на ній;
- обираючи двох об'єктів (клацніть на одному, а потім на другому) дозволяє отримати всі точки перетину (якщо це можливо).

використання ґраток («lattice»), потрібно завантажити пакет «lattice», використовуючи прапорець на вкладці Packages або за допомогою наступної команди:

```
> require(lattice)
```

ґраткові графіки використовують формульний інтерфейс:

```
> plotname( y ~ x | z, data=dataname, groups=grouping_variable, ...)
```

• Ось імена кількох ґраткових графіків:

- histogram (для гістограм)
- bwplot (для коробкових графіків або графіків «ящик з вусами»)
- xyplot (для діаграм розсіювання)
- qqmath (для графіків квантиль-квантиль (Q-Q))

• x це ім'я змінної, яка відкладається по горизонтальній (x) осі.

• y це ім'я змінної, яка відкладається по вертикальній (y) осі. (Для деяких графіків, це поле порожнє, бо R обчислює ці значення із значень x.)

• z змінна стану, використовується для поділу графіка на декілька областей, так званих панелей.

• grouping_variable використовується для відображення різних груп порізному (різні кольори або символи, наприклад) в рамках тієї ж панелі.

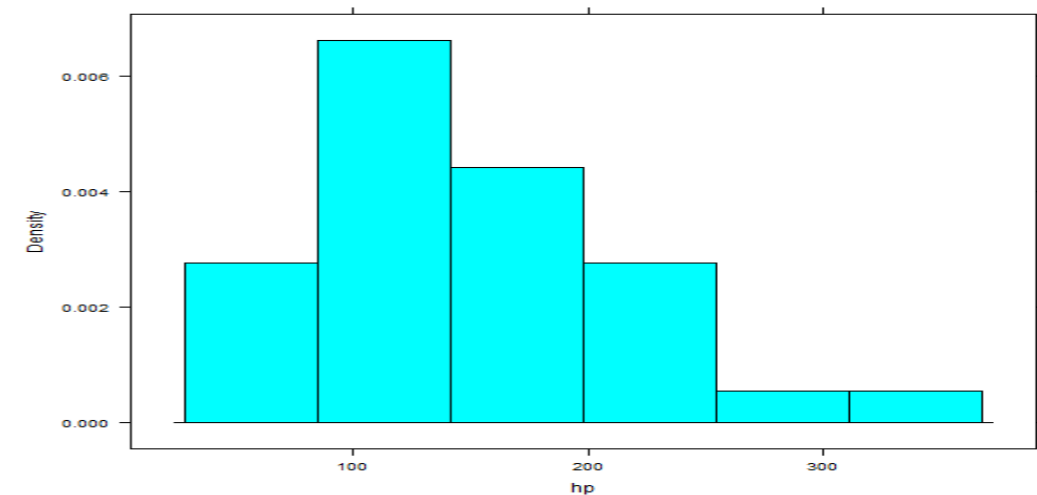
• ... Є багато додаткових аргументів цих функцій, які дозволяють контролювати, як виглядає графік.

Гістограма

Гістограма — спосіб графічного представлення табличних даних. Являє собою діаграму, що складається з прямокутників без розривів між ними.

Наприклад, щоб побудувати гістограму набору даних mtcars по показнику hp, треба:

```
> histogram(~ hp, data=mtcars)
```

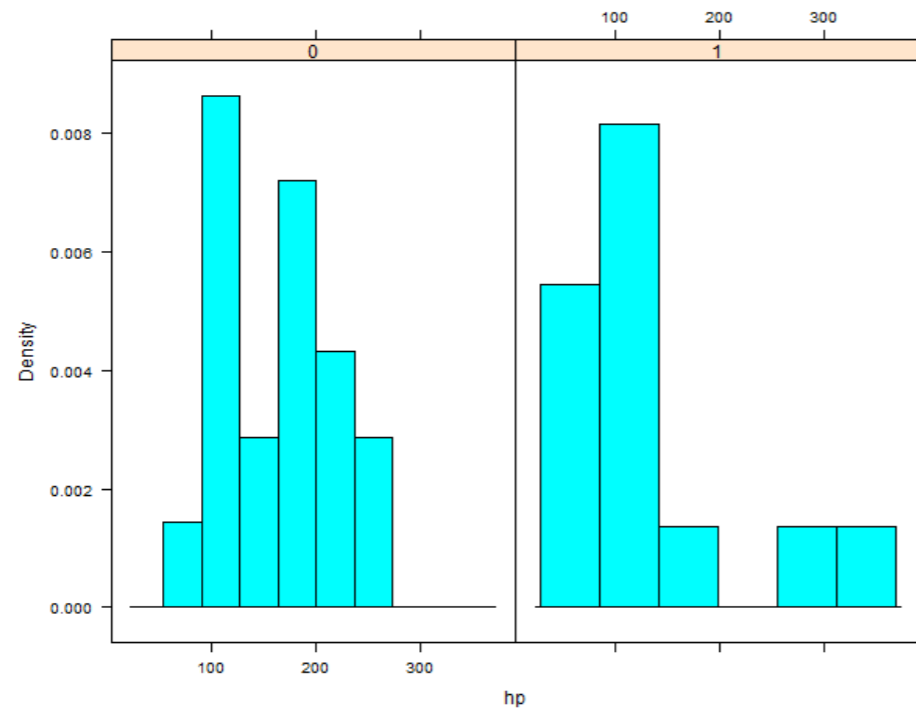


У компонент формули порожній, так як ми дозволяємо R обчислити висоту стовпців.

Ми можемо використовувати змінні стану, щоб побудувати окремі гістограми для кожного значення цієї змінної. Наприклад, щоб побудувати

гістограму набору даних mtcars по показнику hp в залежності від значення параметра am, треба:

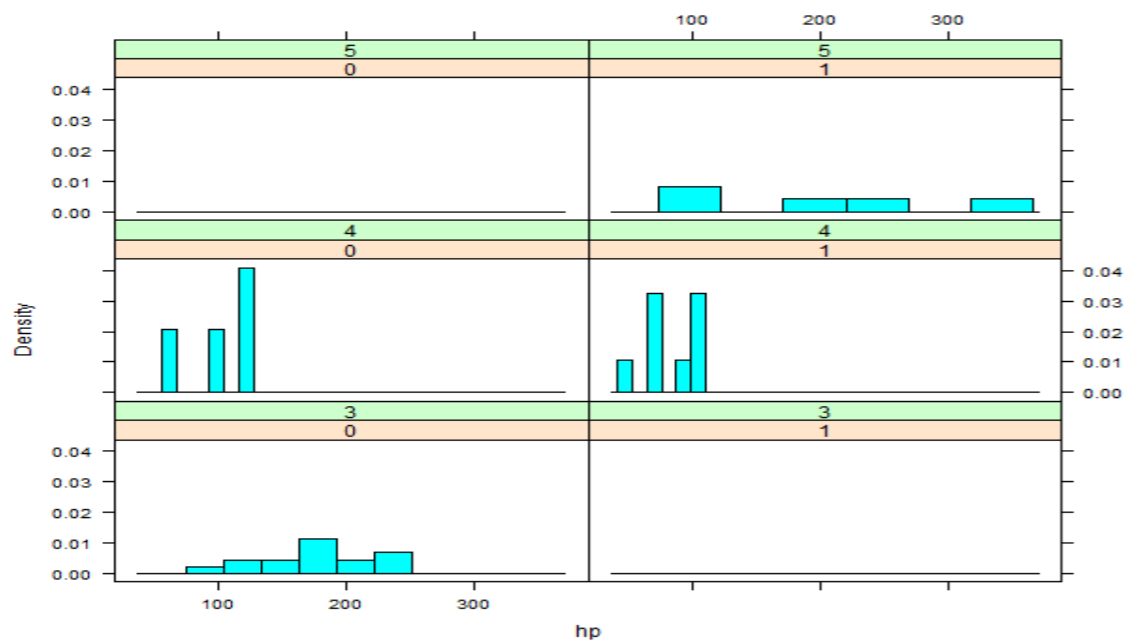
```
> histogram(~ hp | factor(am), data=mtcars)
```



На графіку зображено дві підгрупи, які називаються панелями, та надписи зверху, які називаються смужками. (Смужки можуть бути розміщені на лівій стороні, якщо ви віддаєте перевагу.)

Ми можемо використовувати одразу декілька змінних стану:

```
> histogram(~ hp | factor(am) + factor(gear), data=mtcars)
```



Ще одним із видів гістограм є стовпчикова діаграма. Щоб її побудувати, треба:

розташовані на «Панелі інструментів» (рис. 2.114). Щоб скористатися потрібним інструментом, необхідно його обрати, клацнувши на кнопці, що йому відповідає.



Рис. 2.114 - Панель інструментів

Кожна кнопка панелі представляє цілу групу інструментів, об'єднаних за принципом схожості виконуваних функцій. Щоб дізнатись, які інструменти до неї входять, потрібно клацнути значок у вигляді трикутника, що знаходиться в правому нижньому кутку кожної кнопки. Відобразиться додаткова панель зі списком інструментів, що входять до даної групи (рис. 2.115). Для обрання потрібного інструмента необхідно на ньому клацнути.

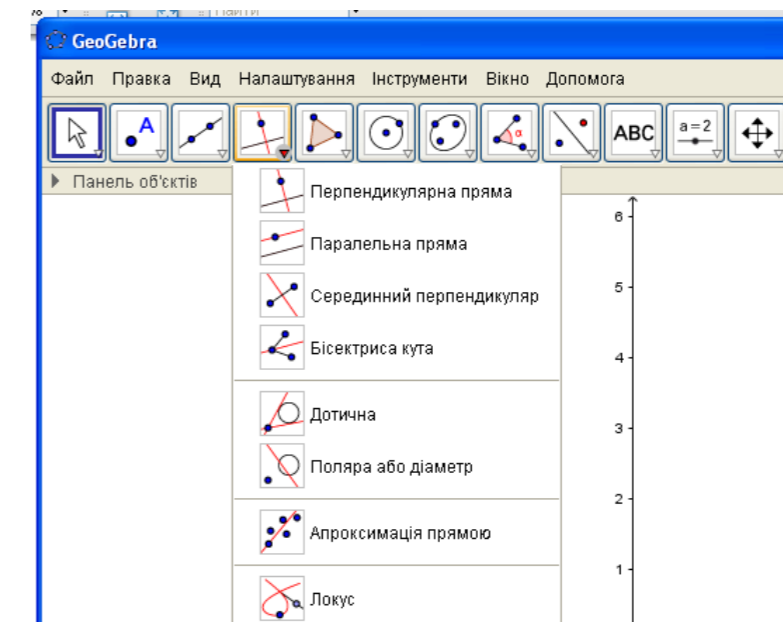



Рис. 2.115 - Додатковий спливаючий список Панелі інструментів

Далі розглянемо основні інструменти середовища GeoGebra та їх призначення.

 - Переміщення. Використовується для обирання (виділення), зняття виділення і переміщення об'єктів на полотні.

Технологія використання:

1. Щоб обрати об'єкт (виділити), клацніть на ньому лівою кнопкою мишки.
2. Для того, щоб зняти виділення, клацніть на полотні за межами виділеного об'єкта.
3. Щоб виділити групу об'єктів, виконайте дії в такій послідовності:
 - перемістіть курсор мишки до кута уявного прямокутника, що охопить усі об'єкти, які потрібно виділити;

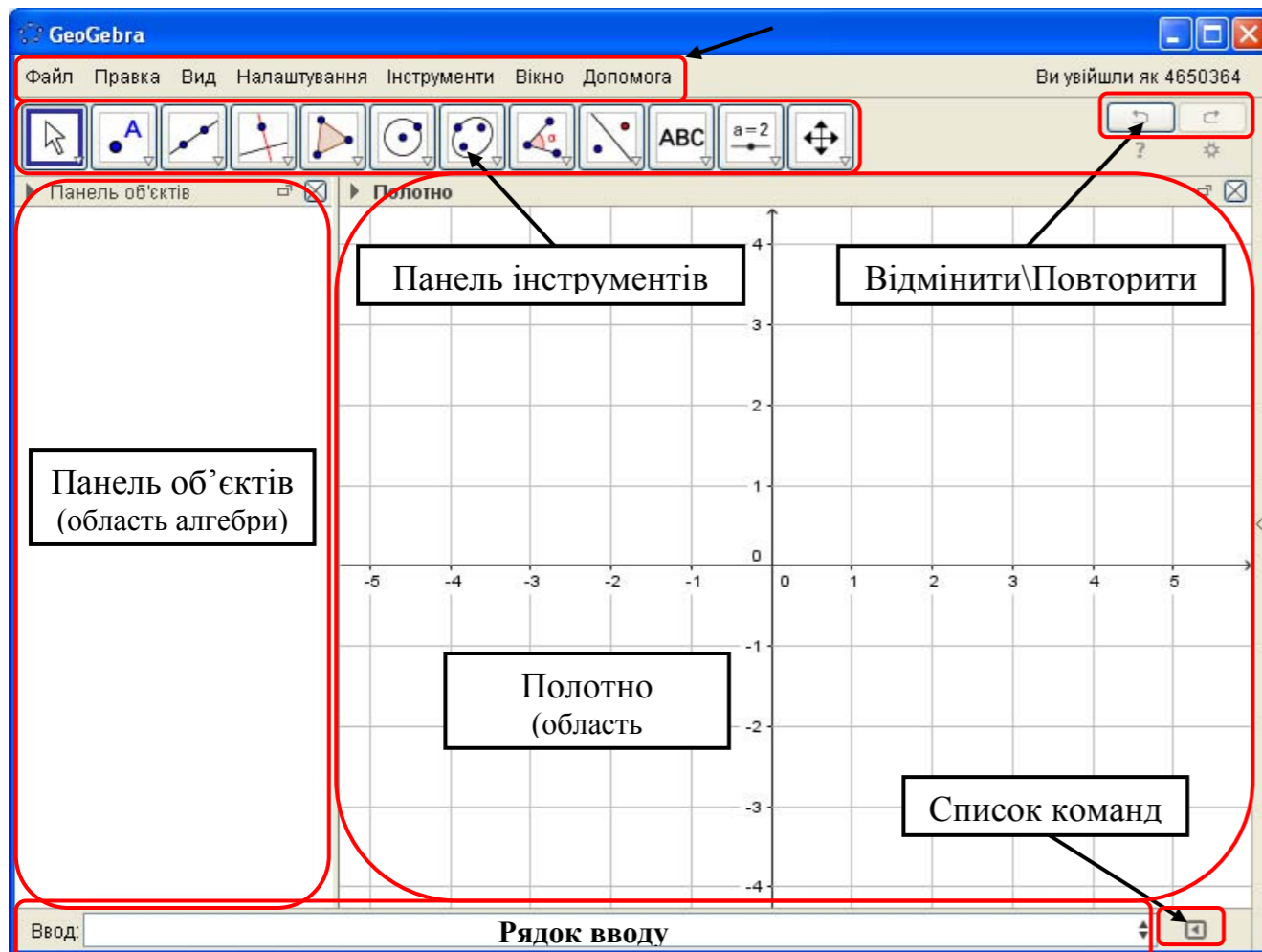


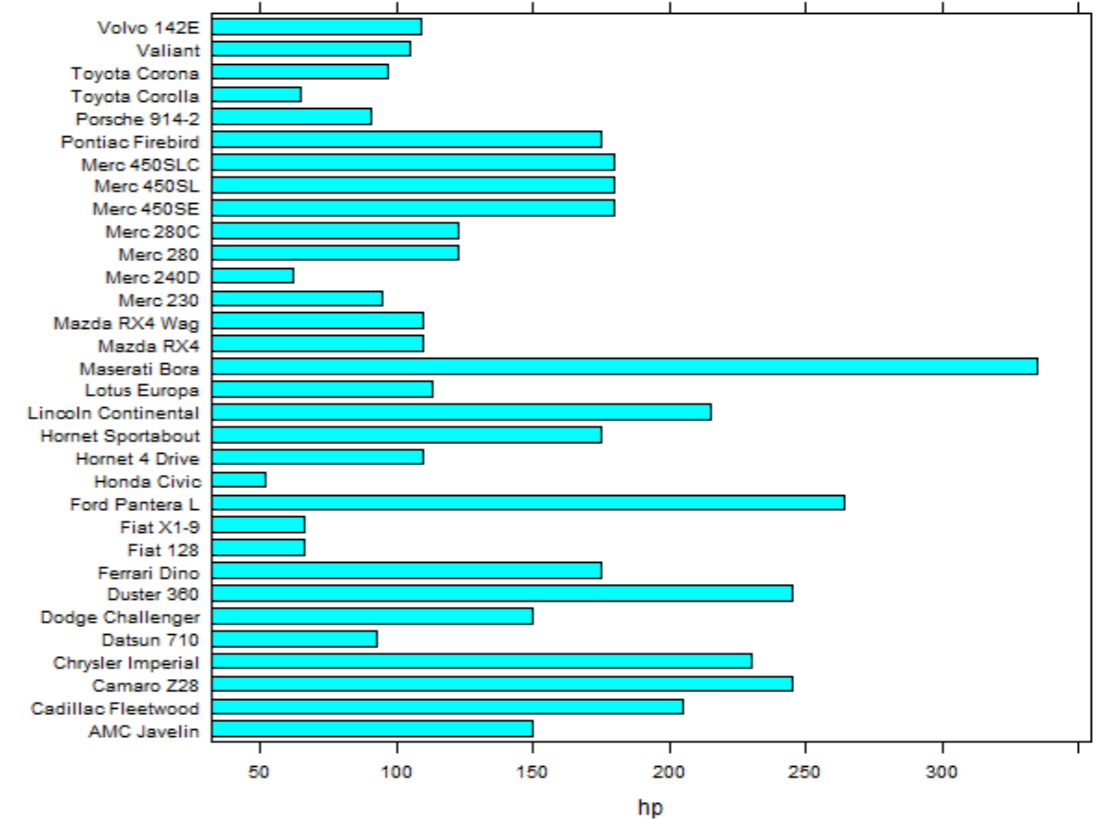
Рис. 2.113

1. Кнопки меню – містять основні команди для роботи з файлами, а також налаштування програми та її інтерфейсу.
2. Панель інструментів – містить кнопки що використовуються для виконання графічних побудов та інших дій на «Полотні».
3. Відмінити\повторити – кнопки призначені для повтору або відміни виконаної операції в програмі GeoGebra.
4. Панель об'єктів – призначена для математичного відображення побудованих на полотні графічних об'єктів (точок, ліній, геометричних фігур, графіків, тощо), а також для редагування їх значень параметрів.
5. Полотно – призначене для побудови, відображення та редагування графічних об'єктів.
6. Рядок вводу – призначений для введення математичних виразів, функцій та параметрів графічних об'єктів.
7. Список команд – спливаюче меню, в якому можна обрати необхідну команду для введення зі списку.

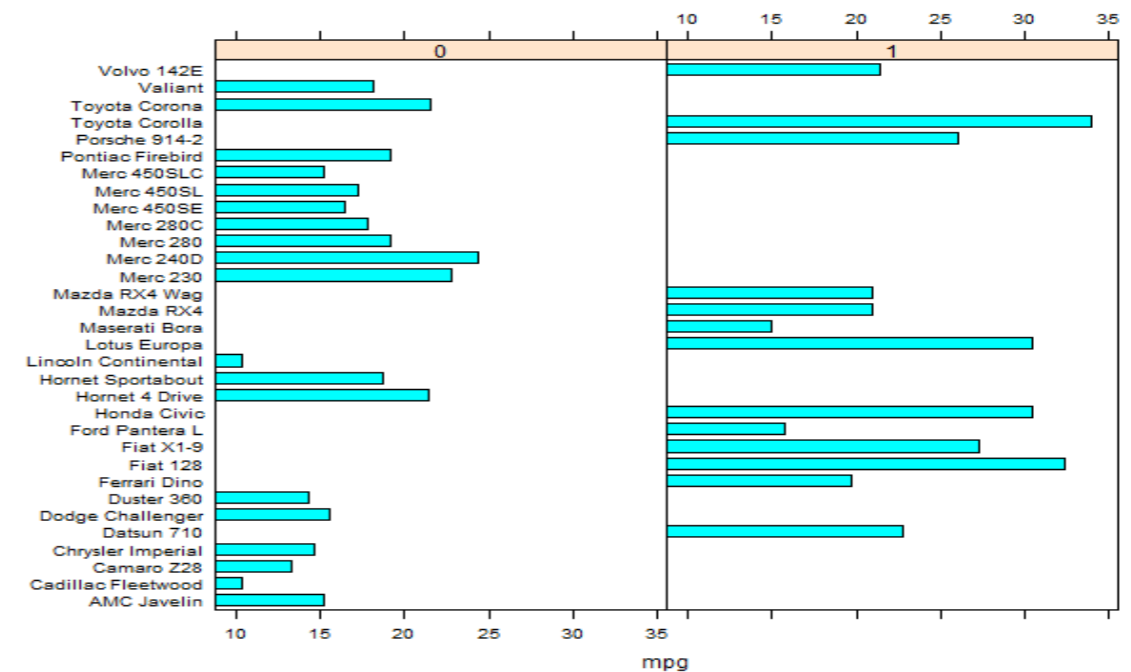
2.5.10. Панель інструментів

Для виконання графічних побудов та інших дій на «Полотні» та на «Полотні 2» використовуються відповідні команди (інструменти), які

```
> barchart(row.names(mtcars) ~ hp, data = mtcars)
```



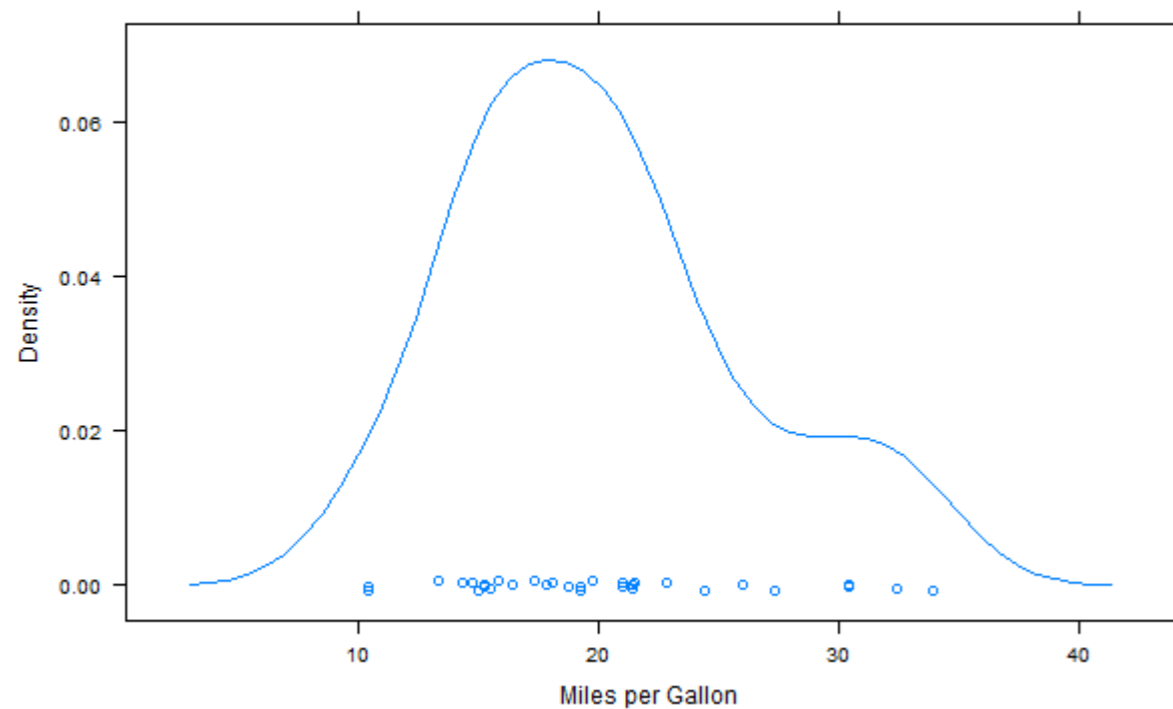
```
> barchart(row.names(mtcars) ~ mpg | factor(am), data = mtcars)
```



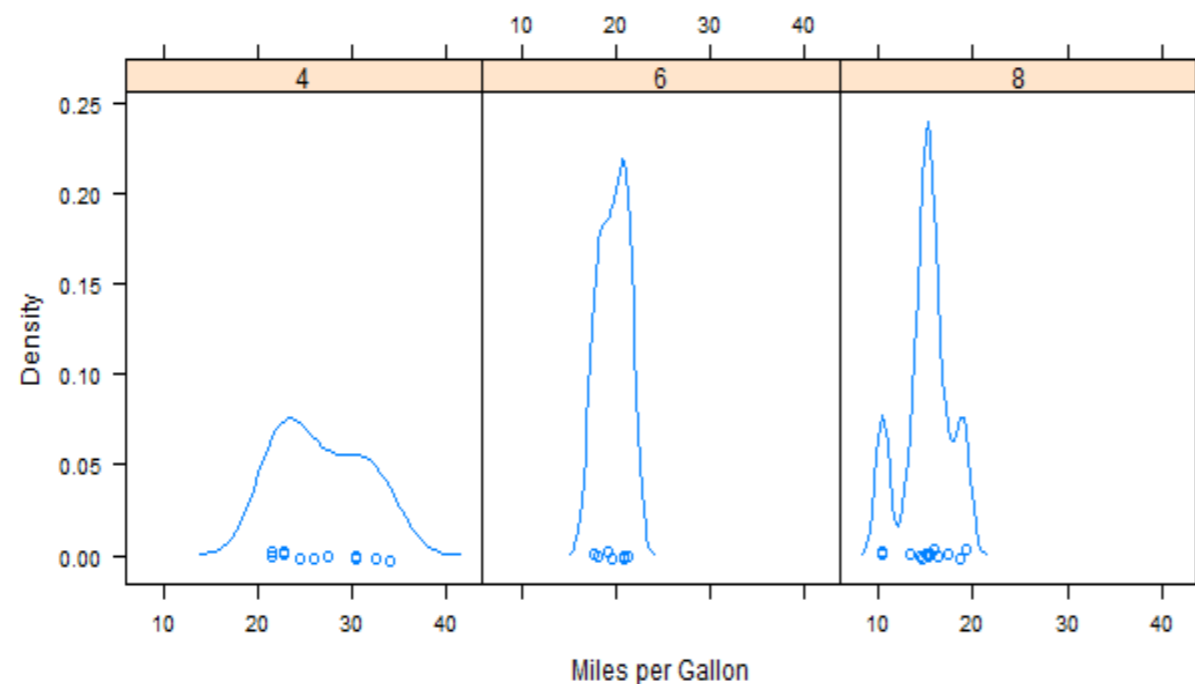
Графік щільності (густини) ймовірності

Функція `densityplot()` буде оцінку щільності ядра в одновимірному наборі даних; тобто оцінку функції густини ймовірностей.

> densityplot(~mpg, data=mtcars, xlab=«Miles per Gallon»)



> densityplot(~mpg|factor(cyl), data=mtcars, xlab=«Miles per Gallon»)



Для більшої зручності, можна побудувати гістограму та графік густини ймовірностей разом:

```
> histogram(~ mpg | factor(cyl), data=mtcars, xlab=«Miles per
Gallon»,type='density',panel=function(...)
{panel.histogram(...);
panel.densityplot(...)}))
```

Скачати файл для інсталяції програми GeoGebra можна з сайту <http://www.geogebra.org/download> (розділ «Завантажити зараз»), обравши потрібний, у залежності від операційної системи користувача. Далі запускаємо інсталяційний файл та здійснюємо процес інсталяції.

GeoGebra також надає можливість використання Portable-версії програми, яка для використання не потребує інсталяції. Щоб її завантажити, потрібно перейти за гіперпосиланням "Більше GeoGebra завантажень" (http://wiki.geogebra.org/en/Reference:GeoGebra_Installation)

2.5.8. Запуск GeoGebra

ОС Windows надає декілька способів для запуску програми GeoGebra. Розглянемо ті, що найчастіше використовуються.

– За допомогою Головного меню

– Для запуску з "Робочого столу" або з панелі "Швидкий запуск" (знаходиться на Панелі задач) потрібно скористатись значком (за умови їх присутності).

– GeoGebra також можна відкрити, виконавши подвійне натискання лівої клавіші мишки на піктограмі документа (рис. 2.111), створеного за її допомогою. Спочатку завантажиться програма, а потім у вікні з'явиться вказаний документ.

Щоб відкрити онлайн версію GeoGebra, потрібно зайти на сайт системи динамічної математики GeoGebra (<http://www.geogebra.org>), обрати Старт GeoGebra, далі обрати необхідний онлайн модуль (рис. 2.112)



Рис. 2.111

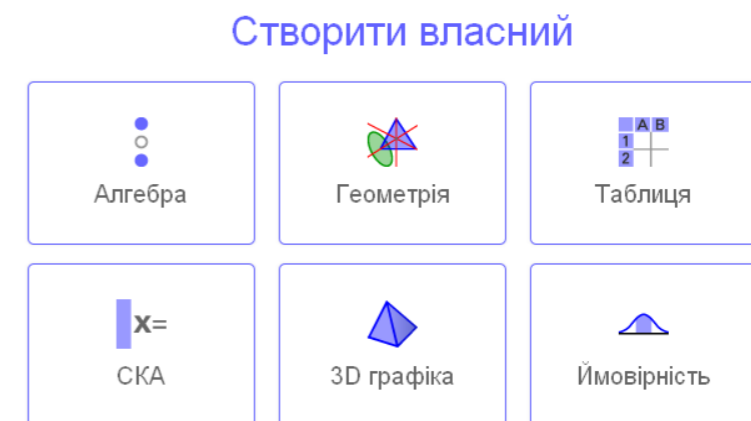


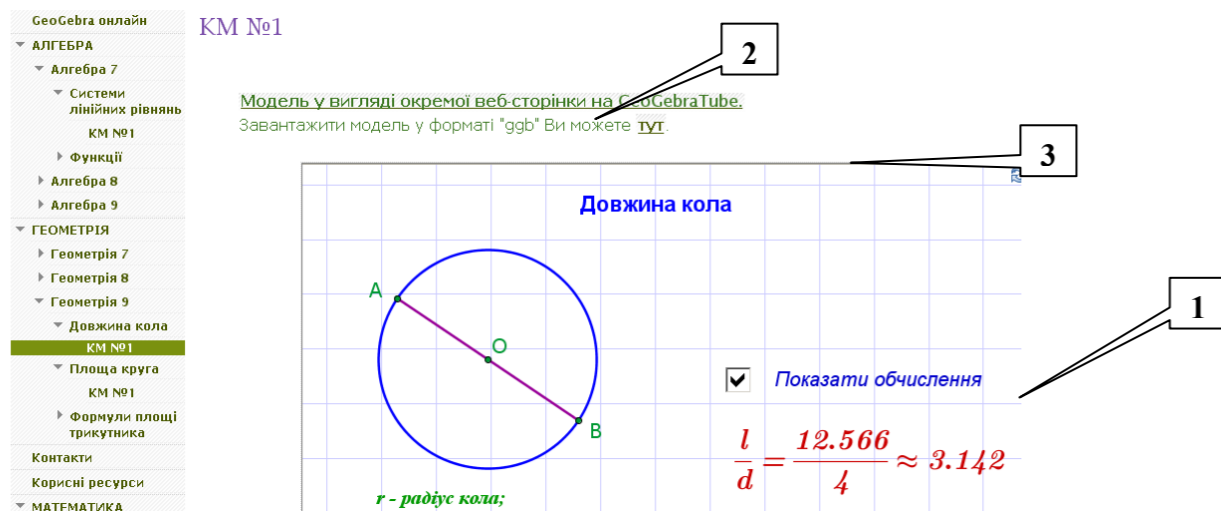
Рис. 2.112

2.5.9. Вікно програми GeoGebra

Після запуску GeoGebra відкривається головне вікно програми. Розглянемо його основні елементи (рис. 2.113)

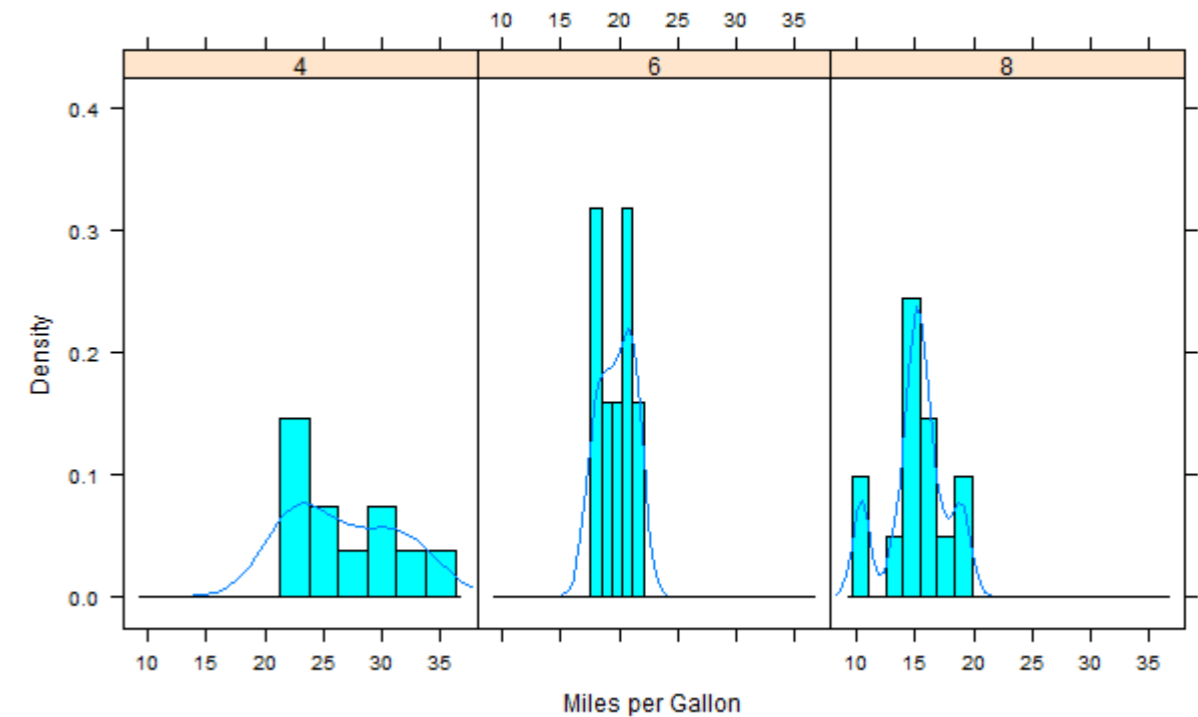


і відповідно трудомістким. Тому важливим є об'єднання зусиль зацікавлених користувачів з метою створення та обміну моделями. Для цього і був створений відповідний веб-сервіс GeoGebraTube (<http://tube.geogebra.org>). З метою забезпечення оптимальних умов для використання комп'ютерних моделей у центрі "Інститут GeoGebra Чернігів, Україна" (<https://sites.google.com/site/geogebrachernigiv>) було створено ін-тернет-ресурс "Бібліотека комп'ютерних моделей" (<https://sites.google.com/site/biblkompmo>). Продовжується робота з його наповнення та вдосконалення. Комп'ютерні моделі, що на ньому представлені, систематизовані відповідно до розділів діючої програми з математики. Для кожної моделі відведено окрему веб-сторінку, на якій розташовано модель у вигляді гаджету Google (1, рис. 2.110), посилання на модель у вигляді окремої вебсторінки (2, рис. 2.110), що зберігається на GeoGebraTube, відомості про розробників та авторів перекладу і адаптації, а також можуть міститись методичні рекомендації щодо її викорис-тання, посилання на добірки завдань до моделі тощо. Щоб завантажити модель у форматі "ggb" («рідному» форматі GeoGebra), яка також міс-титься на сервері GeoGebra, потрібно скористатись відповідним посилан-ням (3, рис. 2.110). Сайти бібліотек створені у системі Google Sites. Взяти участь у поповненні бібліотеки можуть усі зацікавлені користувачі. Вчитель (викладач, учень, студент) має можливість скористатися бібліотекою у режимі онлайн або використати модель у вигляді інтерактивної веб-сторінки чи у форматі "ggb", попередньо завантаживши їх. Кожний користувач може модернізувати існуючу модель з метою оптимального її пристосування до потреб конкретної навчальної ситуації.



2.5.7. Інсталяція GeoGebra

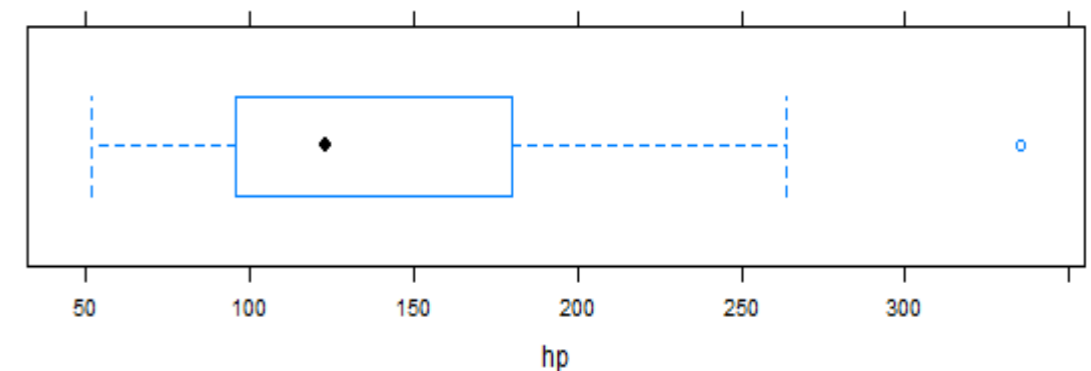
Перед інсталяцією GeoGebra потрібно встановити (у випадку його відсутності на комп'ютері) програмне забезпечення Java, використавши веб-інсталяцію, або інсталювати, попередньо завантаживши його з сайту <http://www.java.com/ru>.



Графік «ящик з вусами»

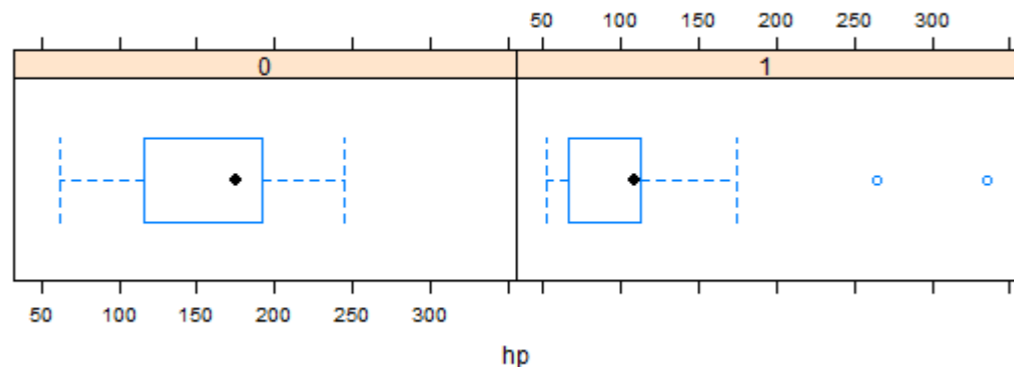
Функція `bwplot` генерує графік типу «ящик з вусами». Ця функція використовується таким самим чином, як і функція що будує гістограми.

```
> bwplot(~ hp, data=mtcars)
```



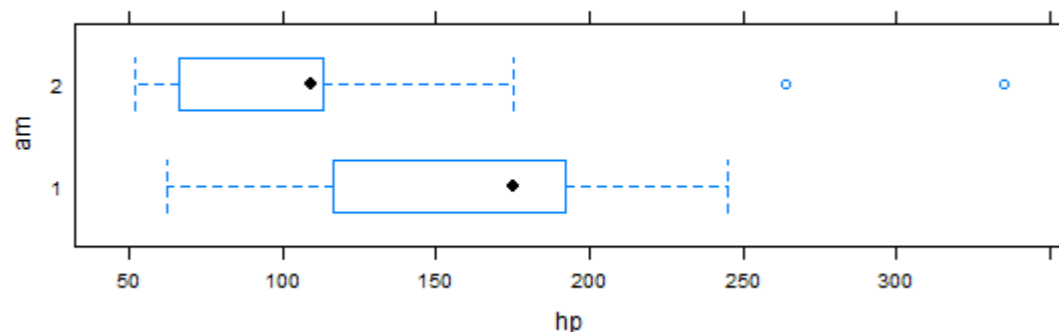
Ми можемо використовувати умовні змінні, так як ми робили і для гістограм:

```
> bwplot(~ hp | format(am), data=mtcars)
```

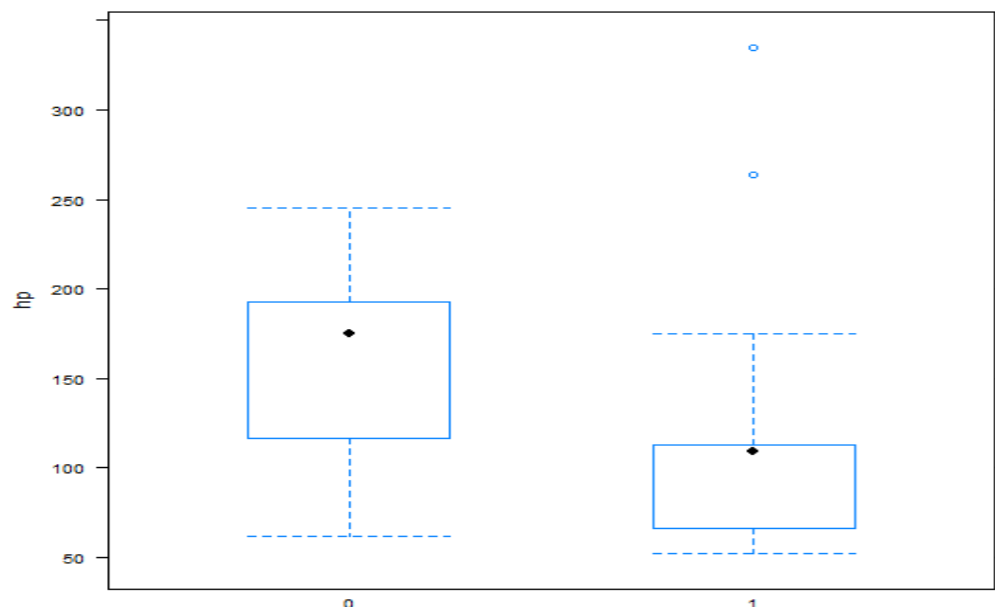


Але є кращі способи зробити це:

```
> bwplot(am ~ hp, data=mtcars)
```



```
> bwplot(hp ~ format(am), data=mtcars)
```



Ми можемо поєднати цей спосіб з умовними змінними, якщо потрібно:

```
> bwplot(hp ~ format(cyl) | format(am), data=mtcars)
```

можна ускладнити, доручивши учням самостійно створити дану модель (чи моделі), а потім провести дослідження. План дослідження учням надається. Сильним учням можна запропонувати скласти такий план самостійно.

За допомогою ІКМ, поданої на рис. 2.109, (<https://sites.google.com/site/biblkompmod/matematika/geometria/geometria-7/trikutniki/vlast-kutiv-trik/km-no3>) учні після проведення відповідних досліджень (самостійних або під керівництвом учителя) не тільки зможуть «відкрити» теорему про суму кутів трикутника, але й знайти ідею доведення цієї теореми та самостійно її довести.

У математиці дуже часто при розв'язуванні складної задачі на обчислення (в кінцевому результаті повинні отримати число) потрібно розв'язати кілька простих, алгоритм розв'язання яких учням уже добре відомий. Наведемо декілька прикладів.

1. Обчислити площу трикутника, якщо відомі його сторони.
2. Знайти гіпотенузу прямокутного трикутника за його катетами.
3. Знайти суму членів арифметичної (чи геометричної) прогресії.
4. Знайти сторону трикутника, якщо відомі дві інші його сторони та кут між ними.

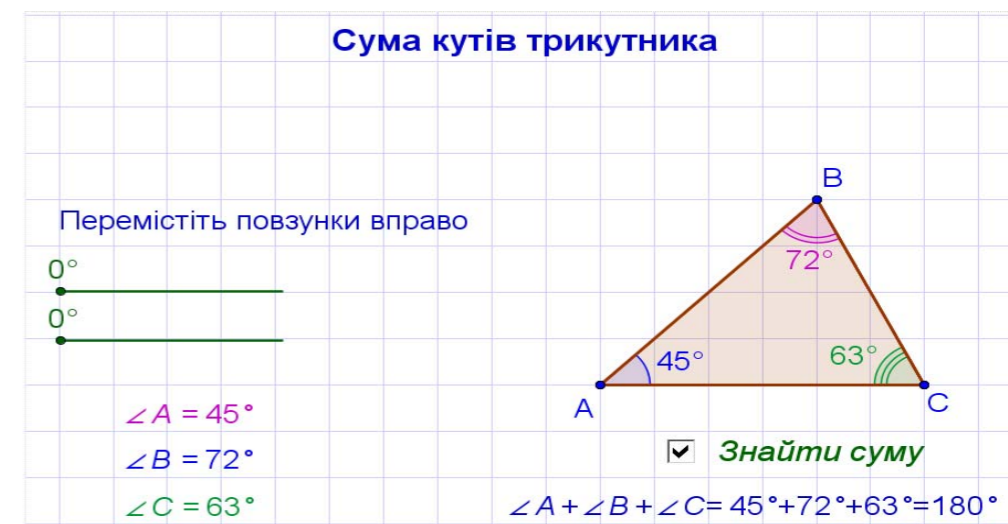


Рис. 2.109

Автоматизувати процес розв'язування подібних задач можна за допомогою раніше створених моделей, створення яких можна доручити учням під час вивчення відповідних тем. Створення подібних моделей сприятиме засвоєнню відповідних математичних формул та алгоритмів, формуватиме навички раціонального використання прикладних програм для розв'язування математичних задач, розвиватиме творчі здібності учнів.

2.5.6. Веб-сервіси "GeoGebraTube" та "Бібліотека комп'ютерних моделей"

Процес створення інтерактивних комп'ютерних моделей є творчим і в залежності від конкретної моделі та її призначення може бути доволі складним

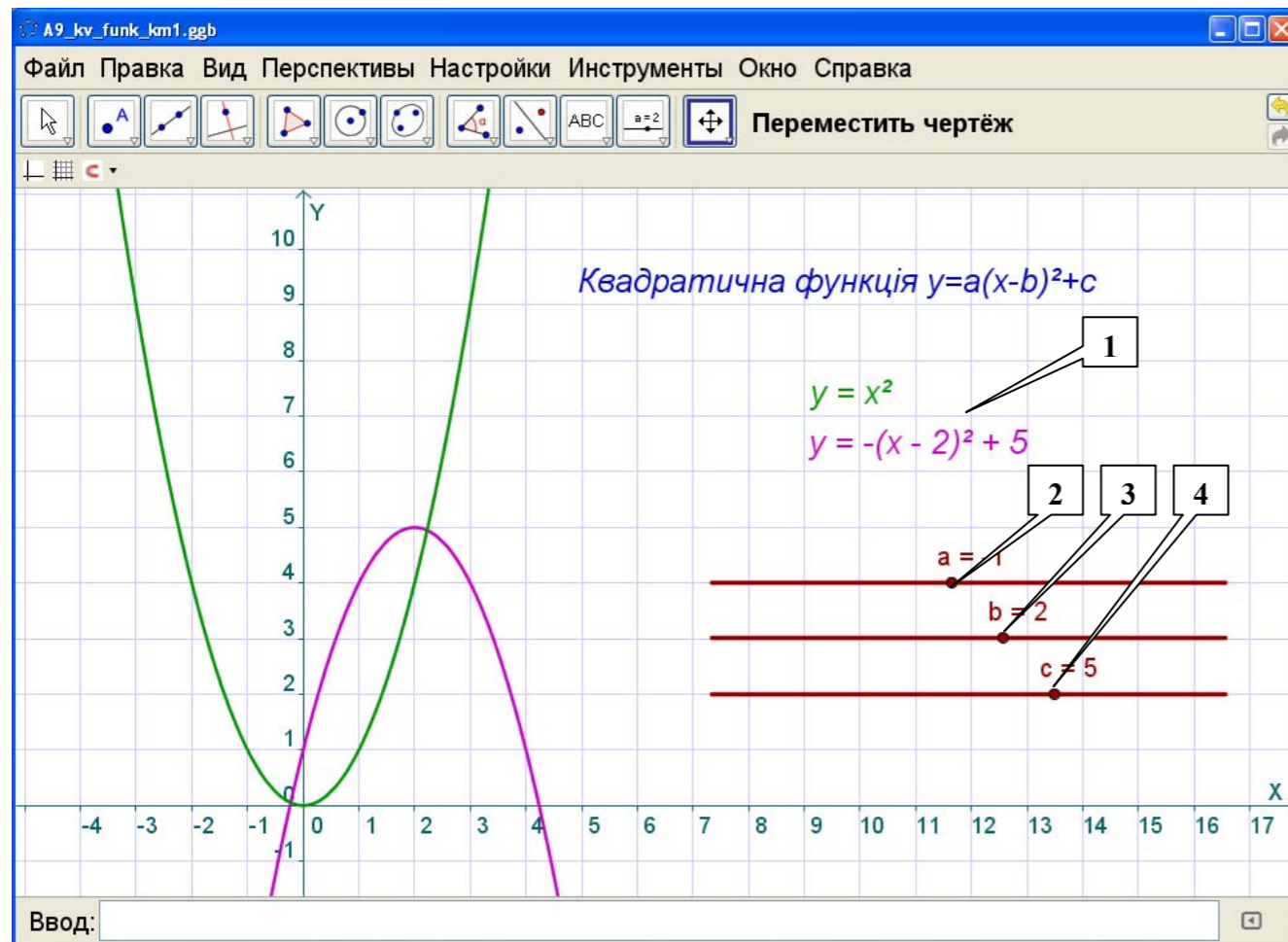


Рис. 2.108 - Комп'ютерна модель

Її можна використовувати як електронний інтерактивний наочний посібник під час вивчення теми «Квадратична функція» у 9-му класі у якості динамічної таблиці. При зміні значень параметрів a , b , і c (за допомогою повзунків 1, 2, 3 (рис. 2.108)) зазнають відповідних змін графік функції та формула (завдяки використанню динамічного тексту (4 на рис. 2.108)), що йому відповідає. Використання моделі буде ефективнішим, якщо вчитель запропонує учням на уроці провести за допомогою даної моделі (або кількох аналогічних моделей, наприклад, таких, як) невеличке дослідження з метою «відкриття» правил отримання графіків функцій $y=af(x)$, $y=f(x)+c$ та $y=f(x+b)$ з графіка функції $y=f(x)$.

Завдання може бути таким.

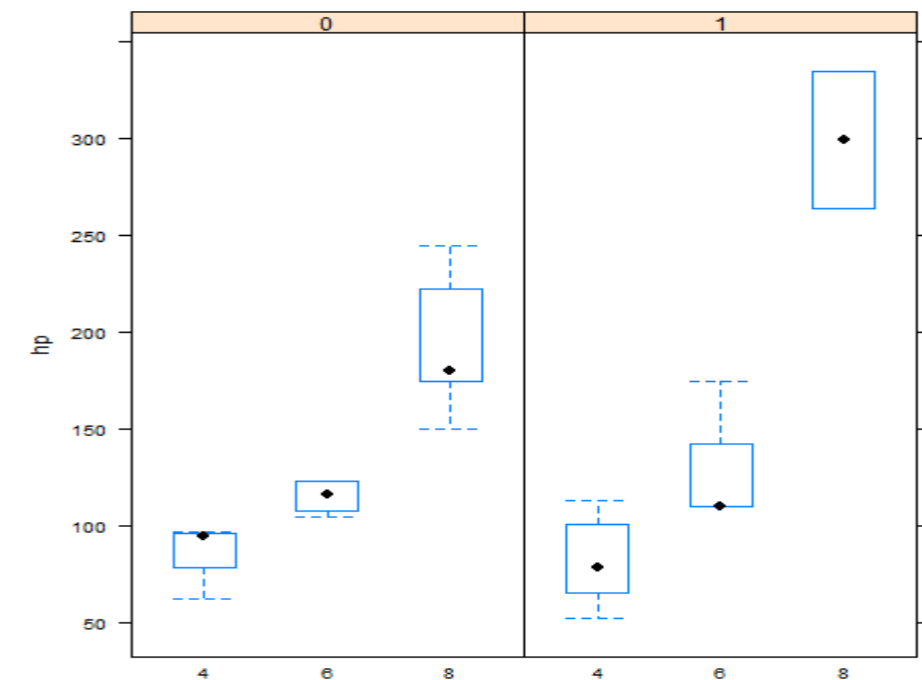
Виконайте дослідження відповідно до плану.

1. Змініть за допомогою повзунка значення параметра a .
2. Надайте йому спочатку додатних, а потім від'ємних значень.
3. Що відбувається з графіком функції? Зробіть висновки.
4. Аналогічні дії виконайте з параметрами c та b .

5. Сформулюйте правила отримання графіків функцій $y=af(x)$, $y=f(x)+c$ та $y=f(x+b)$ з графіка функції $y=f(x)$.

6. Поясніть, чому ви прийшли до таких висновків.

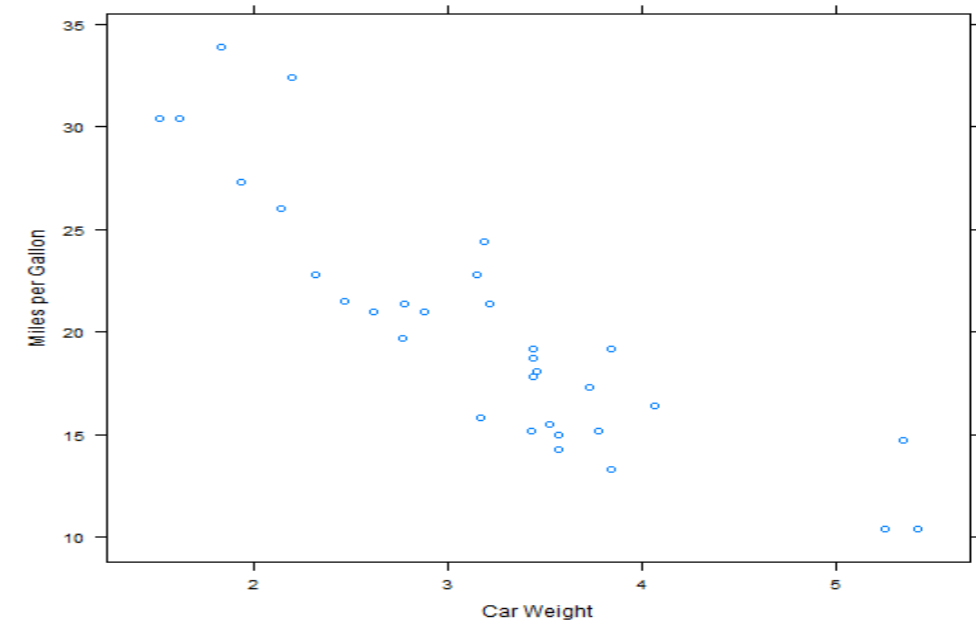
Це ж завдання можна дати як домашнє напередодні вивчення теми. Його



Діаграма розсіювання

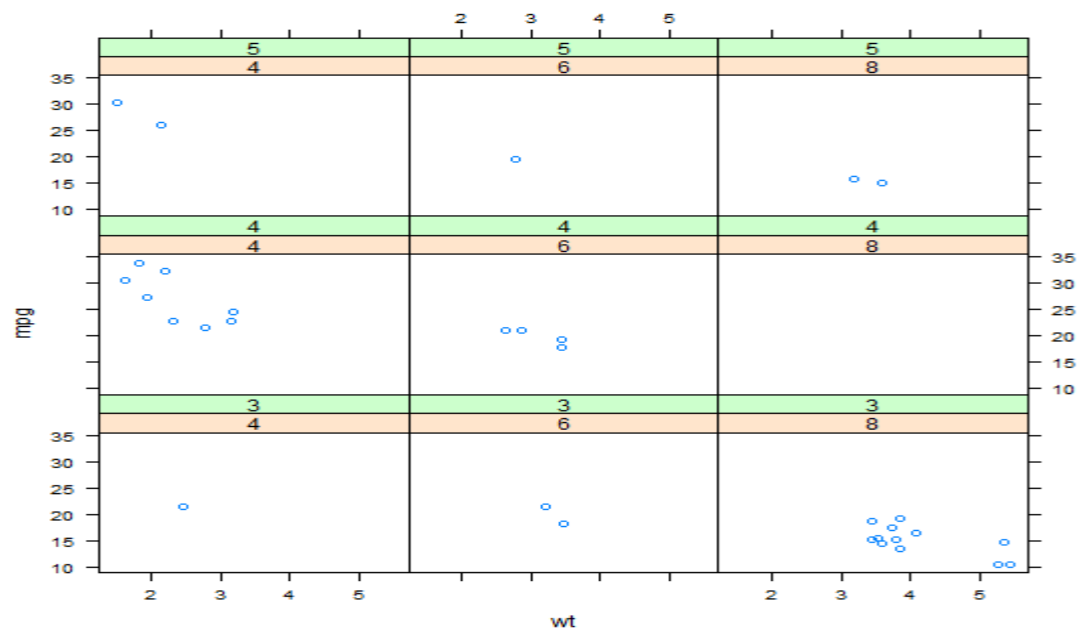
Діаграму розсіювання можна побудувати за допомогою функції `xyplot()`.

```
> xyplot(mpg ~ wt, data=mtcars, ylab=«Miles per Gallon», xlab=«Car Weight»)
```



Знову ж таки, ми можемо використовувати умовні змінні, щоб зробити панель для кожного виду.

```
> xyplot(mpg ~ wt | format(cyl) * format(gear), data=mtcars)
```



Часовий ряд

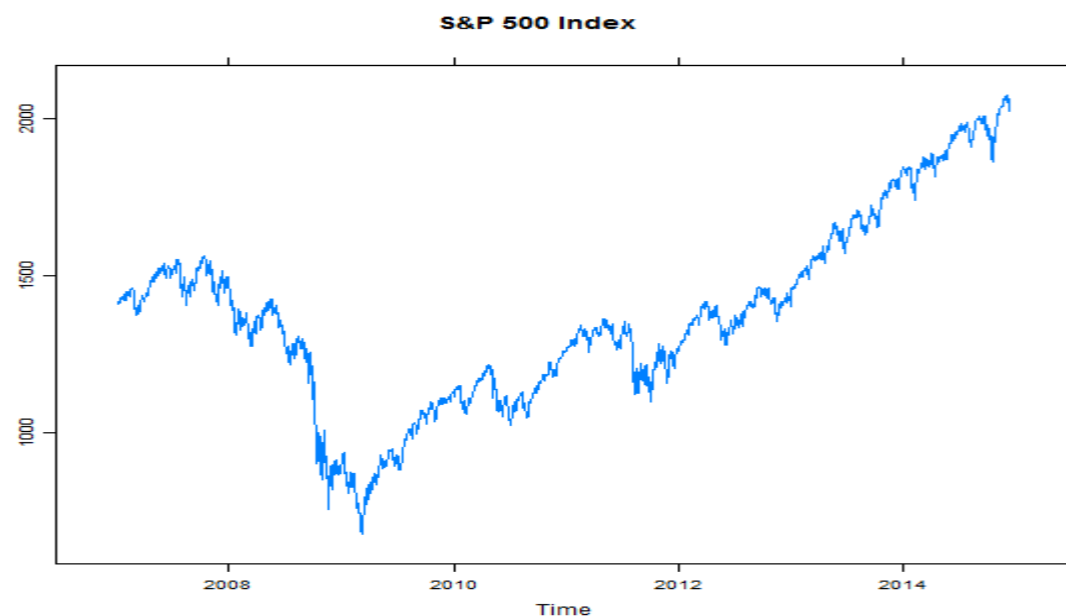
Часовий ряд – це послідовність значень досліджуваної ознаки (статистичного показника), впорядкована у хронологічному порядку.

Функція `xyplot()` має методи для побудови графіків часових рядів (`ts`, `zoo`, `xts`).

Для прикладу використаємо часовий ряд індексу S&P 500.

S&P 500 — фондовий індекс, у кошик якого включено 500 акціонерних компаній США, що мають найбільшу капіталізацію.

```
> install.packages(«quantmod»)
> library(quantmod)
> getSymbols(«^GSPC»)
> xyplot(Ad(GSPC),main=«S&P 500 Index»)
```



Перевага GeoGebra у порівнянні з такими потужними математичними пакетами, як Mathcad, Matlab, Maple, Mathematica та ін. полягає в тому, що GeoGebra поєднує в собі функціональні можливості та інструменти, переважна більшість з яких затребувані саме у процесі вивчення математики у ВНЗ (що не можна сказати, говорячи про вище названі СКМ, тільки незначна частина їхнього функціоналу може знайти своє використання у шкільній математиці), з одного боку, а з іншого, одним з головних принципів у концепції подальшого розвитку GeoGebra є збереження простого у використанні та інтуїтивно зрозумілого інтерфейсу та подальшого його вдосконалення у цьому напрямку.

GeoGebra має у своєму арсеналі великий набір інструментів для створення динамічних комп'ютерних моделей (ДКМ). На сучасному етапі розвитку шкільної математичної освіти використання комп'ютерних моделей (КМ) у навчальному процесі є однією з передумов підвищення його результативності.

Основні напрямки такого використання.

- GeoGebra разом з PowerPoint можна застосовувати для створення якісної наочності (малюнки до задач, теорем, вправи на готових кресленнях тощо).
 - GeoGebra має потужні засоби для розв'язування планіметричних задач.
 - ДКМ, створені за допомогою GG, можна ефективно використовувати для пошуку шляхів та ідей розв'язання планіметричних задач як на обчислення, так і на доведення.
 - Інтерактивні комп'ютерні моделі (ІКМ), розроблені у середовищі GeoGebra, можна застосовувати у якості динамічних наочних посібників як для вивчення нового матеріалу, так і з метою повторення та узагальнення.
 - ІКМ, створені за допомогою GG, можна використовувати для організації евристичного навчання, формування вмінь та навичок дослідницької діяльності, розвитку творчих здібностей дітей.
 - СДМ GeoGebra має у своєму арсеналі засоби для створення комп'ютерних моделей, призначених для автоматизації обчислень.
 - Функціональні можливості GeoGebra дозволяють використовувати її як інструмент для створення вправ на готових кресленнях у вигляді інтерактивних комп'ютерних моделей.
 - ІКМ, призначені для автоматизації процесу створення навчальних вправ і завдань, також можна створювати за допомогою GG.
 - За допомогою СДМ GeoGebra можна створити тренажери (наприклад, для засвоєння геометричних понять) у вигляді ІКМ.
 - У комплексі з іншими засобами (Microsoft Office Word, Microsoft Office PowerPoint, сайти Google, блоги та документи Google, Moodle тощо) GeoGebra є ефективним інструментом для створення інноваційних дидактичних матеріалів, інтерактивних електронних навчальних посібників, інтерактивних дистанційних курсів (присвячених вивченню планіметрії), авторських друкованих дидактичних матеріалів тощо.
- Як приклад, розглянемо комп'ютерну модель, наведену на рис. 2.108 (<https://sites.google.com/site/biblkompmo/mod/matematika/algebra/algebra-9/kvadraticna-funkcia-1/kvadr-funk/kvadraticna-funkcia>).

матеріалів, мультимедійних презентацій навчального призначення тощо. Актуальним є те, що якість зображень під час їх обробки (збільшення, зменшення) залишається високою і фактично не змінюється, а отже, позитивно впливає на дизайн відповідного кінцевого продукту.

СДМ GeoGebra має потужний набір інструментів, за допомогою яких можна розв'язувати різноманітні типи математичних задач. Перерахуємо основні з тих задач, які стосуються вивчення математики у загальноосвітніх навчальних закладах.

Алгебра і початки аналізу:

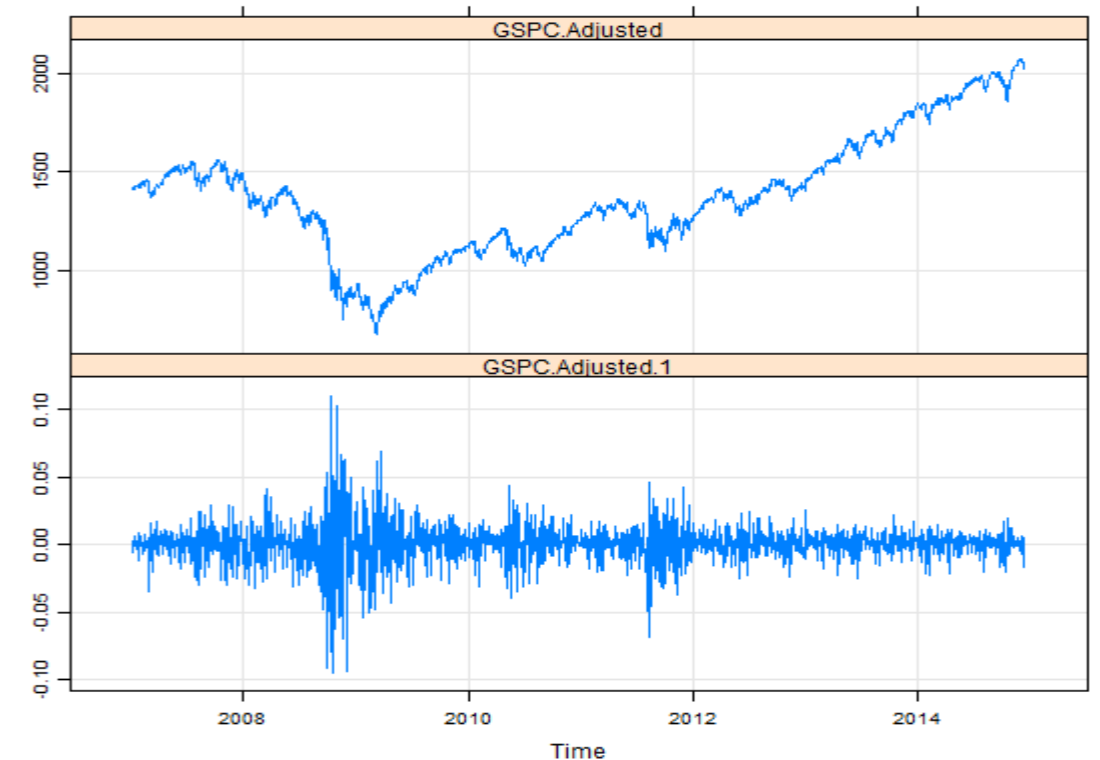
- обчислення значення виразів;
- спрощення дробово-раціональних виразів;
- розкладання многочленів на множники;
- розкладання числа на прості множники;
- знайдення НСД та НСК декількох чисел;
- розв'язування рівнянь та їх систем;
- побудова графіків функцій та рівнянь, які задані аналітично;
- графічне розв'язування рівнянь та їх систем;
- знаходження координат точок перетину графіків двох функцій на заданому проміжку;
- графічне розв'язування нерівностей та їх систем;
- побудова дотичної і нормалі до графіка функції у заданій точці з одночасним знаходженням їх рівнянь.
- трасування графіка, побудова таблиці значень;
- дослідження функції на даному проміжку (відшукування найбільших і найменших значень, екстремум, довжина кривої, нулі функції тощо);
- виконання чисельного інтегрування та його геометрична ілюстрація;
- знаходження первісної та похідної функції.

Геометрія:

- побудова різноманітних геометричних фігур на площині (точок, прямих, променів, ламаних, векторів, кутів, многокутників, правильних многокутників, бісектрис кутів, серединних перпендикулярів, паралельних і перпендикулярних прямих, кіл (за центром і точкою, за центром та радіусом, за трьома точками), дуг кіл та конічних перетинів, дотичних до кола);
- побудова геометричних фігур в просторі (площин, пірамід, призм, сфер, конусів, циліндрів тощо);
- обчислення площ: многокутника, круга, частини площини, обмеженої еліпсом, сектора;
- знаходження: градусної міри кута, довжини відрізка, периметра многокутника, довжини вектора, відстані від точки до прямої, тангенса кута між прямою та додатнім напрямком осі абсцис тощо;
- перетворення фігур на площині: симетрія відносно точки і прямої, поворот навколо точки, гомотетія, паралельне перенесення;
- знаходження точок перетину двох фігур (двох прямих, прямої і кола тощо);
- знаходження середини відрізка, центру кола (еліпса).

Графік рівня S&P500 і графік надходжень:

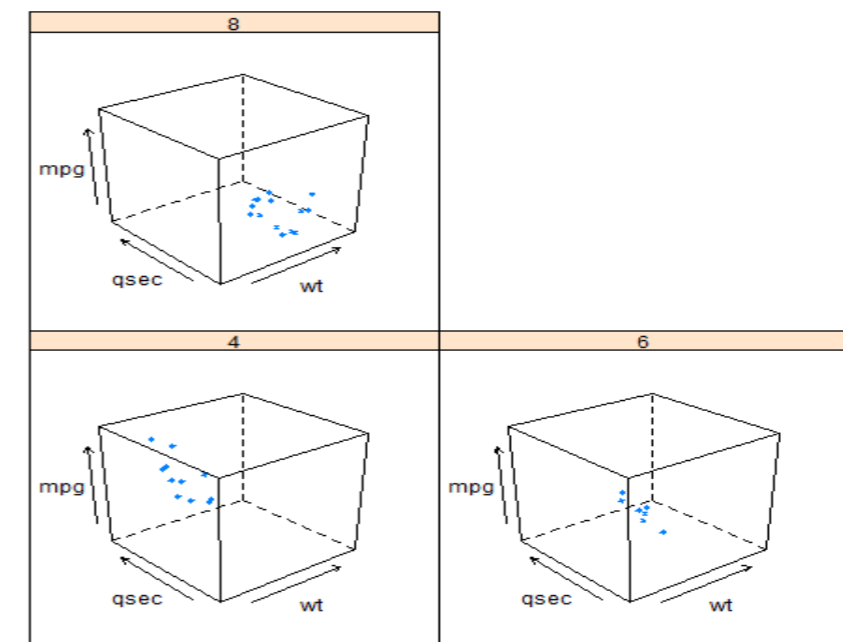
```
> xyplot(merge(Ad(GSPC),diff(log(Ad(GSPC))))),type=c(«l»,»g«))
```



3D графіки

Функція cloud() використовується для побудови 3-D хмари точок.

```
> cloud(mpg ~ wt * qsec | format(cyl),data=mtcars)
```



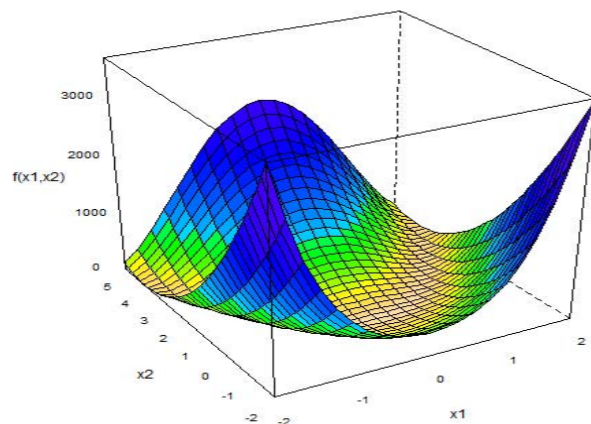
Приклад. Побудуємо графік функції Розенброка (яку ще називають банан)
 $f(x_1, x_2) = (1 - x_1)^2 + 100(x_2 - x_1^2)^2$

```
> banana = function(x) {val = 100*(x[2]-x[1]^2)^2+(1-x[1])^2}
> x <- seq(-2, 2, .2)
> y <- seq(-2, 5, .2)
> YX <- expand.grid(x,y)
> head(YX,3)
  Var1 Var2
1 -2.0  -2
2 -1.8  -2
3 -1.6  -2
> z <- apply(YX,1,banana)
> XYZ <- data.frame(y=YX[,2],x=YX[,1],z=z)
> breaks = as.numeric(quantile(z,(0:1000)/1000))
```

Для побудови поверхні використаємо функцію `wireframe()`.

```
> par.set <- list(axis.line=list(col=«transparent»), clip=list(panel=«off»))

> print(wireframe(z ~ x*y, data = XYZ,
  xlab=«x1»,ylab=«x2», zlab =«\n\nf(x1,x2) »,          scales = list(arrows =
FALSE),
  drape = F,
  shade=F,
  at=breaks,
  col.regions = rev(topo.colors(length(breaks)-1)),
  screen = list(z = 30, x = -60),
  colorkey=F,
  par.settings=par.set,
  zoom=0.95))
```



(<http://tube.geogebra.org>). На ній користувачі мають можливість створювати власні колекції, залишати коментарі та оцінювати якість розміщених матеріалів.

2.5.4. Інститути GeoGebra

Зважаючи на швидке зростання чисельності співтовариства навколо GeoGebra, стало зрозумілим, що і члени співтовариства, і вчителі, які тільки розглядають можливість використання GeoGebra в навчальному процесі, потребують широкої підтримки. Щоб бути в змозі надати таку допомогу і сприяти рефлексивній практиці, з метою активізації досліджень та подальшого розвитку системи динамічної математики GeoGebra, на конференції в Кембриджі у травні 2008 року було засновано інтернаціональну професійну мережу: Міжнародний Інститут GeoGebra (International GeoGebra Institute, IGI) [89]. Його основними цілями є:

- навчання та підтримка вчителів, викладачів і студентів педагогічних вищих навчальних закладів з метою сприяння зростанню їх професійної майстерності;
- розвиток та вдосконалення пакету GeoGebra;
- розробка та вільне розповсюдження інноваційних дидактичних матеріалів, створених за допомогою GeoGebra;
- сприяння співпраці між IGI та регіональними інститутами GeoGebra;
- проведення та підтримка GeoGebra досліджень, зокрема, приділення особливої уваги дослідженням, що пов'язані із вивченням математики та спрямовані на підвищення рівня її викладання. Після заснування IGI розпочався процес створення мережі регіональних осередків GeoGebra. В Україні започатковано регіональні інститути (<http://www.geogebra.org/institutes>) GeoGebra в містах Харкові та Чернігові. «Інститут GeoGebra Харків, Україна» засновано 6 липня 2010 року на базі кафедри інформатики Харківського національного педагогічного університету імені Г. С. Сковороди (<http://kafinfo.org.ua/geogebra>). «Інститут GeoGebra Чернігів, Україна» (<https://sites.google.com/site/geogebra-chernigiv>, <http://www.geogebra.org/geogebra.ch.ua>) засновано 3 лютого 2011 року на базі Чернігівського обласного інституту післядипломної педагогічної освіти (ЧОІППО) імені К. Д. Ушинського (<http://choippo.edu.ua>).

2.5.5. Використання GeoGebra у шкільному курсі математики

Функціональні можливості GeoGebra дозволяють ефективно використовувати її у процесі вивчення математики. Почнемо з найпростішого. За допомогою GeoGebra можна швидко створювати високоякісні графічні зображення математичних об'єктів (графіки функцій, графіки рівнянь, геометричні фігури, формули тощо) та потім їх зберігати у файлах графічних форматів (png; svg) або експортувати до буфера обміну. Після цього такі рисунки можна використовувати при створенні друкованих дидактичних

алгебру, математичний аналіз і статистику. Остання версія GeoGebra 5.0 пропонує кілька динамічно пов'язаних між собою представлень математичних об'єктів: графічне, алгебраїчне та табличне (рис. 2.106) та має у своєму складі систему комп'ютерної алгебри – СКА. Потрібно відмітити, що в процесі розвитку програми, з ростом її функціональних можливостей інтерфейс GeoGebra залишається простим у використанні та інтуїтивно зрозумілим. І цей підхід є одним з головних принципів концепції подальшого розвитку програми.

GeoGebra, яка в даний момент є доступною на більше ніж 65-ьох мовах світу, отримала кілька нагород освітніх програм у Європі та США (наприклад, EASA 2002, digita 2004, Comenius 2004, eTwinning 2006, AECT 2008, BETT 2009 finalist, Tech Award 2009, NTLC Award 2010).

Важливим є те, що для забезпечення підтримки користувачів GeoGebra, організації їх співробітництва та обміну досвідом створені потужні Інтернет-ресурси з використання сучасних веб-технологій (Веб 2.0, хмарних обчислень, Wiki-технологій). Ресурс www.geogebra.org (рис. 2.107) пропонує користувачам GeoGebra:

- постійно оновлювану базу науково-методичних і дидактичних матеріалів у вільному доступі;
- форум користувачів (учнів, студентів, вчителів, викладачів, освітян);
-



Рис. 2.107 – Інтерфейс ресурсу www.geogebra.org

- останні новини щодо заходів та подій у спільноті користувачів GeoGebra з різних куточків світу.

З метою вдосконалення сервісу для зберігання, перегляду, використання та обміну електронними відкритими дидактичними матеріалами, виготовленими за допомогою GeoGebra, було створено платформу GeoGebraTube

2.2.MATLAB

Історично MatLab розроблявся як діалогове середовище для матричних обчислень (MATrix LABoratory). Згодом пакет був оснащений графічною системою, доповнений засобами комп'ютерної алгебри від Maple і посилений бібліотеками команд (Toolboxes), призначеними для ефективної роботи зі спеціальними класами задач.

Програмний продукт MatLab (скорочення від MATrix LABoratory – матрична лабораторія) фірми MathWorks Inc. (www.mathworks.com) – це апробована і надійна інтерактивна програма яка призначена для розв'язування широкого кола математичних задач, орієнтована на роботу з масивами (векторами і матрицями) даних. Система підтримує:

- обчислення з матрицями і векторами,
- комплексну арифметику,
- обчислення з поліномами,
- розв'язання диференціальних і різницевих рівнянь,
- розв'язання лінійних і нелінійних рівнянь,
- різноманітні символні обчислення,
- побудова графіків у декартовій і полярній системах координат,
- побудова різноманітних тривимірних графіків,
- застосування теорії автоматичного управління,
- завдання цифрової обробки сигналів,
- аналіз даних,
- інші високорівневі функції і найрізноманітніші засоби розв'язування математичних завдань, не вдаючись до рутинного програмування [70].

MatLab має розширення у вигляді декількох десятків додаткових програмних пакетів для обчислень і обробки даних в різноманітних областях. Серед цих пакетів-розширень MatLab можна відзначити наступні:

Simulink – пакет моделювання динамічних систем,

Signal Processing Toolbox – пакет для аналізу, моделювання та проектування пристроїв обробки сигналів,

Control System Toolbox – пакет для моделювання, аналізу та проектування систем автоматичного управління,

Symbolic Math Toolbox – пакет символної математики (програма для розв'язання завдань в символному (аналітичному) вигляді),

Image Processing Toolbox – пакет обробки зображень,

Statistics Toolbox – пакет прикладних програм зі статистики та інші додаткові пакети [92].

У зв'язку з такими широкими можливостями MatLab вважається загально визнаним стандартом для науково-технічних обчислень. Він широко використовується в університетах і в науково-дослідних організаціях. MatLab є середовищем для математичних обчислень (перетворень) і в той же час надає користувачеві власну мову програмування. Мовою MatLab можна написати

необхідні користувачам функції і програми (сценарії) у вигляді *M*-файлів з розширенням *m*.

У середовищі системи будь-які і навіть вельми складні обчислення можна виконувати в режимі прямих обчислень, тобто без підготовки програми. У цьому режимі MatLab використовується як потужний калькулятор, який здатний виконувати не тільки звичайні для калькуляторів обчислення, а й операції з векторами і матрицями, комплексними числами, рядами і поліномами й інші складні обчислення. Можна також будувати і виводити графіки різних функцій – від простої синусоїди до складної тривимірної фігури, що дуже важливо і корисно для розуміння результатів обчислень.

Головним недоліком MatLab є його висока ціна. Крім того, MatLab некрайній для рутинних розрахунків. Відсутність інтерактивності робить його незручним при виконанні пошукових, дослідницьких розрахунків для нових, недосліджених масивів даних.

Проблему ціни вирішує альтернативне вільно-розповсюджене математичне забезпечення Chemometrics – спеціальна надбудова для системи Microsoft Excel.

2.2.1. Основи роботи з MatLab

До складу MatLab входять інтерпретатор команд, графічна оболонка, редактор-налаштовувач, бібліотеки команд, компілятор, символічне ядро пакета Maple для проведення аналітичних обчислень, математичні бібліотеки MatLab на C/C++, генератор звітів і багатий інструментаріями (Toolboxes). Інтерфейс MatLab цілком відповідає сучасним канонам (див. рис. 2.16). Він багатівіконний і має ряд засобів прямого доступу до різноманітних компонентів системи. Слід звернути увагу на наступні кнопки панелі інструментів:

- New M-file – виводить порожнє вікно редактора *m*-файлів;
 - Open file – відкриває вікно для завантаження файлів MatLab;
 - Simulink – відкриває вікно браузера бібліотек Simulink;
 - Help – відкриває вікно довідки.
- Ці функції дублюються в дуже простому меню системи MatLab.

У лівій частині вікна системи з'явилися вікна з вкладками Launch Pad/Workspace доступу до компонентів системи і вкладками поточної директорії Current Directory і історії сесії History. Вони забезпечують оперативний контроль за станом системи. Виведені на екран вікна інтерфейсу MatLab можуть бути включені або відключені з пункту меню View.

Вся робота організовується через командне вікно (Command Window), яке з'являється при запуску програми. У процесі роботи дані розміщуються в пам'яті (Workspace) у вигляді матриць.

ньому правою кнопкою миші (Mac OS: Ctrl-клік) і вибравши пункт "Видалити". Потім введіть наступні команди:

$$v = (1, f'(a))$$

$$t: X = T + r v$$

де v це напрям вектора дотичної t . Замість r ви можете також використовувати будь-яке інше буквене позначення як параметр.

Існує додаткова можливість для побудови дотичної за допомогою направляючого вектора: $t = \text{Line}[T, v]$.

Також спробуйте команду "інтеграл [f]"

Програма GeoGebra була створена в 2002 році, і була дипломним магістерським проектом Маркуса Хохенватера під час його навчання в університеті Зальцбурга. Вона була розроблена з метою поєднання можливостей програм динамічної геометрії (наприклад: Cabri Geometry, Geometer's Sketchpad) та систем комп'ютерної алгебри (наприклад: Derive, Maple) в одній простій у використанні системі, призначеній для вивчення та викладання математики.

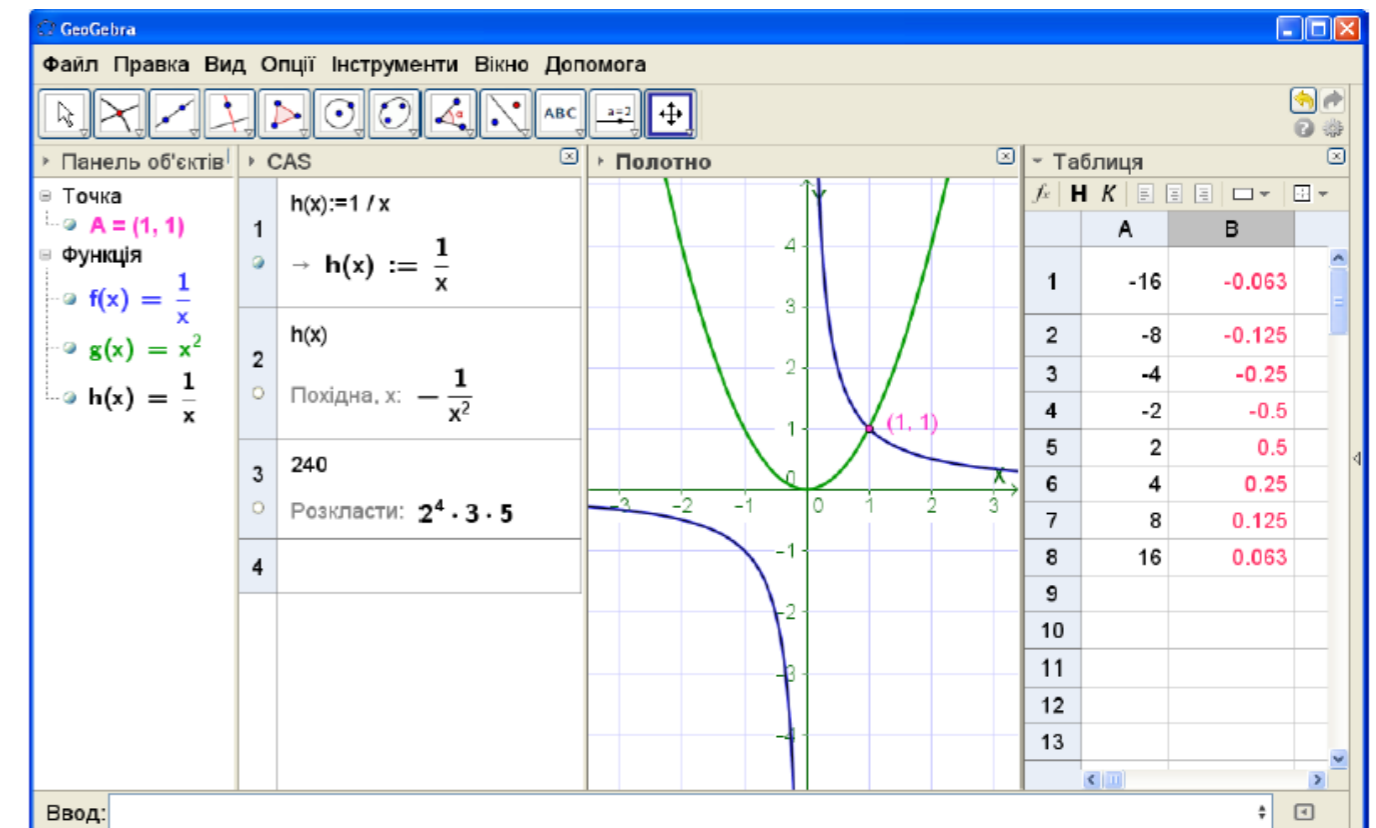



Рис. 2.106 – Інтерфейс програми GeoGebra

Протягом наступних років GeoGebra перетворилась на міжнародний проект з відкритим кодом, що активно розвивається, і над яким зараз працює інтернаціональна команда з більше ніж 30 розробників та більше ніж 140 перекладачів. На даний момент СДМ GeoGebra – це вільний педагогічний програмний продукт, призначений для вивчення і викладання математики в середніх та вищих навчальних закладах, який поєднує динамічну геометрію,

Версія 1: Точка на функції. Вставте функцію $f(x) = \sin(x)$ в поле введення тексту та натисніть клавішу введення enter.

 Виберіть режим "точка" і натисніть на функцію f. Це створює точку A на f.

Потім виберіть режим "касательные" і натисніть на точку A і функцію f. Зміни ім'я дотичній на t (правою кнопкою миші (Mac OS: Ctrl-клік), "перейменувати"). Наступна команда s = нахил [t] показує нахил дотичній. Після вибору режиму "Перемещение", перетягніть точку A за допомогою миші і спостерігайте рух дотичній. Введіть команду B = (x(A), s) і перейти на слід цієї точки (натисніть на B правою кнопкою миші (Mac OS: Ctrl-клік) і виберіть "залишати слід".) X(A) дає вам координати x точки A. Виберіть режим "Перемещение" і перетягніть A за допомогою миші - B залишить слід. Остання команда Похідна [f] Отримуємо похідну [f] - [f]'

Поради:

Вставте різні функції, наприклад, $f(x) = x^3 - 2x^2$ в полі введення тексту; відразу її похідна і дотична будуть відображатися.

Виберіть режим "Переміщення" і перетягніть функції графіка за допомогою миші. Спостерігайте зміна рівняння функції та її похідної.

Версія 2: точка x = Зараз ми збираємося зробити ще один варіант останнього побудови. Тому, вибирайте "Файл - Створити", щоб отримати новий майданчик для малювання. Далі, введіть наступні команди в текстове поле введення і натискайте клавішу введення Enter після кожного рядка.

```
f(x) = sin(x)
a = 2
T = (a, f(a))
t = Касательная[a, f]
s = наклон[t]
B = (x(T), s)
Производная[f]
```

Виберіть режим "Переместить" і натисніть на число a в лівому віконці. Ви можете змінювати його, натискаючи на клавіші зі стрілками. У той же час, точка T і дотична будуть рухатися по функції f.

Ви також можете змінювати значення числа a за допомогою повзунка: для цього у вікні алгебри треба вибрати мишкою кількість a і у вікні для малювання, клацнувши правою кнопкою миші, вибрати "Показати об'єкт".

Порада: повзунки, а також клавіші зі стрілками, дуже корисні для вивчення параметрів p і q в квадратних рівняннях $y = x^2 + px + q$

2.5.3. Дотичні без даної команди

GeoGebra в змозі мати справу з векторами, а також параметричним представленням ліній.

Тому можна побудувати дотичну t без команди "Дотична []". Щоб переконатися в цьому, видаліть дотичну з вашого креслення, клацнувши по

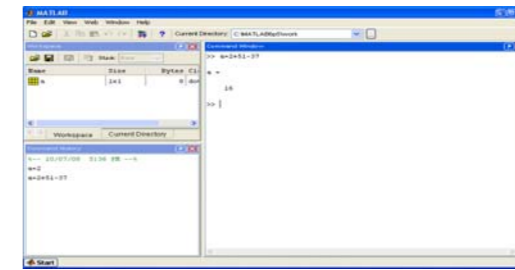


Рис.2.16 – Інтерфейс програми MatLab

Всі розрахунки в MatLab виконуються з подвійною точністю, а для представлення чисел на екрані є різні формати. Потрібний формат може бути визначений в меню (**File/Preferences**) або за допомогою команди **format**. Існують наступні способи подання чисел (табл.2.1) [22].

Таблиця 2.1

Формати виведення на екран

Формат	Подання
Short (default)	Число відображається з 4 цифрами після десяткової точки або в форматі short e
short e	Число в експоненційній формі з мантисою з 5 цифр і показником з 3 цифр
rat	Подання у вигляді раціонального дробового числа
long	Число з 16 десятковими цифрами
long e	Число в експоненційній формі з мантисою з 16 цифр і показником з 3 цифр
hex	Число в шістнадцятковій формі

Змінні в MatLab не потрібно попередньо описувати, вказуючи їх тип. Усі дані зберігаються у вигляді масивів: числові змінні (внутрішній тип numeric), текстові рядки (char), чарунки (Cell) і структури (struct).

Двовимірний масив – це матриця, одновимірний – вектор, а скаляр – матриця розміру 1x1. Ім'я змінної має починатися з літери, за нею можуть йти букви, цифри і символ підкреслення. Допустимі імена будь-якої довжини, але MatLab ідентифікує їх за першими 31 символами і розрізняє великі і малі літери. У MatLab є ряд констант (табл.2.2) [22].

Таблиця 2.2

Зарезервовані імена констант

Ім'я	Опис
ans	Результат останньої операції
i, j	Уявна одиниця (корінь квадратний з -1)
pi	Число π (зберігається у виді 3.141592653589793)
eps	Машинна точність
realmax	Максимальне дійсне число

Ім'я	Опис
realmin	Мінімальне дійсне число
inf	Нескінченність
NaN	Нечислова змінна (наприклад, типу 0/0 або inf/inf)
end	Найбільше значення індексу розмірності масиву

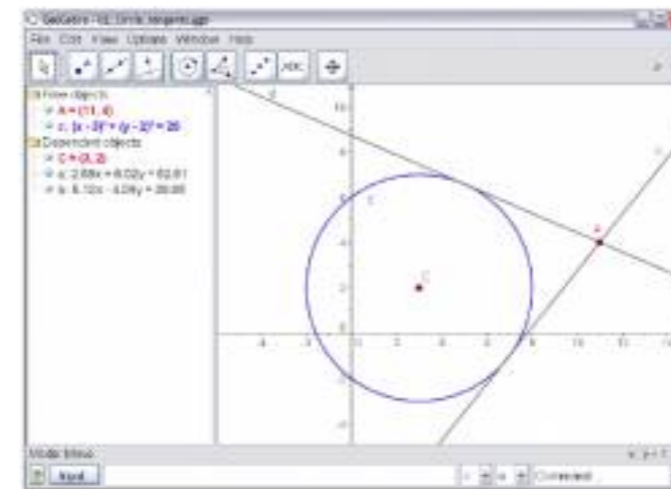
Таблиця 2.3

Спеціальні символи

Символ	Призначення
[]	Квадратні дужки використовуються при заданні матриць і векторів
	Пропуск служить для поділу елементів матриць
,	Кома застосовується для поділу елементів матриць і оператора у рядку введення
;	Крапка з комою відокремлює рядки матриць, а крапка з комою в кінці оператора (команди) скасовує виведення результату на екран
:	Двокрапка використовується для вказівки діапазону (інтервалу зміни величини) і в якості знаку групової операції над елементами матриць
()	Круглі дужки застосовуються для задання порядку виконання математичних операцій, а також для вказівки аргументів функцій і індексів матриць
.	Крапка відокремлює дробову частину числа від цілої його частини, а також застосовується в складі комбінованих знаків (. *, . ^, ./, \)
...	Три крапки і більш в кінці рядка відзначають продовження виразу в наступному рядку
%	Знак відсотка означає початок коментаря
'	Апостроф вказує на символні рядки, а для включення самого апострофа в символний рядок потрібно поставити два апострофа поспіль

У командному вікні в режимі діалогу проводяться обчислення. Користувач вводить команди або запускає на виконання файли з текстами на мові MatLab. Інтерпретатор обробляє введені значення і видає результати: числові і рядкові дані, попередження і повідомлення про помилки. Рядок введення помічений знаком >>.

При роботі з MatLab в командному режимі діє найпростіший рядковий редактор. Зверніть особливу увагу на застосування клавіш Up і Down (стрілки курсору «Вгору» та «Вниз»). Вони використовуються для підстановки після



2.5.2. Використання для побудови поля введення тексту і миші

Вставте рівняння кола c: $(x - 3)^2 + (y - 2)^2 = 25$ в полі введення тексту і натисніть клавішу введення (показник може бути знайдений у списку праворуч в полі введення). Введіть команду C = Center [c] в полі введення тексту. Побудуйте точку A з координатами = (11, 4).

Тепер виберіть режим "касательные" і натисніть на точку A і окружність C.

Після вибору режиму "Переместить", перетягніть точку A за допомогою миші і спостерігайте рух дотичних. Також спробуйте перемістити окружність C і подивитися на її рівняння у вікні алгебри.

Деякі поради:

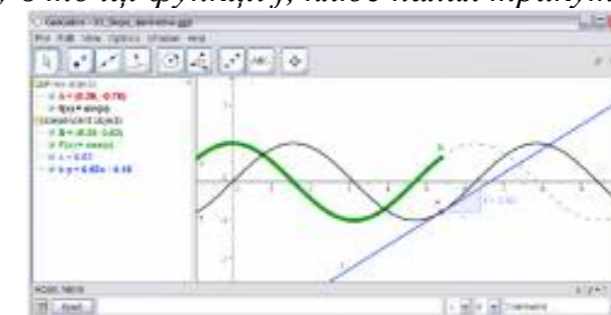
Можна використовувати інструменти у правому меню панелі інструментів, щоб збільшити або зменшити масштаб (клікніть правою кнопкою мишки). Якщо у вас миша з коліщатком, спробуйте використовувати Ctrl + колесо миші для збільшення.

Можна змінити рівняння кола безпосередньо у вікні алгебри двічі клацнувши по ньому мишею.

Більш докладну інформацію про можливості поля введення тексту можна знайти в меню "Довідка", розділ "Algebraic input".

Приклад 3: Похідна та дотична функції

Задача: Використовуйте GeoGebra, щоб побудувати функцію $f(x) = \sin(x)$, її похідну і дотичну в точці функції f, плюс нахил трикутника (дотичній).



Для того щоб змінити положення вікна для малювання, виберіть режим "Перемістити креслення" (останній значок панелі інструментів) і просто використовуйте мишу для переміщення.

Меню " Вид - Протокол " надає таблицю із зазначенням всіх кроків, які ви зробили при побудові креслення. Це дозволяє повторити всі побудова крок за кроком за допомогою клавіш зі стрілками, а також можна змінити порядок різних кроків .

Крім того, ви можете використовувати меню "Вид", щоб приховати або показати небажані колонки.

Додаткова інформація про побудову за допомогою миші можна знайти в меню "Довідка" , розділ " *Geometric input* " .

2.5.1. Використання для побудови текстового поля введення

Зараз ми збираємося зробити ту ж саме побудову, використовуючи текстове поле введення.

Вам потрібен новий майданчик для малювання (меню "Файл - Создать"). Потім введіть наступні команди в поле введення тексту в нижній частині екрана, натискаючи клавішу введення Enter після кожного рядка:

```
A = (2, 1)
B = (12, 5)
C = (8, 11)
Polygon[A, B, C]
l_a = LineBisector[a]
l_b = LineBisector[b]
M = Intersect[l_a, l_b]
Circle[M, A]
```

Додаткові поради:

Автоматичне завершення команди: після введення перших двох букв команди, вона буде відображатися автоматично. Якщо ви хочете прийняти запропоновану команду, натисніть enter, інакше просто продовжуйте друкувати. Це не є необхідним для введення кожної команди, ви можете також вибрати її зі списку команд, що знаходиться в безпосередній близькості від поля введення тексту. Натискання на ікону " Ввод " (внизу зліва) активує режим "Поле вводу". У цьому режимі ви можете натиснути правою кнопкою миші на об'єкт з вікна алгебри або вікна для малювання, щоб скопіювати його в поле введення тексту. Додаткові поради, що стосуються текстового поля введення: натисніть на знак питання в лівому кутку. Ви отримаєте особливо хороші результати від роботи з GeoGebra шляхом об'єднання переваг обох форм введення, мишею і текстового поля введення.

Приклад 2: Дотичні до кола

Задача: Використовуючи GeoGebra, побудувати коло $c: (x - 3)^2 + (y - 2)^2 = 25$ і її дотичні через точку $(11, 4)$

маркера рядка введення >> раніше введених рядків зі спеціального стека, наприклад, для їх виправлення, дублювання або доповнення. При цьому зазначені клавіші забезпечують перегортання раніше введених рядків знизу нагору або зверху вниз.

Імена змінних повинні починатися з літери. Знак = відповідає операції присвоювання. Натискання клавіші Enter змушує систему обчислити вираз і показати результат. Якщо запис оператора не закінчується символом «;», то результат виводиться в командне вікно, в іншому випадку – не виводиться. Якщо оператор не містить знаку присвоювання «=», то значення результату присвоюється системною змінною ans (див. рис. 2.17).

Всі значення змінних, обчислені протягом поточного сеансу роботи, зберігаються в спеціально зарезервованій області пам'яті комп'ютера, званої робочим простором системи MatLab (Workspace).

Для перегляду значення будь-якої змінної з поточного робочого простору системи досить набрати її ім'я і натиснути клавішу Enter.

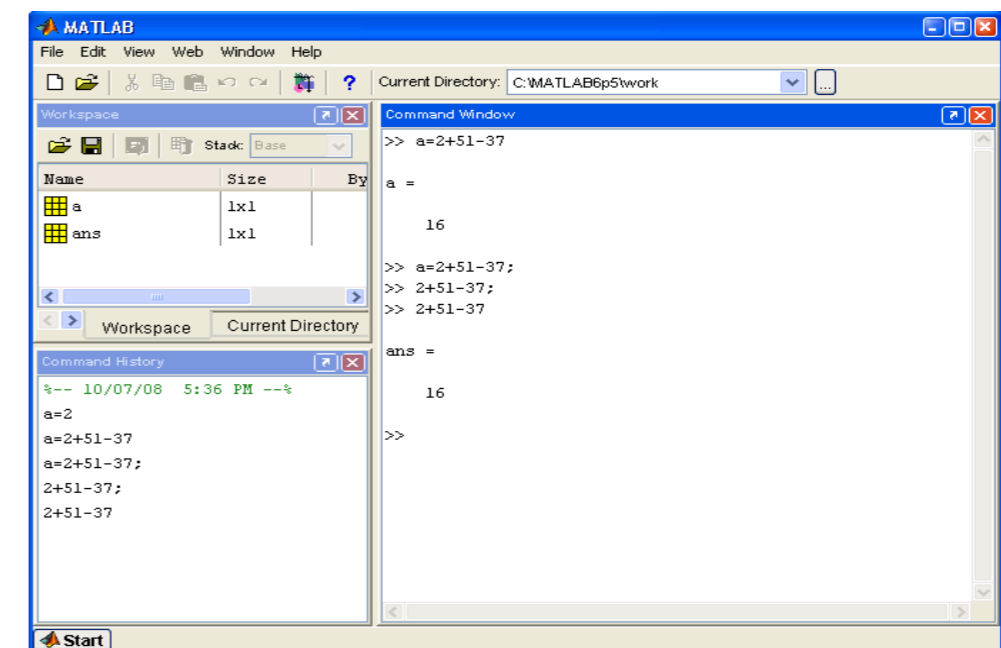


Рис. 2.17 – Демонстрація виконання команди присвоювання

Після закінчення сеансу роботи з системою MatLab усі раніше обчислені змінні втрачаються. Щоб зберегти в файлі на диску комп'ютера вміст робочого простору системи MatLab, потрібно виконати команду меню File\Save Workspace As По замовчуванню розширення імені файлу mat, тому такі файли прийнято називати Mat-файлами.

Система MatLab працює як з дійсними, так і з комплексними числами. Перед використанням операцій з комплексними числами необхідно визначити змінну $i = \text{sqrt}(-1)$ або $j = \text{sqrt}(-1)$. В арифметичних виразах застосовуються такі знаки операцій:

- + – додавання;
- – віднімання;
- * – множення;

- / – правостороннє ділення матриць;
- \ – лівостороннє ділення матриць;
- ^ – піднесення до степеня [22].

Система MatLab дозволяє обчислювати різні математичні функції. Наступні елементарні алгебраїчні функції мають як аргумент одне або два дійсних (x, y) або одне комплексне (z) число (табл. 2.4) [22].

Таблиця 2.4

Елементарні алгебраїчні функції

Функція	Опис
abs (z), abs (x)	Обчислення модуля комплексного числа z або абсолютного значення дійсного числа x
angle (z)	Обчислення аргумента z
sqrt (z), sqrt (x)	Обчислення квадратного кореня чисел z і x
real (z)	Обчислення дійсної частини комплексного числа z
imag (z)	Обчислення уявної частини комплексного числа z
round (x)	Округлення до цілого
fix (x)	Округлення до найближчого цілого в бік нуля
rem (x, y)	Обчислення залишку від ділення x на y
exp (z)	Обчислення e в степені x
log (z)	Обчислення натурального логарифма числа x
log10 (z)	Обчислення десяткового логарифма числа x

Система MatLab надає можливості для обчислення наступних тригонометричних і обернених тригонометричних функцій змінної x (табл.2.18).

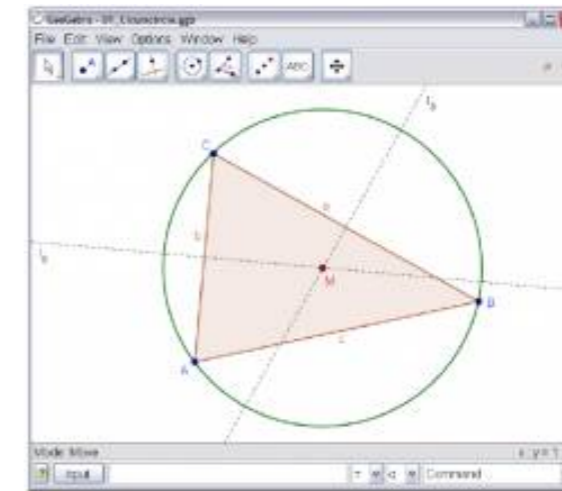
Таблиця 2.5

Тригонометричних функцій

Функція	Опис
sin (x)	Обчислення синуса
cos (x)	Обчислення косинуса
tan (x)	Обчислення тангенса
asin (x)	Обчислення арксинуса
acos (x)	Обчислення арккосинуса
atan (x)	Обчислення арктангенса
atan2 (y, x)	Обчислення арктангенса за координатами точки

Приклад 1. Обчислити вираз $(25+17)*7$.

Це можна зробити таким чином. Спочатку набираємо послідовність $25+17$ і натискаємо <Enter>. Одержуємо на екрані результат у виді $ans=42$. Тепер записуємо послідовність $ans*7$ і натискаємо <Enter>. Одержуємо $ans=294$ (рис. 2.18 а). Щоб запобігти виведенню проміжного результату дії $25+17$, достатньо після запису цієї послідовності додати символ ' ; '. Тоді будемо мати результати у виді, поданому на рис. 2.18 б [53].



Як побудувати:

Виберіть режим "Багатокутник" на панелі інструментів (натисніть на маленьку стрілку - третій значок ліворуч). Тепер натисніть на вікно для малювання три рази, щоб створити вершини А, В і С. Завершіть побудову трикутника, натиснувши на вершину А ще раз.

Потім виберіть режим "Серединний перпендикуляр" і побудуйте два серединних перпендикуляра, натиснувши на дві сторони трикутника.

У режимі "Перетин двох об'єктів" ви можете натиснути на перетин двох серединних перпендикулярів, щоб отримати центр кола. Назвемо цю точку "М", для цього клацніть по ній правою кнопкою миші і виберіть (перейменувати) в меню.

Щоб закінчити побудову, ви повинні вибрати режим "окружність по центру і точці" і натиснути спочатку в центр, потім на будь-яку вершину трикутника.

Тепер виберіть режим "Переміщення" і використовуючи мишу, можете змінити положення будь-якої з вершин - ви зрозумієте сенс "динамічної геометрії".

Деякі поради :

Спробуйте використовувати значок "Скасувати" на правій стороні панелі інструментів.

Щоб приховати об'єкт , клацніть правою кнопкою миші на ньому (в Mac OS : Ctrl - клік) і зніміть прапорець " Показати об'єкт".

Властивості об'єктів (колір, тип лінії, ...) можуть бути легко змінені : правою кнопкою миші (Mac OS : Ctrl - клік) знову натисніть на об'єкт і виберіть " Properties ", " Властивості " в контекстному меню.

У меню "Вид" вікна алгебри осі і сітки можуть бути приховані або показані.

напружує. Адже все, що від нас вимагається при установці — періодично натискати кнопку «Next». А під час першого запуску GeoGebra — вибрати потрібну мову в меню (*Options — Language*).

Інтерфейс GeoGebra простий, чистий і зрозумілий. Все зроблено зі справжньою математичною точністю і геометричною акуратністю.

Наведені приклади дозволять ознайомитись з основними функціями програми.

Приклад 1: Трикутник вписаний в коло

Приклад 2: Дотичні до кола

Приклад 3: Похідна і дотична функції

Після запуску GeoGebra з'являється вікно, що зображено на рис. 2.105.

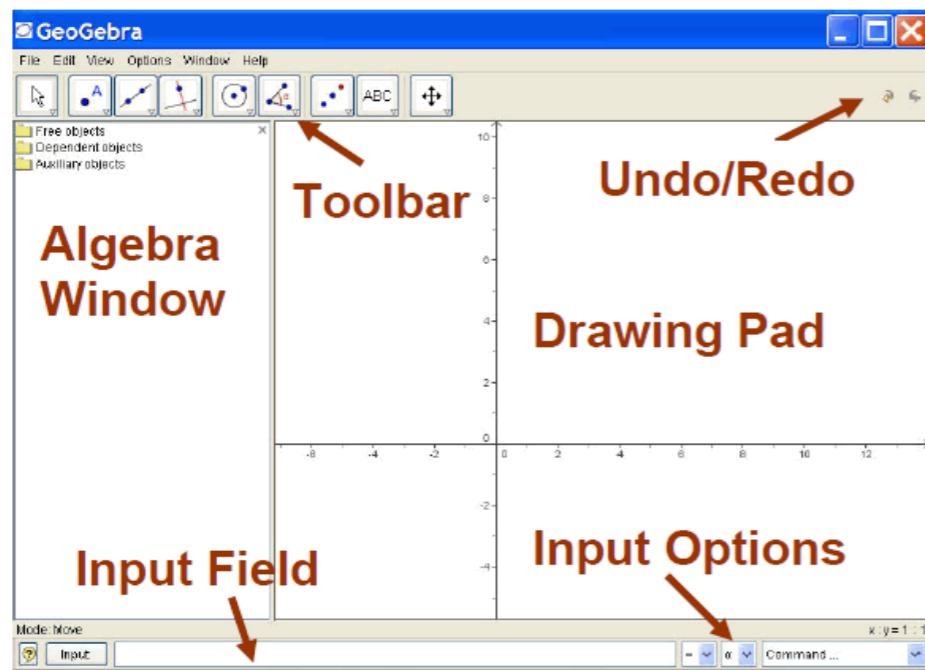
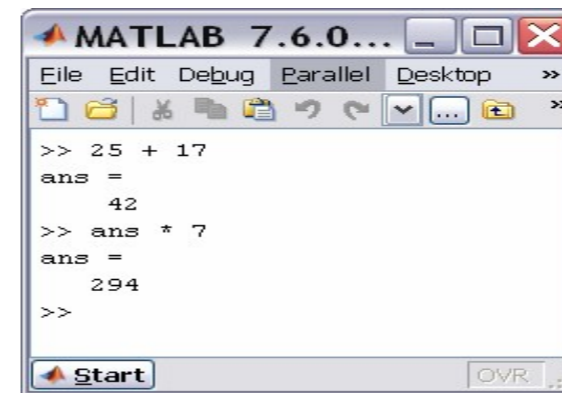


Рис.2.105 – Головне вікно GeoGebra

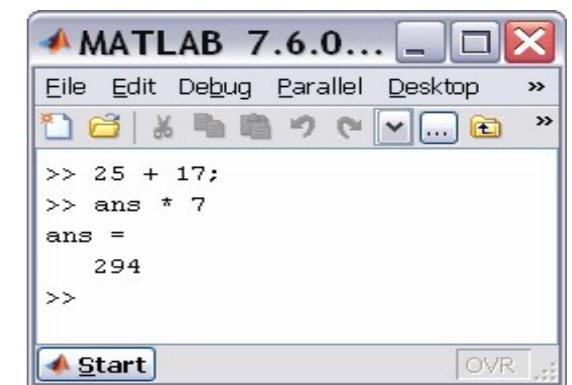
За допомогою панелі інструментів (toolbar) можна створювати об'єкти у вікні для малювання (drawing pad) за допомогою миші. У той же самий час відповідні координати і рівняння будуть показані у вікні алгебри (algebra window). Вхідна текстова область (input text field) використовується для введення координат, рівнянь, команд і функцій відповідно, Input Field - поле введення, Input Options - параметри введення; вони негайно відображаються на кресленні, тобто в блокноті-вікні для малювання після натискання клавіші ENTER.

Приклад 1: Побудова кола навколо трикутника

Задача: Побудувати коло навколо трикутника A, B, C з використанням GeoGebra



а)



б)

Рис. 2.18 – Обчислення виразу $(25+17)*7$

2.2.2. Операції з векторами і матрицями в системі MatLab

За замовчуванням всі числові змінні в MatLab вважаються матрицями, отже скалярна величина є матриця першого порядку, а вектори є матрицями, що складаються з одного стовпчика або одного рядка. Матрицю можна ввести, задавши її елементи або прочитавши дані з файлу, а також в результаті звернення до стандартної або написаної користувачем функції.

Матричні дані розміщуються в пам'яті послідовно по стовпцях. Елементи матриці в межах рядка відокремлюються пробілами або комами. Безпосереднє завдання матриці можна здійснити кількома способами. Наприклад, вектором-стовпцем, тобто матриця, розмірність якої дорівнює одиниці, може бути присвоєна змінної A введенням одного рядка:

```
>> A = [7 + 4i; 4; 3.2] % Введення вектора-стовпчика
```

```
A =
 7.0000 + 4.0000i
 4.0000
 3.2000
```

або введенням кількох рядків

```
>> A = [% Введення вектора по рядках
7 + 4i
4 3.2];
```

Вектори можуть бути сформовані як діапазони за допомогою двокрапки, які поділяють стартове значення, крок і граничне значення. Якщо величина кроку відсутня, то за замовчуванням його значення дорівнює одиниці.

В результаті $n : m : k$ буде сформований вектор, останній елемент якого не більше k для додатного кроку m , і не менше – для від'ємного: $[n, n + m, n + m + m, \dots]$ наприклад:

```
>> A = 1:2:5
a = 1 3 5
```

Завдання діапазону використовується також при організації циклу.

У таблиці 2.6 представлений деякий набір функцій для створення матриць спеціального виду [64].

Таблица 2.6

Функції опису матриць

Функція	Опис
eye (m, n)	Одинична матриця розмірності $m \times n$
zeros (m, n)	Нульова матриця розмірності $m \times n$
ones (m, n)	Матриця, що складається з одних одиниць розмірності $m \times n$
rand (m, n)	Повертає матрицю випадкових чисел рівномірно розподілених в діапазоні від 0 до 1, розмірність $m \times n$
randn (m, n)	Повертає матрицю розмірності $m \times n$, що складаються з випадкових чисел, що мають Гаусовим розподіл
tril (A), triu (A)	Виділення нижньої трикутної і верхньої трикутної частин матриці A
inv (A)	Знаходження оберненої матриці A
det (A)	Знаходження визначника (детермінанта) квадратної матриці A

Звернення до елемента матриці виконується за правилом, – в круглих дужках після імені матриці даються індекси, які повинні бути додатними цілими числами, що вказують номер рядка і через кому, номер стовпця. Наприклад, $A(2,1)$ означає елемент з другого рядка першого стовпчика матриці A.

Для подальших прикладів введемо матрицю 2x2:

```
>> A=[1 2+5*i; 4.6 3]
```

```
A =
 1.0000    2.0000 + 5.0000i
 4.6000    3.0000
```

Щоб змінити елемент матриці, йому потрібно присвоїти нове значення:

```
>> A(2,2)=10 % Второй элемент второй строки
```

```
A =
 1.0000    2.0000 + 5.0000i
 4.6000    10.0000
```

Розмір матриці можна уточнити по команді **size**, а результат команди **size** можна використовувати для організації нової матриці.

Наприклад, нульова матриця того ж порядку, що і матриця A, буде сформована за командою

```
>> A2=zeros(size(A))
```

```
A2 =
 0 0
 0 0
```

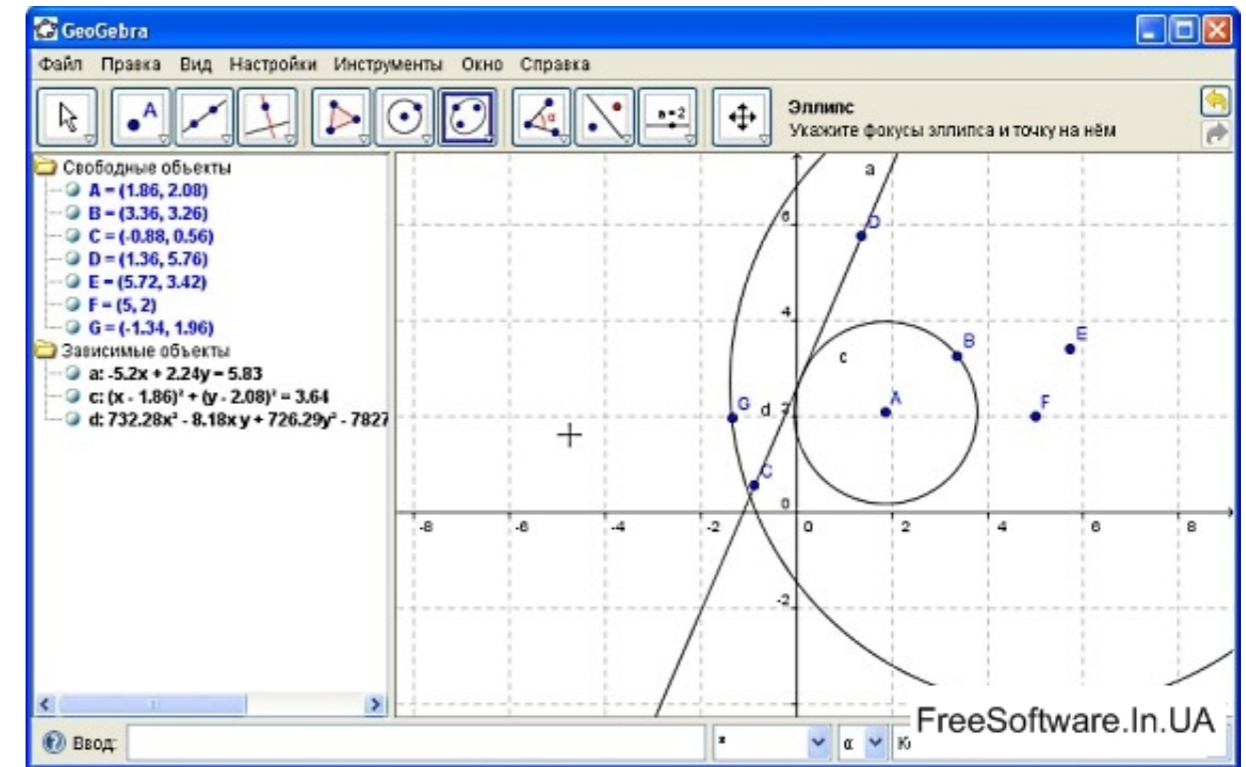
За допомогою двокрапки легко виділити частину матриці. Наприклад, вектор з перших двох елементів другого стовпця матриці A задається виразом:

```
>> A(1:2, 2)
ans =
 2.0000 + 5.0000i
 10.0000
```

Двокрапка сама по собі означає рядок або стовпець цілком. Для видалення елемента вектору досить привласнити йому порожній масив – пару квадратних дужок [2,4]. Щоб викреслити один або декілька рядків (стовпців)

2.5. GeoGebra

GeoGebra — безкоштовна програма, що надає можливість створення динамічних («живих») креслень для використання на різних рівнях навчання геометрії, алгебри, планіметрії та інших суміжних дисциплін. Програма володіє багатьма можливостями роботи з функціями (побудова графіків, обчислення коренів, екстремумів, інтегралів і т. д.):



На відміну від інших програм для динамічного маніпулювання геометричними об'єктами, ідея GeoGebra полягає в інтерактивному поєднанні геометричного, алгебраїчного і числового подання. Ви можете створювати конструкції з точками, векторами, лініями, кінчними перерізами, а також математичними функціями, а потім динамічно змінювати їх.

Крім того, GeoGebra дозволяє напряму вводити рівняння і маніпулювати координатами. Таким чином, можна легко складати графіки функцій, працювати зі слайдерами для підбору необхідних параметрів, шукати символічні похідні, і використовувати потужні команди на кшталт кореня і послідовності.

Для запуску і використання програми GeoGebra нам знадобиться Java, тому якщо вона у вас ще не встановлена загляньте [сюди](#). Звичайно, комусь це може не сподобатися, але зате така архітектура дає можливість запуску програми в більшості популярних операційних систем. І не тільки як окрему програму, але і як [онлайн-застосування](#).

Інсталяція програми не містить ніяких сюрпризів. Все досить інтелігентно, чисто і «по-математичному» :). На жаль серед мов, на яких може проводитися установка, нема української (ні російської), але це не сильно


```

-->A=[7.8 -1.6 2.4 1.9; -2 8.4 5.5 -3.8; 4.7 9 1.6 -3; 1.8 1.5 -2.8 4];

-->b=[7; -3.1; 2.9; 8.5];

-->A1=A; A1(:,1)=b; d(1)=det(A1); A2=A; A2(:,2)=b; d(2)=det(A2);

-->A3=A; A3(:,3)=b; d(3)=det(A3); A4=A; A4(:,4)=b; d(4)=det(A4);

-->x=d/det(A)

x =

    0.6294332
    0.5377544
   -0.0443315
    1.6090651

```

2.4.31. Рішення СЛАР методом Гауса

Суть методу Гауса в тому, що вихідна система еквівалентними перетвореннями приводиться до вигляду, зручного для отримання рішення.

Метод Гауса складається з двох етапів. На першому, званому прямим ходом, матриця системи за допомогою лінійних перетворень (перестановок, складання рівнянь, їх домноження на число) приводиться до трикутного вигляду. На другому етапі (зворотний хід) ця матриця перетвориться в одиничну. В результаті всіх цих перетворень правий стовпець міститиме рішення системи.

Приклад.

```

-->C=rref([A b])

C =

    1.    0.    0.    0.    0.6294332
    0.    1.    0.    0.    0.5377544
    0.    0.    1.    0.   -0.0443315
    0.    0.    0.    1.    1.6090651

-->x=C(:,5)

x =

    0.6294332
    0.5377544
   -0.0443315
    1.6090651

```

матриці потрібно вказати діапазон рядків (стовпчиків), що видаляються для однієї розмірності і поставити двокрапку для іншої розмірності. Для знаходження довжини вектору можна скористатися також командою `length`.

Набір арифметичних операцій в MatLab для роботи з матрицями складається зі стандартних операцій додавання – віднімання, множення – ділення, операції піднесення до степеня і доповнені спеціальними матричними операціями (табл.2.7). Якщо операція застосовується до матриць, розміри яких не узгоджені, то буде виведено повідомлення про помилку [64].

Для поелементного виконання операцій множення, ділення і піднесення до степеня застосовуються комбіновані знаки (крапка і знак операції). Наприклад, якщо за матрицею стоїть знак (^), то вона підноситься до степеня, а комбінація (. ^) Означає піднесення до степеня кожного елемента матриці. При множенні (додавання, віднімання, ділення) матриці на число відповідна операція завжди проводиться поелементно [64].

Таблиця 2.7

Знаки операцій	
Символ	Призначення
+,-	Символи плюс і мінус позначають знак числа або операцію додавання і віднімання матриць, до того ж матриці повинні бути однієї розмірності
*	Знак множення означає матричне множення, для поелементного множення матриці застосовується комбінований знак (.*)
'	Апостроф позначає операцію транспонування (разом з комплексним спряженим), транспонування без обчислення спряженого позначається за допомогою комбінованого знака (.')
/	Ліве ділення
\	Праве розділення
^	Оператор піднесення до степеня, для поелементного піднесення до степеня застосовується комбінований знак (.^)

Проілюструємо відмінність звичайного і поелементного множень за допомогою наступного прикладу.

Введемо матрицю H розміру 2×2 і матрицю D – одиничну тієї ж розмірності:

```

>> H=[0 1; 2 3], D=ones(size(H))

H =
    0    1
    2    3

D =
    1    1
    1    1

```

Перемножимо матриці, використовуючи звичайне множення:

```
>> H*D
ans =
    1    1
    5    5
```

Тепер застосуємо поелементну операцію:

```
>> H.*D
ans =
    0    1
    2    3
```

Система MatLab має ряд функцій, призначених для обробки даних, заданих в матричній або векторній формі (таблиця 2.8) [64].

Таблиця 2.8

Функції для роботи з матрицями

Функція	Опис
size(A)	Повертає масив, що складається з числа рядків і числа стовпців матриці.
sum(A)	Повертає суму всіх елементів по стовпцю
mean(A)	Повертає середнє значення стовпця матриці
std(A)	Повертає середньоквадратичне відхилення стовпця матриці
min(A), max(A)	Повертає мінімум і максимум відповідно, за стовпцем матриці
sort(A)	Сортує стовпець матриці по зростанню
prod(A)	Обчислює добуток всіх елементів стовпців

Наведемо приклади для даних команд. Спочатку вводимо рядок:

```
>> s = 'Привет'
s =
```

Привет

Відзначимо, що для введення російських букв слід вибрати в меню File/Preferences/Command Windows Font шрифт з російським кодуванням.

```
>> h = [v + 'от MatLab']
v =
```

Привет от MatLab

Той же результат вийде, якщо замість змінної v використовувати строкову змінну s.

Для перетворення численних даних в змінні рядка існують команди перетворення. У таблиці 2.9 наведені функції роботи зі змінними рядка і їм обернені, а повний список можна отримати по команді help strfun [64].

Таблиця 2.9

Функції роботи зі змінними рядка

Функція	Дія
num2str	Перетворення числа в рядок
int2str	Перетворення цілого числа в рядок
mat2str	Перетворення матриці в рядок

```
-->A=[7.8 -1.6 2.4 1.9; -2 8.4 5.5 -3.8; 4.7 9 1.6 -3; 1.8 1.5 -2.8 4];
```

```
-->b=[7; -3.1; 2.9; 8.5];
```

```
-->x=inv(A)*b
```

x =

0.6294332

0.5377544

- 0.0443315

1.6090651

```
-->(A*x)'
```

ans =

7. - 3.1 2.9 8.5

Зверніть увагу, по-перше, стовпець правих частин вводиться саме як стовпець, а не рядок, по-друге, при можливості завжди робіть перевірку. З цією метою ми вихідну матрицю системи помножили на знайдене рішення і отримали, як і повинні були, правий стовпець. Правда, на екран він виведений для зручності у вигляді рядка. Для цього використовувалася операція транспонування.

При вирішенні практичних завдань іноді буває необхідно не просто вирішити систему, а зробити це певним способом, щоб скористатися проміжними результатами, отриманими при його використанні.

Рішення СЛАР методом Крамера

За правилом Крамера рішення системи будується наступним чином:

$$x_i = \Delta_i / \Delta,$$

де Δ - визначник матриці системи, Δ_i - визначник матриці, отриманої з вихідної заміною i-го стовпця матриці системи стовпцем вільних членів.

Приклад.

Ортогональна матриця - це матриця, транспонована до якої є зворотною: $M' = M^{-1}$. Переконаємося, що в нашому прикладі вийшло саме так.

Продовження прикладу

```
-->Q*Q'
```

ans =

```

1.          - 3.886D-16  - 2.220D-16  - 1.665D-16
- 3.886D-16    1.          1.527D-16  - 2.012D-16
- 2.220D-16    1.527D-16    1.          0.
- 1.665D-16  - 2.012D-16    0.          1.

```

2.4.29. Система лінійних алгебраїчних рівнянь (СЛАР)

Система n рівнянь з m невідомими

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_m = b_1$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_m = b_2$$

.....

$$a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nm}x_m = b_n$$

називається системою лінійних алгебраїчних рівнянь. Тут x_j - невідомі, a_{ij} - коефіцієнти при невідомих, b_n -вільні члени рівнянь, $i = 1, \dots, n, j = 1, \dots, m$.

Система може бути представлена в матричному вигляді

$$Ax = b$$

У випадку, якщо кількість невідомих дорівнює кількості рівнянь, а визначник матриці системи A і вектор вільних членів b не рівні нулю, система має одне і тільки одне рішення. Ми будемо розглядати тільки такі системи.

Завдання рішення СЛАР настільки поширена в математиці, техніці і природничих науках, що Scilab надає кілька способів її вирішення.

2.4.30. Рішення СЛАР методом зворотної матриці

Скористаємося матричним представленням системи $Ax = b$. Якщо визначник матриці не дорівнює нулю, рішення визначається наступним чином:

$$x = A^{-1}b.$$

Тут A^{-1} - обернена матриця, обчислення якої в Scilab реалізується функцією `inv`.

Приклад.

str2mat	Об'єднання рядків в матрицю
str2num	Перетворення рядка в число
strcat	Об'єднання рядків

Інтегрування MatLab і MS Excel дозволяє користувачеві MS Excel звертатися до численних функцій MatLab для обробки даних, різних обчислень і візуалізації результату. Надбудова `exclink.xla` реалізує дане розширення можливостей MS Excel. Для зв'язку MatLab і MS Excel визначені спеціальні функції.

Перед тим як налаштувати MS Excel на спільну роботу з MatLab, слід переконатися, що MS Excel Link входить до встановленої версії MatLab. У підкаталозі `exclink` основного каталогу MatLab або підкаталогу `toolbox` повинен знаходитися файл з надбудовою `exclink.xla`. Запустіть MS Excel і в меню `Tools` виберіть пункт `Add-ins`. Відкриється діалогове вікно, що містить інформацію про доступні в даний момент надбудови. Використовуючи кнопку `Browse`, вкажіть шлях до файлу `exclink.xla`. У списку надбудов діалогового вікна з'явиться рядок `MS Excel Link 2.0 for use with MatLab` з встановленим прапором. Натисніть `OK`, необхідна надбудова додана в MS Excel.

Панель інструментів MS Excel Link містить три кнопки: `putmatrix`, `getmatrix`, `evalstring`. Ці кнопки реалізують основні дії, необхідні для здійснення взаємозв'язків між MS Excel і MatLab – обмін матричними даними, і виконання команд MatLab з середовища MS Excel. При повторних запусках MS Excel надбудова `exclink.xla` підключається автоматично.

Узгоджена робота MS Excel і MatLab вимагає ще декількох установок, які прийняті в MS Excel за замовчуванням (але можуть бути змінені). У меню `Tools` перейдіть до пункту `Options`, при цьому відкривається діалогове вікно `Options`. Виберіть вкладку `General` і переконайтеся, що прапор `RIC1 reference style` вимкнений, тобто чарунки нумеруються `A1, A2` і т. д. На вкладці `Edit` повинен бути встановлений прапор `Move selection after Enter` [10].

Для здійснення такого обміну даними між MatLab і MS Excel запустіть MS Excel, перевірте, що виконані всі необхідні налаштування (MatLab повинен бути закритий). Введіть в чарунки з `A1` по `C3` матрицю, для виділення десяткових знаків використовують крапки відповідно до вимог MS Excel.

	A	B	C
1	5.5	1.6	-0.8
2	2.3	6.1	0.2
3	0.1	0.4	3.9

Рис.2.19 – Вікно MS Excel

Виділіть на аркуші дані поля і натисніть кнопку `putmatrix`, з'являється вікно MS Excel з попередженням про те, що MatLab не запущено. Натисніть `OK`, дочекайтеся відкриття MatLab.

З'являється діалогове вікно MS Excel з рядком введення, призначеним для визначення імені змінної робочого середовища MatLab, в яке слід

експортувати дані з виділених чарунок MS Excel. Введіть наприклад, M і закрийте вікно за допомогою кнопки ОК. Перейдіть до командного вікна MatLab і переконайтеся, що в робочому середовищі утворилась змінна M , що містить масив три на три:

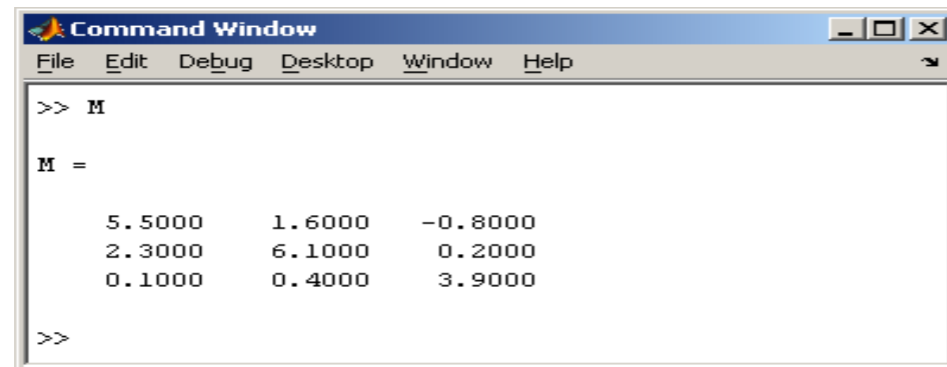


Рис.2.20 – Вікно МАТАВ

Виконайте деякі операції в МАТАВ з матрицею M , наприклад, знайдіть обернену до неї.

Виклик `inv` для знаходження оберненої матриці, як і будь-яку іншу команду MatLab можна здійснити прямо з MS Excel. Натискання на кнопку `evalstring`, розташовану на панелі Excel Link, призводить до появи діалогового вікна, в рядку введення якого слід набрати команду MatLab

$IM = \text{inv}(M)$.

Результат аналогічний отриманому при виконанні команди в середовищі MatLab.

Поверніться в MS Excel, зробіть поточною чарунку $A5$ і натисніть кнопку `getmatrix`. З'являється діалогове вікно з рядком введення, до якого потрібно ввести ім'я змінної, що імпортується в MS Excel. У даному випадку такою змінною є IM . Натисніть ОК, в чарунках з $A5$ по $A7$ будуть введені елементи оберненої матриці.

Отже, для експорту матриці в MatLab слід виділити відповідні чарунки аркуша MS Excel, а для імпорту досить вказати одну клітинку, яка буде верхнім лівим елементом імпортованого масиву. Інші елементи запишуться в чарунку листа відповідно до розмірів масиву, переписуючи в них відповідні дані, тому слід дотримуватися обережності при імпорті масивів [10].

Вищеописаний підхід є найпростішим способом обміну інформацією між додатками – вихідні дані містяться в MS Excel, потім експортуються в MatLab, обробляються там певним чином і результат імпортується в MS Excel. Користувач переносить дані за допомогою кнопок панелі інструментів MS Excel Link. Інформація може бути представлена у вигляді матриці, тобто прямокутної області робочого листа. Чарунки, розташовані в рядку або стовпчику, експортуються, відповідно, в вектор-рядок і вектор-стовпець MatLab. Аналогічно відбувається і імпорт векторів-рядків і векторів-стовпців у MS Excel.

```
-->M=rand(4,4);
```

```
-->[L,U]=lu(M)
```

```
U =
    0.9329616    0.2922267    0.5935095    0.6325745
    0.          0.2652853    0.1364844    0.0607448
    0.          0.          0.0364000   -0.0832093
    0.          0.          0.          -0.0032653

L =
    0.9468651   -0.2340817    1.          0.
    0.6994001    0.4080837    0.3255362    1.
    0.3297125    1.          0.          0.
    1.          0.          0.          0.
```

QR - розкладання - це розкладання квадратної матриці на ортогональну (Q) і верхню трикутну (R) матриці $A = QR$. У Scilab це розкладання реалізується командою `qr(A)`.

Приклад.

```
-->M=rand(4,4);
```

```
-->[Q,R]=qr(M)
```

```
R =
- 1.1143899   -0.5042206   -0.7585314   -0.8388711
    0.          0.2874919    0.1047526    0.6775043
    0.          0.          0.7396454   -0.0912502
    0.          0.          0.          0.4599353

Q =
- 0.3636029    0.2804236    0.6396761   -0.6164170
- 0.8241916   -0.0026575   -0.5583673   -0.0944831
- 0.0392443    0.9073716   -0.0171462    0.4181419
- 0.4323899   -0.3131010    0.5279644    0.6605001
```

```
-->a=[1 2 3]; b=[2 4 1];
```

```
-->S=sum(a.*b)
```

```
S =
```

```
13.
```

```
-->V=[det([a(2:3);b(2:3)]) -det([a(1) a(3); b(1) b(3)]) det([a(1:2);b(1:2)])]
```

```
V =
```

```
- 10.    5.    0.
```

```
-->cosa=S/(norm(a,2)*norm(b,2))
```

```
cosa =
```

```
0.7581754
```

2.4.28. Матричні розкладання

Рішення ряду завдань лінійної алгебри вимагають подання матриці в тому чи іншому вигляді. Перерахування цих завдань і огляд всіх типів розкладів не входить в наші плани, розглянемо тільки два основних, які часто використовуються.

Трикутним розкладанням квадратної матриці M , при якому остання представляється у вигляді $M = LU$, де U - верхня трикутна матриця, а L може бути приведена до нижньої трикутної простою перестановкою рядків, причому на головній діагоналі при цьому будуть стояти одиниці, називається LU-розкладанням і здійснюється за допомогою команди `lu (M)`.

Приклад.

2.2.3. Програмування в середовищі MatLab

В MatLab особливе значення мають файли двох типів – з розширеннями `.mat` і `.m`. Перші є бінарними файлами, в яких можуть зберігатися значення змінних, другі являють собою текстові файли, що містять зовнішні програми, визначення команд і функцій системи. Саме до них належить більшість команд і функцій, в тому числі ті, що задаються користувачем для розв’язання своїх специфічних завдань.

Багатовіконний редактор-налаштовувач з порожнім вікном редагування `m`-файлів можна викликати командою `Edit` з командного рядка або командою меню `File> New> M-file` (рис. 2.21) [45].

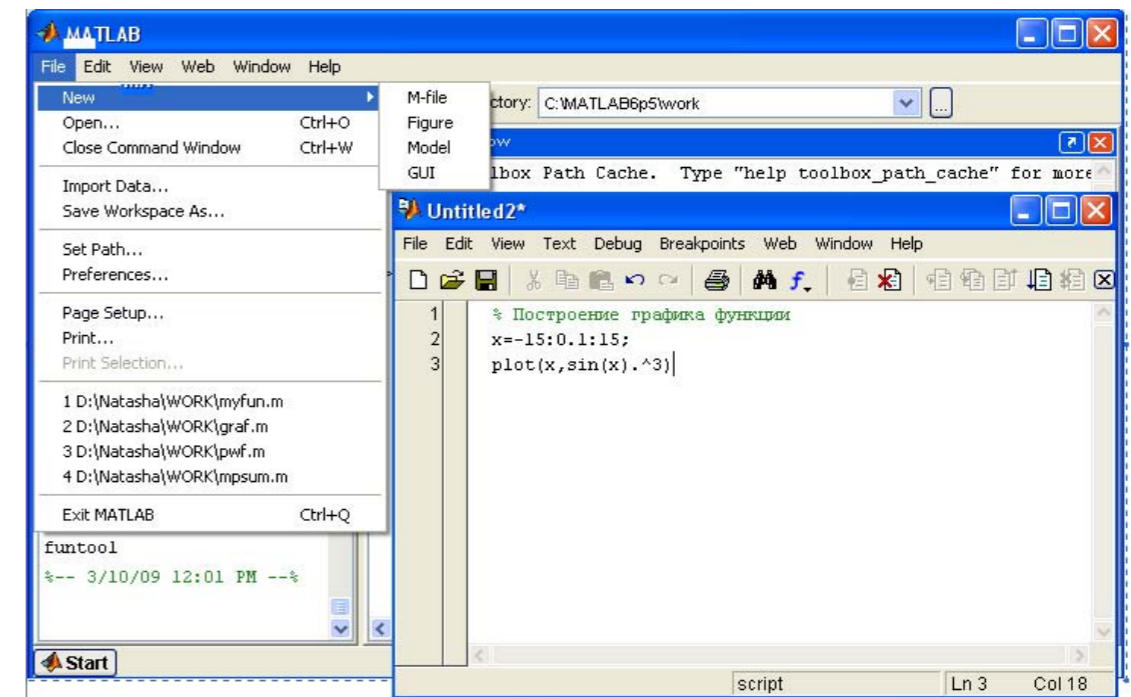


Рис.2.21 – Багатовіконний редактор-налаштовувач

Після цього у вікні редактора можна створити свій файл, а також користуватися засобами його налагодження і запуску. Для запуску файлу його необхідно записати на диск, використовуючи команду `Save as` в меню `File` редактора. Редактор-налаштовувач `m`-файлів виконує синтаксичну перевірку програмного коду зі зворотним відліком. При цьому використовується наступне кольорове виділення:

- ключові слова мови програмування – синій колір;
- оператори, константи і змінні – чорний колір;
- коментарі після знака `%` – зелений колір;
- символічні змінні (в апострофах) – коричневий колір;
- синтаксичні помилки – червоний колір.

Завдяки кольоровому виділенню ймовірність синтаксичних помилок різко знижується.

`M`-файли, створювані редактором-налаштовувачем, діляться на два класи: файли-сценарії, які не мають вхідних параметрів і файли-функції, що мають

вхідні параметри. Файл-сценарій (*script*-файл), є просто записом серії команд без вхідних і вихідних параметрів. Він має наступну структуру:

```
% Основний коментар
% Додатковий коментар
Тіло файлу з будь-якими виразами
```

Важливими є такі властивості файлів-сценаріїв:

- 1) вони не мають вхідних і вихідних аргументів;
- 2) працюють з даними з робочою областю;
- 3) у процесі виконання не компілюються;

4) представляють собою зафіксовану у вигляді файлу послідовність операцій, повністю аналогічну тій, що використовується в сесії.

Розглянемо наступний файл-сценарій (рис. 2.22):

Перші три рядки тут – це коментар, інші – тіло файлу. Зверніть увагу на можливість задання коментаря російською мовою. Знак% в коментарях повинен починатися з першої позиції рядка. Необхідно відзначити, що такий файл можна запустити без попередньої підготовки, що зводиться до завдання значень змінних *xmin* і *xmax*, використаним в тілі файлу. Це наслідок першого властивості файлів-сценаріїв – вони працюють з даними з робочої області [45].

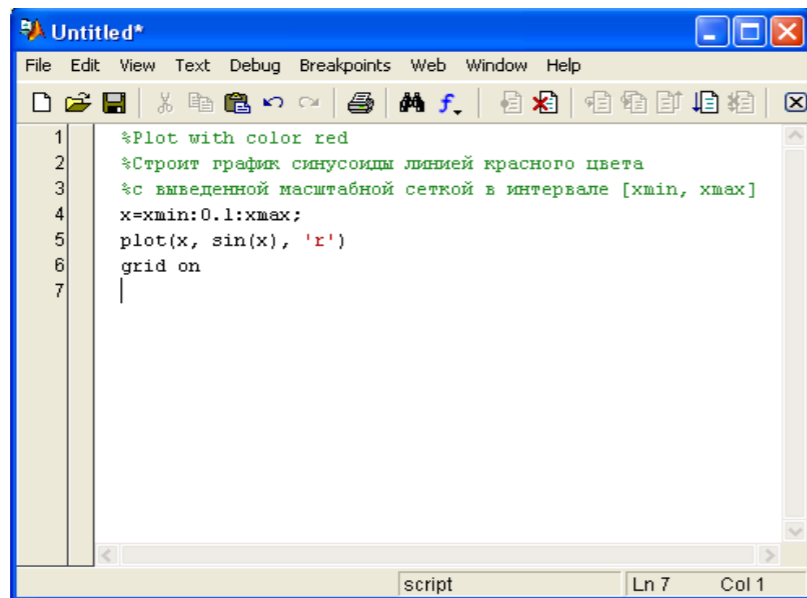


Рис. 2.22 – Створення файлу-сценарію в MatLab

Імена файлів-сценаріїв можна використовувати в якості параметрів функцій, оскільки файли-сценарії не повертають значень. Можна сказати, що файл-сценарій – це найпростіша програма на мові програмування MatLab.

M-файл-функція є типовим об'єктом мови програмування системи MatLab. Одночасно він є повноцінним модулем з точки зору структурного програмування, оскільки містить вхідні та вихідні параметри і використовує апарат локальних змінних. Структура такого модуля з одним вихідним параметром виглядає наступним чином:

```
function var=f_name (Перелік _ параметрів)
%Основний коментар
%Додатковий коментар
```

```
-->v=[1 1 2]
```

```
v =
```

```
1. 1. 2.
```

```
-->l=norm(v,2)
```

```
l =
```

```
2.4494897
```

```
-->[v(1)/l, v(2)/l, v(3)/l]
```

```
ans =
```

```
0.4082483 0.4082483 0.8164966
```

Розглянемо два вектори

$$\vec{a}=(a_1, a_2, a_3) \text{ и } \vec{b}=(b_1, b_2, b_3)$$

Обчислимо їх скалярний, векторний добуток і кут між ними за формулами

$$\vec{a} \cdot \vec{b} = a_1 b_1 + a_2 b_2 + a_3 b_3$$

$$\vec{a} \times \vec{b} = \begin{bmatrix} \vec{i} & \vec{j} & \vec{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix} = \vec{i}(a_2 b_3 - a_3 b_2) + \vec{j}(a_3 b_1 - a_1 b_3) + \vec{k}(a_1 b_2 - a_2 b_1)$$

$$\cos(\widehat{\vec{a}, \vec{b}}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}| |\vec{b}|}$$

Приклад.

Приклад

```
-->vn=[1 2 3]; vk=[2 3 5]; v=vk-vn
```

```
v =
```

```
1. 1. 2.
```

```
-->l=sqrt(sum(v.^2))
```

```
l =
```

```
2.4494897
```

```
-->[v(1)/l, v(2)/l, v(3)/l]
```

```
ans =
```

```
0.4082483 0.4082483 0.8164966
```

У прикладі ми вираховували довжину вектора за наведеною формулою. У Scilab для цих цілей існує вбудована функція `norm(V, 2)`.

Приклад.

```
-->l=norm(v, 2)
```

```
l =
```

```
2.4494897
```

Як видно, результат виходить той же.

Чому ж прийнято вважати, що вектор визначається тільки трьома числами - координатами його кінця? Справа в тому, що основні характеристики вектора - довжина і напрям - не залежать від того, в якій саме точці він починається і не змінюються, якщо паралельним перенесенням перенести вектор в початок системи координат. А при цьому, дійсно, вектор визначається тільки координатами свого кінця оскільки координати його початку - нульові. Далі ми будемо мати справу саме з такими векторами. Попередній приклад спроститься.

Приклад.

Тіло файлу с будь-якими виразами

var=вираз

M-файл-функція має такі властивості:

1) вона починається з оголошення `function`, після якого вказується ім'я змінної `var` – вихідного параметра, ім'я самої функції `f_name` і список її вхідних параметрів;

2) функція повертає своє значення і може використовуватися в математичних виразах;

3) усі змінні, наявні в тілі файлу-функції, є локальними, тобто діють тільки в межах тіла функції;

4) файл-функція є самостійним програмним модулем, який пов'язаний з іншими модулями через свої вхідні та вихідні параметри;

5) правила виведення коментарів ті ж, що у файлів-сценаріїв;

6) при виявленні файлу-функції вона компілюється і потім виконується, а створені машинні коди зберігаються в робочій області системи MatLab.

Остання конструкція `var = вираз` вводиться, якщо потрібно, щоб функція повертала результат обчислень. Наведена форма файлу-функції характерна для функції з одним вихідним параметром. Якщо вихідних параметрів більше, то вони вказуються в квадратних дужках після слова `function`. При цьому структура модуля має наступний вигляд:

`function[var1,var2,...]= f_name (Перелік _ параметрів)`

%Основний коментар

%Додатковий коментар

Тіло файлу с будь-якими виразами

var1=вираз

var2=вираз

Якщо функція яка використовується має єдиний вхідний параметр, але має ряд вихідних параметрів, то для повернення значення буде використовуватися перший з них. Це часто приводить до помилок в математичних обчисленнях. Тому, як зазначалося, дана функція використовується як окремий елемент програм виду `[var1, var2] = f_name (Перелік _ параметрів)`. Після його застосування змінні виходу `var1, var2` визначені і їх можна використовувати в подальших математичних виразах і інших сегментах програми.

Для організації діалогового введення і виведення використовуються наступні оператори, представлені в таблиці 2.10.

Таблиця 2.10

Оператори діалогового введення / виведення

Оператор	Синтаксис	Призначення
INPUT	<code>x = input('<запрошення>')</code>	Для введення даних з клавіатури
DISP	<code>disp (<змінна або текст в апострофах>)</code>	Для виведення на дисплей

Наведемо простий приклад діалогової програми, яка служить для багаторазового обчислення довжини кола за введеним користувачем значенням радіуса r (рис.2.23).

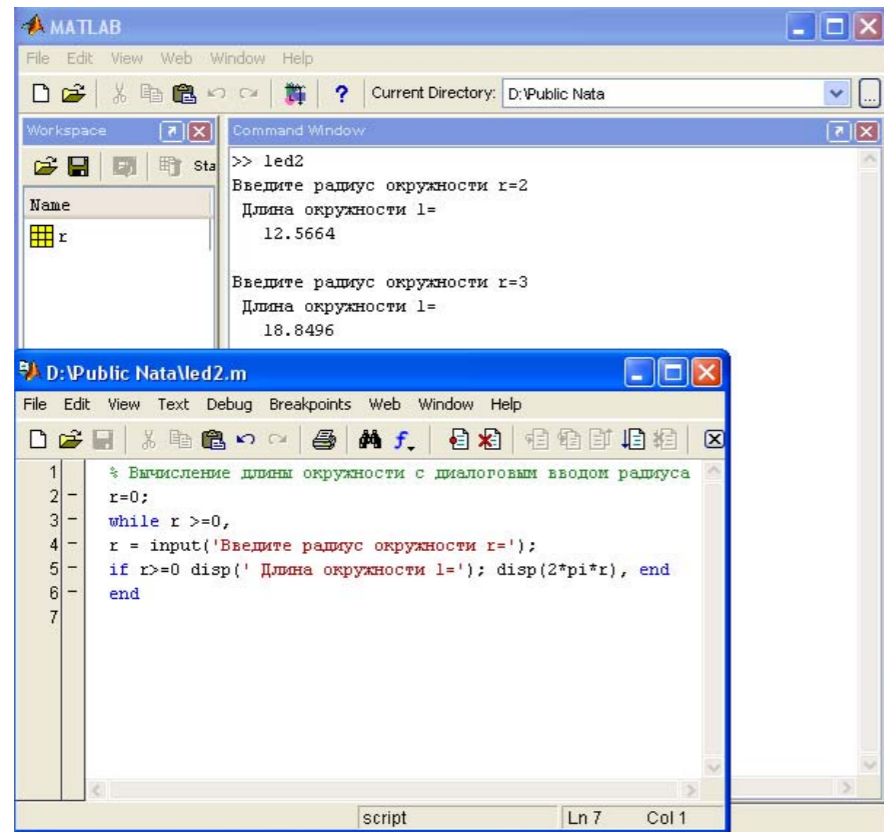


Рис. 2.23 – Приклад діалогової програми

Для організації розгалужень служать умовні оператори.

Конструкції умовних операторів:

1) **if** <умова>
<оператори>
end

Оператори (тіло виразу) виконуються тільки в тому випадку, якщо умова істинна, якщо умова помилкова, то тіло виразу не виконується.

2) **if** <умова>
<Оператори 1>
else
<Оператори 2>
end

Якщо хід програми повинен змінюватися в залежності від декількох умов, то слід використовувати повну конструкцію **if-elseif-else**. Кожна з гілок **elseif** в цьому випадку повинна містити умову виконання блоку операторів, розміщених після неї. Важливо розуміти, що умови перевіряються поступово, перша виконана умова призводить до роботи відповідного блоку, виходу з конструкції **if-elseif-else** і переходу до оператора, наступного за **end**. В останній гілці **else** не повинно бути жодної умови. Оператори, що знаходяться між **else** і **end**, працюють в тому випадку, якщо всі умови виявилися невиконаними.

\wedge - Піднесення до степеня. При цьому цілочисельний показник ступеня n може бути як позитивним, так і негативним. У першому випадку матриця множиться n раз сама на себе, у другому - подібна дія проводиться над матрицею, оберненою до початкової

Приклад.

-->A=[1 3 2; 7 -2 4; 9 6 2]

A =
1. 3. 2.
7. -2. 4.
9. 6. 2.

-->A^2
ans =

40. 9. 18.
29. 49. 14.
69. 27. 46.

-->A^(-2)
ans =

0.0751482 0.0028842 - 0.0302836
- 0.0147412 0.0239545 - 0.0015222
- 0.1040699 - 0.0183865 0.0680580

-->A_obr=inv(A);

-->A_obr^2
ans =

0.0751482 0.0028842 - 0.0302836
- 0.0147412 0.0239545 - 0.0015222
- 0.1040699 - 0.0183865 0.0680580

2.4.27. Лінійна і векторна алгебра

Робота з векторами

Розглянемо найбільш відомі та поширені завдання з курсу вищої математики, що стосуються роботи з векторами. Нагадаємо, що вектор - це спрямований відрізок і його положення в просторі визначається координатами його початку (x_1, y_1, z_1) і кінця (x_2, y_2, z_2) - шістьма числами. Обчислення основних характеристик вектора - його довжина і косинуси напрямних кутів - визначаються формулами:

$$l = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2},$$

$$\cos \alpha = \frac{x_2 - x_1}{l}, \quad \cos \beta = \frac{y_2 - y_1}{l}, \quad \cos \gamma = \frac{z_2 - z_1}{l}.$$


```

-->a
a =
    1.    2.    3.
-->sin(a)
ans =
    0.8414710    0.9092974    0.1411200
-->[sin(a(1)) sin(a(2)) sin(a(3))]
ans =
    0.8414710    0.9092974    0.1411200

```

2.4.26. Дії над матрицями

' - Транспонування, в результаті якого рядка матриці стають стовпцями і навпаки;

+ - Поповнення числа до всіх елементів матриці і матричне складання, певне для матриць однієї розмірності;

- - Віднімання числа від всіх елементів матриці і матричне віднімання, певне для матриць однієї розмірності;

* - Множення на число і матричне множення, яке виконується по правилу «рядок на стовпець» і допустимий, якщо кількість рядків у другій матриці збігається з кількістю стовпців у першому;

/ - Ділення на число.

Приклад.

```

-->A=[1 2 3; 3 7 4; 2 -6 1]
A =
    1.    2.    3.
    3.    7.    4.
    2.   -6.    1.

```

```

-->B=[3 7 1; 9 -2 0; 5 1 5];

```

```

-->A+1.5
ans =
    2.5    3.5    4.5
    4.5    8.5    5.5
    3.5   -4.5    2.5

```

```

-->C=B-1
C =
    2.    6.    0.
    8.   -3.   -1.
    4.    0.    4.

```

```

-->A*B
ans =
    36.    6.    16.
    92.   11.   23.
   -43.   27.    7.

```

```

-->C'
ans =
    2.    8.    4.
    6.   -3.    0.
    0.   -1.    4.

```

Приклад 3.1. Написати файл-функцію для обчислення кусочно-заданої функції:

$$f(x) = \begin{cases} 1 - e^{-1-x}, & x < -1; \\ x^2 - x - 2, & -1 \leq x \leq 2; \\ 2 - x, & x > 2. \end{cases}$$

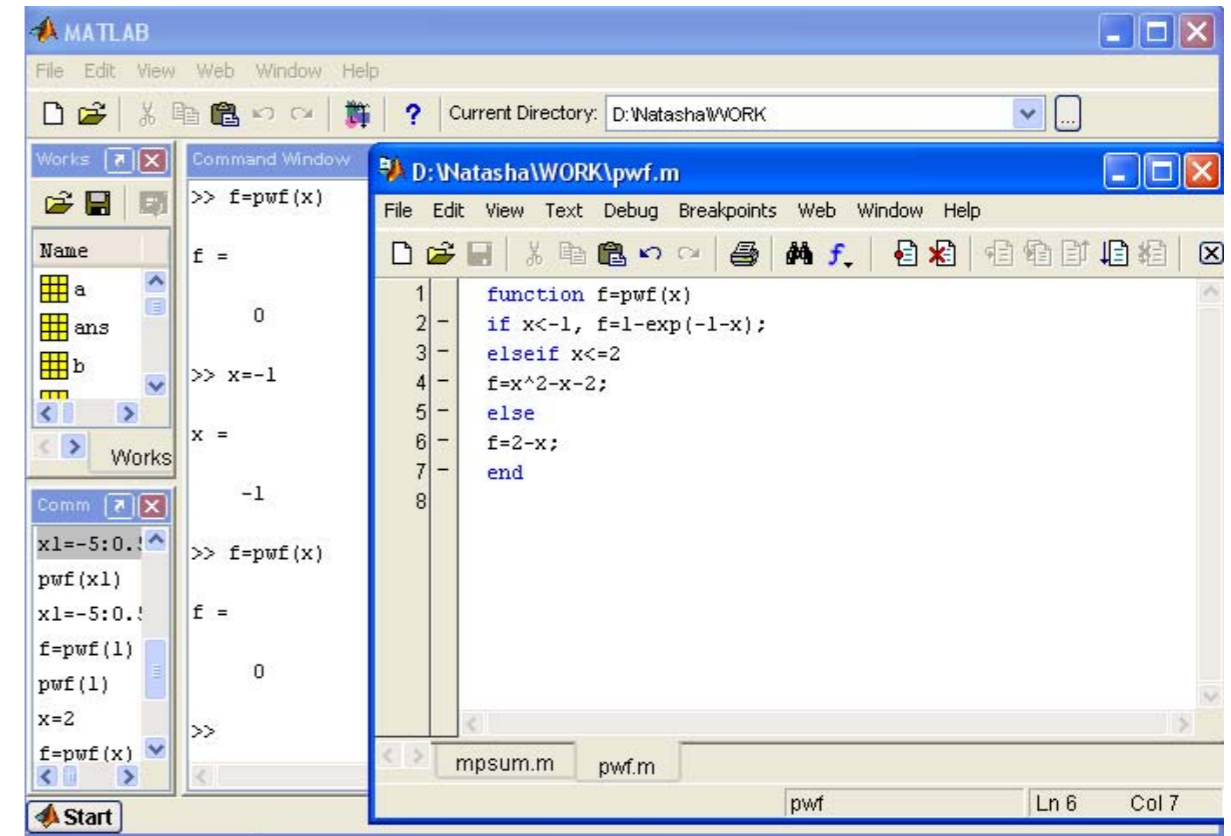


Рис. 2.24 – Лістинг програми для обчислення значення функції

В системі MatLab можуть застосовуватися наступні оператори порівняння, наведені в таблиці 2.11 [45].

Таблиця 2.11

Оператори порівняння		
Символ	Призначення	Ім'я функції
<	Менше	lt
>=	Більше або дорівнює	ge
>	Більше	gt
<=	Менше або дорівнює	le
==	Дорівнює	eq
~=	Не дорівнює	ne

Операції (==, ~=) проводять порівняння дійсних і уявних частин комплексних чисел, а операції (>, <,>=, <=) – тільки для дійсних частин.

Логічні операції можна записати у вигляді функцій (табл. 2.12) [45].

Таблиця 2.12

Логічні операції

Символ	Призначення	Ім'я функції
&	Логічне «і»	and
	Логічне «або»	or
~	Заперечення	not

Результатом логічних операцій є числа 0 (false) і 1 (true).

В системі MatLab є два різновиди операторів циклу – умовний і арифметичний. Для повторення операторів нефіксоване число раз використовується оператор циклу з передумовою:

```
while <умова>
<Оператори>
end
```

Оператори виконуються, якщо змінна <умова> має ненульові елементи.

Арифметичний оператор циклу має такий вигляд:

```
for <ім'я> = <ПЗ>: <Крок>: <КЗ>
<Оператори>
end,
```

де <ім'я> – ім'я керуючої змінної циклу,

<ПЗ> – початкове значення керуючої змінної,

<КЗ> – кінцеве значення керуючої змінної,

<Крок> – приріст значень змінної <ім'я> від значення <ПЗ> до значення <КЗ>. Якщо параметр <Крок> не вказано, за замовчуванням його значення приймається рівним одиниці.

При роботі з циклом for допустиме використання оператора переривання циклу break. При роботі даного оператора робота циклу завершується, і управління передається на наступний після кінця циклу оператор.

Хід роботи програми може визначатися значенням деякої змінної (перемикача). Такий альтернативний спосіб розгалуження програми заснований на використанні оператора перемикачання switch. Оператор switch містить блоки, що починаються зі слова case, після кожного case записується через пробіл те значення перемикача, при якому виконується даний блок. Останній блок починається зі слова otherwise, його оператори працюють в тому випадку, коли жоден з блоків case не був виконаний. Якщо хоча б один з блоків case виконаний, то відбувається вихід з оператора switch і перехід до оператора, наступного за end.

Приклад 3.2. Потрібно знайти кількість одиниць і мінус одиниць в заданому масиві і, крім того, знайти суму всіх елементів, відмінних від одиниці і мінус одиниці. Лістинг програми містить файл-функцію, яка по заданому масиву повертає число мінус одиниць в першому вихідному аргументі, число одиниць – у другому, а суму – в третьому (рис.2.25).

По-друге, поелементне множення означає, що кожен елемент одного рядка (стовпця) множиться на елемент з таким же номером іншого рядка (стовпчика). Природно, при цьому рядки (стовпці) повинні бути одного розміру. Причому, не допускається поелементне множення рядка на стовпець і навпаки. Те ж справедливо і для інших поелементних дій: ділення і піднесення до степеня.

По-третє, відзначимо дії, що принципово відрізняються від перерахованих тим, що змінюють тип масиву:

- транспонування ('), в результаті якого рядок стає стовпчиком і навпаки;
- множення рядка на стовпець тієї ж довжини, при якому виходить просто число;

- множення стовпця на рядок, що дає матрицю.

Приклад.

```
-->a=[1 2 3]; b=[4; 5; 6];
```

```
-->a*3
ans =
     3     6     9
```

```
-->a.*b
!--error 9999
inconsistent element-wise operation
```

```
-->a.*b'
ans =
     4    10    18
```

```
-->a*b
ans =
    32
```

```
-->b*a
ans =
     4     8    12
     5    10    15
     6    12    18
```

Якщо до деякого заданому вектору застосувати математичну функцію, то результатом буде новий вектор тієї ж розмірності з елементами, перетвореними у відповідності із заданою функцією.

Приклад.

Приклад. Формування матриці з матриць

```

-->U=[1 3; 2 4]
U =
    1.    3.
    2.    4.

-->U1=[U U]
U1 =
    1.    3.    1.    3.
    2.    4.    2.    4.

-->U2=[U; U]
U2 =
    1.    3.
    2.    4.
    1.    3.
    2.    4.

-->U3=[U U; U U]
U3 =
    1.    3.    1.    3.
    2.    4.    2.    4.
    1.    3.    1.    3.
    2.    4.    2.    4.

-->U4=[2*U U; U-1 U+2]
U4 =
    2.    6.    1.    3.
    4.    8.    2.    4.
    0.    2.    3.    5.
    1.    3.    4.    6.

```

2.4.25. Дії над векторами

При роботі з рядками і стовпцями в Scilab припустимі наступні математичні дії:

- + - Додавання
- Віднімання
- * - Множення на число
- / - Ділення на число
- ^ - Піднесення до степеня
- . * - Поелементне множення
- ./ - Поелементне ділення
- . ^ - Поелементне зведення в ступінь

Зробимо деякі зауваження.

По-перше, операції додавання / віднімання справедливі тільки для стовпців або тільки для рядків однакової довжини. У разі складання / віднімання вектора з числом дію буде вироблено з кожним елементом масиву.

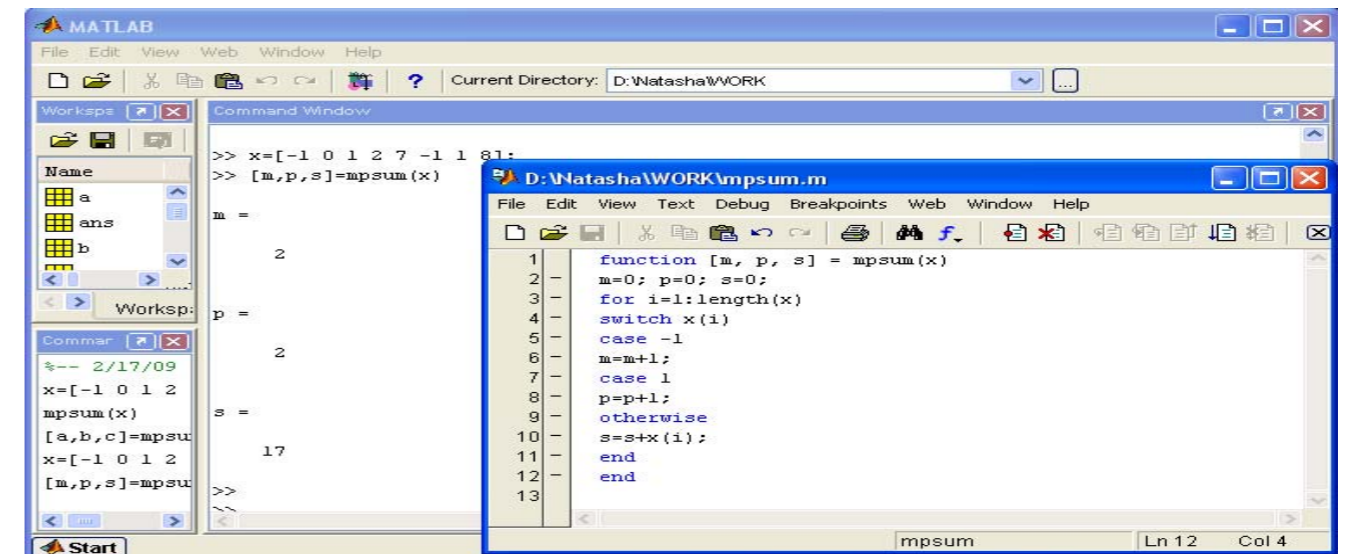


Рис. 2.25 – Лістинг програми

Для зупинки програми використовується оператор pause. Він використовується в наступних формах:

- a) pause – зупиняє обчислення до натискання будь-якої клавіші;
- b) pause (N) – зупиняє обчислення на N секунд;
- c) pause on – включає режим відпрацювання пауз;
- d) pause off – вимикає режим відпрацювання пауз.

2.2.4. Робота з графікою засобами MatLab

Одна з переваг системи MatLab – велика кількість засобів графіки, починаючи від команд побудови простих графіків функцій однієї змінної в Декартовій системі координат і закінчуючи комбінованими і презентаційними графіками з елементами анімації, а також засобами проектування графічного інтерфейсу користувача (GUI). Особлива увага в системі приділена тривимірній графіці з функціональним розфарбуванням відображених фігур і імітацією різних світлових ефектів.

Для відображення функцій однієї змінної $y(x)$ використовуються графіки в Декартовій (прямокутній) системі координат. При цьому зазвичай будуються дві осі – горизонтальна X і вертикальна Y , і задаються координати x і y , що визначають вузлові точки функції $y(x)$. Оскільки MatLab – матрична система, сукупність точок $y(x)$ задається векторами X і Y однакового розміру.

Команда $\text{plot}(X, Y)$ служить для побудови графіків функцій у Декартовій системі координат, координати точок (x, y) беруться з векторів однакового розміру Y і X . Якщо X або Y – матриця, то будується сімейство графіків за даними, що містяться в колонках матриці.

Команда $\text{plot}(X, Y, S)$ аналогічна команді $\text{plot}(X, Y)$, але тип лінії графіка можна задавати за допомогою константи рядка S .

Значеннями константи S можуть бути наступні символи, які представлені в таблиці 2.13 [12].

Таблиця 2.13

Задання типу лінії	
Маркер типу лінії	
Маркер	Тип лінії
-	Неперервна
--	Штрихова
:	Пунктирна (крапкова)
-.	Штрих-пунктирна
Маркер коліру графика	
Маркер	Колір графика
c	Блакитний
m	Фіолетовий
y	Жовтий
R	Червоний
g	Зелений
b	Синій
w	Білий
k	Чорний
Тип виділеної крапки	
Маркер	Тип крапки
.	Крапка
+	Плюс
*	Зірочка
o	Кружечок (вказують латинську букву o)
x	Хрестик (вказують латинську букву x)

Таким чином, за допомогою константи рядка S можна змінювати колір лінії, представляти вузлові точки різними відмітками (точка, коло, хрест, трикутник з різною орієнтацією вершини і т. д.) і змінювати тип лінії графіка.

Приклад 4.1. За допомогою пакету MatLab знайти наближений розв'язок задачі одновимірної мінімізації функції $f(x) = 3x^4 + 5x^3 - 10x^2 + 6x$ на відрізку $[-4; 2]$.

На рис. 2.26 подано графічну інтерпретацію розв'язку поставленої задачі, одержану в результаті виконання наступних команд:

```
>> x = -4:0.1:2;
>> y = 3*x.^4 + 5*x.^3 - 10*x.^2 + 6*x
>> plot(x,y) >> grid on.
```

```
--> b = A(:,3)
b =
    4.
    2.
    7.
    3.
```

• виділимо 2-й рядок

```
--> c = A(2,:)
c =
    5.    9.    2.    6.    3.
```

• виділимо перші три елементи 4-го рядка

```
--> d = A(4,1:3)
d =
    2.    6.    3.
```

• виділимо 2-й і 3-й елементи 5-го стовпчика

```
--> f = A(2:3,5)
f =
    3.
    5.
```

• виділимо підматрицю 3×2 елементів матриці A , розташованих на перетині 1-й, 2-й, 3-й рядки і 3-го, 4-го стовпця

```
--> G = A(1:3,3:4)
G =
    4.    3.
    2.    6.
    7.    3.
```

Інший спосіб формування матриць - це використання раніше заданих матриць і векторів.

Приклад. Формування матриці з рядків

```
--> a1 = [7 9 3]; a2 = [1 3 5]; a3 = [-4 5 8];
--> a = [a1 a2 a3]
a =
    7.    9.    3.    1.    3.    5.   -4.    5.    8.
--> A = [a1; a2; a3]
A =
    7.    9.    3.
    1.    3.    5.
   -4.    5.    8.
```

```
-->x(1)+y(2)-z(3)*z1(3)
ans =
    3.04
-->sin(z2(1))
ans =
    0.2955202
-->exp(x(4))-3*z(2)
ans =
    3.9496475
```

2.4.24. Формування матриці

Введення двовимірного масиву (матриці) здійснюється через підрядник. Елементи рядка відокремлюються між собою пробілами або комами, а рядки - крапкою з комою:

$A = [a_{11}, a_{12}, \dots, a_{1n}; a_{21}, a_{22}, \dots, a_{2n}; \dots; a_{n1}, a_{n2}, \dots, a_{nn}]$

Обов'язкова вимога: кількість елементів в кожному рядку повинно бути однаковим.

Звернутися до елементу матриці можна, вказавши в круглих дужках через кому номера рядка і стовпчика, на перетині яких знаходиться елемент:

$A(\text{індекс1}, \text{індекс2})$

Приклад. Введення матриці A і обчислення з її елементами

```
-->A=[11 7 9; 2 8 17; -4 5 6]
A =
```

```
    11.    7.    9.
     2.    8.   17.
    - 4.    5.    6.
```

```
-->A(1,2)^2+A(2,2)/A(3,1)
ans =
    47.
```

У ряді випадків потрібно отримати доступ не до одного елементу, а до групи. Це робиться за допомогою двокрапки.

Приклад. виділення під матриць:

- задаємо матрицю A

```
-->A=[1 2 4 3 7; 5 9 2 6 3; 1 9 7 3 5; 2 6 3 5 1]
A =
```

```
    1.    2.    4.    3.    7.
     5.    9.    2.    6.    3.
     1.    9.    7.    3.    5.
     2.    6.    3.    5.    1.
```

- виділимо її 3-й стовпець

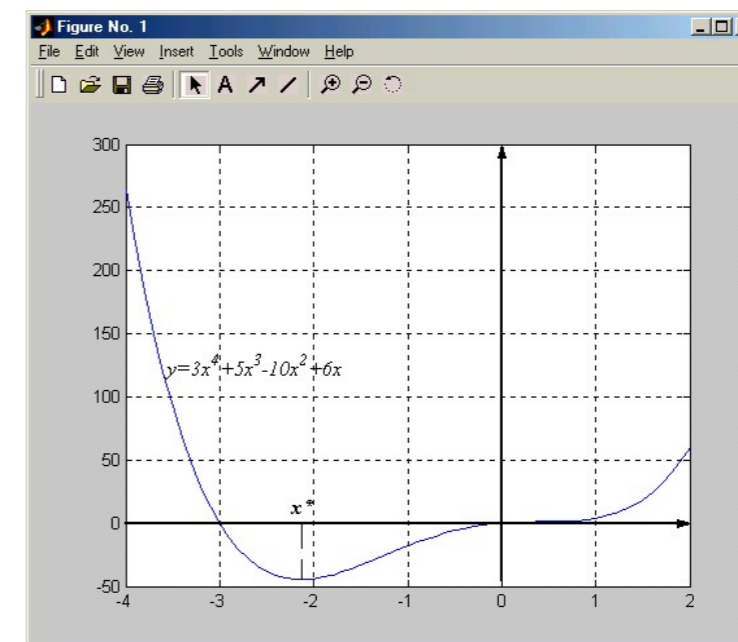


Рис.2.26 – Графік функції $f(x) = 3x^4 + 5x^3 - 10x^2 + 6x$

Наведений приклад демонструє широкі можливості для дослідницької роботи студентів на лабораторних заняттях з використанням пакету MatLab.

Команда `plot(X1, Y1, S1, X2, Y2, S2, X3, Y3, S3, ...)` будує на одному графіку ряд ліній, представлених даними виду (X, Y, S) , де X і Y – вектори або матриці, а S – рядки. За допомогою такої конструкції можлива побудова, наприклад, графіка функції лінією, колір якої відрізняється від кольору вузлових точок. При відсутності вказівки на колір ліній і точок він буде вибраний автоматично з таблиці кольорів (білий виключається). Якщо ліній більше шести, то вибір кольорів повторюється [12].

Іноді потрібно порівняти поведінку двох функцій, значення яких сильно відрізняються одне від одного. Графік функції з невеликими значеннями практично зливається з віссю абсцис, і встановити його вид не вдається. У цій ситуації допомагає функція `plotyy`, яка виводить графіки у вікно з двома вертикальними осями, що мають відповідний масштаб.

Тривимірні поверхні зазвичай описуються функцією двох змінних $z(x, y)$. Специфіка побудови тривимірних графіків вимагає не просто задання ряду значень x і y , тобто векторів x і y . Вона вимагає визначення для X і Y двовимірних масивів – матриць.

Для створення таких масивів служить функція `meshgrid`. В основному вона використовується спільно з функціями побудови графіків тривимірних поверхонь. Функція `meshgrid` записується в наступних формах:

$[X, Y, Z] = \text{meshgrid}(x, y, z)$ – повертає тривимірні масиви, які використані для обчислення функцій трьох змінних і побудови тривимірних графіків;

$[X, Y] = \text{meshgrid}(x, y)$ – перетворює область, задану векторами x і y , в масиви X і Y , які можуть бути використані для обчислення функції двох змінних і побудови тривимірних графіків. Рядки вихідного масиву X є копіями вектору x , а стовпці Y – копіями вектору y .

Команда `plot3 (...)` є аналогом команди `plot(...)`, але вона відноситься до функції двох змінних $z(x,y)$. Вона будує аксонометричне зображення тривимірних поверхонь і представлена наступними формами:

`plot3 (x, y, z)` – будує масив точок, представлених векторами x , y і z , з'єднуючи їх відрізками прямих. Ця команда має обмежене застосування;

`plot3 (X, Y, Z, S)` – забезпечує побудови зі специфікацією стилю ліній і точок;

`plot3(x1, y1, z1, s1, x2, y2, z2, s2, ...)` – будує на одному рисунку графіки декількох функцій $z1(x1, y1)$, $z2(x2, y2)$ і т. д. з специфікацією ліній і маркерів кожної з них [12].

Найбільш представницькими і наочними є сітчасті графіки поверхонь із заданим або функціональним розфарбуванням. У назві їх команд присутнє слово `mesh`. Є три групи таких команд:

`mesh(X, Y, Z, C)` – виводить в графічне вікно сітчасту поверхню $Z(X, Y)$ з кольорами вузлів поверхні, заданими масивом C ;

`mesh(X, Y, Z)` – аналог попередньої команди при $C=Z$.

В даному випадку використовується функціональне розфарбування, при якому колір задається висотою поверхні. Функція `mesh` повертає дескриптор для об'єкта класу `surface`. Нижче наводиться приклад застосування команди `mesh` (рис.2.27):

```
>> [X, Y] = meshgrid([-3:0.15:3]);
>> Z = X.^2 + Y.^2;
>> mesh(X, Y, Z)
```

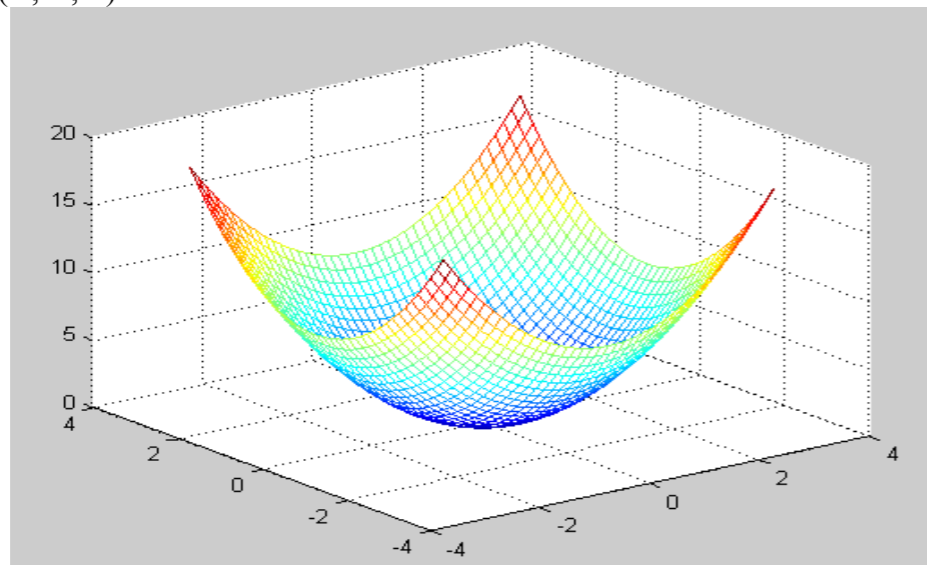


Рис. 2.27 – Графік поверхні, створений командою `mesh(X, Y, Z)`

Після того як графік уже побудований, MatLab дозволяє виконати його форматування або оформлення в потрібному вигляді. Так, для установки титульного напису використовується наступна команда `title('string')` – установка на двовимірних і тривимірних графіках титульного напису, заданого константою рядка `'string'`.

Для установки написів біля осей x , y і z використовуються наступні команди: `xlabel('String')`, `ylabel('String')` (рис.2.28).

чином, що перший елемент масиву X (1) дорівнює X_n , другий $X(2) = X(1) + dX = X_n + dX$, третій $X(3) = X(2) + dX = X_n + 2dX$ і так далі.

Таким способом ми отримуємо масив у вигляді рядка, в чому легко переконатися, вивівши його значення на екран. В цьому випадку дужки необов'язкові, тому ми їх і не використовували раніше (для завдання інтервалу аргументу при побудові графіків). Однак нехтування ними в іншій ситуації призводить до помилки. Для формування масиву використовуються квадратні дужки. Якщо ж в подібній ситуації потрібно представити масив у вигляді стовпчика, використовують операцію транспонування:

$X = [X_n: dX: X_k]'$

Зауважимо, що якщо параметр dX (крок) відсутня в запису масиву

$X = [X_n: X_k]$

його значення за замовчуванням приймається рівним 1.

Ще один спосіб завдання масиву в Scilab - поелементне введення. Елементи рядка відокремлюються один від одного пробілами або комами, при введенні стовпчика крапку з комою.

Приклад. Введення рядків і стовпців:

```
-->x=[1.2:0.2:2.4]
x =
    1.2    1.4    1.6    1.8    2.0    2.2    2.4

-->y=1:4
y =
    1.    2.    3.    4.

-->z1=[0.3 0.7 -0.4]
z1 =
    0.3    0.7   -0.4

-->z2=[0.3,0.7,-0.4]
z2 =
    0.3    0.7   -0.4

-->z=[0.3;0.7;-0.4]
z =
    0.3
    0.7
   -0.4
```

Звернутися до елементу масиву можна, вказавши ім'я масиву і порядковий номер елемента (індекс) в круглих дужках:

ім'я (індекс)

Таким чином елементи масиву можуть бути використані в математичних виразах:

```

-->x0=[0;0];

-->[RBmin,Xmin]=optim(fn,x0)
Xmin =

    0.9999955
    0.9999910
RBmin =

    2.0090-11

```

Таким чином, функція Розенброка має мінімум в точці (1; 1), рівний 0. Результат відповідає теоретичному.

2.4.22. Масиви і матриці

Коли доводиться працювати з великою кількістю однотипних даних, зручно їх пронумерувати і об'єднати під одним ім'ям. Така сукупність називається масивом.

Наприклад, як зберігати в пам'яті комп'ютера зростання кількості всіх студентів потоку? Можна кожному з них присвоїти ім'я і зберігати як набір змінних: зростання Іванова - H_iv, Петрова - H_pt, Сидорова - H_sd і так далі. Однак в цьому випадку дуже незручно працювати з цими даними.

Правильний підхід - об'єднати всі дані в масив із загальним ім'ям, скажімо, H, в якому першим елементом H (1) буде зростання Іванова, другим H (2) - Петрова і так далі. Таким чином, для доступу до даних масиву, що зберігаються в певному елементі, досить вказати ім'я масиву та порядковий номер (індекс) цього елемента.

Всі математичні пакети, в тому числі, звичайно, і Scilab, мають у своєму розпорядженні потужний арсенал засобів для обробки масивів і таке завдання, як згадане вище знаходження середнього стає елементарним.

Масиви можуть бути одновимірними (вектор-стовпці і вектор-рядка), двовимірними (матриці), тривимірними (тензори третього рангу) і так далі.

2.4.23. Формування одновимірних масивів

Строго кажучи, ми вже працювали з масивами в попередніх розділах. Справа в тому, що Scilab до будь-якого об'єкту відноситься як до масиву. Наприклад, число для Scilab - це масив, що складається з одного елемента.

Однак, нам вже довелося оперувати і справжніми одновимірними масивами - при побудові графіків. Коли ми ставили проміжок побудови, ми тим самим визначали рядок (одновимірний масив) аргументу. Отже, одновимірний масив може бути заданий наступним чином:

```
X = [Xn: dX: Xk]
```

Тут X - ім'я масиву, Xn, Xk - початковий і кінцевий його значення, dX - крок для формування кожного наступного елемента масиву. Виходить, таким

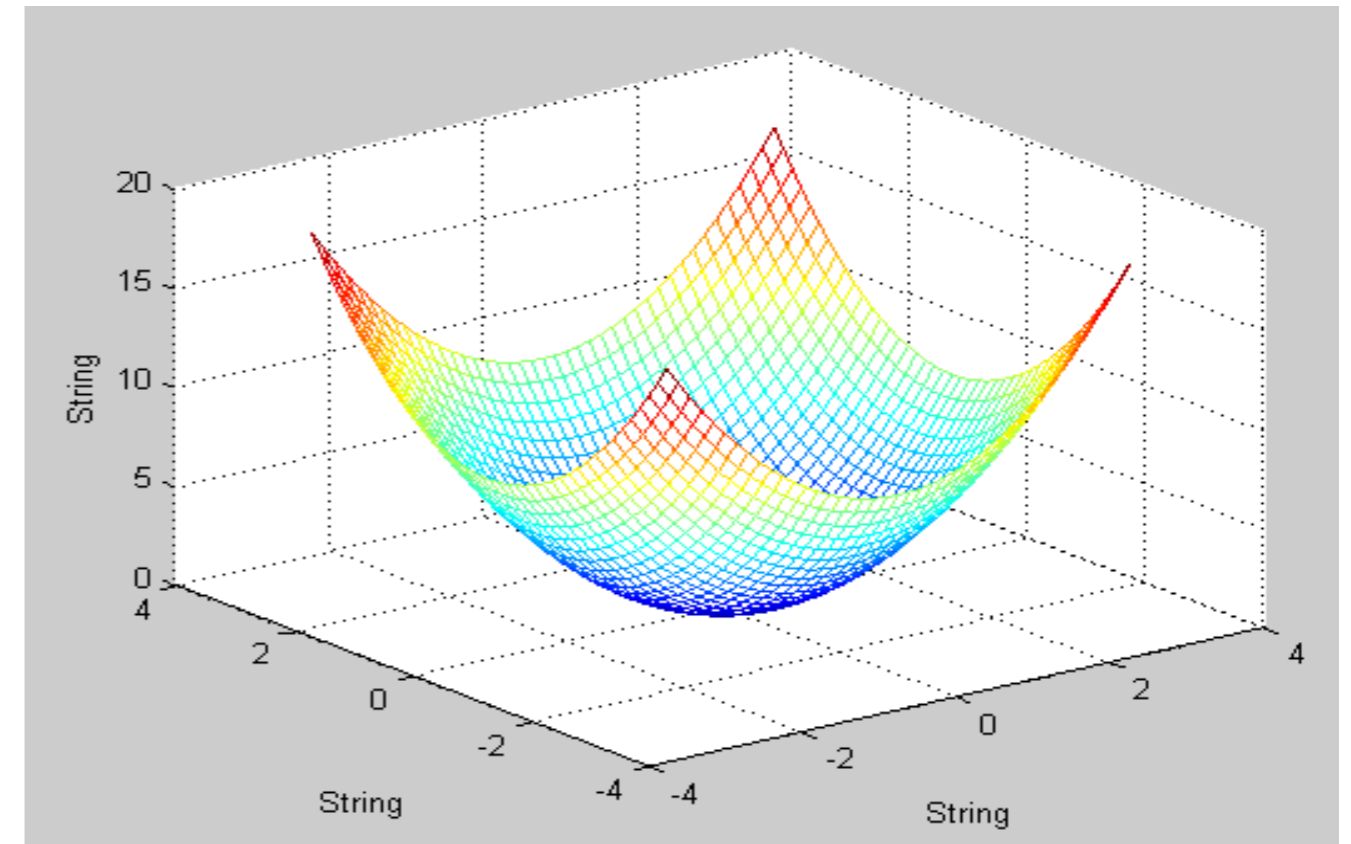


Рис. 2.28 – Установка написів з використанням команд: xlabel('String'), ylabel('String'), zlabel ('String')

Часто виникає необхідність додавання тексту в певне місце графіка, наприклад для позначення тієї чи іншої кривої графіка. Для цього використовується команда text [23]:

text(X,Y, 'string') – додає в двовимірний графік текст, заданий константою рядка 'string', так що початок тексту розташований в точці з координатами (X,Y). Якщо X і Y задані як одновимірні масиви, то напис поміщається в усі позиції [x (i), y (i)];

text (X,Y,Z, 'string') – додає в тривимірний графік текст, заданий константою рядка 'string', так що початок тексту розташований в позиції, заданій координатами X, Y і Z.

Дуже зручний спосіб введення тексту надає команда gtext:

gtext('string') – задає виведений на графік текст у вигляді константи рядка 'string' і виводить на графік переміщуваний мишею маркер у вигляді хрестика. Установивши маркер в потрібне місце, досить клацнути клавішею миші для виведення тексту.

Пояснення у вигляді відрізків ліній з довідковими написами, що розміщується всередині графіка або біля нього, називається легендою. Для створення легенди використовуються різні варіанти команди legend (рис.2.29):

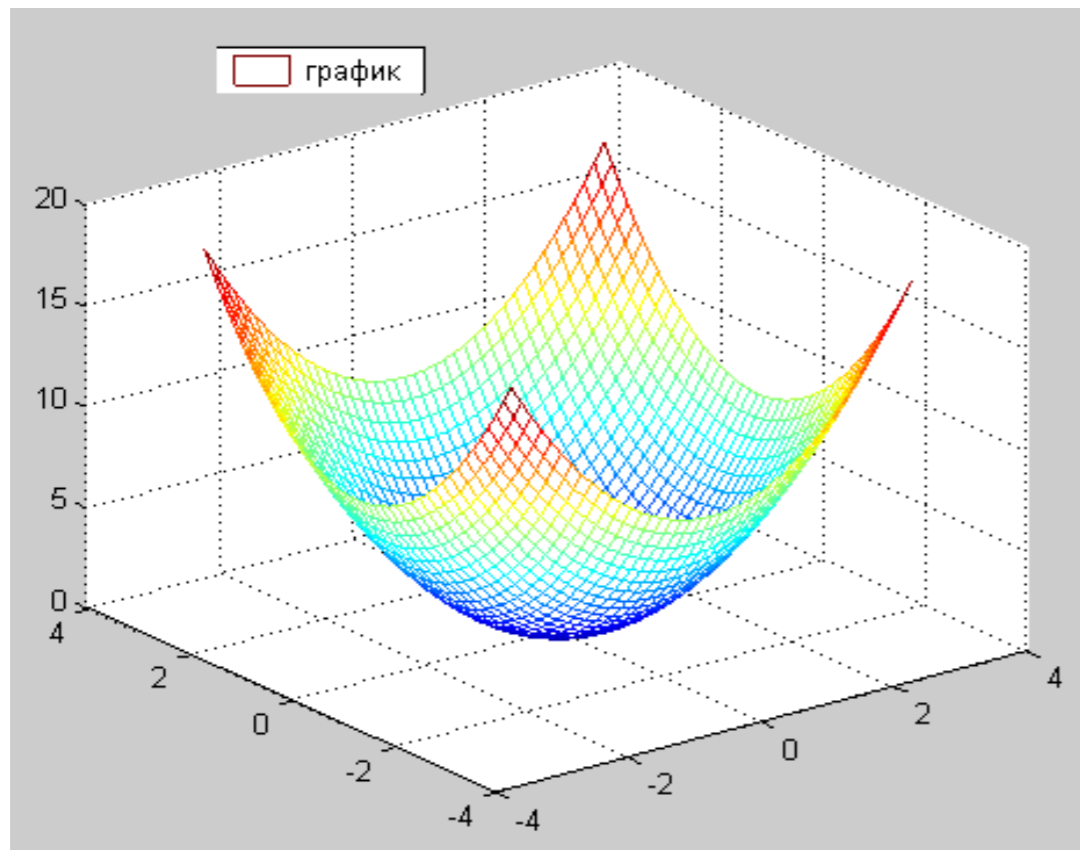


Рис. 2.29 – Графік з поясненнями

`legend(string1, string2, ..., strings)` – додає до поточного графіку легенду у вигляді рядків, зазначених у списку параметрів;

`>> legend ('график')`

`legend (Pos)` – розміщує легенду в точно зазначене місце, специфіцироване параметром Pos:

Pos=0 – краще місце, обирається автоматично;

Pos=1 – верхній правий кут;

Pos=2 – верхній лівий кут;

Pos=3 – нижній лівий кут;

Pos=4 – нижній правий кут;

Pos=-1 – праворуч від графіка [53].

При додаванні легенди слід врахувати, що порядок і кількість аргументів команди `legend` повинні відповідати порядку виведення графіків і їх кількості.

Зазвичай графіки виводяться в режимі автоматичного масштабування.

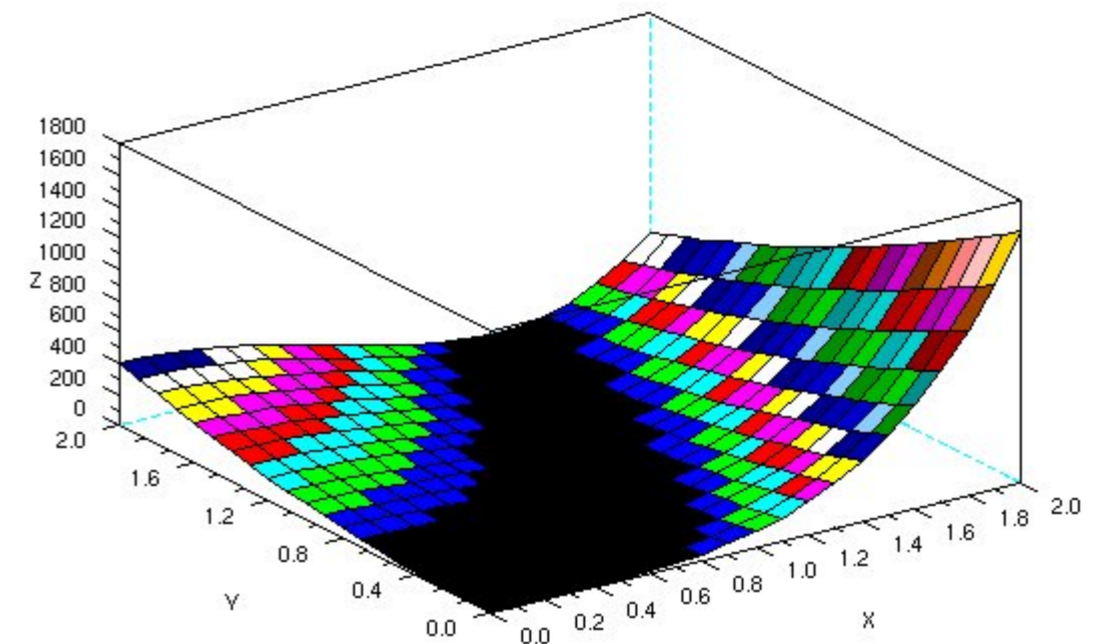
Наступні команди класу `axis` змінюють цю ситуацію:

`axis([XMIN XMAX YMIN YMAX])` – установка діапазонів координат по осях x та y для поточного двовимірного графіка;

`axis([XMIN XMAX YMIN YMAX ZMIN ZMAX])` – установка діапазонів координат по осях x , y і z поточного тривимірного графіка;

`axis auto` – установка параметрів осей за замовчуванням.

У математичній, фізичній та іншій літературі при побудові графіків у доповнення до розмітки осей часто використовують масштабну сітку. Команди `grid` дозволяють задавати побудову сітки або скасовувати цю побудову:



Це не функція одного аргументу, і графічно знайти мінімум у неї набагато складніше, тим не менш, за забарвленням видно, що в обраній галузі є сенс пошукати мінімум. Тому задамося початковим наближенням $(0, 0)$, оскільки в околиці цієї точки найбільше чорного кольору.

2.4.21. Пошук мінімуму

Початкове наближення вибрано, переходимо до безпосереднього визначення точки мінімуму. Для зручності програмування перепишемо функцію Розенброка «в термінах» Scilab.

$$f = 100(x(2) - x(1))^2 + (1 - x(1))^2$$

Так як функція немає від однієї змінної, а від вектора, простіше поставити її за допомогою конструкції `function`:

```
-->function y=RB(x)
-->y=100*(x(2)-x(1))^2+(1-x(1))^2;
-->endfunction
```

Так як тут використовуються вже числа, а не стовпці значень, можна використовувати звичайний оператор піднесення до степеня.

Задамо функцію `fn` в точності так само, як ми це робили для випадку функції однієї змінної.

```
-->function [RBmain, RBdif, id]=fn(x, id)
-->RBmain=RB(x);
-->RBdif=numdiff(RB, x);
-->endfunction
```

Задаємо початкове наближення і знаходимо точку мінімуму:

Таким чином, точки мінімуму в нашому випадку: (-3,8487884; -95,889413) і (1,8068586; 1,0725284).

Підкреслимо, основною проблемою є правильний вибір початкового наближення. Але це проблема не пакета SciLab, а математична.

Зауважимо, що при завданні fun функції Fmain і Fdif можна було ставити явно, проте при цьому вам доведеться самостійно обчислювати похідну. Інша справа, що в даному випадку це не складає труднощів.

```
-->function [fmain,fdif,ind]=fun1(x,ind)
-->fmain=x^4+3*x^3-13*x^2-6*x+26;
-->fdif=4*x^3+9*x^2-26*x-6;
-->endfunction
```

Результат, звичайно, буде той же:

```
-->[fm1,xm1]=optim(fun1,-4)
xm1 =
- 3.8407084
fm1 =
- 95.089413
-->[fm2,xm2]=optim(fun1,2)
xm2 =
1.8068586
fm2 =
1.0725284
```

2.4.20. Побудова тривимірних графіків

Графік функції двох змінних, заданої, як в нашому випадку, в явному вигляді, найпростіше побудувати за допомогою команди surf.

```
-->[x y]=meshgrid(0:0.1:2,0:0.1:2);
-->z=100*(y-x.^2).^2+(1-x).^2;
-->surf(x,y,z)
```

Код досить простий. Спочатку задаються область зміни аргументів за допомогою команди meshgrid, так звану сітку. Потім в вузлах цієї сітки обчислюються значення функції, і якщо все зроблено правильно, команда surf дає шуканий результат.

Звертаємо вашу увагу, оскільки x і y є стовпцями, під час запису функції слід використовувати поелементні дії. У нашому випадку це зведення в ступінь: «.^»

Графік має вигляд:

grid on – додає сітку до поточного графіку;

grid off – видаляє сітку [53].

У багатьох випадках вимагається побудова багатьох накладених один на одного графіків в одному і тому ж вікні. Для цього служить команда продовження графічних побудов hold. Вона використовується в наступних формах:

hold on – забезпечує продовження виведення графіків в поточне вікно, що дозволяє додавати наступні графіки до вже існуючих;

hold off – скасовує режим продовження графічних побудов;

Буває, що в одному вікні треба розташувати кілька координатних осей з різними графіками без накладання їх однієї на одну. Для цього використовуються команди subplot, що застосовуються перед побудовою графіків:

subplot(m, n, p) – розбиває графічне вікно на $m \times n$ підвікон, при цьому m – число підвікон по горизонталі, n – число підвікон по вертикалі, а p – номер підвікна, в яке буде виводитися поточний графік (підвікна рахують послідовно по рядках).

Проілюструємо роботу функції subplot (див. рис. 2.30):

```
>>subplot(3, 2, 1); plot(x,y);
>> subplot(3, 2, 4); plot(x,y);
>> subplot(3, 2, 5); plot(x,y);
```

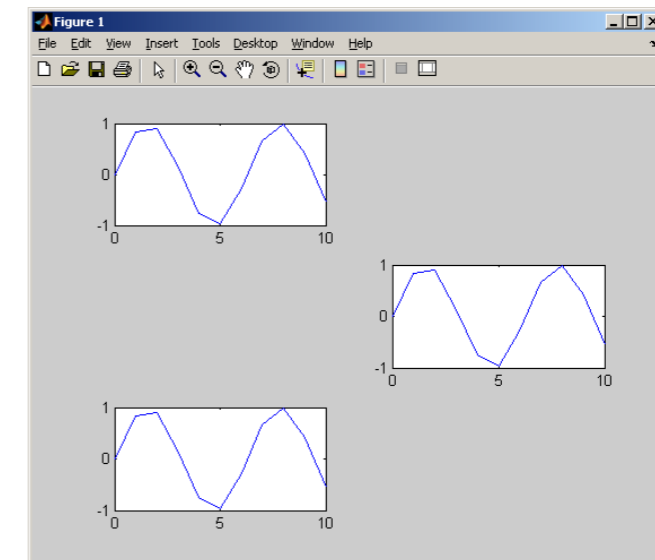


Рис. 2.30 – Робота функції subplot

Було сформовано 3 рядки і два стовпці полів для виведення графіків. Звернення до кожного конкретного поля відбувається із зазначенням його номера. Нумерація відбувається зліва направо і знизу вгору

2.2.5. Розв'язання типових завдань алгебри і аналізу

В системі MatLab для розв'язання систем лінійних рівнянь передбачаються знаки операцій / і \. Щоб розв'язати систему лінійних рівнянь виду:

$$A \cdot Y = B, \quad (2.1)$$

де A – задана квадратна матриця розміром $N \times N$,

B – заданий вектор-стовпець довжини N , досить застосувати операцію \backslash і обчислити вираз $A \backslash B$.

Операція \backslash називається лівим діленням матриць і при застосуванні до матриць A і B у вигляді $A \backslash B$, приблизно еквівалентна обчисленню висловлювання $\text{inv}(A) \cdot B$. Тут під $\text{inv}(A)$ розуміють обчислення матриці, оберненої до матриці A . Операцію $/$ називають правим діленням матриці. Вираз A/B наближено відповідає обчисленню виразу $B \cdot \text{inv}(A)$. Отже, ця операція дозволяє

розв'язувати системи лінійних рівнянь виду $Y \cdot A = B$ [21].

Приклад 1. Розв'язати систему лінійних рівнянь методами Крамера, Гауса та за допомогою лівого ділення:

$$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 30, \\ -x_1 + 2x_2 - 3x_3 + 4x_4 = 10, \\ x_2 - x_3 + x_4 = 3, \\ x_1 + x_2 + x_3 + x_4 = 10. \end{cases}$$

Дії, які необхідно виконати для розв'язування поставленої задачі методом

Крамера:

```
% Решим систему методом Крамера
```

```
A = [1 2 3 4; -1 2 -3 4; 0 1 -1 1; 1 1 1 1];
```

```
b = [30;10;3;10];
```

```
% Проверим не вырожденность системы
```

```
rank(A)
```

```
>> ans = -4
```

```
% По правилу Крамера
```

```
A1 = A;
```

```
A2 = A;
```

```
A3 = A;
```

```
A4 = A;
```

```
A1(:,1) = b;
```

```
A2(:,2) = b;
```

```
A3(:,3) = b;
```

```
A4(:,4) = b;
```

```
x1 = det(A1) / det(A);
```

```
x2 = det(A2) / det(A);
```

```
x3 = det(A3) / det(A);
```

```
x4 = det(A4) / det(A);
```

```
x=[x1;x2;x3;x4];
```

```
% Проверим решение A*x - b
```

```
>> ans = 0
```

```
0
```

```
0
```

```
0
```

Тепер добре видно точки мінімуму. Перша знаходиться в околиці -4, друга - близько 2. Для знаходження більш точного значення мінімуму функції в Scilab служить команда

```
[Fmin, xmin] = optim (fun, x0)
```

Тут x_0 - початкове наближення. Команда `optim` повертає абсцису точки мінімуму x_{min} і значення функції в цій точці f_{min} . Головною особливістю команди `optim` є структура її основний аргумент - функції `fun`.

Припустимо, що функція $f(x)$, мінімум якої ми шукаємо, задана за допомогою команди `deff` або конструкції `function` (так воно і є, оскільки ми ж будували її графік). Тоді `fun` визначається наступним чином:

```
function [Fmain, Fdif, ind] = fun (x, ind)
```

```
Fmain = f (x);
```

```
Fdif = numdiff (f, x);
```

```
endfunction
```

Функція `fun`, таким чином, повертає саму функцію `Fmain` і її похідну `Fdif`. Параметр `ind` ми не чіпаємо: це внутрішній параметр і він потрібен SciLab для зв'язку `fun` з командою `optim`.

Застосуємо тепер ці відомості до нашого завдання.

```
-->deff('y=f(x)', 'y=x^4+3*x^3-13*x^2-6*x+26');
```

```
-->function [fmain,fdif,ind]=fun(x,ind)
```

```
-->fmain=f(x);
```

```
-->fdif=numdiff(f,x);
```

```
-->endfunction
```

Тепер знаходимо значення точок мінімуму, скориставшись з'ясованим раніше початковим наближенням.

```
-->[fmin1,xmin1]=optim(fun,-4)
```

```
xmin1 =
```

```
- 3.8407084
```

```
fmin1 =
```

```
- 95.089413
```

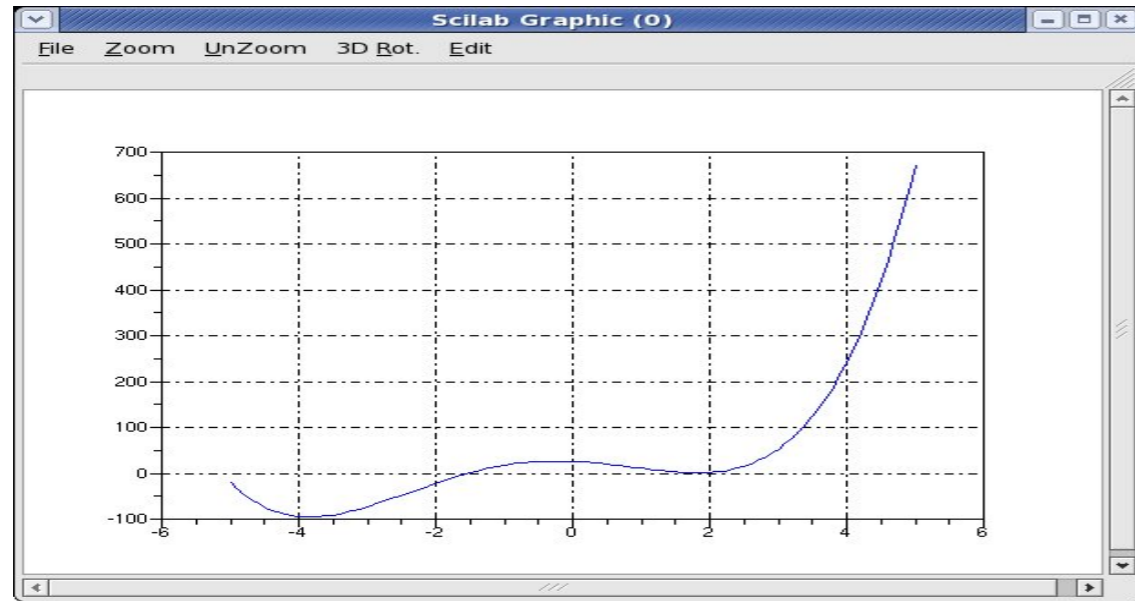
```
-->[fmin2,xmin2]=optim(fun,2)
```

```
xmin2 =
```

```
1.8068586
```

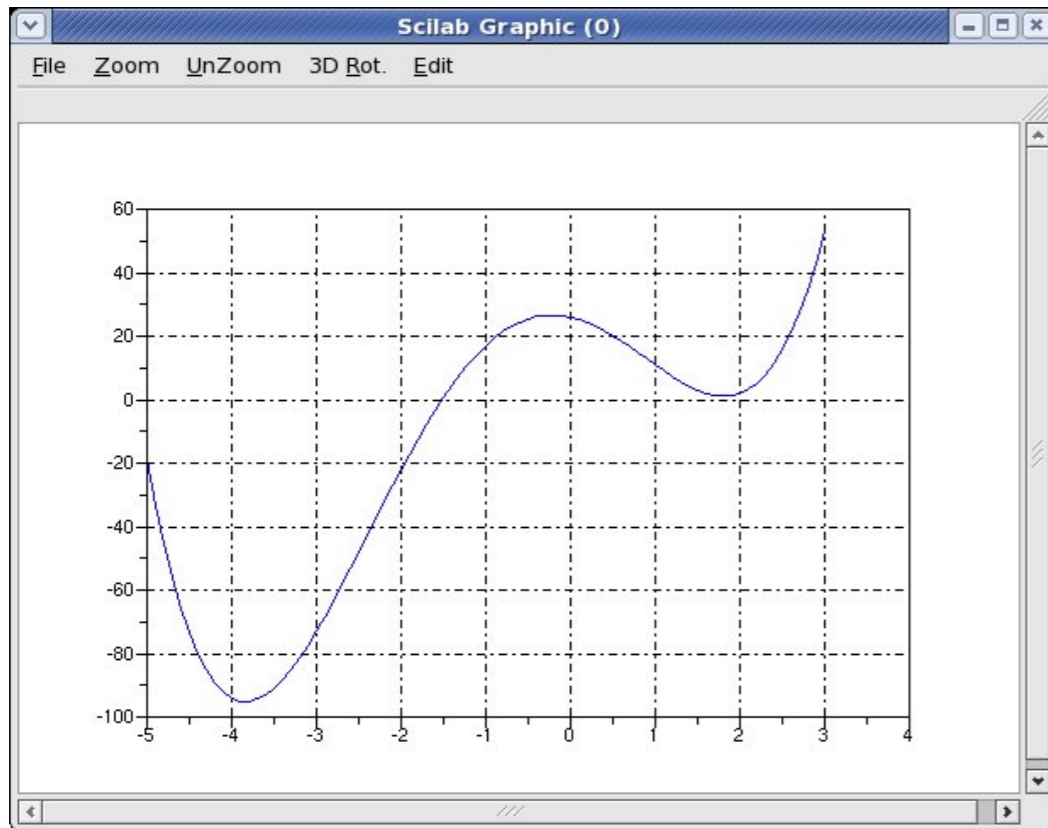
```
fmin2 =
```

```
1.0725284
```



Видно, що функція має два локальних мінімуми, але для більш наочного її подання має сенс розглянути поведінку функції на меншому інтервалі, оскільки «хвіст» графіка праворуч малоінформативний.

```
-->def('y=f(x)', 'y=x^4+3*x^3-13*x^2-6*x+26');
-->x=-5:0.1:3;
-->plot(x,f(x))
-->xgrid()
```



Дії, які необхідно виконати для розв'язування поставленої задачі методом Гауса:

```
% Решим систему Ax=b методом Гаусса
% Для этого, сформируем расширенную систему
A = [1 2 3 4; -1 2 -3 4; 0 1 -1 1; 1 1 1 1];
b = [30;10;3;10];
C = [A b];
% Приведем ее к ступенчатому виду, выполнив прямые и обратные шаги метода Гаусса
D = rref(C)
>> D = 1 0 0 0 1
      0 1 0 0 2
      0 0 1 0 3
      0 0 0 0 4
% Последний столбик - решение
x = D(:,5);
% Проверим решение A*x - b
>> ans = 0
      0
      0
      0
```

Дії, які необхідно виконати для розв'язування поставленої задачі за допомогою лівого ділення.

Заповнимо матрицю і вектор-стовпець правої частини (права частина повинна бути саме стовпчиком, інакше виведеться помилка про несумісність овпадении розмірностей)

```
A = [1 2 3 4
      -1 2 -3 4
      0 1 -1 1
      1 1 1 1];
b = [30;10;3;10];
і використаємо знак лівого ділення
x = A\b
x =
  1
  2
  3
  4
```

Замість знака лівого ділення можна було викликати функцію `mldivide`

```
x = mldivide(A, b)
```

Результат буде такий же. Виконуючи перевірку, впевнімося у тому, що результат правильний.

```
b - A*x
ans =
  0
  0
  0
  0
```

Розв'язання рівняння $F(x)=0$, або знаходження нулів функції, здійснюється за допомогою функції `fzero(name, x0)`. В якості першого аргументу їй передається ім'я функції, яка задає початкове рівняння, другим аргументом служить початкове наближення до кореня. Розглянемо наступний приклад [21].

Приклад 2. Визначити нулі функції $\cos(x)$ на відрізку від 0 до π . В якості початкового наближення прийmemo $x_0=1$.

```
>> X = fzero ('cos', 1)
x = 1.5708
```

Якщо потрібно знайти корінь рівняння в якому використана функція, відмінна від стандартної (вбудованої в систему MatLab), то потрібно присвоїти деяке ім'я виразу, що обчислює функцію.

Приклад 3. Знайти корені рівняння $\cos(x)=x$.

Задача еквівалентна знаходженню нулів функції, які знаходяться за формулою $y = \cos(x)-x$, що не має в рамках системи MatLab фіксованого імені. У цьому випадку потрібно створити *Mat*-функцію виду

```
function y = MyFunction1 (x)
y = cos (x)-x
```

Після цього можна скористатися функцією `fzero`:

```
>> X = fzero ('MyFunction1', pi/2)
x = 0.7391
```

Якщо знайдено абсолютно точне значення кореня, то значення функції в цій точці дорівнює нулю. Таким чином, величину функції в наближено знайденому нулі побічно характеризує похибка результату. Щоб керувати похибкою, потрібно здійснювати виклик функції `fzero` з трьома аргументами `fzero (name, x0, tol)`, де `tol` задає необхідну величину похибки. Необхідно відзначити, що функція `fzero` знаходить нулі тільки для функцій однієї дійсної змінної, які приймають лише дійсні значення. Однак часто буває необхідно знайти комплексні корені функцій дійсної змінної, особливо у випадку многочленів. Для цієї мети в системі MatLab існує спеціальна функція `roots(p)`, якій в якості аргументу передається масив коефіцієнтів многочлена (p).

Приклад 4. Знайти корені рівняння $x^4 - 3x^3 + 3x^2 - 3x + 2 = 0$.

Для многочлена $p = x^4 - 3x^3 + 3x^2 - 3x + 2$ потрібно спочатку сформувати масив його коефіцієнтів (розташували їх в порядку спадання степенів x):

```
>> P = [1 -3 3 -3 2];
>> R = roots (p)
r = 2.0000
-0.0000 + 1.0000i
-0.0000 -1.0000i
1.0000
```

У задачі про знаходження нулів функції складним моментом є знаходження початкового наближення до нуля функції, а також апріорна оцінка їх кількості. Тому важливо паралельно із застосуванням функцій типу `roots` або `fzero` візуалізувати поведінку шуканих функцій на тому чи іншому відрізку значень аргументу [21].

В системі MatLab є спеціальні функції для пошуку мінімуму заданих функцій. При цьому можливий пошук мінімуму як для функції однієї дійсної

В обох випадках результат має порядок 10^{-15} , тобто практично дорівнює 0, що підтверджує правильність розв'язку.

Даний приклад ілюструє той факт, що завдання розв'язання системи нелінійних рівнянь не є тривіальною і вимагає вдумливого і акуратного підходу на кожному кроці, і тільки в цьому випадку рішення буде успішним.

2.4.18. Завдання оптимізації

Це найбільш поширений в інженерній практиці клас задач. Дійсно, проектуючи установку, технологію або процес, ми прагнемо мінімізувати деякі витрати або навантаження, максимально збільшити позитивний ефект, найбільшим чином скоротити виробничий цикл або, навпаки, збільшити його з метою досягнення найкращих якісних показників продукту.

Зверніть увагу, які прикметники використовуються: найбільший, максимальний, мінімальний, найкращий. Всі вони характеризують завдання оптимізації. Економічні розрахунки - взагалі часто-густо рішення таких задач, деякі з яких вже стали класичними. Наприклад, завдання про розподіл ресурсів, завдання комівояжера.

2.4.19. Локальний мінімум функцій однієї змінної

В якості найпростішої оптимізаційної задачі розглянемо пошук локального мінімуму функції однієї змінної. Завдання полягає в пошуку координат точки локального мінімуму заданої функції.

Як і у випадку з пошуком коренів нелінійного рівняння, рішення здійснюється в два етапи. Спочатку потрібно побудувати графік, щоб переконатися в існуванні мінімуму і вибрати початкове наближення для абсциси шуканої точки. Потім знаходяться точне значення координат точки мінімуму.

Розглянемо конкретний приклад.

Знайти мінімум функції
 $f(x) = x^4 + 3x^3 - 13x^2 - 6x + 26$

Будуємо графік функції на довільному інтервалі., Який згодом при необхідності можна буде скорегувати.

```
-->deff('y=f(x)', 'y=x^4+3*x^3-13*x^2-6*x+26');
-->x=-5:0.1:5;
-->plot(x,f(x))
-->xgrid()
```

Отримуємо:

Тут $x(1) = x$, $x(2) = y$. При цьому функція виглядає так:

$$\begin{cases} x(1)^2 + x(2)^2 - 1 = 0 \\ x(2) - x(1)^3 = 0 \end{cases}$$

Мінлива eq(1) містить перше рівняння, eq(2) друге. Тепер залишається знайти коріння:

```
-->res1=fsolve([-0.8 -0.6],f)
```

```
res1 =
```

```
- 0.8260314 - 0.5636242
```

```
-->res2=fsolve([0.8 0.6],f)
```

```
res2 =
```

```
0.8260314 0.5636242
```

Зробимо перевірку:

```
-->f(res1)
```

```
ans =
```

```
1.0D-15 *
```

```
- 0.3330669
```

```
0.1110223
```

```
-->f(res2)
```

```
ans =
```

```
1.0D-15 *
```

```
- 0.3330669
```

```
- 0.1110223
```

змінної, так і для функцій багатьох змінних. Для функцій однієї змінної їх мінімуми знаходить функція $xmin = fmin(name, x_0, x_1)$. Тут $name$ – ім'я функції, в якій знаходять мінімуми, а x_0 і x_1 задають відрізок пошуку. Для пошуку мінімуму функції декількох змінних застосовується функція $fmns: xmins = fmns(name, x_0)$. Тут $name$ – ім'ям функції декількох змінних, для якої знаходять мінімум, а x_0 – це вектор її аргументів, з якого починається пошук. Для ілюстративного прикладу створимо просту функцію двох змінних, що має мінімум в точці (0,0).

```
function y = MyFunc2(x)
```

```
y = x(1)^2 + x(2)^2;
```

Після цього можна викликати функцію $fmns$, яка наближено знаходить вектор $xmin$ координат точки мінімуму:

```
>> Xmin = fmns('MyFunc2', [1,1]);
```

```
>> Xmin(1)
```

```
ans = -2.1023e-005
```

```
>> Xmin(2)
```

```
ans = 2.5484e-005
```

Для обчислення інтегралів методом трапеції в системі MatLab передбачена функція $trapz$: $Integ = trapz(x, y)$. Точність обчислення інтеграла залежить від величини кроку інтегрування: чим менше цей крок, тим більша точність [45].

Приклад 5. Обчислити інтеграл $\int_0^{\pi} \cos(x)dx$ методом трапецій з кроком

інтегрування $\pi/10$.

```
>> Dx = pi / 10;
```

```
>> X = 0: dx: pi;
```

```
>> Y = cos(x);
```

```
>> I1 = trapz(x, y);
```

```
I1 = 5.5511e-017
```

Зазвичай для досягнення високої точності потрібно виконувати інтегрування з дуже малим кроком, а контроль досягнутої точності здійснювати шляхом порівняння послідовних результатів. При одному і тому ж кроці інтегрування методи більш високих порядків точності досягають більш точних результатів. У системі MatLab методи інтегрування більш високих порядків точності реалізуються функціями: $quad$ (метод Сімпсона) і $quad8$ (метод Ньютона-Котеса 8-го порядку точності) [76].

Подвійні інтеграли в системі MatLab обчислюються за допомогою спеціальної функції $dblquad$.

Приклад 6. Обчислити інтеграл виду $\int_0^1 \int_0^2 (x \sin(y) + y \sin(x)) dx dy$.

Запишемо підінтегральну mat -функцію і викличемо функцію $dblquad$:

```
function z = integ(x, y)
```

```
z = x.*sin(y) + y.*sin(x);
```

```
>> J = dblquad('integ', 0, 1, 1, 2);
```

```
J = 1.1678
```

Можливості вбудованого пакету символьних обчислень і операцій Symbolic Math Toolbox досить великі, розглянемо лише деякі його можливості [23]. Символьний об'єкт створюється за допомогою функції `syms`. Команда створює три символьні змінні x , a й b

»`Syms x a b`

Конструювання символьних функцій від змінних класу `sym` об'єкт здійснюється з використанням звичайних арифметичних операцій та позначень для вбудованих математичних функцій. Запис формули для виразу в один рядок не завжди є зручним, більш природним є введення виразу в командне вікно за допомогою функції `pretty` [23]:

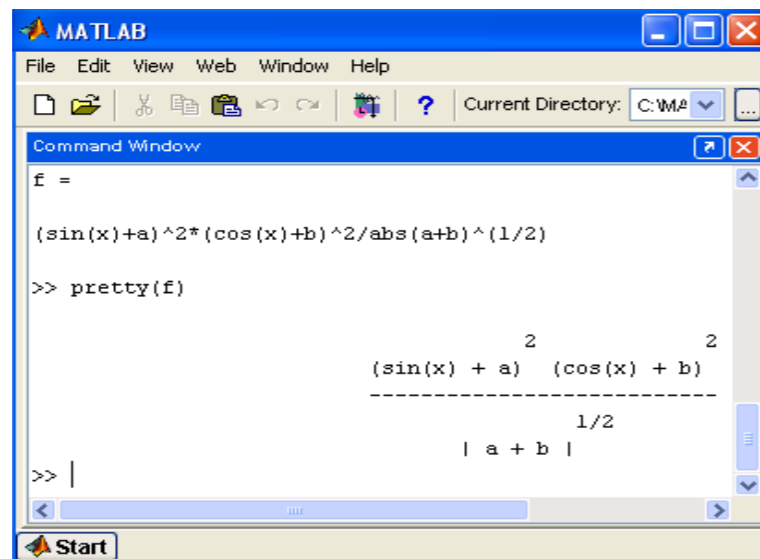


Рис. 2.31 – Демонстрація роботи функції `pretty`

Приклад 5. Спростити вираз $\frac{x^2 - y^2}{x - y}$, використовуючи функцію `simplify`.

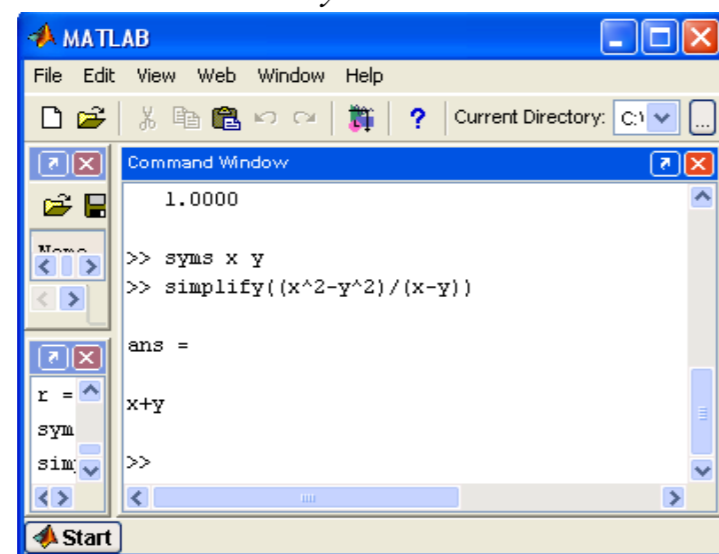


Рис.2.32 – Демонстрація роботи функції `simplify`

Символьну функцію можна створити без попереднього оголошення змінних за допомогою `syms`, вхідним аргументом якої виступає рядок з виразом, виділеним апострофом:

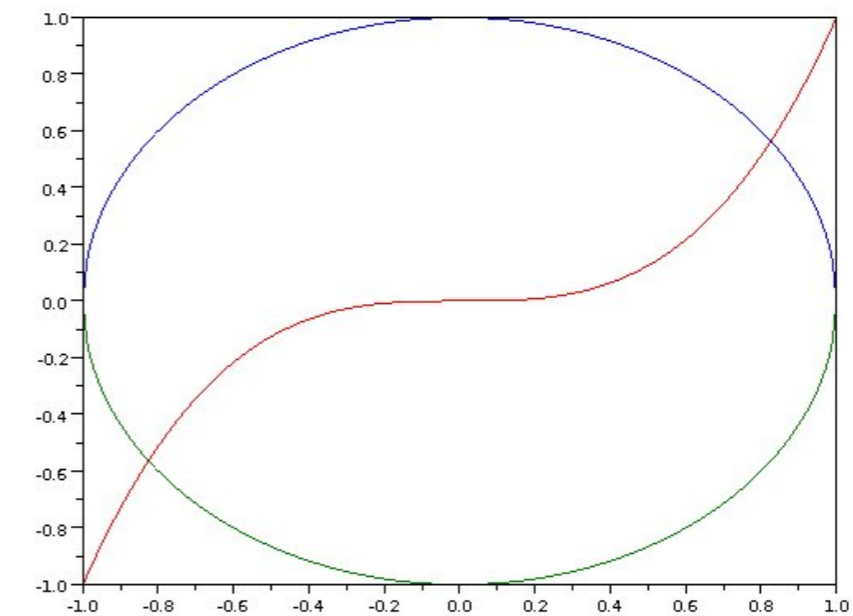
```
-->deff('z=y1(x)', 'z=sqrt(1-x^2)');
```

```
-->deff('t=y2(x)', 't=-sqrt(1-x^2)');
```

```
-->deff('p=y(x)', 'p=x^3');
```

```
-->x=-1:0.01:1;
```

```
-->plot(x,y1(x),x,y2(x),x,y(x))
```



Для отримання еліпса нам довелося будувати окремо його верхню і нижню дугу, а по-друге, розглядається проміжок x , відповідний області визначення: $[-1; 1]$.

Видно, що система має два кореня. В якості початкового наближення для першого візьмемо значення $(-0.8; -0.6)$, для другого: $(0.8; 0.6)$.

Тепер переходимо до другого етапу розв'язання - безпосередньому знаходженню коренів. Для цього потрібно задати функцію, яка описує рівняння системи, які повинні бути представлені в наступному вигляді:

```
-->function [eq]=f(x)
```

```
-->eq(1)=x(1)^2+x(2)^2-1;
```

```
-->eq(2)=x(2)-x(1)^3;
```

```
-->endfunction
```

2.4.17. Системи рівнянь

У загальному випадку задача розв'язання системи двох нелінійних рівнянь з двома невідомими має наступний вигляд:

$$\begin{cases} f(x, y) = 0 \\ g(x, y) = 0 \end{cases} \quad (*)$$

Формально можна виписати і систему трьох рівнянь для трьох невідомих, і чотирьох, проте слід зауважити, що в інженерній практиці нелінійні системи числом більше двох невідомі.

Рішенням системи нелінійних рівнянь (*) є пара чисел (x^*, y^*) , яка при підстановці в рівняння перетворює їх в тотожності. Рішенням може бути не одна пара, а кілька. В цьому випадку говорять, що система має кілька рішень. Природно, можлива ситуація, коли система взагалі не має розв'язків.

Геометрично розв'язання системи - координати точок перетину кривих $f(x, y) = 0$ і $g(x, y) = 0$.

Розв'язання проводяться в два етапи. Спочатку необхідно упевнитися в тому, що розв'язок системи існує. Для цього будуються графіки функцій $f(x, y) = 0$ і $g(x, y) = 0$ і з'ясовується, чи є у них точки перетину. Наближені координати цих точок будуть використовуватися в якості початкових наближень на другому етапі безпосереднього обчислення кореня. Розглянемо конкретну ситуацію.

Приклад. Розв'язати систему рівнянь:

$$\begin{cases} x^2 + y^2 = 1 \\ y = x^3 \end{cases}$$

Перш за все, зведемо систему до стандартного вигляду, переносячи в обох рівняннях всі члени в ліву частину:

$$\begin{cases} x^2 + y^2 - 1 = 0 \\ y - x^3 = 0 \end{cases}$$

Спочатку з'ясуємо, чи має ця система розв'язок. Виражаємо з кожного рівняння y (х) і побудуємо графік цих функцій.

З першого рівняння

$$y^2 = 1 - x^2, \quad y = \pm \sqrt{1 - x^2}.$$

Область визначення цієї функції $x \in [-1; 1]$.

З другого рівняння $y = x^3$. Будуємо графіки:

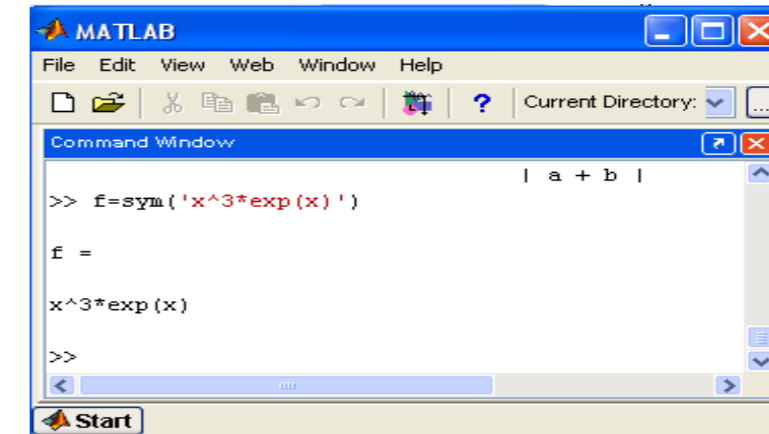


Рис.2.33 – Демонстрація створення символічної змінної без попереднього оголошення

Спрощення тригонометричних, логарифмічних, експоненційних функцій і поліномів здійснюється функцією `expand`, формат звернення до якої має наступний вид: `rez=expand(S)`, де S – символічний вираз, який необхідно спростити, `rez` – результат спрощення, наприклад:

```
>> syms x y;
>> rez1=expand(sin(x+y))
rez1 = sin(x)*cos(y)+cos(x)*sin(y)
```

За допомогою функції `factor` можна розкладати многочлени на прості множники, а цілі числа представити, як добуток простих чисел [23]:

```
>> factor(sym('x^5 - 1'))
ans =
(x-1)*(x^4+x^3+x^2+x+1)
```

Функція `subs` здійснює підстановку нових виразів для вказаних символічних змінних:

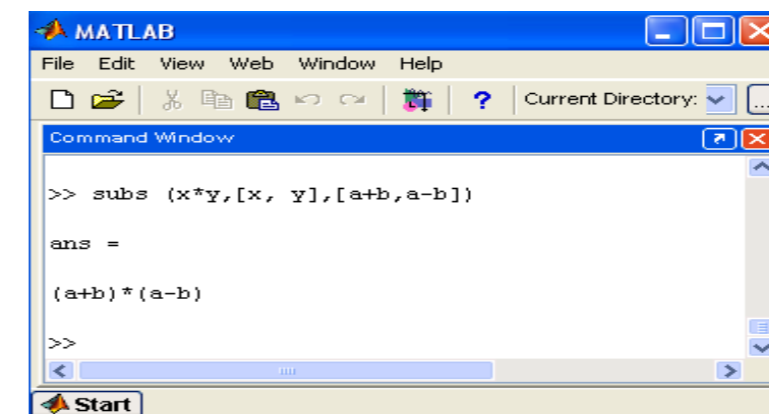


Рис.2.34 – Демонстрація роботи функції subs

Symbolic Math Toolbox дозволяє працювати як з невизначеними інтегралами, так і з визначеними. Невизначені інтеграли від символічних функцій обчислюються за допомогою функції `int`, в якості вхідних аргументів вказують символічну функцію та змінну, за якою відбувається інтегрування (див. рис.2.35).

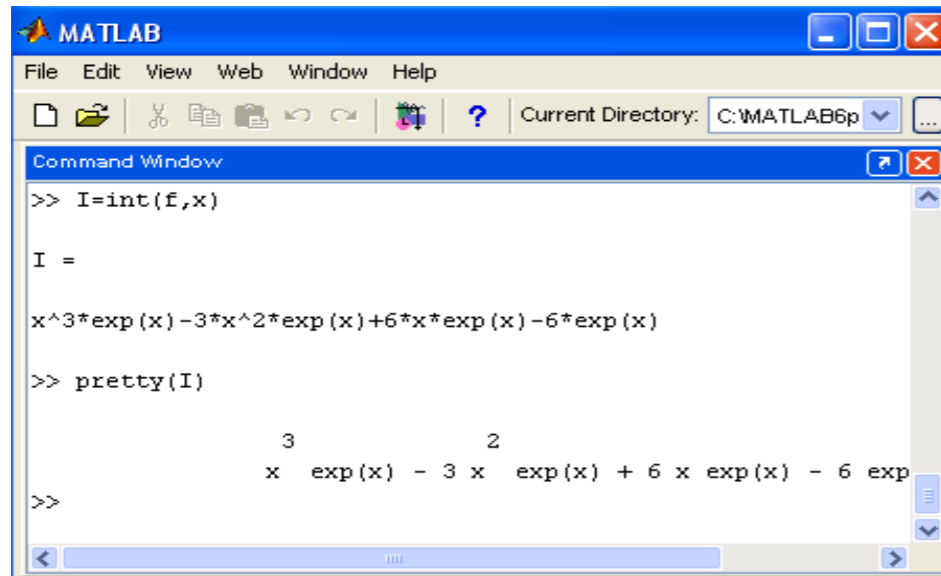


Рис. 2.35 – Демонстрація роботи функції `int`

Для знаходження визначеного інтеграла в символьному виді необхідно знайти нижню і верхню границі інтегрування, відповідно, в третьому і четвертому аргументах `int`:

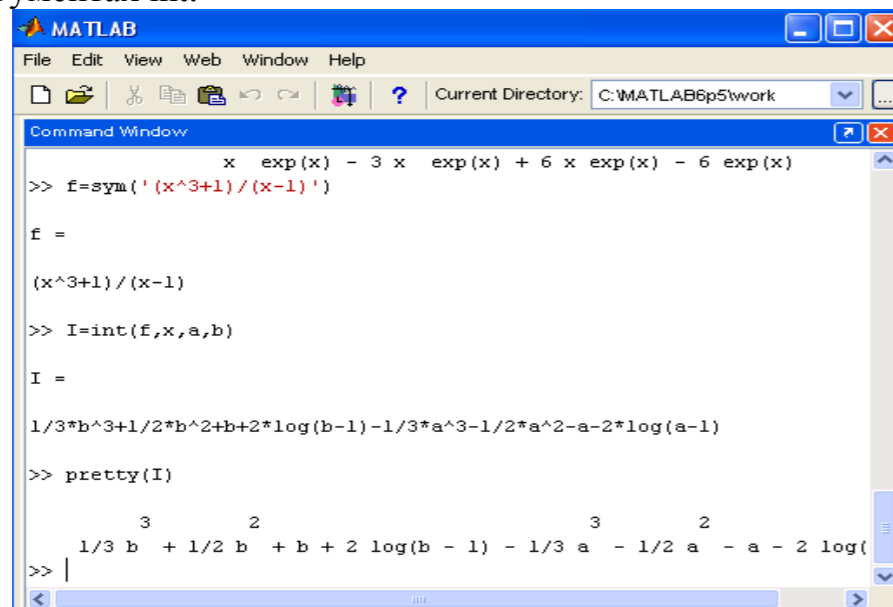


Рис. 2.36 – Демонстрація роботи функції `int` для знаходження визначеного інтеграла

Перерахуємо ще декілька функцій, які часто використовуються при символьних обчисленнях:

`inv` – обчислює обернену матрицю;

`limit` – обчислює границі;

`taylor` – здійснює розкладання функцій в ряд Тейлора;

`solve` – розв’язує алгебраїчні рівняння та системи алгебраїчних рівнянь

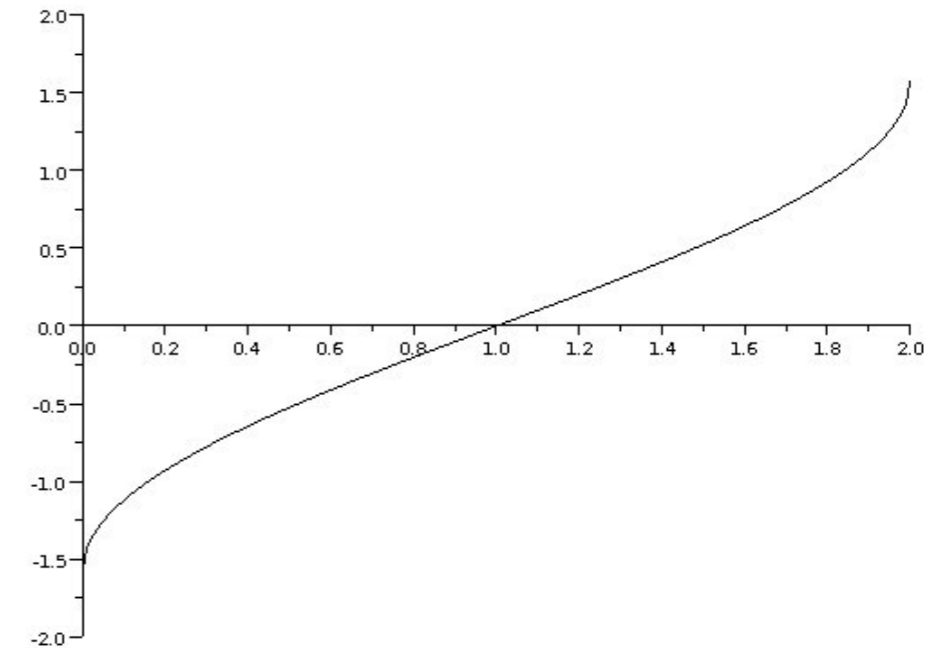
[23].

Для роботи з символьними даними передбачена оболонка `funtool`. Вона являє собою інтерактивний графічний калькулятор, що дозволяє швидко

```
-->deff('t=h(x)', 't=asin(x-1)');
```

```
-->x=0:0.01:2;
```

```
-->plot2d(x,h(x),axesflag=4)
```



```
xx =
```

```
1.
```

```
-->xk=fsolve(1.2,h)
```

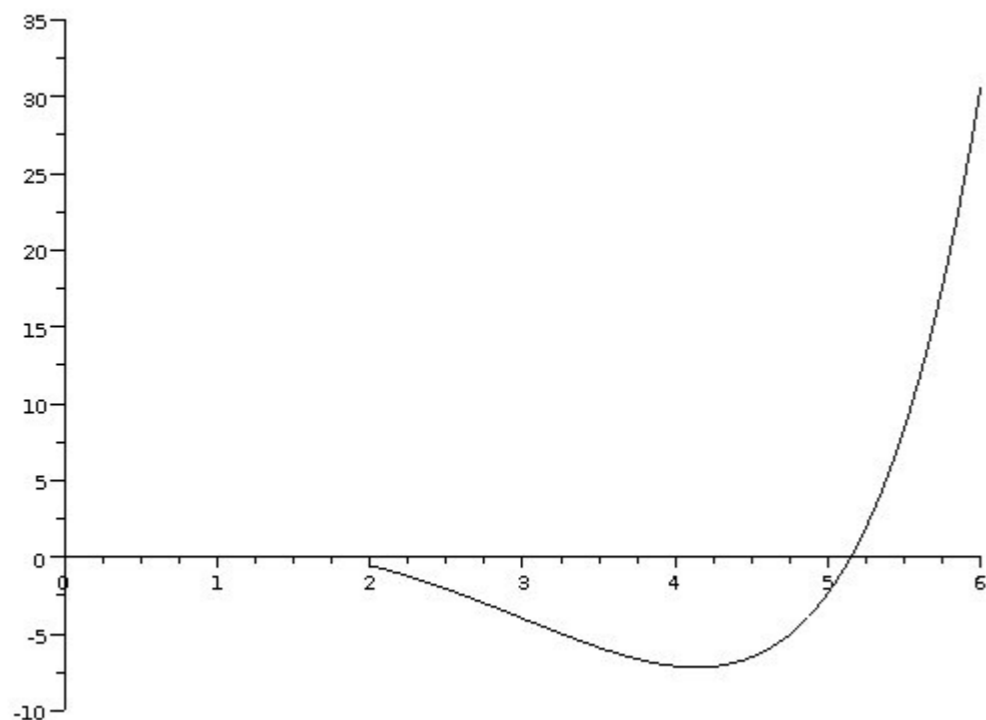
```
xk =
```

```
1.
```

```
-->h(xk)
```

```
ans =
```

```
0.
```

Дійсно, є ще корінь в районі $x = 5$. Знайдемо його:

```
-->x3=fsolve(5,g)
```

x3 =

5.1476865

```
-->g(x3)
```

ans =

7.105D-15

Отже, рівняння має три корені:

$x_1 = 0,5778406$, $x_2 = 1,7638701$, $x_3 = 5,1476865$.

Знайти корені рівняння $\arcsin(x-1) = 0$.

При побудові графіка потрібно врахувати область визначення арксинуса. Аргумент цієї функції повинен знаходитися в межах від -1 до 1 ($-1 \leq x-1 \leq 1$), значить, $x \in [0, 2]$. Подальше просто: задаємо функцію, будуємо на цьому проміжку графік, визначаємо початкове наближення, знаходимо корінь, робимо перевірку.

побудувати графіки двох функцій $f(x)$ і $g(x)$. Інтерфейс даного додатку представлений на рис. 2.37. При запуску вводяться три автономних вікна: два графічні і керуюче. Керуюче вікно містить два поля для вводу функцій, поле вводу обмежень для змінної x , поле вводу коефіцієнту, що здійснює масштабування.

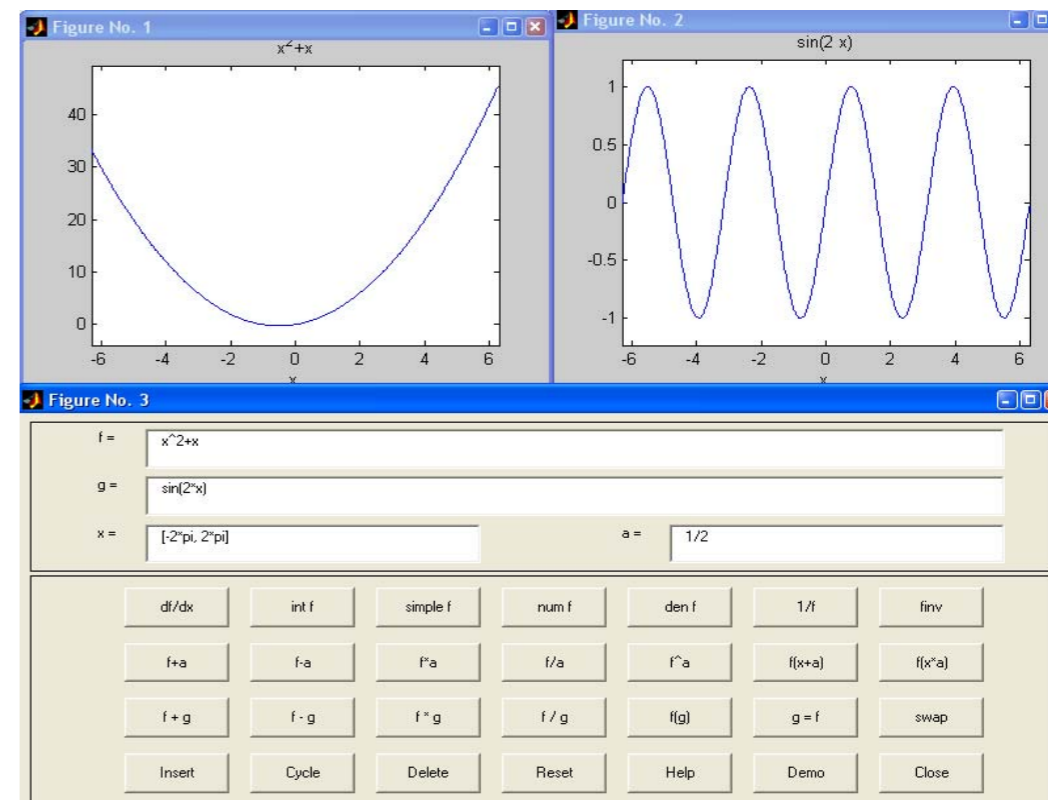


Рис. 2.37 – Інтерфейс додатку funtool

2.2.6. Апроксимація та інтерполяція даних

Під апроксимацією зазвичай розуміють опис деякої, часом не заданої явно, залежності або сукупності даних за допомогою іншої, зазвичай більш простої або більш одноманітної залежності. Часто дані знаходяться у вигляді окремих вузлових точок, координати яких задаються таблицею даних. Результат апроксимації може не проходити через вузлові точки. Для обробки даних MatLab використовує різні функції інтерполяції та апроксимації даних [12].

Одна з найбільш відомих апроксимацій – поліноміальна. У системі MatLab визначені функції апроксимації даних поліномами за методом найменших квадратів (МНК) – поліноміальної регресії, а саме це виконує функція, наведена нижче:

$\text{polyfit}(x, y, n)$ – повертає вектор коефіцієнтів полінома $p(x)$ степеня n , який з найменшою середньоквадратичною похибкою апроксимує функцію $y(x)$.

Під інтерполяцією розуміють обчислення значень функції $f(x)$ в проміжках між вузловими точками. Лінійна, квадратична і поліноміальна інтерполяція реалізуються при поліноміальній апроксимації. У ряді випадків

дуже зручна сплайнова інтерполяція і апроксимація таблично заданих функцій. При ній проміжні точки знаходять за відрізками поліномів третього степеня – це кубічна сплайнова інтерполяція.

При цьому зазвичай такі поліноми обчислюються так, щоб не тільки їх значення збігалися з координатами вузлових точок, але також, щоб у вузлових точках були неперервні похідні першого і другого порядків. Така поведінка характерна для гнучкої лінійки, закріпленої в вузлових точках, звідки і походить назва spline (сплайн) для цього виду інтерполяції (апроксимації).

Для одновимірної таблично інтерполяції використовується функція `interp1`:

$y_i = \text{interp1}(x, y, x_i)$ – повертає вектор y_i , що містить елементи, відповідні елементам x_i і отримані інтерполяцією вектори x і y . Вектор x визначає точки, в яких задано значення y .

$y_i = \text{interp1}(x, y, x_i, \text{method})$ – дозволяє за допомогою параметра `method` задати метод інтерполяції:

'nearest' – східчаста інтерполяція;

'linear' – лінійна інтерполяція (прийнята за замовчуванням);

'spline' – кубічна сплайн-інтерполяція;

'cubic' або 'pchip' – інтерполяція многочленами Ерміта [10].

Сплайн-інтерполяція використовується для представлення даних відрізками поліномів невисокого степеня – найчастіше третього. При цьому кубічна інтерполяція забезпечує неперервність першої та другої похідних результату інтерполяції в вузлових точках. Реалізується сплайн-інтерполяція наступною функцією:

$y_i = \text{spline}(x, y, x_i)$ – використовує вектори x і y , що містять аргументи функції і її значення, і вектор x_i , що задає нові точки.

Розв'язання більшості завдань інтерполяції та апроксимації функцій і табличних даних зазвичай супроводжується їх візуалізацією. Вона, як правило, полягає в побудові вузлових точок функції (або табличних даних) і в побудові функції апроксимації або інтерполяції (рис.2.38).

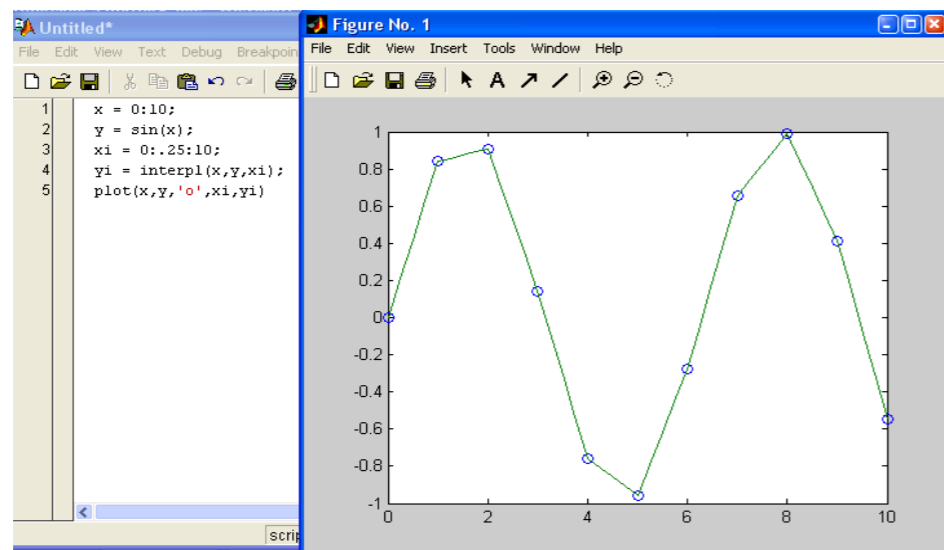
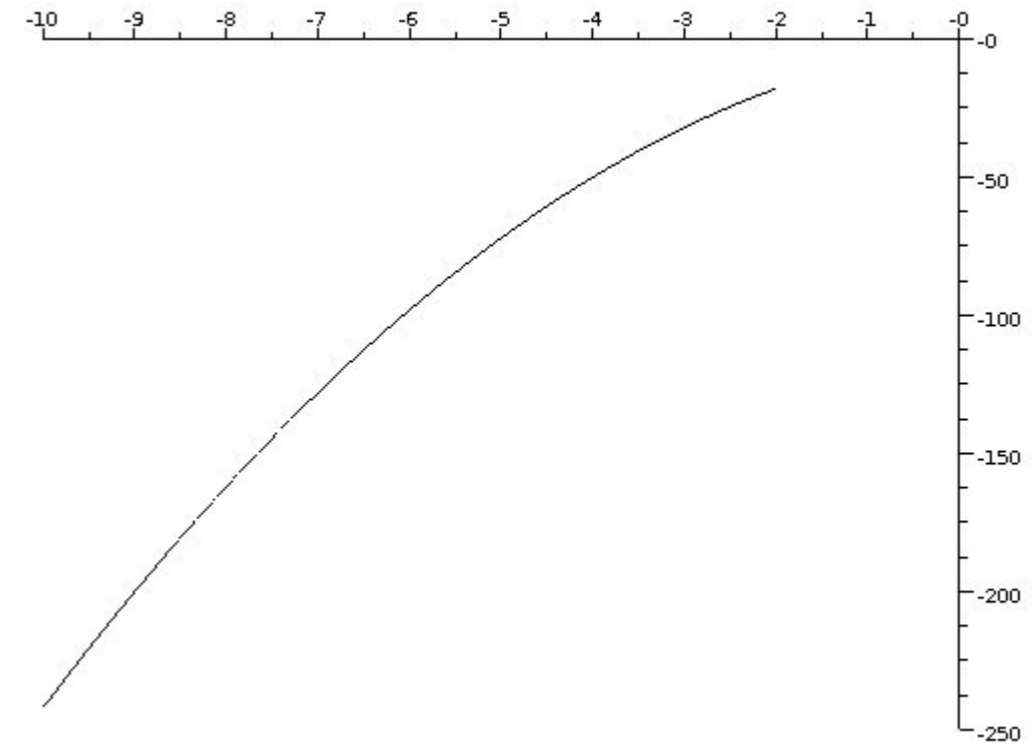
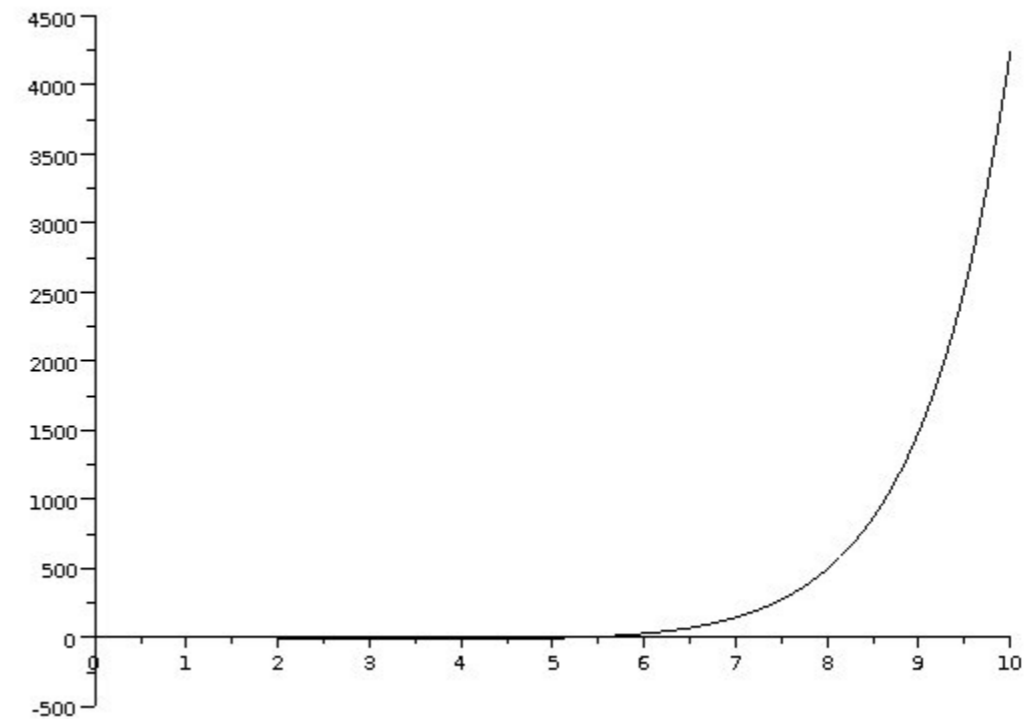


Рис. 2.38 – Приклад візуалізації процесу інтерполяції



Все ясно: функція спадає і коренів тут, по всій видимості, немає. Дивимося інтервал правіше 2:



Очевидно, функція після $x = 6$ різко йде в нескінченність, але її поведінка між 2 і 6 заслуговує більш уважного розгляду:

Очевидно, що на цьому проміжку два кореня: один поблизу 0,5, інший - між 1,5 і 2,0. Візьмемо в якості початкового наближення для першого кореня 0,5, другого - 1,75.

Підкреслимо, що тут потрібні не точні значення, а наближені, тому результат би не змінився, якби ми взяли замість 0,5, наприклад, значення 0,4, а замість 1,75 - 2. Переконайтеся в цьому самостійно.

Отже, знаходимо корені:

```
-->x1=fsolve(0.5,g)
```

```
x1 =
```

```
0.5778406
```

```
-->x2=fsolve(1.75,g)
```

```
x2 =
```

```
1.7638701
```

Виходить, що перший корінь $x_1 = 0,5778406$, другий $x_2 = 1,7638701$. Зробимо перевірку:

```
-->g(x1), g(x2)
```

```
ans =
```

```
- 1.110D-16
```

```
ans =
```

```
0.
```

У першому випадку підстановка дало $\sim 10^{-16}$, у другому - точний 0. Отже, рішення знайдено вірно.

З'ясуємо, чи існують інші корені рівняння. Розглянемо область лівіше -2:

В MatLab поєднання функцій апроксимації з графічною візуалізацією доведено до логічного кінця – передбачена апроксимація рядом методів для точок функції, графік якої побудований. І все це виконується безпосередньо у вікні редактора графіки Property Editor. Для цього в позиції Tools графічного вікна є дві нові команди:

Basic Fitting – основні види апроксимації (регресії);

Data Statistics – статистичні параметри даних [12].

Команда Basic Fitting відкриває вікно, що дає доступ до ряду видів апроксимації і регресії: сплайнової, ермітової і поліноміальної зі степенями від 1 (лінійна апроксимація) до 10. У тому числі зі степенем 2 (квадратична апроксимація) і 3 (кубічна апроксимація) (рис. 2.39).

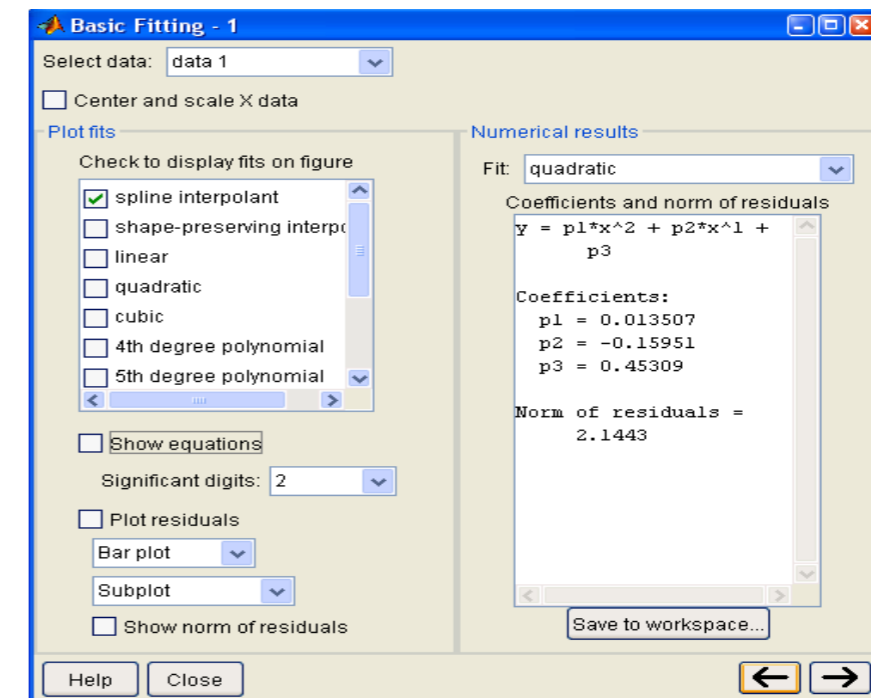


Рис. 2.39 – Вікно доступу до видів апроксимації і регресії

Команда Data Statistics відкриває вікно з результатами простішої статистичної обробки даних (рис.2.40).

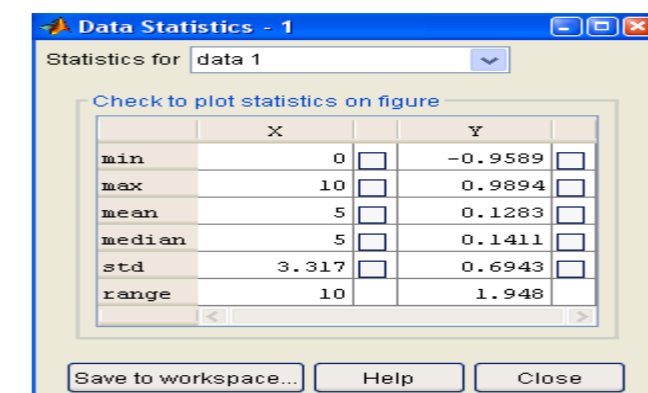


Рис. 2.40 – Вікно результатів статистичної обробки
2.2.7. Методи розв'язання звичайних диференціальних рівнянь

Аналіз поведінки багатьох систем і пристроїв в динаміці базуються на розв'язанні систем звичайних диференціальних рівнянь (ЗДР). Їх, як правило, представляють у вигляді системи диференціальних рівнянь першого порядку в формі Коші:

$$\frac{dy}{dt} = y';$$

$$y' = f(x, y), \quad y(t_0, t_{end}) = b.$$

де t_0, t_{end} – початкові і кінцеві точки інтервалів.

Параметр t не обов'язково означає час, хоча найчастіше розв'язування диференціальних рівнянь знаходиться в часовій області. Вектор b задає початкові і кінцеві умови.

Для розв'язання систем ЗДР в MatLab реалізовані різні методи. Їх реалізації мають назву вирішувачі ЗДР. Вирішувачі реалізують наступні методи розв'язання диференціальних рівнянь, до того ж для розв'язання жорстких систем рівнянь рекомендується використовувати, тільки спеціальні вирішувачі ode45, ode23:

ode45 – однокрокові явні методи Рунге-Кутта 4-го і 5-го порядку. Це класичний метод, який рекомендується для початкової спроби розв'язання.

ode23 – однокрокові явні методи Рунге-Кутта 2-го і 4-го порядку [21].

В розглянутих нижче функціях для розв'язання систем диференціальних рівнянь використовують наступні позначення і правила:

options – аргумент, створений функцією odeset (ще одна функція дозволяє вивести параметри, встановлені за замовченням);

tspan – вектор, який визначає інтервал інтегрування $[t_0 \ t_{final}]$. Для отримання розв'язків у конкретні моменти часу $t_0, t_1, \dots, t_{final}$ (розташовані в порядку спадання або зростання) необхідно використовувати tspan = $[t_0 \ t_1 \ \dots \ t_{final}]$;

y_0 – вектор початкових умов;

p_1, p_2, \dots – довільні параметри, які передаються до функції F ;

T, Y – матриця розв'язки Y , де кожний рядок відповідає часу, що повернений у вектор-стовпчик T .

Розглянемо функції для розв'язання систем диференціальних рівнянь:

$[T, Y] = \text{solver}(@F, \text{tspan}, y_0)$ – де замість solver підставляємо ім'я конкретного вирішувача – інтегрує систему диференціальних рівнянь виду $y' = F(t, y)$ на інтервалі tspan з початковими умовами y_0 , @F – дескриптор ODE-функції. Кожний рядок в масиві розв'язків Y відповідає значенню часу, який повертається у векторі-стовпчику T ;

$[T, Y] = \text{solver}(@F, \text{tspan}, y_0, \text{options})$ – дає розв'язок, що є подібним описаному вище, але з параметрами, які визначені значеннями аргументу options, створеного функцією odeset.

Технологія розв'язання диференціальних рівнянь в системі MatLab така:

1) Створення m -файлу. Незалежно від виду системи він має вид:

```
-->xr=fsolve(0.5,f)
```

```
xr =
```

```
0.5
```

```
-->f(xr)
```

```
ans =
```

```
0.
```

Дійсно, корінь виявився рівним 0,5, що і підтвердила перевірка: підстановка знайденого значення xr в функцію f дає 0.

Приклад. Знайти корінь рівняння

$$f(x) = \frac{e^x}{5} - 2(x-1)^2$$

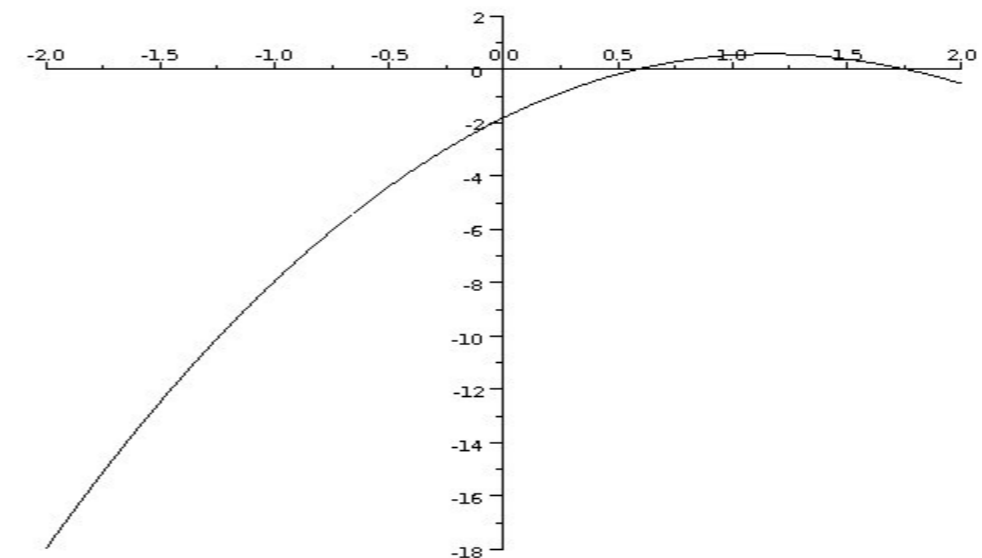
Область визначення знову не обмежена, тому чинимо так само, як і в попередньому прикладі:

```
-->deff('z=g(x)', 'z=exp(x)/5-2*(x-1)^2');
```

```
-->x=-2:0.01:2;
```

```
-->plot2d(x,g(x), axesflag=4)
```

Графік:



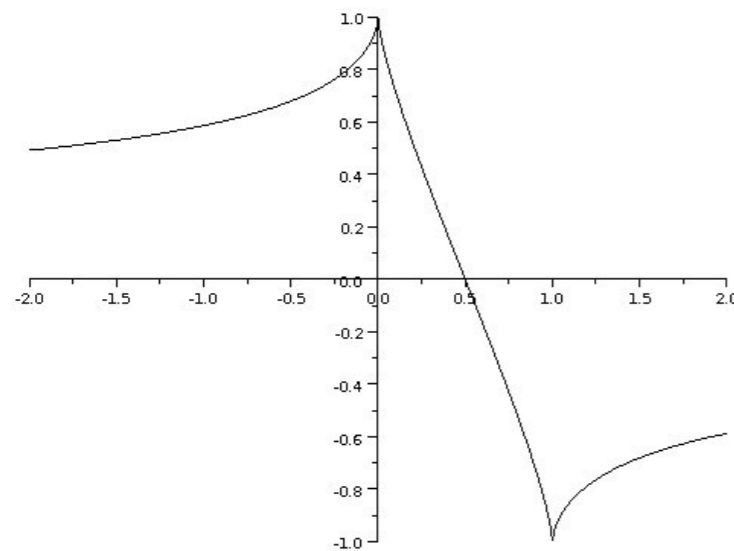
$$f(x) = \sqrt[3]{(x-1)^2} - \sqrt[3]{x^2} = 0$$

Область визначення функції - вся числова пряма, тому з побудовою графіка проблем не виникає. Задаємо функцію і будуємо її графік:

```
-->def f('y=f(x)', 'y=((x-1)^2)^(1/3)-(x^2)^(1/3)');
```

```
-->x=-2:0.01:2;
```

```
-->plot2d(x,f(x),axesflag=4)
```



Перш за все звертаємо увагу на задання функції:

- підкореневий вираз, як складний аргумент, береться в дужки;
- корінь кубічний від деякого виразу, це, по суті - зведення цього виразу в ступінь 1/3, причому ступінь в цьому випадку теж повинна полягати в дужки: '...', ^ (1/3). Дійсно, запис ... ^ 1/3 - це, відповідно до пріоритету математичних операцій, зведення в ступінь 1 і розподіл результату на 3.

Далі. А чому ми взяли проміжок зміни x від -2 до 2? Чесно кажучи, просто навмання. Так як x може бути будь-яким, ми вибрали такий проміжок. Якби кореня на ньому не виявилось, ми б розглянули область правіше: $x \in [2; 4]$, потім $x \in [4; 6]$ і так далі, або «пішли» б в іншу сторону: $x \in [-4; -2]$, $x \in [-6; -4]$, Поки б не знайшли такий інтервал, на якому графік перетинає вісь x, тобто рівняння має корінь.

Нам пощастило відразу: корінь виявився на першому ж обраному проміжку. Більш того, очевидно, що він або дорівнює, або знаходиться поблизу значення 0,5. Значить, його можна вибрати в якості початкового наближення.

Тепер шукаємо точне значення кореня:

```
function dy = solverDE(t, y)
dy = zeros(n, 1);
dy(1) = f1(t, y(1), y(2), ..., y(n));
dy(2) = f2(t, y(1), y(2), ..., y(n));
.....
dy(n) = fn(t, y(1), y(2), ..., y(n));
```

2) Отриманий розв'язок і супроводжуючий його графік:

```
>> [T, Y] = solver('solverDE', [t0 tfinal], [y10 y20 ... yn0]);
>> plot(T, Y)
```

Приклад 6.1. Розв'язати диференціальне рівняння:

$$y''' - 2y'' - y' + 2y = 0 \quad \text{з одиничними початковими умовами.}$$

Дане диференціальне рівняння другого порядку зведемо до системи диференціальних рівнянь першого порядку:

$$\begin{cases} \frac{dy_1}{dt} = y_2; \\ \frac{dy_2}{dt} = y_3; \\ \frac{dy_3}{dt} = 2y_3 + y_2 - 2y_1, \end{cases}$$

з початковими умовами $y_1(0)=1, y_2(0)=1, y_3(0)=1$.

Вектор dy/dt правих частин системи рівнянь, обчислюємо за допомогою власної функції ex21 (рис. 2.41):

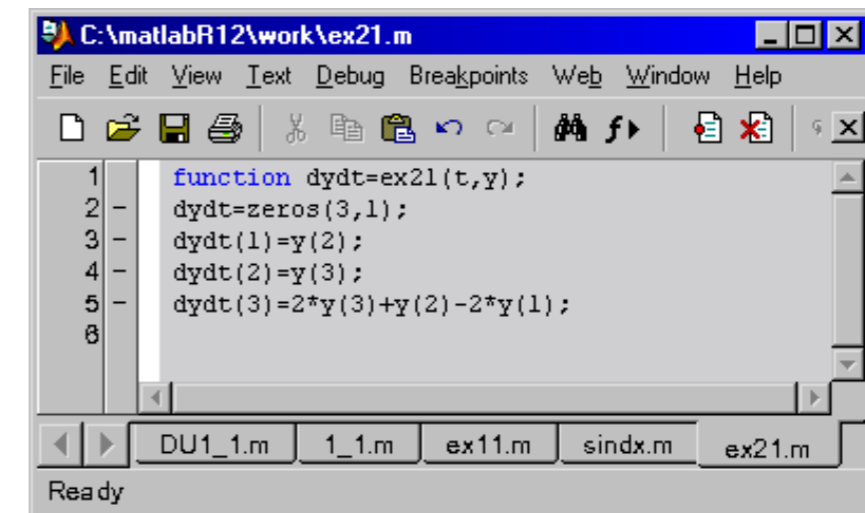


Рис. 2.41 - Приклад створення функції для розв'язання системи ЗДР

Тепер можна викликати функцію ode45, що знаходить розв'язок нашої системи диференціальних рівнянь з початковими умовами [1,1,1] на відрізку [0,20] (рис. 2.42)

```
>> y0=[1 1 1];
>> tspan=[0 20];
>> [T,Y]=ode45('ex21',tspan,y0);
```

>> plot(T,Y)

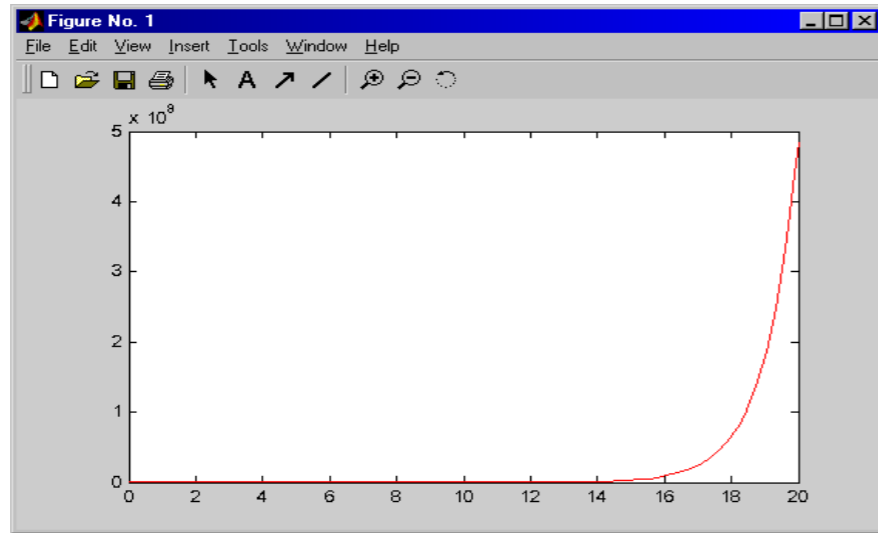


Рис. 2.42 – Результат роботи програми

Для розв'язання диференціальних рівнянь в MatLab зарезервована функція dsolve, яка має наступні формати звернення і повертає аналітичний розв'язок системи диференціальних рівнянь з початковими умовами:

$y = \text{dsolve}('Dy(x)'),$ де $Dy(x)$ – рівняння, y – розв'язки, які повертаються функцією dsolve.

$y = \text{dsolve}('Dy(x)', 'ПУ'),$ де $Dy(x)$ – рівняння, ПУ – початкові умови.

Перша похідна функції позначається Dy , друга похідна – $D2y$ і так далі. Функція dsolve призначена також для розв'язання систем диференціальних рівнянь. У цьому випадку вона має наступний формат звернення:

$[f, g] = \text{dsolve}('Df(x), Dg(x)', 'ПУ'),$ де $Df(x), Dg(x)$ – система рівнянь, ПУ – початкові умови [75].

Приклад 6.2. Розв'язати диференціальне рівняння (6.1) з використанням функції dsolve (рис. 2.43).

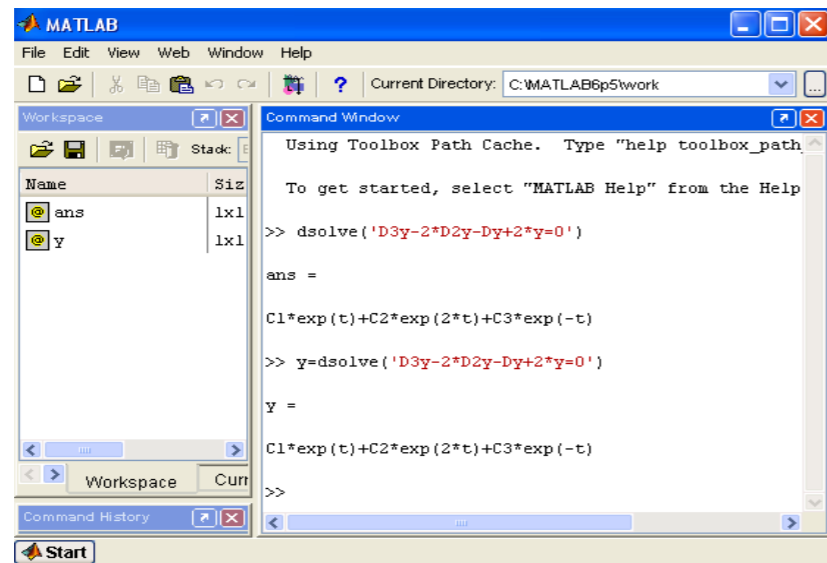
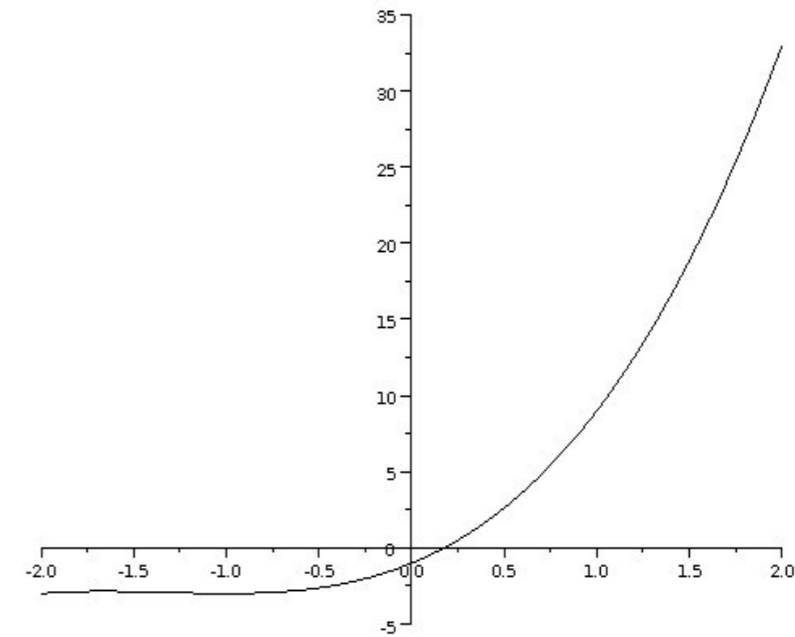


Рис. 2.43 – Приклад використання команди dsolve



2.4.16. Трансцендентні рівняння

Якщо в рівнянні $f(x) = 0$ невідома x входить в аргумент трансцендентних функцій, воно називається трансцендентним. Методи рішення трансцендентних рівнянь є універсальними, тобто справедливими для будь-якої функції $f(x)$, що стоїть в лівій частині рівняння. Природно, за універсальність треба платити - алгоритм рішення трансцендентних рівнянь набагато складніше і вимагає обдумування кожного кроку.

Рішення нелінійного рівняння проводиться в два етапи. На початку відокремлюють коріння рівняння, тобто визначають, по-перше, чи існують коріння рівняння і, по-друге, якщо існують, то на яких проміжках. Це необхідно для вибору початкового наближення x_0 до кореня рівняння. Якщо початкове наближення вибране вірно, вважайте, що задача вирішена: залишається скористатися командою

$\text{fsolve}(x_0, f)$

де x_0 - значення початкового наближення, f - функція, яка описувала ліву частину рівняння.

Перший, найбільш важливий і складний етап відділення коренів зручно проводити графічним методом, тим більше з таким інструментом, як Scilab. Продемонструємо весь цикл рішення задачі на конкретних прикладах.

Приклад. Знайти рішення рівняння

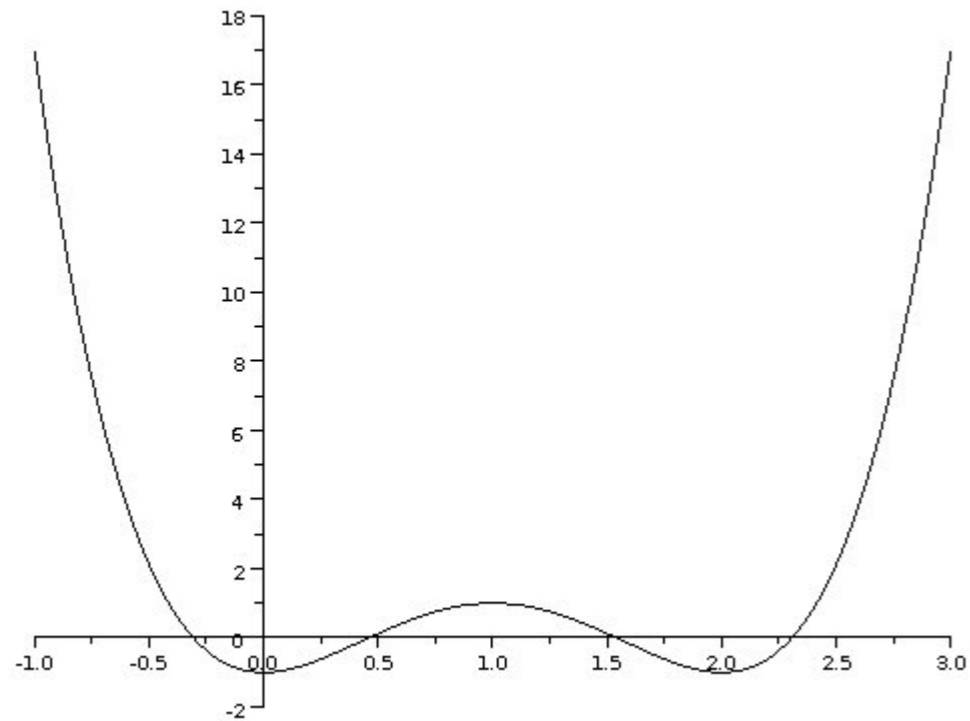
$$\sqrt[3]{(x-1)^2} = \sqrt[3]{x^2}$$

Спочатку наведемо рівняння до виду $f(x) = 0$, перенісши всі його члени в ліву частину:

$$\sqrt[3]{(x-1)^2} - \sqrt[3]{x^2} = 0$$

Значить, в нашому випадку

Побудуємо графік полінома, щоб переконатися, що коріння знайдені вірно.



Приклад. Знайти корені полінома $x^3 + 4x^2 + 5x - 1 = 0$.

```
-->p=poly([-1 5 4 1], 'x', 'c')
```

```
p =
```

```

      2      3
- 1 + 5x + 4x + x

```

```
-->x=roots(p)
```

```
x =
```

```

0.1745594
- 2.0872797 + 1.1713121i
- 2.0872797 - 1.1713121i

```

Поліном має один дійсний і два комплексних кореня, що ілюструє і графік:

MatLab це дуже популярний інструмент для аналізу даних. За даними опитування, проведеного на сайті www.chemometrics.it його використовують до третини всіх дослідників [50, 51].

Нажаль, не дивлячись на те, що МОН України визначило пакет MatLab як базовий для ВНЗ України, використання цього пакету при вивченні математичних дисциплін, за даними проведеного анкетування викладачів, є невиправдано низьким (7% з 70 респондентів) [74].

2.3. Mathcad

Однією з основних галузей застосування персональних комп'ютерів є математичні й науково-технічні розрахунки. Складні обчислювальні задачі, що виникають під час моделювання різних фізичних явищ і процесів, можна розбити на ряд елементарних: обчислення інтегралів, розв'язування алгебраїчних і диференціальних рівнянь та систем і т.д. Для таких задач уже розроблені методи розв'язування, створені математичні системи, доступні для студентів вищих навчальних закладів.

Mathcad — система комп'ютерної алгебри з класу систем автоматизованого проектування, орієнтована на підготовку інтерактивних документів з обчисленнями і візуальним супроводом, відрізняється легкістю використання і застосування для колективної роботи.

Mathcad має простий і інтуїтивний для використання інтерфейс користувача. Для введення формул і даних можна використати як клавіатуру, так і спеціальні панелі інструментів.

Mathcad – один з популярних та зручних пакетів для розв'язування математичних, інженерних та економічних задач. Спочатку Mathcad створювався для розв'язання виключно задач технічної сфери, але з часом стрімко еволюціонував у потужний універсальний математичний пакет для розв'язування практично всяких задач, що пов'язані з розрахунками. Успіх Mathcad на ринку програмних продуктів обумовлений відносною простотою використання та універсальністю додатків. Mathcad не тільки дозволяє вести математичні та економічні розрахунки практично довільної важкості, але й має можливості графічної інтерпретації результатів розрахунків, що особливо важливо при розв'язанні економічних задач. Поєднання текстового, формульного та графічного редакторів з багато поточним ядром дозволяє легко та зрозуміло розв'язувати багато задач, документувати й представляти у зручному вигляді результати розрахунків. Робочий лист може бути збережений в універсальному трансфертному форматі «*.xps», що дозволяє використовувати його у інших додатках, включаючи MS Office.

Система Mathcad існує в кількох основних варіантах:

- Mathcad Standard – ідеальна система для повсякденних технічних обчислень. Призначена для масової аудиторії і широкого використання у навчальному процесі;

- Mathcad Professional – промисловий стандарт прикладного використання математики в технічних додатках. Орієнтована на математиків і науковців, які проводять складні й трудомісткі розрахунки;

- Mathcad Professional Academic – пакет програм для професійного використання математичного апарату з електронними підручниками і ресурсами.

2.3.1. Призначення Mathcad

Mathcad надає користувачу зручне обчислювальне середовище, що поєднує в одній оболонці математичне ядро, текстовий процесор, могутню графічну систему обробки результатів і засоби комунікації. У ній знайшла відображення передова технологія LDI (Live Document Interface), відповідно до якої кожна дія користувача з введення математичного виразу негайно інтерпретується системою і після відповідних обчислень відображається в документі. Після того, як знаково-символьна модель розв'язання записана у виразах Mathcad'у, користувач просто задає потрібні йому дані і негайно отримує результат. Іншим важливим принципом системи є природне, тобто прийняте в математичній літературі, подання формул і виразів. Цей принцип у з'єднанні з розширеними можливостями форматування перетворює Mathcad на могутній інструмент підготовки математичних публікацій, які можна отримувати як у паперовому, так і в електронному вигляді. Основним документом системи є робоча область - Worksheet. У документах Worksheet відображаються як дані, що вводяться користувачем, так і обчислені результати. Кожна формула, текстовий блок, або графік у документі вважаються його окремою областю. З останніми можна проводити традиційні дії — переміщати, видаляти, копіювати і т.д. Mathcad ідеальний інструмент для виконання інженерних та математичних розрахунків.

За допомогою Mathcad можна у символьному та чисельному вигляді:

- виконувати елементарні розрахунки за формулами, використовуючи пакет як інженерний калькулятор;
- розв'язувати нелінійні рівняння та системи;
- розв'язувати задачі лінійної алгебри;
- обробляти експериментальні данні (шляхом інтерполяції та апроксимації методом найменших квадратів);
- диференціювати та інтегрувати;
- розв'язувати задачі оптимізації, а також задачі математичного програмування;
- розв'язувати задачі статистики та теорії ймовірності;
- проводити фінансові розрахунки;
- розв'язувати звичайні диференціальні рівняння та системи;
- розв'язувати диференціальні рівняння у частинних похідних;
- будувати та редагувати різні графіки.

Система MathCad має такі особливості:

```

-->P1+P2
ans =
      2   3
- 2 + x + x + 4x

-->P1*P2
ans =
      2   3   4   5
- 15 - 13x - 7x + 11x + 8x + 4x

-->P1/P2
ans =
      2
3 + 2x + x
-----
      3
- 5 - x + 4x

```

Для знаходження коренів полінома в Scilab призначається функція roots (p), де p - поліном, створений функцією poly і представляє собою ліву частину рівняння P (x) = 0.

Приклад. Знайти корені полінома $2x^4 - 8x^3 + 8x^2 - 1 = 0$.

```

-->f=poly([-1 0 8 -8 2], 'x', 'c')
f =
      2   3   4
- 1 + 8x - 8x + 2x

-->x=roots(f)
x =
      0.4588039
- 0.3065630
      1.5411961
      2.306563

```


2.4.15. Алгебраїчні рівняння

Алгебраїчним рівнянням називається будь-яке рівняння виду $P(x) = 0$, де $P(x)$ - многочлен ненульовий ступеня. Тобто будь-яке алгебраїчне рівняння можна представити у вигляді

$$a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-1}x + a_n = 0.$$

Тут $a_0 \neq 0$, $n \leq 1$, a_i - коефіцієнти алгебраїчного рівняння n -го ступеня. Наприклад, квадратне рівняння $a_0x^2 + a_1x + a_2 = 0$ - не що інше, як рівняння алгебри другого ступеня, кубічну - третього і так далі.

Ми виділили алгебраїчні рівняння в окремий клас з двох причин. По-перше, при роботі з такими рівняннями можна не замислюватися про область визначення. Аргумент x може приймати будь-яке дійсне значення. По-друге, поліноми мають стільки коренів, який порядок полінома, проте деякі з рішень можуть бути комплексними.

Рішення алгебраїчного рівняння в Scilab складається з двох етапів. Спочатку необхідно задати поліном $P(x)$, а потім знайти його коріння.

Завдання полінома в Scilab проводиться за допомогою функції `poly([a], 'x', 'c')`

Тут a - рядок, що містить коефіцієнти полінома, починаючи з a_0 (з вільного члена), $'x'$ показує, яка змінна використовується при запису полінома, $'c'$ - опція, що показує, що в рядку вказані саме коефіцієнти полінома.

Приклад. Задати поліноми $P_1(x) = x^2 + 2x + 3$ і $P_2(x) = 4x^3 - x + 5$.

```
-->P1=poly([3 2 1], 'x', 'c')
```

```
P1 =
```

```
2
```

```
3 + 2x + x
```

```
-->P2=poly([-5 -1 0 4], 'x', 'c')
```

```
P2 =
```

```
3
```

```
- 5 - x + 4x
```

Зверніть увагу, в другому випадку відсутня член з x^2 , тому при завданні полінома відповідні коефіцієнти в рядку дорівнюють нулю.

Зауважимо, що над поліномами можна виробляти різні алгебраїчні дії.

Приклад.

- математичні вирази в MathCad записуються в їх звичній формі, тобто чисельник знаходиться вгорі, а знаменник — унизу. Аналогічним способом записуються будь-які математичні позначення. Це особливо важливо під час аналізу економіко-математичних моделей, форма та зміст яких у цьому разі єдині;

- у середовищі MathCad процес створення «програми» йде паралельно з її налагодженням. Користувач, увівши в MathCad-документ новий вираз, може не тільки відразу підрахувати його числовий вираз при заданих значеннях змінних, а й побудувати графік або поверхню, швидкий погляд на які може безпомилково показати, де криється помилка, якщо вона була допущена у формулі або під час створення самої математичної моделі;

- пакет MathCad доповнено довідником, що стосується основних економіко-математичних і фізико-хімічних формул та констант, які можна автоматично переносити в документ без побоювання внести в них перекручування;

- у систему MathCad інтегровано засоби символічної математики, що дає змогу розв'язувати поставлені задачі не тільки числово, а й аналітично;

- систему MathCad оснащено засобами анімації, завдяки чому можна реалізувати створені моделі не тільки в статистиці (числа, таблиці, графіки), а й у динаміці.

2.3.2. Основні прийоми роботи з пакетом Mathcad

У найпростішому випадку робота з системою Mathcad зводиться до підготовки у вікні редагування завдання на розрахунки і встановлення форматів для їх результатів. Для цього використовуються різні засоби підготовки блоків.

Спілкування користувача з системою Mathcad відбувається на деякій проміжній математично орієнтованій мові візуального програмування. Багато математичних записів на цій мові вводяться просто виводом відповідних шаблонів, операторів і функцій. Ця мова знаходиться дуже близько до звичайної математичної мови опису обчислювальних завдань, що майже не вимагають програмування. Mathcad дуже простий у використанні і легкий у навчанні. Всюди використовується звичний засіб математичного запису. Якщо існує загальноприйнятий спосіб зображення рівняння, математичної операції або графіка, то Mathcad використовує його. Використовується принцип «Те, що ви бачите, це те що ви отримуєте» (What you see is what you get – WYSIWYG). Не існує ніякої скритої інформації, все показується на екрані. Результат виведення на друкування має такий же вигляд, як і на екрані дисплею.

Прості вирази набираються на клавіатурі з використанням стандартних клавіш. Для спеціальних операторів (інтегралів, диференціалів, матриць, тощо) передбачено спеціальні палітри. Mathcad має широкі графічні можливості. Зручна довідкова система: відмітивши покажчиком оператор, функцію або повідомлення про помилку та натиснувши [F1], можна відобразити на екрані

інформацію, що пояснює інформацію з довідкової системи. Довідка вміщує покрокові пояснення з конкретної теми та ілюструючи приклади.

2.3.3. Особливості інтерфейсу користувача

Під *інтерфейсом користувача* будемо розуміти сукупність засобів графічної оболонки MathCad, які забезпечують керування системою як із клавіатури, так і за допомогою миші. Інтерфейс MathCad створено так, що користувач, який має елементарні навички роботи з Windows-додатками, може відразу почати роботу з MathCad.

MathCad має стандартний інтерфейс системи Windows: вікно, рядок меню, панелі інструментів, робоча область, рядок стану, спадаюче, або контекстне меню, діалогові вікна (рис.2.44).

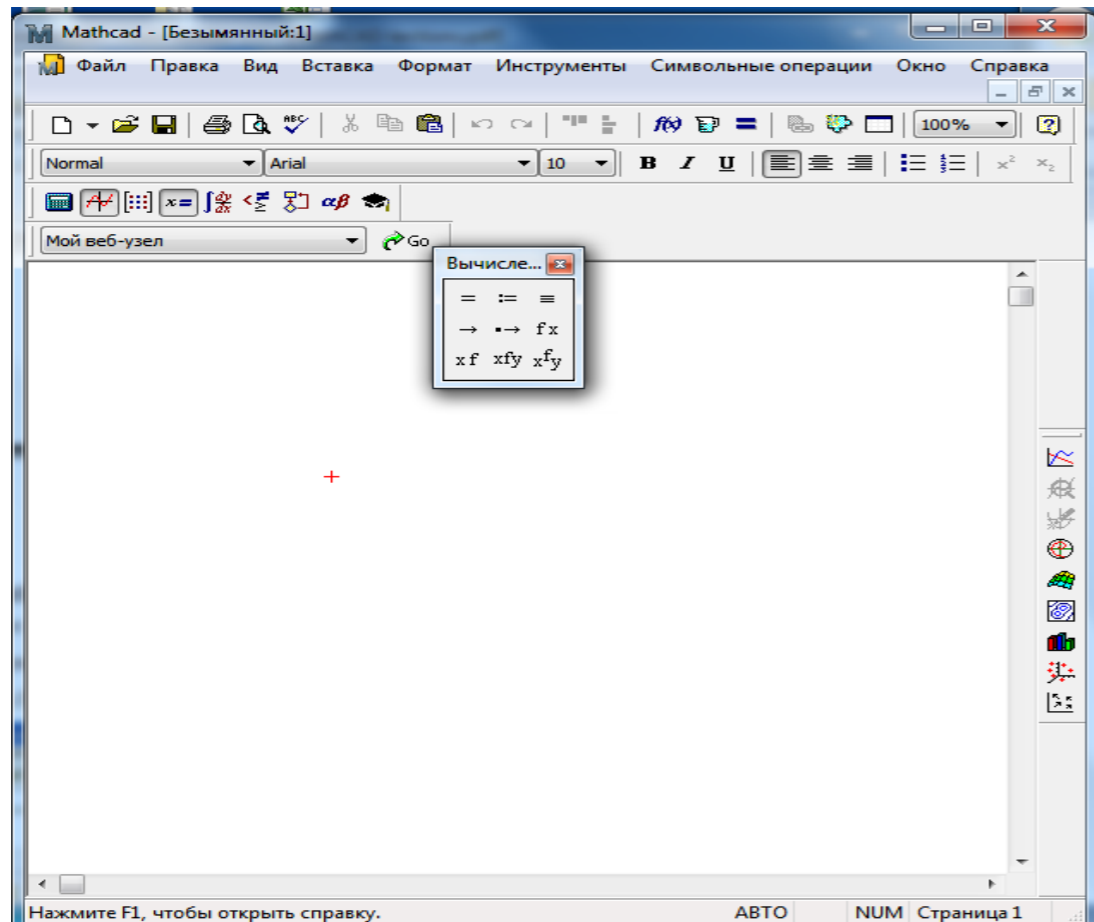


Рис.2.44 – Основне вікно MathCad

Коли запускається програма, відображаються панелі інструментів Стандартная, Форматирование і Математика.

Панель Математика призначена для виведення на екран ще дев'яти панелей (рис.2.45), за допомогою яких відбувається вставка математичних операцій у документи.

Про термінологію

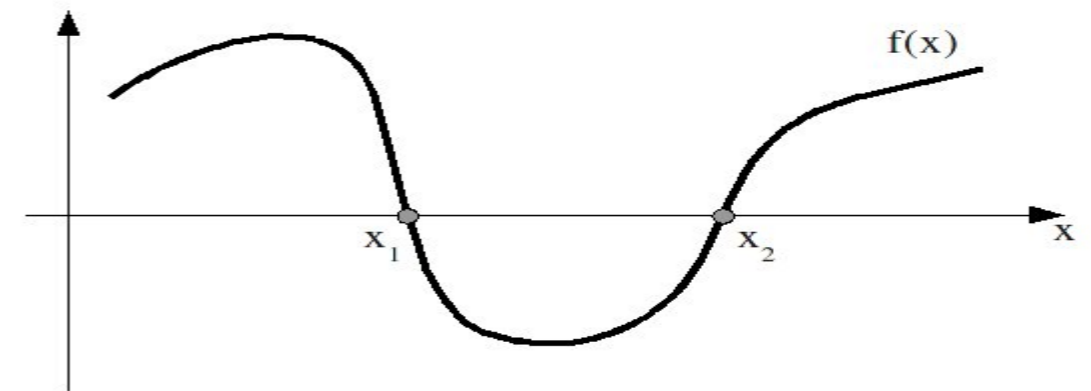
У загальному вигляді нелінійне рівняння записується в такий спосіб:

$$f(x) = 0.$$

Вирішити це рівняння означає: знайти значення $x = x^*$, яке при підстановці в рівняння перетворює його в тотожність. Слід мати на увазі, що оскільки рішення наближене, точного тотожності не вийде, проте величина $f(x^*)$ буде дуже близька до нуля (чим ближче, тим точніше отримане рішення).

Значення x^* називається коренем рівняння. Рівняння може мати один, кілька коренів або зовсім їх не мати. В останньому випадку говорять, що рівняння не має розв'язків.

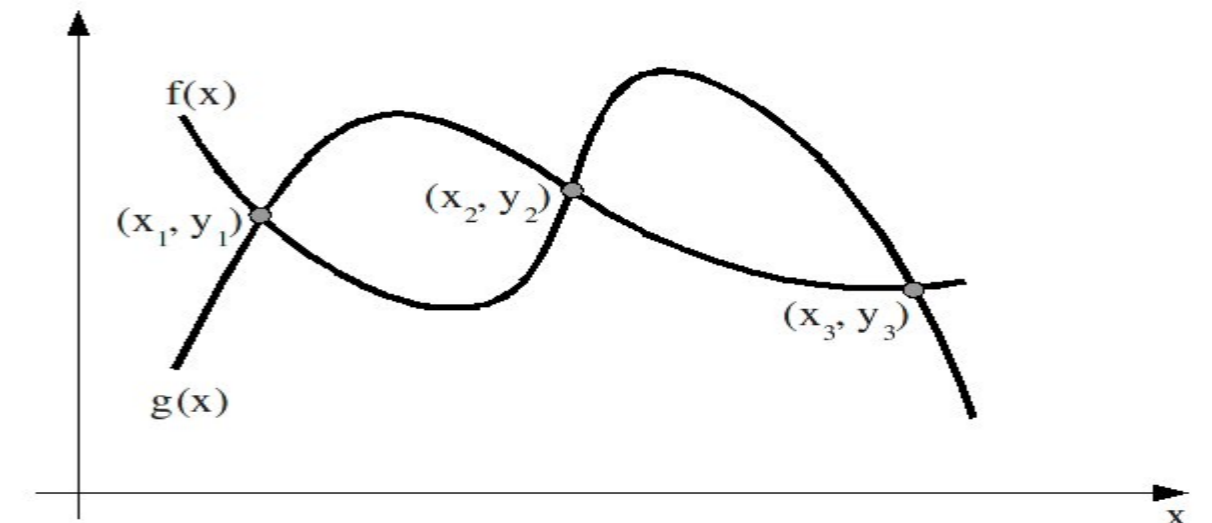
Для подальшого дуже важливий геометричний сенс розв'язання рівняння. Коріння рівняння - це координати точок перетину функції $f(x)$ з віссю x . Це очевидно, оскільки тільки в цих точках (за визначенням) значення функції дорівнює нулю.



Система двох рівнянь в загальному випадку має вигляд:

$$\begin{cases} f(x, y) = 0 \\ g(x, y) = 0 \end{cases}$$

Рішенням системи є пари (або пара) точок (x^*, y^*) , здатні перетворювати обидва рівняння в тотожності (точніше, майже тотожності в силу наближеності результату). Геометрично рішення системи являє собою координати точок перетину функцій $f(x, y)$ і $g(x, y)$.



```

-->deff('y=f(x)', 'y=(x+3)^2+2*x');
-->x0=[1 1.8 2.5];

-->numdiff(f,x0)
ans =

    10.    0.    0.
     0.   11.6   0.
     0.    0.   13.

-->//Proverka

-->deff('y1=f1(x)', 'y1=2*(x+3)+2');

-->f1(x0)
ans =

    10.    11.6    13.

```

2.4.13. Частинна похідна

Нарешті, команда numdiff легко справляється і з взяттям частинних похідних.

Приклад. Задана функція кількох змінних:

Обчислити $\partial f / \partial x_1$, $\partial f / \partial x_2$, $\partial f / \partial x_3$ в точці (1, 2, 3).

Код і результат:

```

-->deff('s=f(X)', 's=X(1)^2*X(2)+X(2)^2*X(3)+X(3)^2*X(1)');
-->X=[1 2 3];

-->numdiff(f,X)
ans =

    13.    13.    10.

```

2.4.14. Нелінійні рівняння та системи

Завдання розв'язання нелінійних рівнянь і їх систем є вельми нетривіальним і не має точного рішення. Тому можливості, що надаються для неї Scilab, засновані на наближених методах обчислень. Однак ми повинні застерегти вас від бездумного їх використання. Так як універсального алгоритму розв'язання не існує, це може призвести або до невдачі в пошуку рішення, або, що ще гірше, до отримання помилкового результату.

У зв'язку з цим ми детально розглянемо різні типи рівнянь і покажемо, яким саме чином слід діяти в тій чи іншій ситуації.

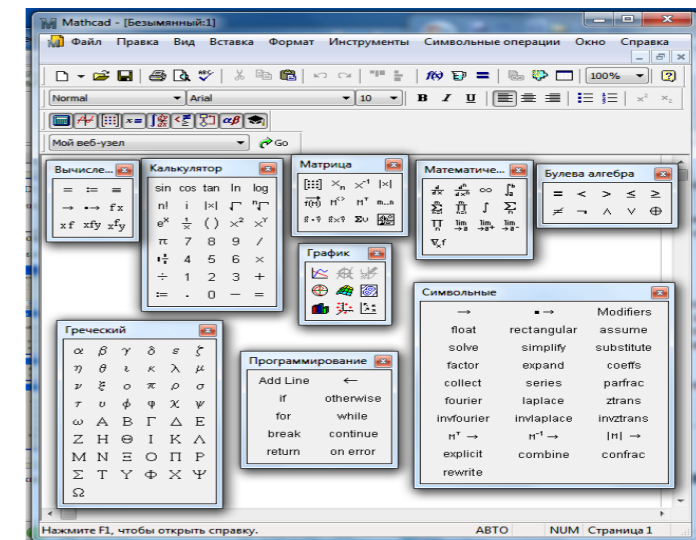


Рис.2.45 – Математична панель інструментів

2.3.4. Головне меню MathCad

Верхній рядок — рядок заголовка. Він відображає назву завантаженого або такого, що вводиться з клавіатури, документа. У лівій частині рядка міститься стандартна кнопка керування вікном, у правій - кнопка згортання вікна в піктограму і розгортання його на весь екран, а також кнопка закриття системи MathCad. Головне меню MathCad складається з таких пунктів:

- Файл — робота з файлами, мережею Internet та електронною поштою;
- Правка — редагування документів;
- Вид — зміна засобів огляду і включення елементів інтерфейсу;
- Вставка — вставлення об'єктів та їхніх шаблонів (включаючи графіку);
- Формат — зміна формату (параметрів) об'єкта;
- Інструменти — керування процесом обчислень;
- Символьные операции — вибір операцій символічного процесора;
- Окно — керування вікнами системи;
- Справка — робота з довідковою базою даних про систему.

Кожний пункт головного меню містить підменю зі списком доступних і недоступних (але можливих надалі) операцій (команд). Доступні в даний момент операції подаються чітким шрифтом, а недоступні — шрифтом із характерним затіненням.

2.3.5. Використання інструментальних і вбудованих панелей MathCad

Робота з документами MathCad не потребує обов'язкового використання опцій головного меню, тому що основні з них дублюються кнопками швидкого керування, їх можна виводити на екран або прибирати з нього за допомогою відповідних опцій позиції Вид головного меню Windows. Найчастіше використовуються дві такі панелі - панель стандартна (дублює найпоширеніші команди та операції) і панель форматування (забезпечує вибір типу, розміру шрифтів і засобу вирівнювання текстових коментарів). Третій рядок вікна

системи MathCad займає панель математика. Вона містить кілька груп кнопок керування з піктограмами, кожна з яких дублює одну з найважливіших операцій головного меню. Для набору відповідних математичних знаків використовуються вбудовані панелі, в оригіналі — палітри. Вони дають змогу виводити на екран заголовки — шаблони математичних знаків (наприклад, символи арифметичних операцій, матриць, знаків інтегралів, похідних і т.д.). Вбудовані панелі з'являються у вікні редагування документів при активізації відповідних піктограм. Більшість кнопок на палітрах виводять звичайні та спеціальні математичні знаки й оператори переміщенням їхніх шаблонів у місці розташування курсору на документі. Потрібний лише точний опис алгоритму для розв'язання задачі на звичайній математичній мові.

У найпростішому випадку робота із системою MathCad зводиться до підготовки у вікні редагування завдання на обчислення і встановлення форматів для результатів, допускається запровадження формул і тексту в будь-якому місці робочого документа. Кожен математичний вираз або фрагмент тексту є блоком, під яким розуміється обмежена область у MathCad-документі. З кожним блоком можна працювати самостійно: переміщати, копіювати і т.д. При цьому MathCad створює невидимий прямокутник, що обмежує кожний із блоків. У MathCad-документі використовуються блоки трьох типів: обчислювальні (або математичні), текстові та графічні.

Спілкування користувача з системою MathCad відбувається на вхідній мові, що є проміжною, математично орієнтованою мовою візуального програмування. Математичні записи в цій мові вводяться просто виведенням шаблонів відповідних операторів і функцій. Вона настільки наближена до звичайної математичної мови опису обчислювальних задач, що практично не потребує їх програмування. Потрібний лише точний опис алгоритму розв'язання задачі.

Оператори — це спеціальні MathCad-символи, що вказують на виконання тих або інших операцій над даними - операндами.

Останні можуть бути подані константами або змінними — об'єктами з іменами, які містять дані певних типу і значення. Оператори вводяться як із клавіатури, так і за допомогою вбудованих панелей.

Функція — об'єкт вхідної мови, що має ім'я і параметри, які вказуються в круглих дужках.

Ім'я MathCad-функції ототожнюється з відповідною математичною функцією, наприклад $\sin(x)$. Характерною рисою функції є повернення значення (результату обчислення функції) у відповідь на звернення до неї. Оператори та функції використовуються для створення *математичних виразів* — формул, які можуть обчислюватися в числовому або символьному вигляді. Зі змінними пов'язано поняття *присвоювання* їм значень. Символ присвоювання позначається як «:=», його можна набрати з вікна «Калькулятор» (наприклад, $x := 11$) під час виконання числових операцій і як « \Rightarrow » (жирне значення знаку дорівнює, яке знаходиться у віконці «Булева алгебра» (наприклад, « $a = c$ ») під час роботи з символьним процесором (рис.2.46).

Тут f - підінтегральна функція, представлена в символьному вигляді (як рядок), x - змінна інтегрування, теж у вигляді символу, а a і b - межі інтегрування.

Приклад. обчислимо інтеграл

$$\int_0^{\pi} \cos(\sin(x)) dx$$

Код і результат:

```
-->S=integrate('cos(sin(x))','x',0,%pi)
S =
2.4039394
```

2.4.12. Обчислення похідної

Вельми поширена в математичному аналізі операція обчислення похідної функції в точці вирішується в Scilab за допомогою команди

```
numdiff(f, x)
```

де f - ім'я функції, x - точка, в якій обчислюється похідна.

Приклад. Обчислити похідну функції $f(x) = (x + 3)^2 + 2x$ в точці $x = 2$.

Код і результат:

```
-->def f('y=f(x)', 'y=(x+3)^2+2*x');
-->pt=numdiff(f,2)
pt =
12.
```

Перевірка. Похідна функції обчислюється аналітично і знаходиться її значення в шуканій точці, яка повинна співпасти з знайденим вище.

```
-->x=2; f1=2*(x+3)+2
f1 =
12.
```

При обчисленні функції однієї змінної відразу в декількох точках результатом роботи команди numdiff буде матриця, на головній діагоналі якої розташовані шукані значення.

Приклад. Обчислити похідну функції $f(x) = (x + 3)^2 + 2x$ в точках $\{1; 1,8; 2,5\}$.

Код і результат:

Точність результату

Як визначити, наскільки точною є відповідь? Ніяк. Якщо, звичайно, невідомо справжнє значення (а якщо відомо, навіщо його шукати?). Але навіть інтуїтивно зрозуміло, що чим більше точок задіяно в розрахунку, тим ближче до істини буде результат.

Приклад. Розглянемо інтеграл

$$\int_1^6 \sqrt[3]{3x^2 - 2} dx$$

Обчислимо його значення за допомогою функції `intg`, а потім - за допомогою `inttrap`, розбивши відрізок інтегрування на 5, 10 і 20 частей. Код і результат:

```
-->def f('y=f(x)', 'y=(3*x^2-2)^(1/3)');
-->I=intg(1,6,f)
I =
    15.806989
-->//*****
-->a=1; b=6;
-->//*****
-->N=5; h=(b-a)/N; x=a:h:b; y=(3*x^2-2)^(1/3); I1=inttrap(x,y)
I1 =
    15.707151
-->//*****
-->N=10; h=(b-a)/N; x=a:h:b; y=(3*x^2-2)^(1/3); I2=inttrap(x,y)
I2 =
    15.779032
-->//*****
-->N=20; h=(b-a)/N; x=a:h:b; y=(3*x^2-2)^(1/3); I3=inttrap(x,y)
I3 =
    15.799595
```

Видно, що збільшення числа вузлових точок призводить до більш точному результату.

2.4.11. Інтегрування по квадратурі

Обчислювальний алгоритм квадратурних формул Ньютона - Котеса реалізований функцією `integrate(t, x, a, b)`.

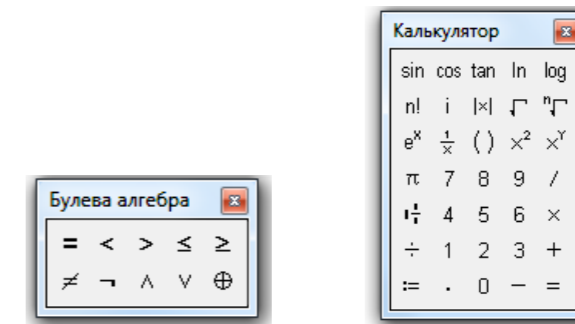


Рис.2.46 – Вікна вбудованих панелей

Характерним для MathCad є ідентичність подання документа, видимого на екрані, і його друку. Таку відповідність називають іноді принципом *WYSIWYG* («що бачу те й одержую»). Застосування шаблонів для введення математичних виразів, які мають звичайний вигляд, істотно полегшує рутинну роботу користувача і робить його інтерфейс ще більш дружнім (особливо це стосується початківців).

2.3.6. Робота з формульним редактором

Фактично система Mathcad інтегрує у собі три редактори: формульний, текстовий та графічний. За для пуску формульного редактора достатньо встановити курсор у довільному вільному місці вікна редагування та клацнути лівою клавішею. З'явиться візир у вигляді маленького червоного хрестика. Його можна переміщувати клавішами переміщення курсору. Візир не треба плутати з курсором миші – він живе своїм життям.

Візир вказує на місце з якого можна починати набирати формули – обчислювальні блоки. Підготовка обчислювальних блоків спрощується завдяки можливості введення шаблона того чи іншого оператора.

Математичні вирази в Mathcad набираються не так, як в текстовому процесорі. У документі доступні:

- латинські букви та цифри, для визначення змінних і чисел, набираються з клавіатури;
- грецькі вставляються за допомогою панелі інструментів Греческий (Greek), можна також ввести відповідну латинську букву та натиснути клавіші `<Ctrl>+<G>`, після цього латинська буква перетворюється на грецьку.
- оператори можуть бути вставлені або з математичних панелей інструментів, або відповідним набором клавіш.

Кожний математичний вираз, графік або текст утворюють блоки, вони мають прямокутні межі, які ми не бачимо. Для спостереження за межами блоків можна скористатися командою `Регион` у позиції `Вид` головного меню. При цьому блоки будуть представлені прямокутниками з білим фоном, решта фону основного меню буде сірим (рис.2.47). Перекриття блоків небажане. Для того, щоб позбутися цього можна використати команду `Вирівнять області` у позиції головного меню `Формат`.

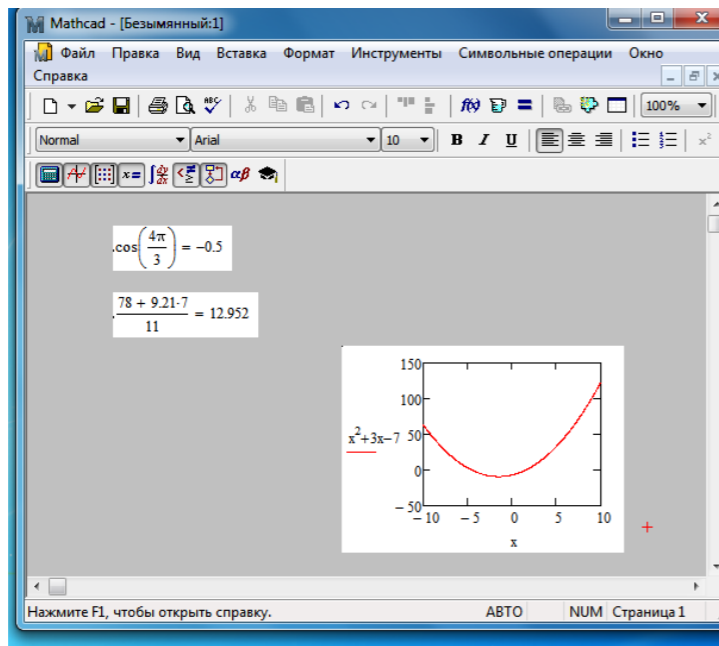


Рис. 2.47 – Зображення блоків

Блоки виконуються строго за чергою з переглядом їх зліва направо та зверху до низу. При виконанні документа блоки розглядаються в порядку, що вказаний вище, вони розпізнаються, компілюються та виконуються. При виконанні складних обчислень для припинення роботи можна натиснути клавішу Esc. Для поновлення роботи натиснути F9, або зробити активною піктограму із зображенням жирного знака рівності.

Математична панель інструментів включається (виключається) командою Вид ► Панель інструментів ► Математика (рис.2.48).

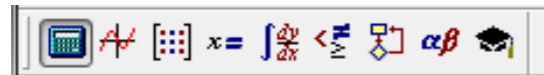



Рис. 2.48 – Математична панель інструментів

Перший значок математичної панелі  управляє панеллю Калькулятор (рис.3), яка призначена для введення тригонометричних, логарифмічних функцій та операторів обчислення кореня, модуля, піднесення до степеня. Ця панель дозволяє вставляти цифри та арифметичні оператори.

При введенні математичних виразів важливо розуміти роль курсору в цьому пакеті. Курсор виділяє символ, або групу символів, з якими буде проводитись операція. Для управління курсором використовують клавіші Insert, Пробел, ←, →

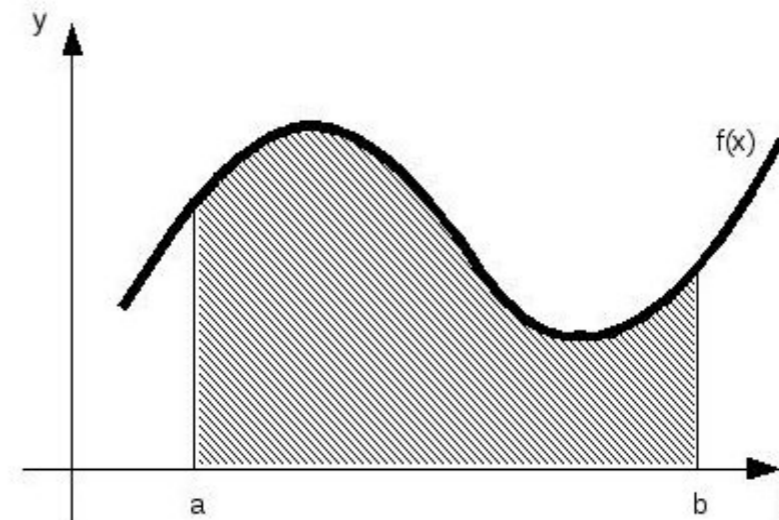
Оператор діє на останній введений операнд. Для розширення операнда використовують клавішу Пробел. Курсор може мати вигляд L (оператор дописується ліворуч від операнда), або J (оператор дописується праворуч від операнда). Переключення між двома курсорами здійснюється клавішею Insert.

Клавіша ← знімає об'єднання з групи символів, якщо курсор має форму J, а клавіша → з групи символів, коли курсор має форму L. Клавіша Delete

Геометрично інтеграл

$$\int_a^b f(x) dx$$

Представляє собою площу криволінійної трапеції, заштриховану на малюнку. Остання обмежується віссю x , лінією $f(x)$ і відрізками $x = a$ і $x = b$:



Явною формули для обчислення такої площі немає. Методи чисельного інтегрування полягають в тому, щоб представити цю область у вигляді сукупності простіших геометричних фігур, площа яких ми обчислювати вміємо. Яких саме - залежить від методу. У методі прямокутників шукана площа представляється, як неважко здогадатися, в вигляді набору прямокутників, в методі трапецій - трапецій, метод Сімпсона - парабол.

У Scilab чисельне інтегрування по найпоширенішому в силу оптимального поєднання простоти і точності методу трапецій реалізований за допомогою функції

`inttrap(x, y)`

Тут (x, y) - заданий набір точок.

Приклад. Обчислити інтеграл від функції, заданої у вигляді таблиці:

x	0,0000	0,3142	0,6283	0,3425	1,2566	1,5708	1,8850	2,1991	2,5133	2,8274	3,1416
y	0,0000	0,3090	0,5878	0,8090	0,9511	1,0000	0,9511	0,8090	0,5878	0,3090	0,0000

Код і результат:

```
-->x=[0 0.3142 0.6283 0.3425 1.2566 1.5708 1.8850 2.1991 2.5133 2.8274 3.1416];
-->y=[0 0.3090 0.5878 0.8090 0.9511 1.0000 0.9511 0.8090 0.5878 0.3090 0];
-->inttrap(x,y)
ans =
2.0925203
```

Інший приклад. Ідеально круглий гумову кульку (або мильний міхур) заповнюється постійним потоком повітря. Як змінюється з плином часу його радіус?

Подібні завдання були не під силу стародавнім математикам, вони розглядали лише вузький клас статичних явищ, коли невідомі величини не зазнавали змін, тобто залишалися незмінними.

Дослідження динамічних явищ стало можливим тільки при появі потужного апарату математичного аналізу.

2.4.9. Інтегрування функцій, заданої формулою

Найбільш універсальною командою обчислення інтеграла від функції $f(x)$

$$\int_a^b f(x) dx$$

є into. Синтаксис її в загальному вигляді виглядає наступним чином:

$$[I, \text{err}] = \text{intg}(a, b, f).$$

Тут a і b - числа, нижній і верхній межі інтегрування. Підінтегральна функція повинна бути зовнішньої, тобто заданої за допомогою операторів `deff` або `function`; в аргументах команди `intg` використовується тільки її ім'я.

Вихідними параметрами команди `intg` є безпосередньо значення інтеграла I і відносна похибка, з якою воно було знайдено. Останній параметр необов'язковий.

Приклад. Розглянемо інтеграл, значення якого відомо з довідників:

$$\int_0^{\pi/4} \frac{x dx}{\sin(x)\sqrt{\cos(x)}} = \frac{\pi}{2} \ln(1+\sqrt{2}) \approx 1,3844584$$

Код і результат:

```
-->deff('y=F(x)', 'y=x/(sin(x)*sqrt(cos(2*x)))');
-->[I,a]=intg(0,%pi/4,F)
a =
    1.9500-10
I =
    1.3844584
```

Таким чином, значення інтеграла - 1,3844584 знайдено з точністю $1,95 \cdot 10^{-10}$. Дуже хороший по точності результат!

2.4.10. Інтегрування таблично заданої функції

У ряді випадків, наприклад, при обробці експериментальних даних, вираз для функції невідомо - вона задається набором точок. У цьому випадку використовується апарат чисельного інтегрування: метод прямокутників, трапецій, Сімпсона, квадратур та інше.

видаляє символ після курсору, якщо він має форму \lrcorner , якщо курсор виглядає як \llcorner , то ця клавіша виділяє розширений операнд, або видаляє наявний символ. Клавіша `Backspace` видаляє символ попереду від курсору, якщо він має форму \llcorner , якщо курсор виглядає як \lrcorner , то ця клавіша виділяє розширений операнд, або видаляє символ. Виділений операнд може бути видалений за допомогою клавіш `Delete` та `Backspace`.

При створенні та редагуванні формул пробел призначений для виділення різних частин формули. Якщо раз за разом натискати клавішу пробілу у формулі, то лінії введення будуть циклічно змінювати своє положення (рис 2.49).

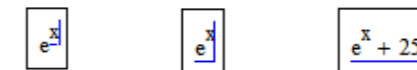


Рис.2.49 – Приклади зміни положення курсору

Коли процесор `Mathcad` з цієї або іншої причини не може обчислити вираз, він замість відповіді і видає повідомлення про помилку. Ім'я функції або змінної, що викликала помилку, у формулі відмічається червоним кольором. Якщо клацнути по такій формулі з'явиться повідомлення про тип помилки.

Якщо змінна або функція визначена, то для обчислення її значення треба набрати її ім'я та знак дорівнює. Праворуч з'явиться результат (рис.2.50)

```
f(x) :=  $\sqrt[5]{x}$ 
x := 1..4
f(x) =


|       |
|-------|
| 1     |
| 1.32  |
| 1.552 |
| 1.741 |


```

Рис. 2.50 – Обчислення значення функції

При введенні імен змінних та функцій з клавіатури, необхідно строго дотримуватись правила їхнього написання з урахуванням регістру (рядкової та великих букв). Математичний редактор `Mathcad` дуже чутливий до регістру символів, тобто x та X – це різні змінні. Якщо ім'я змінної або функції набрано невірно або без урахування регістру, виникає повідомлення про помилку. Вірне написання вбудованих функцій можна подивитися в меню `Справка`.

`Mathcad` відображає формули точно в тому вигляді, як їх друкують у книжках або пишуть у зошиті – по всі довжині екрану. Він підбирає розміри

для риски дробу, дужок та інших математичних символів щоб вони виглядали на екрані так, як їх зазвичай пишуть.

Режим обчислень встановлюється незалежно для кожного документа. Одночасно можуть бути відкриті декілька документів, які обчислюються у різних режимах. За замовчанням при створенні документа включено автоматичний режим обчислень.

Обчислення можна припинити, для цього достатньо натиснути клавішу <Esc>. З'явиться діалогове вікно, у якому треба підтвердити припинення обчислень (ОК). В цьому випадку вираз, який Mathcad не встиг обчислити, будуть помічені в документі червоним кольором. Перервані обчислення оновляться натисканням клавіші <F9>.

Mathcad дозволяє вимкнути обчислення будь-якої формули. Для того щоб не обчислювати вибрану формулу у документі достатньо клацнути правою кнопкою миші на формулі та вибрати в контекстному меню пункт *отключить вычисления* (відключити обчислення). Щоб обчислити відключену формулу треба клацнути правою кнопкою миші по формулі та вибрати у контекстному меню *включить вычисления* (включити обчислення).

Mathcad розуміє, яку операцію треба виконати першою. Так у наступному прикладі Mathcad «знає», що спочатку виконується ділення, а потім віднімання (рис. 2.51).

$$2.6 - \frac{7.6}{0.342} = -19.622$$

Рис.2.51 – Приклад виконання розрахунків

Вираз на екрані можна редагувати, для цього достатньо поставити у необхідному місці покажчик та замінити старі символи на нові. Після встановлення покажчика на вільне поле або інший вираз новий результат буде обчислюватись автоматично (рис. 2.52).

$$2.6 - \frac{7.6}{0.342} = -19.622$$

$$2.6 - \frac{7.6}{3.42} = 0.378$$

Рис.2.52 – Приклад редагування розрахунків

Розглянемо редагування математичного виразу на конкретному прикладі.

Приклад 1. Обчислити значення виразу $\frac{2.5 + 7.28}{0.64 - 0.036} \cdot \cos\left(\frac{\pi}{3}\right)$

На робочому листі по черзі набираємо: клацнути на будь-якому місці лівою кнопкою миші та вводимо за допомогою клавіатури строку:

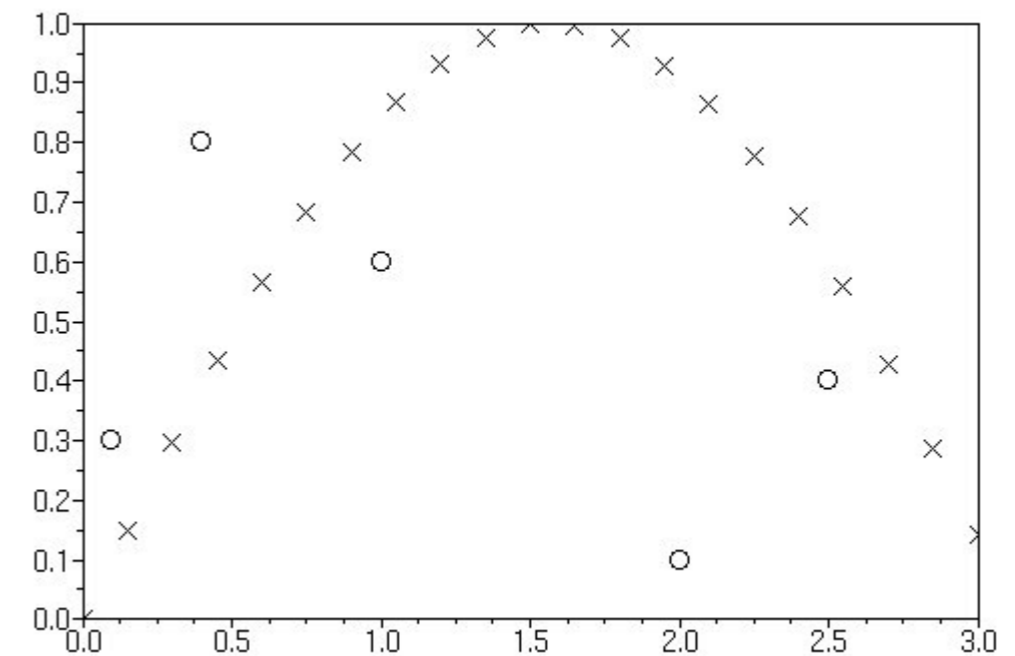
1. 2.5+7.28;
2. Символ /;

Масиви координат можуть бути обчислені або введені безпосередньо.

Приклад. Побудова точкового графіка функції $\sin(x)$ на проміжку $[0; 3]$ з кроком 0,15 хрестиками і нанести на цей же графік пари точок $(0; 0,3)$ $(0,2; 0,5)$ $(0,6; 1)$ $(1,2; 0,7)$ $(1,8; 0)$ кружками.

```
-->x=0:0.15:3; y=sin(x);
-->x1=[0.1, 0.4, 1, 2, 2.5]; y1=[0.3, 0.8, 0.6, 0.1, 0.4];
-->plot2d(x,y,-2)
-->plot2d(x1,y1,-9)
```

Результат

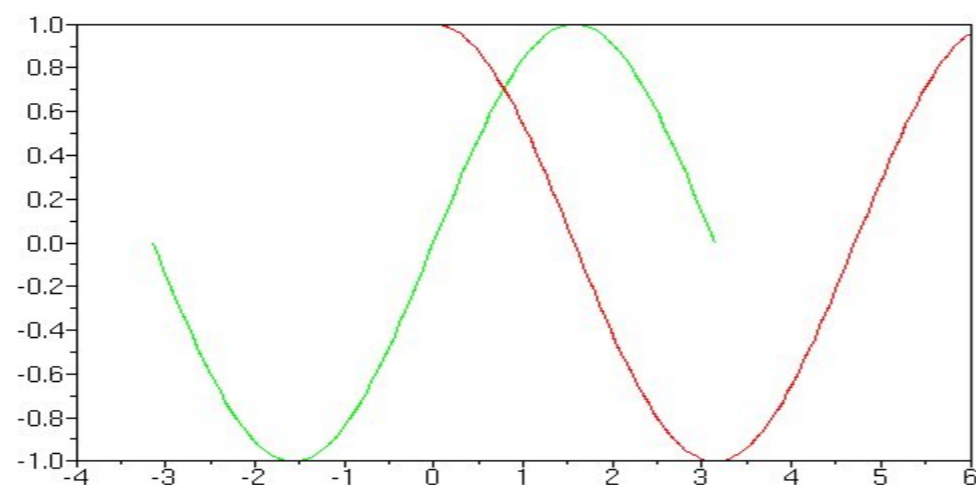


2.4.8. Задачі математичного аналізу

Аналізом називають область математики, засновану на апараті інтегрального та диференціального числення. Цей розділ математики не випадково отримав таку назву, оскільки саме відкриття обчислення нескінченно малих (в термінології Ньютона і Ейлера) дало можливість проводити строгий математичний аналіз природних явищ, що призвело, в свою чергу, до небувалого якісного стрибка в розвитку природничих наук.

Добре ілюструють ситуацію того часу парадокси філософа Зенона, найбільш відомим з яких є «Ахіллес і черепаха». Нагадаємо його суть.

Ахіллес і черепаха починають бігти по прямій лінії в одному напрямку, але черепаха в момент старту знаходиться попереду Ахіллеса, припустимо, на 10 метрів. Зенон доводив, що атлет ніколи не наздожене черепаху, оскільки поки він буде долати розділяє їх відстань, черепаха, нехай на чуть-чуть, але просунеться вперед. І так до нескінченності.



Принципових труднощів, звичайно, немає, однак наочності додаткові обчислення не додають. Тому ми запропонуємо інший, у багатьох випадках більш простий спосіб, а саме: кожен графік будується окремою функцією plot2dв поточному графічному сеансі (тобто не закриваючи графічного вікна). Код попереднього прикладу буде виглядати наступним чином:

```
-->x=-%pi:0.2:%pi; f1=sin(x);
-->y=0:0.2:6; f2=cos(y);
-->plot2d(x,f1,style=color('green'))
-->plot2d(y,f2,style=color(255,0,0))
```

Результат буде тим же, а код, як ми бачимо, простий і зрозумілий. Однак, слід мати на увазі, що при такому підході від опису ліній (легенди) доведеться відмовитися.

Точковий графік

Багато прикладних задач (наприклад, обробка і аналіз експериментальних даних) вимагають побудови точкового графіка. Для цих цілей існує наступна форма оператора plot2d:

```
plot2d (x, y, d)
```

де x, y - масиви координат точок, які потрібно відобразити на графіку, d – від’ємне число, що визначає тип маркера:

Значення	Опис	Значення	Опис	Значення	Опис
-0	точка	-5	незафарбований ромб	-10	зірочка
-1	плюс	-6	трикутник вершиною вгору	-11	квадрат
-2	хрестик	-7	трикутник вершиною вниз	-12	трикутник вершиною право
-3	плюс, вписаний в коло	-8	плюс, вписаний в ромб	-13	трикутник вершиною вліво
-4	зафарбований ромб	-9	коло	-14	п’ятикутна зірка

3. $0.64 - 0.036$;

4. $\frac{2.5 + 7.28}{0.64 - 0.036}$;

5. $\frac{2.5 + 7.28}{0.64 - 0.036} \cdot \cos\left(\frac{\pi}{3}\right)$

6. Виводимо результат, натискаючи знак =;

7. $\frac{2.5 + 7.28}{0.64 - 0.036} \cdot \cos\left(\frac{\pi}{3}\right) = 8.096$

Цей приклад демонструє особливості роботи MathCad. Він відображає формули точно в тому вигляді, як їх печатають у книгах, або пишуть на дошці – по всій площині екрану. MathCad підбирає розміри для дужок, рисок дроби та інших математичних символів, щоб вони виглядали на екрані так, як зазвичай пишуть у зошиті. MathCad розуміє яку операцію треба виконати першою.

Mathcad може встановлювати формат виведення чисел, тобто змінювати кількість десяткових знаків, змінювати експоненціальний вигляд представлення чисел на звичайний запис з десятковою крапкою. Для цього достатньо клацнути лівою кнопкою миші на таблиці, виділити її суцільною контурною лінією, вибрати пункт Результат з меню Формат та у діалоговому віні, що з’явиться, встановити необхідні параметри (рис.2.53).

$x := 1..7$

$g(x) := \sqrt[3]{x^2 + 3}$

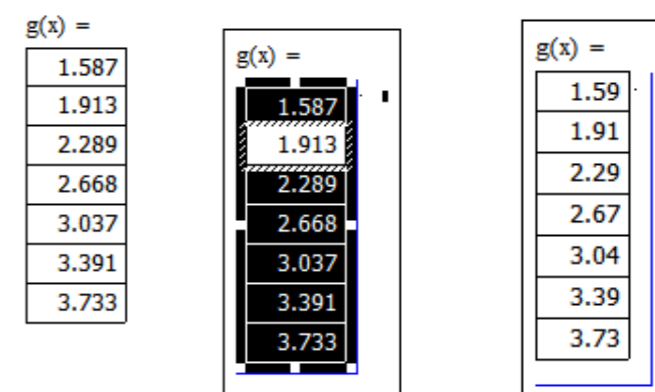


Рис.2.53 – Встановлення формату виведення чисел

При введенні математичних виразів можна використовувати математичні оператори та функції, які представлені в меню панелі інструментів Стандартная:

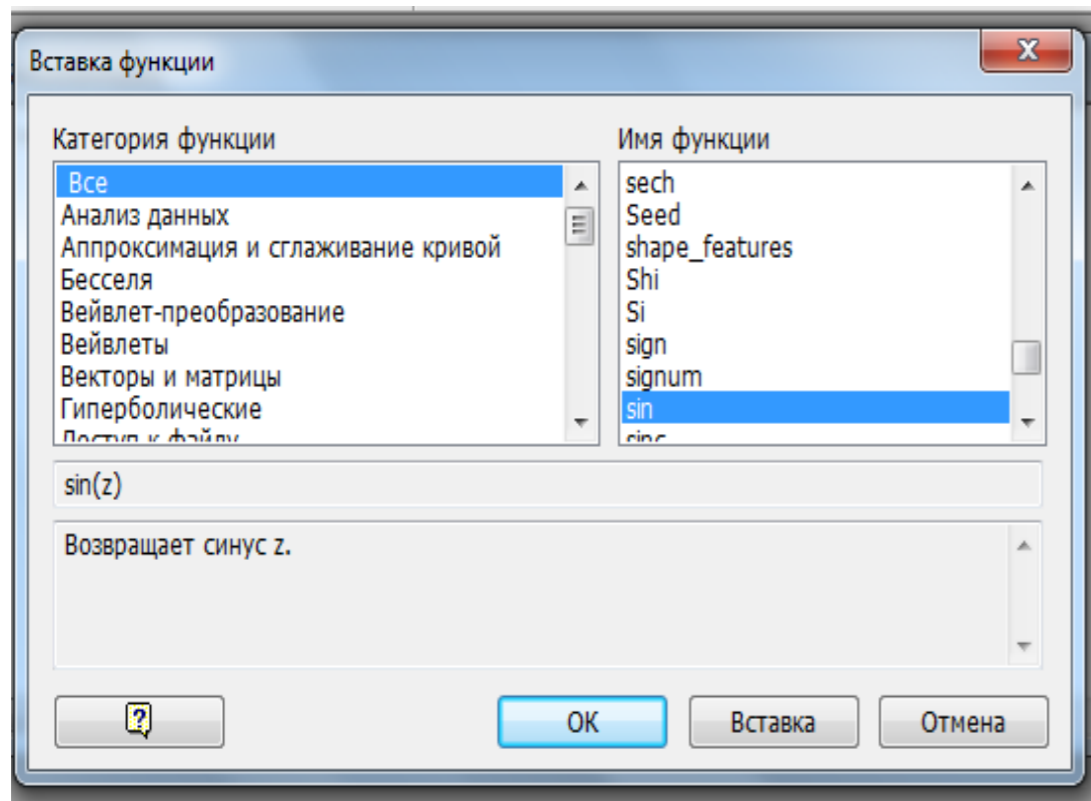


Рис.2.54 – Диалогове вікно для вибирання та вставлення функцій

2.3.7. Робота з текстовим редактором

Поряд з формульним редактором MathCad має доволі розвинуті засоби оформлення тексту. Призначення текстових областей у документі MathCad може бути як у якості простих коментаріїв, так і як елемент оформлення документів.

Текст можна розмістити у будь-якому вільному місці документу MathCad. Але коли користувач поміщує курсор введення у порожнє місце документу та просто вводить символи, MathCad за замовчанням інтерпретує їх як початок формули. Текстом ці символи стають автоматично якщо натиснути пробіл після першого слова.

Для того, щоб до початку введення вказати програмі, що потрібно створити не формульний, а текстовий регіон, достатньо, попередньо натиснути клавішу << (англ.). В результаті на місці курсору введення з'явиться новий текстовий регіон, який має характерне виділення (рис.2.55). Курсор приймає при цьому вигляд вертикальної лінії червоного кольору, яка називається лінією введення тексту.



Рис.2.55 – Текстовий регіон

Для побудови графіків декількох функцій в одному графічному вікні команда plot2d формально виглядає так само, як і в найпростішому випадку:

plot2d (X, Y)

Принципова різниця полягає в тому, що тут X і Y - матриці значень аргументів і значень функцій відповідно. Обов'язкова умова - кількість рядків у цих матриць має бути однакою. Це означає однакову кількість точок, в яких обчислюються значення всіх функцій. У разі загального інтервалу зробити це неважко.

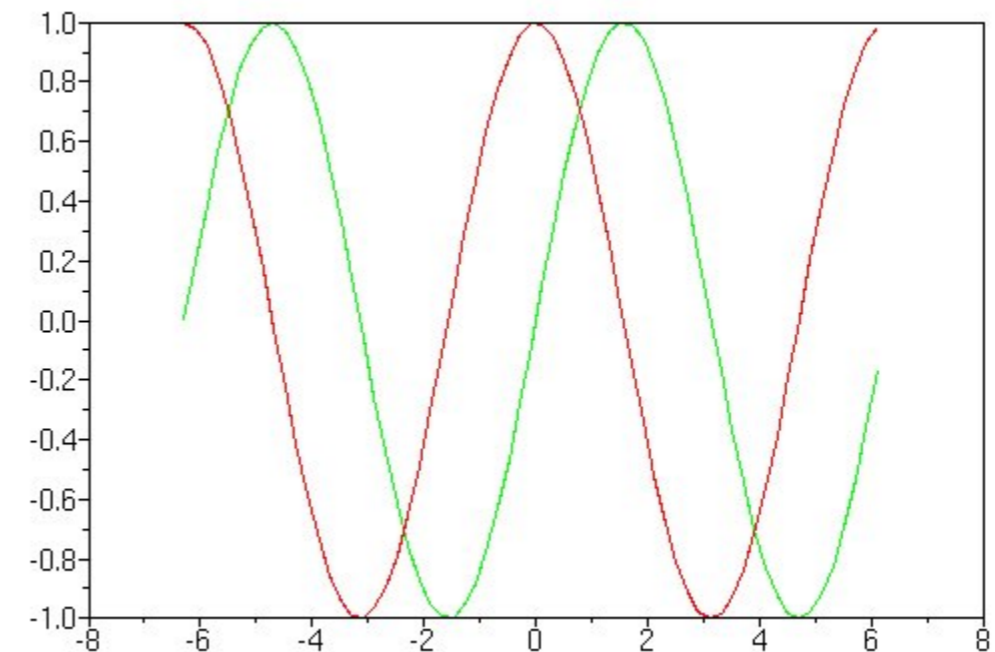
Приклад. Побудова графіка функцій sin (x) і cos (x) на проміжку [-2π; 2π].

```
-->x=-2*%pi:0.2:2*%pi; f1=sin(x);f2=cos(x);
```

```
-->plot2d(x,[f1;f2]',style=[color('green'),color(255,0,0)])
```

Тут ми об'єднали значення обох функцій в матрицю ([f1; f2]) і транспонували її, перетворивши рядки в стовпці. При цьому ми ще «пофарбували» графік в різні кольори різними способами.

Результат



А що робити, якщо інтервал побудови різний? Доведеться ставити кількість проміжків розбиття і обчислювати крок зміни для кожного аргументу.

Приклад. Побудова графіка функції sin (x) на проміжку [-π; π] і cos (x) - на [0; 6].

```
-->N=100; hx=2*%pi/N; hy=6/N;
```

```
-->x=-%pi:hx:%pi; f1=sin(x);
```

```
-->y=0:hy:6; f2=cos(y);
```

```
-->plot2d([x;y]',[f1;f2]',style=[color('green'),color(255,0,0)])
```

Результат

yellow	жовтий	blue	синій
cyan	голубий	black	чорний
red	червоний	brown	коричневий
green	зелений	gray	сірий

Повний список доступних кольорів (їх назва і RGB-код) наведено в таблиці кольорів (color_list) в довідці Scilab.

rect встановлює розмір вікна навколо графіка; значення цього ключа задається у вигляді вектора $rect = [xmin, ymin, xmax, ymax]$, де $xmin, xmax$ - інтервал відображення графічного вікна по осі x , $ymin, ymax$ - по y .

Axesflag задає наявність рамки навколо графіка; базисні значення цього параметра: 0 - немає рамки, 1 - зображення рамки, вісь у зліва, 2 - зображення рамки без підпису осей, 3 - зображення осей, вісь у справа, 5 - зображення осей у вигляді перехрестя на початку координат. За замовчуванням прийнято значення 1.

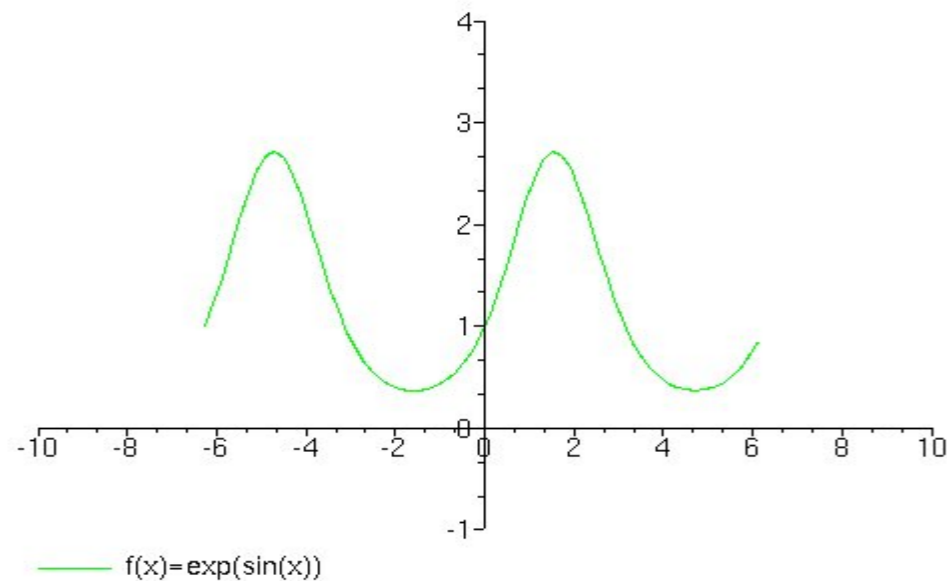
max встановлює число значень координатних осей; його значення є масивом $max = [nx, Ny]$, де $Nx (Ny)$ - число основних поділів осі $X (Y)$, які підписані на графіку, $px (py)$ - число проміжних поділок.

leg визначає легенду для кожного графіка; задається в вигляді $leg = 'leg1 @ leg2 @ leg3 ...'$, де $leg1$ - опис першого графіка, $leg2$ - другого і так далі.

Приклад. Побудова графіка функції $f(x) = \exp(\sin(x))$ на проміжку $[-2\pi; 2\pi]$ з наступним оформленням: лінія графіка зеленого кольору, розміри графічного вікна: від -10 до 10 по осі x і від -1 до 4 по осі y , осі повинні бути зображені у вигляді перехрестя, кількість основних міток: 11 по осі x і 6 по осі y , допоміжних - по 2 для обох осей, відобразити легенду.

```
-->x=-2*pi:0.2:2*pi; f=exp(sin(x));
-->plot2d(x,f,style=color('green'),rect=[-10,-1,10,4],...
-->axesflag=5,max=[2,11,2,6],leg='f(x)=exp(sin(x))')
```

Результат



Plot2d: кілька графіків

Створити текстовий регіон можна за допомогою команди (Вставка/Текстова область). Наразі можна просто вводити любий текст у текстовий регіон. Текстові коментарі можна вводити довільною мовою.

2.3.8. Масиви

Масивами називають впорядковані послідовності однотипних елементів (чисел, строк, символів). Всі елементи масиву мають однакове ім'я та відрізняються між собою індексами. У Mathcad умовно виділяють два типи масивів: ранжирувані змінні; вектори, матриці та тензори.

Ранжирувані змінні – це різновид векторів, елементи яких визначеним образом залежать від індексу. Найпростіший приклад ранжируваної змінної – це масив з числами, що лежать в деякому діапазоні з вибраним кроком. Для створення ранжируваної змінної з елементами 1,2,3,4,5,6 достатньо ввести ім'я змінної (g) та оператор присвоєння «:=», клацнути лівою кнопкою миші на панелі Матриця на символ та ввести границі ранжируваної змінної (рис.2.56).

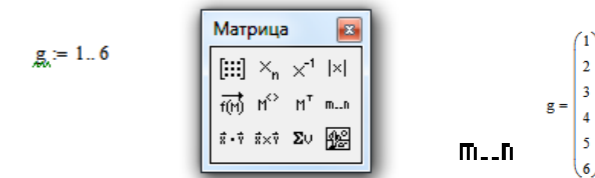


Рис.2.56 – Введення ранжируваної змінної

За замовчанням крок ранжирування дорівнює 1. Для створення ранжируваної змінної з кроком відмінним від 1, треба вказати другий елемент через кому (рис.2.57).

```
h := 1,3..11
```

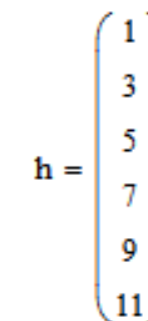


Рис.2.57 – Введення ранжируваної змінної з кроком відмінним від 1

Частіше за все ранжирувані змінні застосовуються для паралельних обчислень – циклів, ітераційних обчислень, або для присвоєння значень елементам інших масивів. При паралельних обчисленнях одна й та ж сама дія здійснюється паралельно над усіма елементами ранжируваної змінної (рис.2.58).

$$h := 1,3..11 \quad f(h) := 5h^3 - 2$$


$$h = \begin{pmatrix} 1 \\ 3 \\ 5 \\ 7 \\ 9 \\ 11 \end{pmatrix} \quad f(h) = \begin{pmatrix} 3 \\ 133 \\ 623 \\ 1.713 \times 10^3 \\ 3.643 \times 10^3 \\ 6.653 \times 10^3 \end{pmatrix} \quad \tan(f(h)) = \begin{pmatrix} -0.143 \\ 1.756 \\ 1.443 \\ 1.098 \\ -2.984 \\ -1.243 \end{pmatrix} +$$

Рис.2.58 – Ранжирувана змінна при паралельних обчисленнях

Якщо ранжирувана змінна використовується для формування іншого масиву, вона частіш за все виступає у ролі індексу, тому повинна змінюватись з кроком 1. За замовчанням нумерація індексів масивів починається з 0. Стартовий індекс масиву задається системною змінною ORIGIN, яка за замовчанням дорівнює 0. За необхідності нумерувати елементи масивів з 1, достатньо присвоїти цієї змінної значення 1. В цьому випадку значення нульового елемента вектора призводить до помилки бо його значення невизначене.

2.3.9. Побудова графіків в Mathcad

Графічні засоби обробки результатів обчислень Mathcad є одними з кращих серед аналогів. Двовимірні графіки можна будувати в декартових і полярних координатах. Для кожної кривої передбачене завдання кольору, товщини, стилю і т.д. Для графіка у цілому задають вигляд осей координат, надписів, масштаби та інші параметри. Mathcad підтримує наступні основні типи тривимірних графіків: графік у вигляді поверхні (заданої явно або параметрично), контурний, точковий, стовбчастий, у вигляді векторного поля. Їх можна відображати у трьох координатних системах: декартовій, сферичній і циліндричній. Крім кольору, товщини ліній і інших традиційних параметрів можна задати обертання, нахил, підсвічування, перспективу, рівень прозорості й інші спеціальні ефекти. Також графіки можна створювати, імпортуючи їх з файлів графічних форматів. За допомогою анімаційних функцій побудовані графіки можна "оживити".

Для побудови графіка потрібно вказати його тип. Перелік основних графіків можна побачити, якщо звернутися до пункту головного меню (Вставити ► Графік). Крім того можна створити графік за допомогою панелі інструментів **Графік**, для цього треба вибрати кнопку  на математичній панелі.

4 - нижній правий,

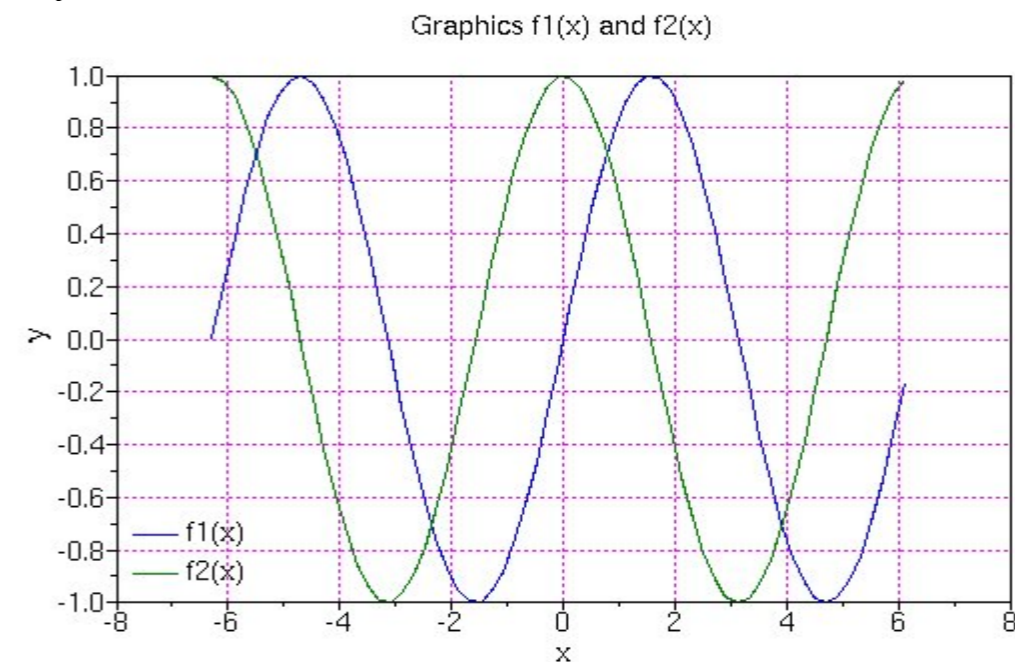
5 - визначається користувачем після зображення графіка.

frame задає рамку для опису: % t - опис укладено в рамку, % f- рамки немає.

Приклад. Побудуємо графік функцій $f(x) = \sin(x)$ і $g(x) = \cos(x)$ на проміжку $[-2\pi; 2\pi]$ і оформимо його, підписавши сам графік, осі і відобразивши опис ліній і сітку.

```
-->x:=-2*pi:0.2:2*pi; f1=sin(x); f2=cos(x);
-->plot(x,f1,x,f2)
-->xtitle('Graphics f1(x) and f2(x)', 'x', 'y')
-->xgrid(6)
-->legend('f1(x)', 'f2(x)', 3, %f)
```

Результат



Функція plot2d

Функція plot2d надає користувачеві більше можливостей для оформлення графіків. Найпростіше звернення до цієї функції здійснюється командою

plot2d (x, y)

За зовнішній вигляд графіка відповідають ключі команди, що задаються у вигляді key = value:

plot2d (x, y), key1 = value1, key2 = value2, key3 = value3, ...)

Можливі значення ключів:

style визначає колір графіка, який задається вказівкою RGB-коду style = color (r, g, b). Кожне із значень r, g або b відповідає «кількості» червоного, зеленого і синього і може змінюватися в межах від 0 до 255. Є більш проста можливість - вказати безпосередньо ім'я кольору: style = color ('ім'я кольору'). Наведемо найбільш поширені назви:

Якщо один з символів не вказано, його значення вибирається за замовчуванням (як правило, синя суцільна лінія), при невизначенні типу символу - він буде відсутній.

Приклад. Побудова трьох графіків:

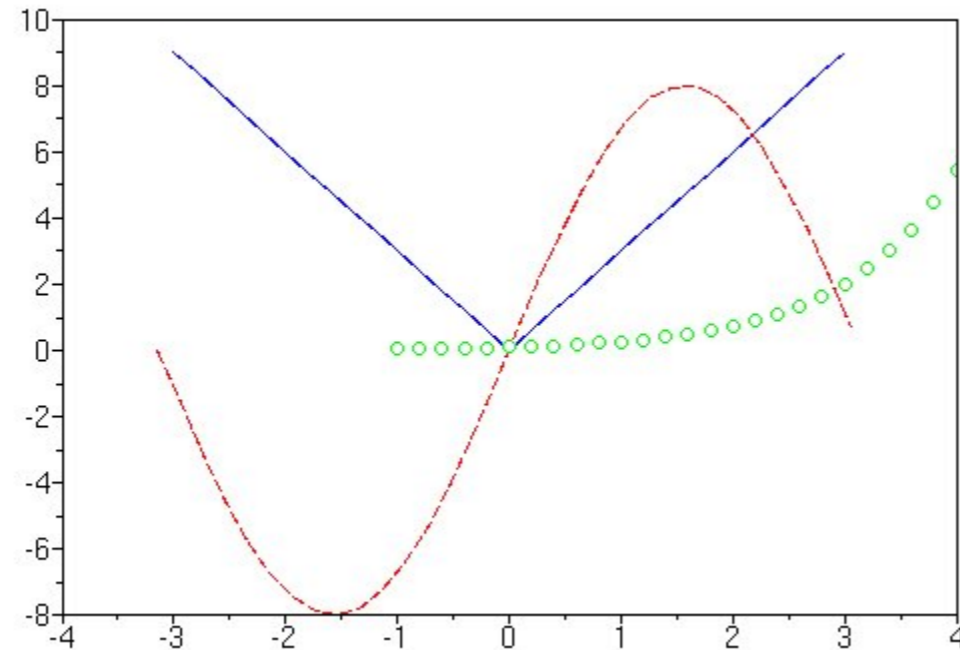
$f_1 = |3x|$ на інтервалі $[-3; 3]$ - суцільна синя лінія

$f_2 = 8\sin(x)$ на інтервалі $[-\pi; \pi]$ - пунктирна червона лінія

$f_3 = e^x / 10$ на інтервалі $[-1; 4]$ графік, намальований зеленими кружальцями.

```
-->x1=-3:0.2:3; f1=abs(3*x1);
-->x2=-%pi:0.2:%pi; f2=8*sin(x2);
-->x3=-1:0.2:4; f3=exp(x3)/10;
-->plot(x1,f1,'b-',x2,f2,'r--',x3,f3,'go')
```

Результат



Оформлення графіків

Для виведення назви графіка і осей використовується команда

`xtitle('plotname', 'xname', 'yname')`

де `plotname` - назва графіка, `xname` - назва осі X, `yname` - назву осі Y.

Щоб відобразити сітку на графіку, слід скористатися оператором `xgrid(n)`

Тут `n` - ціле число, яке відповідає за колір сітки.

Опис ліній (так звана легенда) здійснюється командою

`legend(line1, line2, ..., lineN, place, frame)`

`line1` і інші - назви графіків;

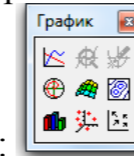
`place` - місце розташування опису:

1 - верхній правий кут графічного вікна,

2 - верхній лівий,

3 - нижній лівий,

Отримаємо панель, за допомогою якої можна побудувати наступні типи



графіків:

- **X-Y Plot** (декартовий графік ) - двох вимірний графік у декартовій системі координат;
- **Polar Plot** (полярний графік ) – графік у полярних координатах;
- **Surface Plot** (графік поверхні ) – трьохвимірний графік (поверхня);
- **Contour Plot** (контурний графік ) – контурний графік трьохвимірної поверхні;
- **3D Scatter Plot** (точковий графік ) – графік у вигляді точок у тривимірному просторі;
- **3D Bar Plot** (тривимірна гистограма ) – графік у вигляді сукупності стовпчиків у тривимірному просторі;
- **Vector Field Plot** (векторне поле ) – графік векторного поля на площині.

Взагалі виклик кожної з цих команд, крім останньої, призведе до вставлення у робочий документ спеціальної графічної області, яка називається шаблоном. Шаблон може мати одне або декілька полів введення. Заповнення цих полів відповідними даними здійснить побудову графіка. На рис.2.59 зображений робочий документ, що містить шаблони для створення графіка у декартовій системі координат та на рис. 2.60 - у просторі.

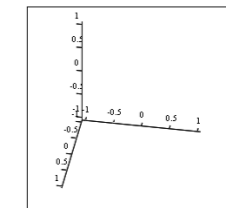
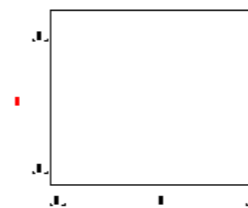


Рис.2.59 – Декартова система координат Рис.2.60 – Система координат у просторі

Шаблон для побудови графіка з'являється у позначеному місці, тобто там, де знаходиться курсор. За потреби його можна переміщувати за допомогою миші.

$$g(x) = \frac{x^4 - 5x^2 + 8}{x^3 + 7}$$

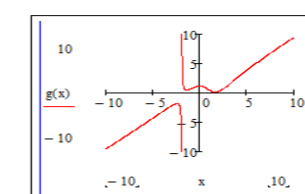


Рис.2.61 – Приклад побудови графіка у декартовій системі координат

Графічні області, як і інші об'єкти, виділяються, поміщують у буфер обміну, копіюють, переміщують або видаляють.

2.3.10. Побудова графіків у декартовій системі координат

Для побудови графічного об'єкту достатньо:

- підготувати початкові дані;
- вказати місце розміщення графічного об'єкту за допомогою курсору;
- з панелі інструментів вибрати потрібний графік;
- у місце заповнювач ввести імена змінних або функції та їхні аргументи;
- клацнути лівою кнопкою миші за межами графічного об'єкта.

В одній системі координат можна побудувати декілька різних залежностей. Для побудови графіків треба у місце заповнювач осі ординат через кому ввести ім'я наступної функції (рис.2.62).

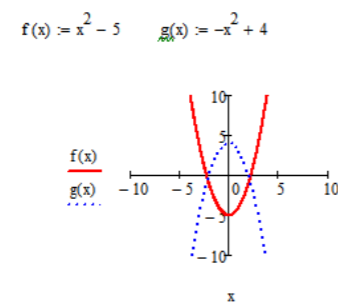


Рис.2.62 – Графік двох функцій від одного аргументу

В Mathcad форматкування слугує для зміни зовнішнього вигляду графіків та включає:

- зміну діапазону осей. Якщо графік створюється вперше, до діапазон осей вибирається автоматично. Для зміни діапазону достатньо клацнути лівою кнопкою миші по графіку, поряд з кожною віссю з'являться два поля з межами діапазону які можна змінювати.

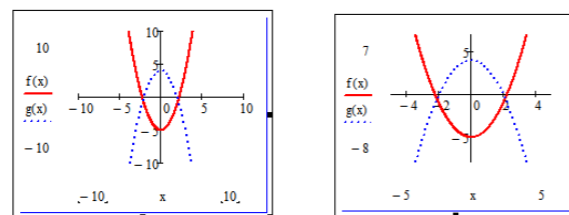


Рис. 2.63

Форматування графіка здійснюється за допомогою діалогового вікна, яке з'являється подвійним клацанням лівою кнопкою миші по графіку (рис.2.64).

Вигляд графіка можна змінювати, додавши при зверненні до функції plot крім основних, ще один аргумент - рядок, що складається з трьох символів, які будуть визначати колір лінії, тип символу, яким буде намальований графік і тип лінії. Звернення до функції plot буде виглядати так:

plot (x1, y1, string1, x2, y2, string2, ...)

Рядок string виглядає наступним чином

'Параметр1 параметр2 параметр3'.

Параметри пишуться один за іншим без роздільників.

Параметр1 визначає колір графіка:

Символ	Опис
y	жовтий
m	рожевий
c	голубий
r	червоний
g	зелений
b	синій
w	білий
k	чорний

Параметр 2 задає символ для малювання графіка:

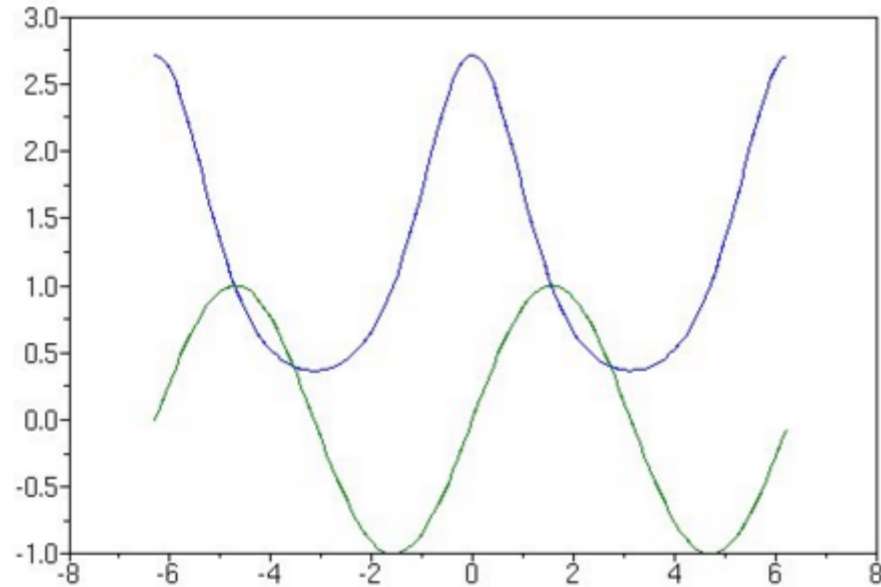
Символ	Опис
.	крапка
o	кружок
x	хрестик
+	знак «плюс»
*	зірочка
s	квадрат
d	Ромб
v	трикутник вершиною вниз
^	трикутник вершиною вгору
<	трикутник вершиною вліво
>	трикутник вершиною вправо
p	пятикутна зірка
h	шестикутна зірка

Параметр3 встановлює тип лінії графіка:

Символ	Опис
-	суцільна
:	штрихпунктирна
--	штрихова

```
-->x=-2*pi:0.1:2*pi; y=exp(cos(x)); z=sin(x);
-->plot(x,y,x,z)
```

Результат

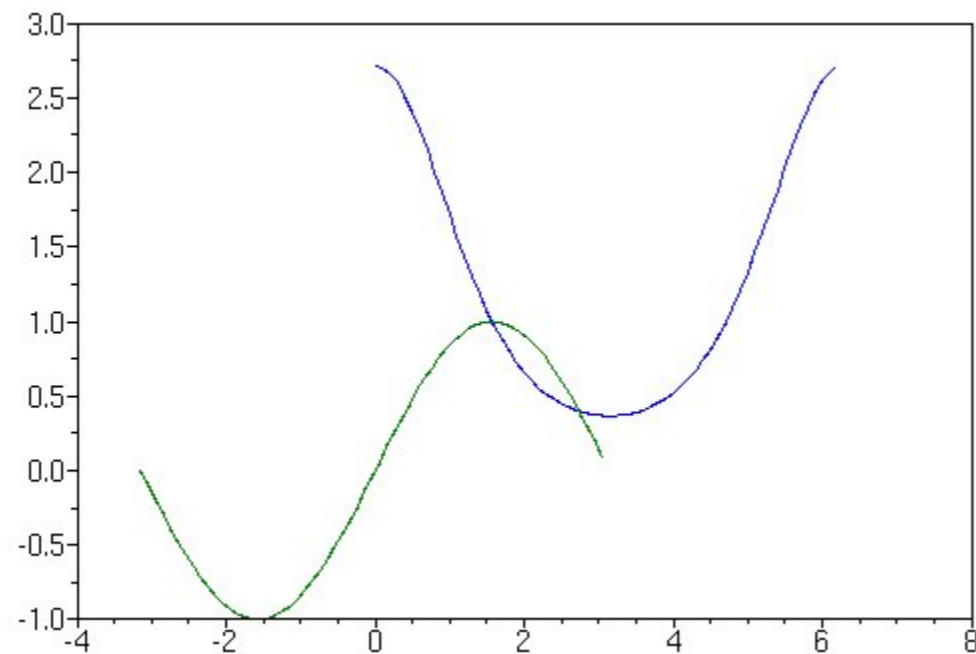


Випадок, коли потрібно побудувати графіки на різних інтервалах, набагато складніше.

Приклад. Побудова графіка функції $y = e^{\cos(x)}$ на інтервалі $[0; 2\pi]$ і $z = \sin(x)$ на проміжку $[-\pi; \pi]$.

```
-->x1=0:0.1:2*pi; y1=exp(cos(x1));
-->x2=-pi:0.1:pi; y2=sin(x2);
-->plot(x1,y1,x2,y2)
```

Результат



Видозміна графіків

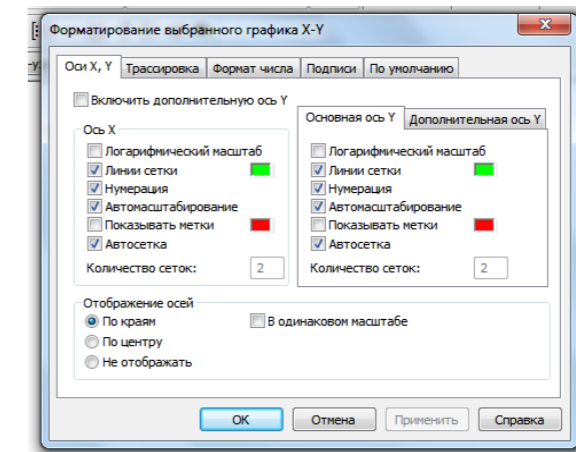
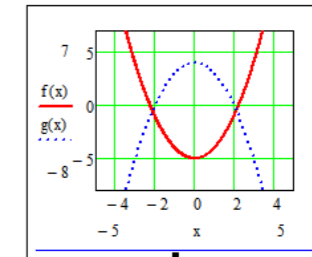


Рис.2.64 – Діалогове вікно форматування графіка X-Y

- нанесення ліній сітки;



- зміна стилю, товщини та кольору ліній;
- завдання стилю осей – прямокутник (Boxed), перетин (Crossed), або відсутність осей (None);
- створення символу для позначення окремих точок даних;
- визначення типу представлення ряду даних, тощо.

За допомогою Mathcad можна будувати графіки функцій, які задано параметрично, наприклад необхідно побудувати графік функції:

$$x = \frac{3at}{1+t^3}, y = \frac{3at^2}{1+t^3}, \text{ якщо } -\infty < t < -1 \text{ та } -1 < t < +\infty.$$

Побудова графіка зображено на рис.2.65

$$a = \frac{1}{2} \quad x(t) = \frac{3a \cdot t}{1+t^3} \quad y(t) = \frac{3a \cdot t^2}{1+t^3}$$

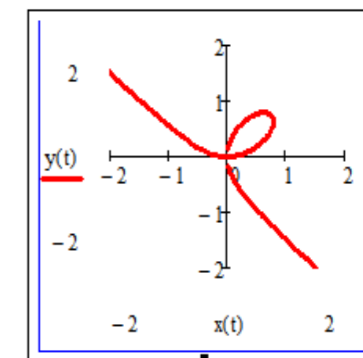



Рис.2.65 – Графік функції, заданої параметрично

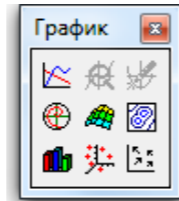
2.3.11. Трьохвимірні графіки

Графік поверхні (трьохвимірний або 3-D графік) – це графік, положення точки в якому визначається значеннями трьох координат. Функція в просторі задається:

- формулою (або декількома формулами), причому функція може бути явною та неявною;
- таблицею;
- просторовим графіком.

Для побудови графіка поверхні достатньо:

- визначити задану функцію;
- натиснути на кнопку  панелі **Графік**;
- ввести ім'я функції у шаблон побудови графіка;
- клацнути лівою кнопкою миші за межами графіка.



$$z(x,y) := 3y^2 - 5x^4$$

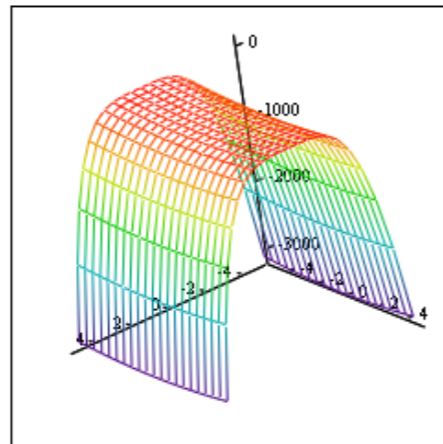

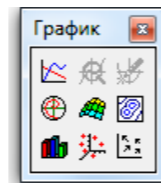


Рис.2.66 – Швидка побудова графіка

За допомогою Mathcad також можна побудувати графік функції, що задана матрицею, для цього потрібно:

- визначити матрицю;
- визвати шаблон для створення трьохвимірного графіка для цього натиснути на кнопку  на панелі;
- ввести ім'я матриці у поле введення шаблону;
- клацнути лівою кнопкою миші за межами графіка.



2.4.7. Побудова двовимірних графіків

Функція plot

Для побудови двовимірних графіків функції однієї змінної виду $y = f(x)$ в Scilab існує команда plot, звернення до якої в найпростішому випадку виглядає наступним чином:

plot (x, y)

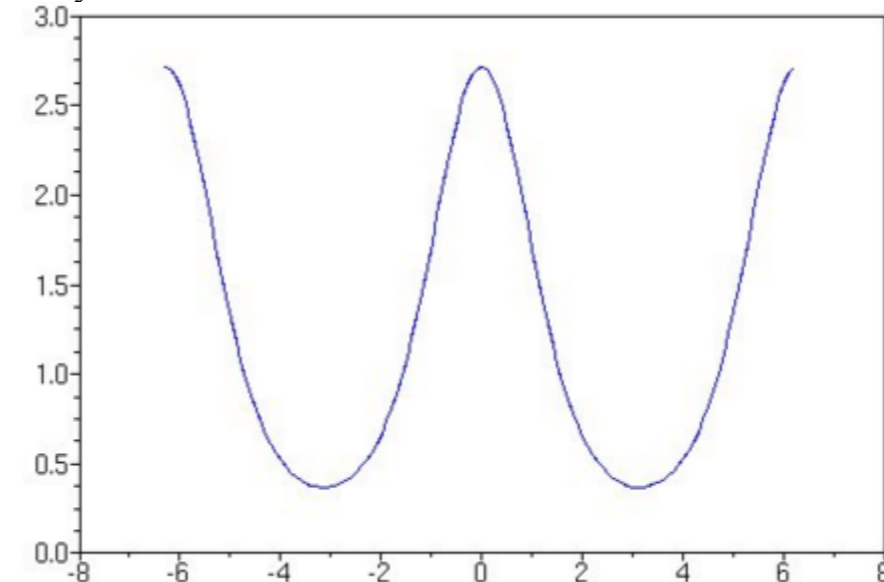
Тут x - список значень незалежної змінної, а y - список значень функції f в цих точках. Отже, для того, щоб побудувати графік в Scilab, необхідно створити два списки: значень змінної і значень функції в цих точках.

При цьому слід врахувати, що графік, побудований за допомогою функції plot, є ламаною лінією, що сполучає значення функції, обчислені в заданих точках (x, y). Значить, щоб отримати гладкий графік, необхідно обчислити значення функції у великій кількості точок.

Приклад. Побудова графіка функції $y = \exp(\cos(x))$ на проміжку $[-2\pi; 2\pi]$.

```
-->x=-2*pi:0.1:2*pi; y=exp(cos(x));
-->plot(x,y)
```

Результат



Plot: кілька графіків

При побудові декількох графіків досить просто поставити кілька функцій plot в поточному сеансі, тобто не закриваючи графічного вікна. Однак в цьому випадку графіки будуть одного кольору.

Зручніше (і правильніше) використовувати звернення до функції plot такого вигляду:

plot (x1, y1, x2, y2, x3, y3, ...)

Тут x1, y1 - список значень змінної і першої функції, x2, y2 - другої, x3, y3- третьої і так далі. Таким чином будуються графіки практично будь-якої кількості функцій.

Приклад. Побудова графіків функції $y = \exp(\cos(x))$ і $z = \sin(x)$ на проміжку $[-2\pi; 2\pi]$.


```

-->a=2; b=3;
-->if (a>0)&(b>0)
-->c=a+b;
-->elseif ((a>0)&(b<0))|((a<0)&(b>0))
-->c=a*b;
-->else
-->c=(a+b)/2;
-->end

-->c
c =
    5.

-->a=2; b=-3;
-->if (a>0)&(b>0)
-->c=a+b;
-->elseif ((a>0)&(b<0))|((a<0)&(b>0))
-->c=a*b;
-->else
-->c=(a+b)/2;
-->end

-->c
c =
    - 6.

-->a=-2; b=3;
-->if (a>0)&(b>0)
-->c=a+b;
-->elseif ((a>0)&(b<0))|((a<0)&(b>0))
-->c=a*b;
-->else
-->c=(a+b)/2;
-->end

-->c
c =
    - 6.

-->a=-2; b=-3;
-->if (a>0)&(b>0)
-->c=a+b;
-->elseif ((a>0)&(b<0))|((a<0)&(b>0))
-->c=a*b;
-->else
-->c=(a+b)/2;
-->end

-->c
c =
    - 2.5

```

Тут ми розглянули всі чотири можливих випадку для вибору змінних a і b. Зверніть увагу, як записується умова «одна з змінних a і b від'ємна».

$$M := \begin{pmatrix} 1 & 4 & 7 & -3 & 5 \\ 2 & -8 & 5 & 9 & 2 \\ 7 & 5 & 8 & 0 & 2 \\ -5 & 7 & 9 & 4 & 2 \end{pmatrix}$$

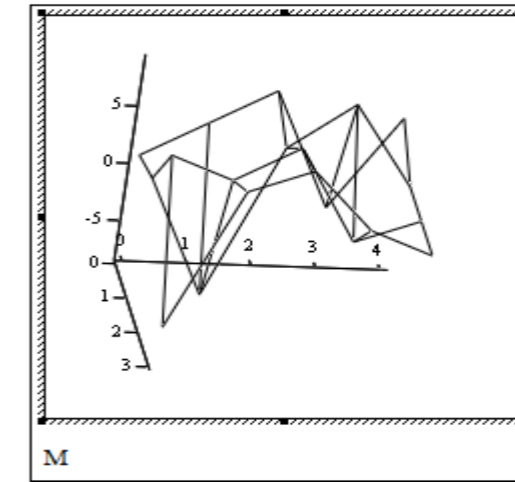


Рис.2.67 – Графік матриці

2.3.12. Форматування трьохвимірних графіків

Вікно форматування трьохвимірного графіка відкривається подвійним клацанням по області побудови графіка

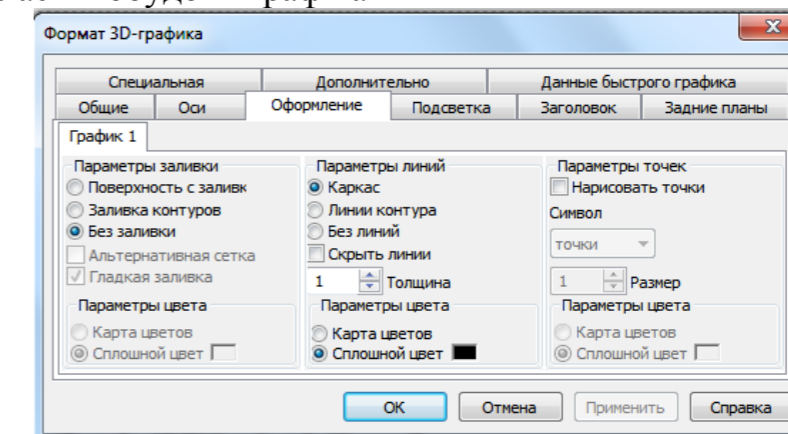


Рис.2.68 – Вікно форматування трьохвимірного графіка

Параметри, що впливають на зовнішній вигляд графіка згруповані за принципом дії на декількох вкладках :

- Общие – загальні параметри зображення;
- Оси – параметри координатних осей;
- Оформление – параметри відображення графіка (колір лінії, тип точок, які використовуються при побудові);
- Подсветка – умови освітлення та вибір схеми освітлення;
- Заголовки – створення заголовків та їхні параметри;
- Задние планы – додаткові параметри (перспектива, світові ефекти, тощо).

За допомогою групи параметрів *Общие* можна вказати:

- кут повороту (від 0° до 360°);
- кут нахилу (від 0° до 180°);
- кут обертання (від 0° до 360°);
- відносний розмір (одиниця за замовчанням).

Група *Оси* задає:

- осі по периметру;
- осі в куті;
- відсутність осей;
- однаковий масштаб за усіма осями.

Група *Оформление* дає змогу змінювати:

- параметри заливки;
- параметри ліній;
- параметри точок.

$$z(x,y) = 5x^3 + 4x \cdot y^2$$

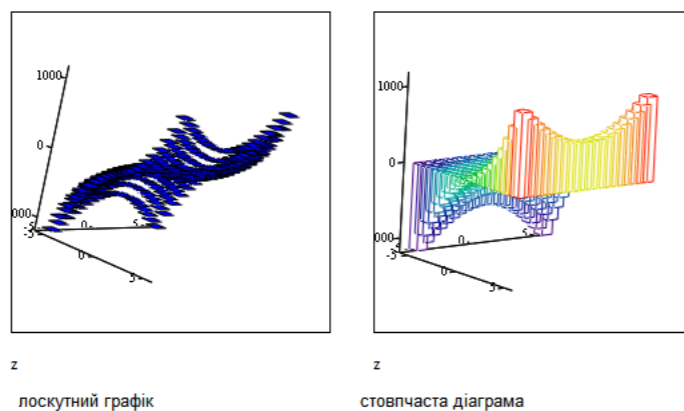


Рис.2.69 – Приклади створення різних діаграм

Іноді Mathcad успішно справляється з побудовою графіків функцій, що мають розриви. Але це робиться специфічно: розриви просто ігноруються та їхнє існування на графіку не відображається.

Наприклад при побудові графіка функції $z = \frac{xy}{\cos xy}$ повинні бути розриви у

точках коли знаменник перетворюється на нуль, тобто коли $xy = \frac{\pi}{2}$ маємо неперервний графік (рис.2.70).

```

-->a=2; b=3;
-->if (a>0)&(b>0)
-->c=a+b;
-->else
-->c=a*b;
-->end

-->c
c =
    5.

-->a=2; b=-3;
-->if (a>0)&(b>0)
-->c=a+b;
-->else
-->c=a*b;
-->end

-->c
c =
    - 6.

```

При вирішенні практичних завдань, як правило, недостатньо вибору виконання або невиконання однієї умови. В цьому випадку можна користуватися вкладеним оператором *if*, тобто в гілці *else* написати новий оператор *if*, але краще застосувати розширену форму оператора *if*:

```

if умова1
оператор1
elseif умова2
оператор2
elseif умова3
оператор3
...
elseif умова n
операторn
else
оператори
end

```

Алгоритм роботи цієї конструкції наступний: якщо умова1 істинна, то виконуються оператор1, інакше перевіряється умова2, якщо вона істинна, то виконуються оператор2, інакше перевіряється умова3 і т. Д. Нарешті, якщо жодна з умов по гілках *else* і *elseif* не виконується, то виконуються оператори по гілці *else*.

Приклад. Задані два числа *a* і *b*. Обчислити їх суму, якщо $a > 0$ і $b > 0$, добуток, якщо одне з них від'ємне і середнє значення, якщо від'ємні обидва.

оператори1
else
оператори2
end

Тут умова - логічне вираження, оператори1, оператори2 оператори мови Scilab або вбудовані функції. Оператор if працює по наступному алгоритму: якщо умова істинна, то виконуються оператори1, якщо помилково – оператори 2.

Приклад. Задані два числа a і b. Обчислити їх суму, якщо $a > b$ і добуток в іншому випадку.

```
-->a=2.4;b=6.3;

-->if a>b
-->c=a+b;
-->else
-->c=a*b;
-->end

-->c
c =
    15.12

-->a=7.4;b=6.3;

-->if a>b
-->c=a+b;
-->else
-->c=a*b;
-->end

-->c
c =
    13.7
```

Для того, щоб припинити видачу поточних результатів, ми завершили відповідні рядки крапкою з комою.

Для завдання умов використовуються оператори:

- & (And) - логічне «і»
- | (Or) - логічне «або»
- ~ (Not) - логічний заперечення
- <- Менше
- > - Більше
- == - рівність
- <> (~ =) - Не дорівнює
- >= - Більше або дорівнює
- <= - Менше або дорівнює

У випадку складних умов складові його елементарні умови полягають в дужки.

Приклад. Задані два числа a і b. Обчислити їх суму, якщо $a > 0$ і $b > 0$ одночасно і добуток в іншому випадку.

$$z(x,y) := \frac{x \cdot y}{\cos(x \cdot y)}$$

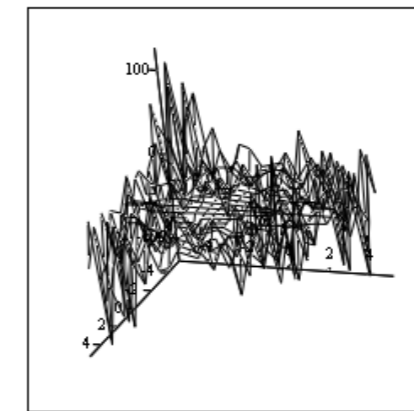





Рис.2.70 – Графік функції $z = \frac{xy}{\cos xy}$

Справа у тому, що Mathcad будує графіки дискретно, обчислюючи значення функції у визначених точках, а потім з'єднує їх прямими. Отже чим менше відстань між сусідніми обчисленими значеннями функції, тим більше вірогідність що при побудові графіка Mathcad стикнеться з точкою розриву та відмовиться її будувати, або як на вказаному прикладі буде її ігнорувати. Тому при побудові графіків розривних функцій треба враховувати область допустимих значень.

2.3.13. Робота з матрицями і векторами в Mathcad

Працювати з матрицями та векторами можна звернувшись до кнопки на панелі математичних інструментів  (рис.2.71). Ця панель містить наступні кнопки:

-  - визначення розміру матриці;
- \times_n - введення елемента масиву;
- \times^{-1} - обчислення матриці оберненої до даної;
- $|\times|$ - обчислення визначника матриці;
- $\vec{r}(\vec{r})$ - оператор векторизації;
- $M^{<>}$ - визначення стовпчика матриці;
- M^T - транспонування матриці;
- $m..n$ - визначення ранжируваної змінної;
- $\vec{r} \cdot \vec{v}$ - обчислення скалярного добутку векторів;
- $\vec{r} \times \vec{v}$ - обчислення векторного добутку векторів;
- ΣU - обчислення суми компонент вектора;
-  - візуалізація цифрової інформації, що зберігається в матриці:

обробка малюнків, які представлені матрицями.

Для виконання будь-якої операції над матрицями за допомогою кнопок панелі інструментів достатньо натиснути на відповідній кнопці й записати у полі введення ім'я матриці. Фрагмент робочого листа зображений на рис.2.71.

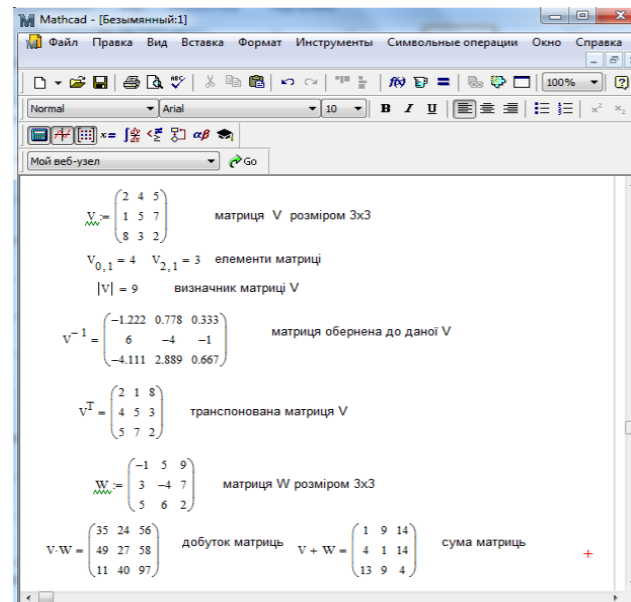


Рис.2.71 – Приклади операцій над матрицями

Робота з векторами також не повинна визивати труднощів. Фрагмент робочого листа на рис.2.72 містить дії з векторами, які відомі з математичного апарата векторних обчислень.

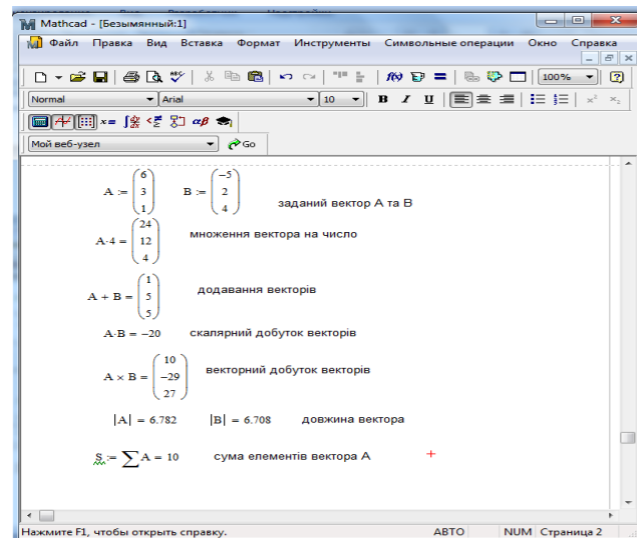


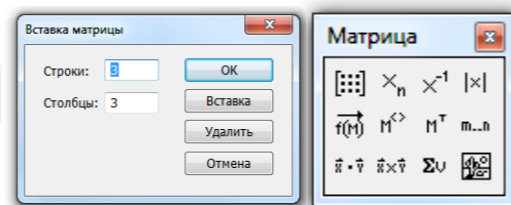
Рис.2.72 – Приклади операцій над векторами

Для створення вектора або матриці у Mathcad

достатньо на панелі інструментів вибрати 

та задати за допомогою діалогового вікна

кількість рядків та стовпчиків.



Тут аналогічно попередньому ім'я 1, ..., ім'я - список вихідних параметрів; ім.'я_функції - ім'я з яким ця функція буде викликатися, змінна_1, ..., змінна_M - вхідні параметри. Визначення вихідних змінних відбувається в тілі функції.

Приклад. Визначте функцію $f(x) = \sin(\cos(x))$ і обчисліть її значення в точках $\{-1,3; 2,1; 3,7\}$.

```
-->deff('f=fun(x)', 'f=sin(cos(x))')
-->[fun(-1.3) fun(2.1) fun(3.7)]
ans =
    0.2643200 - 0.4836727 - 0.7500251

-->function f=fun(x)
-->f=sin(cos(x))
-->endfunction
Warning :redefining function: fun

-->[fun(-1.3) fun(2.1) fun(3.7)]
ans =
    0.2643200 - 0.4836727 - 0.7500251
```

Приклад. Визначте функцію, в якій одночасно задаються $f(x, y) = \cos(x+y)$ і $g(x, y) = \sin(x-y)$ і обчисліть її значення в точці $(-2,5; 1,7)$.

```
-->deff('[f,g]=fnc(x,y)', 'f=cos(x+y);g=sin(x-y)')
-->[f1,g1]=fnc(-2.5,1.7)
g1 =
    0.8715758
f1 =
    0.6967067

-->function [f,g]=fnc2(x,y)
-->f=cos(x+y)
-->g=sin(x-y)
-->endfunction

-->[f2,g2]=fnc2(-2.5,1.7)
g2 =
    0.8715758
f2 =
    0.6967067
```

2.4.6. Логічні оператори

При вирішенні завдань досить часто доводиться мати справу з алгоритмом розгалуження - хід обчислення залежить від виконання або невиконання будь-якої умови: якщо ... то ... інакше (в іншому випадку). Тому ми вирішили розглянути цю конструкцію в даному розділі.

Одним з основних операторів, які реалізують розгалуження в більшості мов програмування, є умовний оператор if. Існує звичайна і розширена форми оператора if в Scilab. Звичайний if має вигляд:

if умова

$$a = 2\operatorname{tg}(x) + 7 \left(\frac{e^x}{\sin(x)+3} - 3.4 \ln(x) \right)$$

$$b = 4.6^x + 8.5 \left(\frac{e^x}{\sin(x)+3} - 3.4 \ln(x) \right) - 2$$

$$c = \frac{3x}{2} - 4.7 \left(\frac{e^x}{\sin(x)+3} - 3.4 \ln(x) + \sin(x) \right)$$

при різних значеннях змінної x .

Для спрощення обчислень і, до речі, скорочення часу розрахунку, що дуже істотно для великих проектів, має сенс ввести функцію

$$f(x) = \frac{e^x}{\sin(x)+3} - 3.4 \ln(x)$$

У цьому разі висловлення для змінних значно спрощуються:

$$a = 2\operatorname{tg}(x) + 7f(x)$$

$$b = 4.6^x + 8.5f(x) - 2$$

$$c = \frac{3x}{2} - 4.7(f(x) + \sin(x))$$

У Scilab функція $f(x)$ оформляється як функція користувача. Зробити це можна двома способами. Але перш, ніж їх розглядати, зупинимося на особливостях поняття функція в програмуванні.

В математиці все просто (або ж здається, що просто). Функція має ім'я і аргументи, а її значення є число.

У програмуванні аргументи функції називають вхідними параметрами. Функція також має ім'я, проте її значення повинно бути присвоєно змінній, яка визначається всередині функції. Ця змінна називається вихідним параметром, а її значення - повернутим значенням. Але найдивніше в тому, що в описі функції допустимо існування кількох вихідних параметрів, тобто функція може мати не одне, а кілька значень одночасно. Звичайно, в ряді випадків це дуже зручно!

Перейдемо до розгляду способів визначення функції в Scilab.

Спосіб 1. Застосування оператора `deff`, що має в загальному випадку вигляд:

`deff('ім'я1,...,ім'яN) = ім'я_функції (змінна_1,...,змінна_M)',`
`'ім'я1=вираз1;...;ім'яN = втразN'`

де ім'я1, ..., ім'я - список вихідних параметрів, тобто змінних, яким буде присвоєно кінцевий результат обчислень, ім'я_функції - ім'я з яким ця функція буде викликатися, змінні_1, ..., змінна_M - вхідні параметри, ім'я1 = втраз1; ...; ім'яN = втразN- визначення (обчислення) вихідних змінних.

Спосіб 2. Застосування конструкції `function`, синтаксис якої наступний:

`function[ім'я1,...,ім'яN]=ім'я_функції(змінна_1,...,змінна_M)`
тіло функції
`endfunction`

В отриманій матриці заповнюємо місцезаповнювачі необхідними даними:

$$A := \begin{pmatrix} 3 & 5 & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare \\ \blacksquare & \blacksquare & \blacksquare \end{pmatrix}$$

Матриці, вектори та ранжирувані змінні відображаються по-різному в залежності від того, як заданий масив (рис.2.73).

$$i := 0..5 \quad x_i := 0.2 \cdot i$$

$$x_i = \begin{pmatrix} 0 \\ 0.2 \\ 0.4 \\ 0.6 \\ 0.8 \\ 1 \end{pmatrix} + x = \begin{pmatrix} 0 \\ 0.2 \\ 0.4 \\ 0.6 \\ 0.8 \\ 1 \end{pmatrix}$$

Рис.2.73 – Завдання ранжируваної змінної та вектора

На відміну від звичного позначення членів матриці, коли перший індекс вказує на рядок, а другий на стовпчик і a_{11} , a_{12} означає, що ми маємо перший елемент у першому стовпчику та перший елемент у другому стовпчику, в Mathcad ці елементи будуть мати індекси a_{00} , a_{01} . Тобто підрахунок ведеться не від 1, а від 0. Між індексами у Mathcad ставимо крапку.

Для виконання будь-якої операції над матрицями достатньо клацнути на відповідну кнопку на панелі інструментів.

В Mathcad з матрицями можна виконувати символічні операції. Меню символічних операцій містить три функції – транспонування, обчислення оберненої матриці та визначника (рис.2.74).

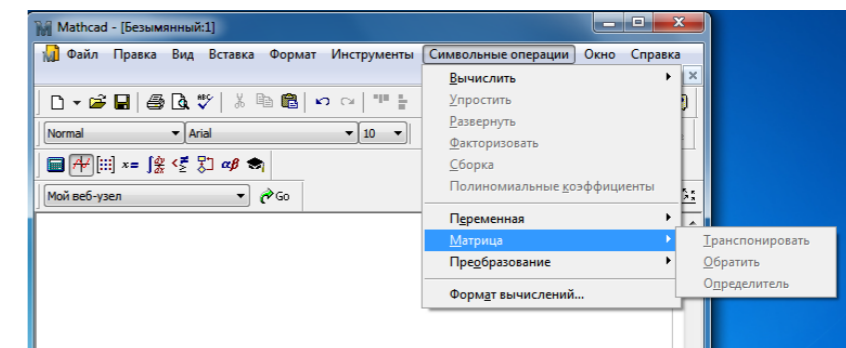


Рис.2.74 – Меню символічних операцій з матрицями.

Для здійснення операцій з використанням меню достатньо виділити матрицю, а потім вибрати потрібний пункт меню клацнувши мишею (рис.2.75).

$$\begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad \text{задана матриця}$$

$$\begin{pmatrix} a & c \\ b & d \end{pmatrix} \quad \text{транспонована матриця}$$

$$a \cdot d - b \cdot c \quad \text{визначник}$$

$$\begin{pmatrix} \frac{d}{a \cdot d - b \cdot c} & -\frac{c}{a \cdot d - b \cdot c} \\ -\frac{b}{a \cdot d - b \cdot c} & \frac{a}{a \cdot d - b \cdot c} \end{pmatrix} \quad \text{обернена матриця}$$

Рис.2.75 – Символьні операції над матрицею

Крім панелі інструментів для роботи з матрицями призначено спеціальні функції, які можна поділити на групи:

- функції визначення матриць та операції з блоками матриць;
- функції обчислення різних чисельних характеристик матриць;
- функції, що реалізують чисельні алгоритми розв'язування задач;
- функції сортування.

Розглянемо деякі з них:

2.3.14. Функції визначення матриць та операції з блоками матриць

• `matrix(m,n,f)` – створює та заповнює матрицю $A=\{a_{ij}\}$. Що має розмір $m \times n$, кожний елемент якої дорівнює значенню функції $f(i,j)$.

$$f(x,y) := \cos(x) + \sin(y)$$

$$A := \text{matrix}(4,3,f) = \begin{pmatrix} 1 & 1.841 & 1.909 \\ 0.54 & 1.382 & 1.45 \\ -0.416 & 0.425 & 0.493 \\ -0.99 & -0.149 & -0.081 \end{pmatrix}$$

• `diag(d)` – створює діагональну матрицю, елементи головної діагоналі якої зберігаються у векторі d .

$$i := 1..4 \quad d_i := 5 \cdot i \quad D := \text{diag}(d)$$

$$D = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 15 & 0 \\ 0 & 0 & 0 & 0 & 20 \end{pmatrix} \quad +$$

acos(x)	арккосинус числа x
atan(x)	арктангенс числа x
exp(x)	експонента числа x
log(x)	натуральний логарифм числа x
log10(x)	десятковий логарифм числа x
log2(x)	логарифм зв основою 2 числа x
sqrt(x)	квадратний корінь числа x
abs(x)	модуль числа x

При роботі з функціями необхідно пам'ятати, що функція без аргументу не існує! Як не дивно, цей очевидний факт часто ігнорується студентами і можна замість, скажімо, $\sin(x)$, побачити такий перл: $\sin \cdot x$. Програмою це сприймається саме так, як написано: змінна \sin помножити на змінну x . Природно, видається помилка!

Запам'ятайте! При записи будь-яких функцій після її імені обов'язкові дужки, в яких записаний аргумент.

Приклад. При $x = 1.34$ та $y = 0.45$ обчислити

$$z = \sqrt{\sin^2 x + e^y}$$

$$\text{-->} x=1.34; y=0.45;$$

$$\text{-->} z=\text{sqrt}(\sin(x)^2+\text{exp}(y))$$

$$z =$$

$$1.5861855$$

Зверніть увагу, як записати квадрат синуса!

2.4.5. Функції користувача

Навіщо створювати власні функції? Невже для розрахунків недостатньо наявних?

Ці питання цілком правомірні і не є надуманими або риторичними. Проте, в літературі вони чомусь, як правило, обходяться стороною.

Тут ми вкажемо лише одну, але цілком достатню причину: зручність. У багатьох випадках обчислення проводити набагато зручніше, простіше і наочніше, якщо ввести функцію користувача. Наведемо дуже простий приклад.

Необхідно обчислити величини:

2.4.3. Системні змінні

Якщо користувач не вказав в командному рядку ім'я змінної, результат буде присвоєно системній змінній з ім'ям ans (від англ. Answer - відповідь). Змінну ans можна використовувати в подальших обчисленнях, але її значення буде змінюватися кожного разу після виконання команди без оператора присвоєння.

У Scilab існують інші системні змінні (правильніше - системні константи). Їх запис починається символом %:

- % i - уявна одиниця ($\sqrt{-1}$);
- % pi - число $\pi = 3.141592653589793$;
- % e- число $e = 2.7182818$;
- % inf - машинний символ нескінченності (∞);
- % NaN - невизначений результат ($0/0, \infty / \infty$ і т. П.);
- % eps - умовний нуль % eps = 2.220E-16.

Приклад. Розрахунки з системними змінними

```
-->2*pi
ans =
    6.2831853

-->ans-4
ans =
    2.2831853

-->%e^ans
ans =
    9.8078718
```

2.4.4. Функції

Будь-математичний пакет або мову програмування оперує двома типами функцій: вбудованими та визначеними користувачем.

У загальному вигляді звернення до функції має вигляд:

ім'я_змінної = ім'я_функції (arg1 [, arg2, ...])

Тут ім'я_змінної - змінна, куди запишуться результати роботи функції; ім'я_функцій - ім'я вбудованої або введеної раніше користувачем функції; arg1, arg2 і т.д. - Аргументи функції.

Вбудовані функції

Функцій цього типу в Scilab цілком достатньо для проведення, аналізу та оформлення інженерних обчислень. Їх розгляд почнемо з математичних функцій, як найбільш поширених на практиці.

sin(x)	синус числа x
cos(x)	косинус числа x
tan(x)	тангенс числа x
cotg(x)	котангенс числа x
asin(x)	арксинус числа x

•identity(n) – створює одиничну матрицю порядку n.

$$E := \text{identity}(4) \quad E = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

1. Функції обчислення різних чисельних характеристик матриць:

- last(v) – номер останньої компоненти вектора;
- length(v) – кількість компонент вектора;
- rows(A) – кількість рядків в матриці;
- cols(A) – кількість стовпчиків в матриці;
- max(A) – найбільший елемент матриці (вектора);
- min(A) – найменший елемент матриці (вектора);
- mean(A) – середнє значення матриці (вектора);
- rank(A) – ранг матриці, тощо.

2. Функції, що реалізують чисельні алгоритми розв'язування задач:

- rref(A) – призводить матрицю A до ступінчастого вигляду шляхом елементарних операцій з її рядками;
- eigenvals(A) – обчислює власне значення квадратної матриці A;
- eigenvecs(A) – обчислює власні вектори квадратної матриці A;
- lsolve(A,b) – розв'язує систему лінійних алгебраїчних рівнянь $A \cdot x = b$, тощо.

3. Функції сортування:

- Sort(v) – сортує елементи вектора у порядку зростання їхнього значення;
- Reverse(v) – переставляє елементи вектора в оберненому порядку;
- Csort(A,n) – переставляє рядки матриці A таким чином, щоб відсортувався n-й стовпчик;
- Rsort(A,n) - переставляє стовпчики матриці A таким чином, щоб відсортувався n-й рядок.

2.3.15. Розв'язування систем лінійних рівнянь

Розв'язати систему лінійних рівнянь

$$2x + y - 5z + k = 8$$

$$x - 3y - 6k = 9$$

$$2y - z + 2k = -5$$

$$x + 4y - 7z + 6k = 0$$

Задамо матрицю коефіцієнтів A та вектор правих частин B:

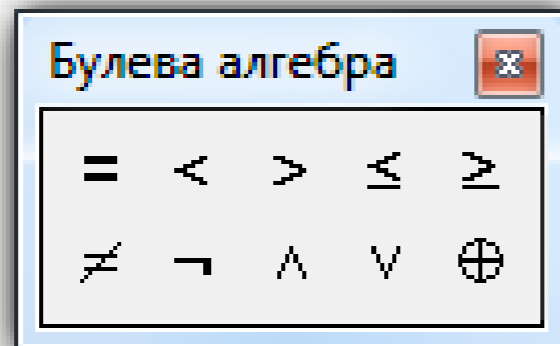
$$A := \begin{pmatrix} 2 & 1 & -5 & 1 \\ 1 & -3 & 0 & -6 \\ 0 & 2 & -1 & 2 \\ 1 & 4 & -7 & 6 \end{pmatrix} \quad B := \begin{pmatrix} 8 \\ 9 \\ -5 \\ 0 \end{pmatrix}$$

Для знаходження розв'язку системи використовуємо обчислювальну функцію `lsolve(A,B)`

$$\text{lsolve}(A,B) = \begin{pmatrix} 3 \\ -4 \\ -1 \\ 1 \end{pmatrix}$$

Можна розв'язати систему використовуючи метод розв'язуючого блоку `Given/Find`, для цього:

- задаємо початкові наближення для невідомих (довільні числа);
- вводимо ключове слово `Given`;
- переписуємо систему рівнянь у вигляді логічних рівностей, використовуючи знак `=`, який знаходиться на панелі:



- виводимо результат, використовуючи функцію `Find()`, причому число аргументів функції `Find()` повинно бути рівним кількості змінних (рис.2.76).

```
-->7.4E-3+0.085
ans =
    0.0924
-->0.0074+0.085
ans =
    0.0924
-->0.0074+0.085;
-->■
```

Зверніть увагу: якщо результат дії команди відобразити не потрібно, то її набір слід завершити символом «;»

2.4.2. Змінні

Користувач може визначити змінну, щоб використовувати її в подальших розрахунках. «Визначити змінну» означає поставити її ім'я і значення. З цією метою використовується оператор присвоєвання, в загальному випадку має вигляд:

`ім'я_змінної = значення_змінної`

У Scilab допускається використання в імені змінної безлічі символів, однак ми рекомендуємо використовувати тільки латинські букви і цифри, причому першою повинна бути буква. У цьому випадку Ви будете гарантовані від помилок, тим більше що максимальна довжина імені - 24 символи. Система розрізняє великі і малі літери в іменах змінних, тобто `ABC`, `abc`, `Abc` - це імена різних змінних.

Вираз у правій частині оператора присвоєвання може бути числом, арифметичним виразом, символьним виразом або рядком символів. В останньому випадку вираз справа береться в лапки.

Приклад. Визначити змінні `a = 18,4`; `b = 3 * 4,2 - 1,6`; `c = 2a + b`; `d = Hello!`

```
-->a=18.4
a =
    18.4
-->b=3*4.2-1.6
b =
    11.
-->c=2*a+b
c =
    47.8
-->d='Hello!'
d =
    Hello!
```


Для виконання найпростіших арифметичних операцій Scilab використовує такі оператори:

- + Додавання;
- Віднімання;
- * Множення;
- / Ділення зліва направо;
- \ Ділення справа наліво;
- ^ Піднесення до степеня.

Щоб обчислити значення арифметичного виразу, необхідно ввести його в командний рядок і натиснути Enter.

При записі виразу слід пам'ятати про пріоритет виконання арифметичних операцій. У разі необхідності його зміни використовуйте круглі дужки.

Приклад. обчислити вираз

$$\frac{3 \cdot (5,9 + 4,1)}{7,8 + 2,1}$$

```

-->3*(5.9+4.1)/(7.8+2.1)
ans =
3.030303

```

Десяткова кома в Scilab кодується точкою.

Якщо обчислюєш вираз довгий і бажано перенести його запис на наступний рядок, то в кінці незавершеного рядку необхідно ввести три (або більше) точки. Після цього можна натиснути Enter і продовжувати набір решти на наступному рядку. Ось як можна було закодувати попередній вираз:

```

-->3*(5.9+4.1)/...
-->(7.8+2.1)
ans =
3.030303

```

Числові результати можуть бути представлені з фіксованою (наприклад, 4.12, 6.05, -17.5489) або з плаваючою (наприклад, -3.2E-6, -6.42E + 2) точкою. Числа в форматі з плаваючою точкою представлені в експоненційній формі $mE \pm p$, де m мантиса (ціле або дробове число з десятковою крапкою), p порядок (ціле число). Для того, щоб перевести число в експоненційної формі до звичайного поданням з фіксованою точкою, необхідно мантису помножити на десять в ступені порядку.

Наприклад,

$$-6.42E+2 = -6.42 \cdot 10^2 = -642$$

$$3.2E-6 = 3.2 \cdot 10^{-6} = 0.0000032$$

Приклад. Обчислити вираз $7,4 \cdot 10^{-3} + 0,085$

```

x := 3   y := 3   z := 3   k := 3
Given

2x + y - 5z + k = 8
x - 3y - 6k = 9
2y - z + 2k = -5
x + 4y - 7z + 6k = 0

otvet := Find(x,y,z,k)

```

$$\text{otvet} = \begin{pmatrix} 3 \\ -4 \\ -1 \\ 1 \end{pmatrix} +$$

Рис.2.76. Розв'язок системи рівнянь за допомогою блоку Given/Find.

2.3.16. Оператори в Mathcad

Крім найпростіших операторів (+, -, /, ^, $\sqrt{\quad}$) в Mathcad існують і інші групи:

- арифметичні оператори;
- обчислювальні оператори;
- логічні оператори та оператори відношення.

У доповнення до простих арифметичних операторів у Mathcad існують ще:

- ! (обчислення факторіалу);
- $\sqrt[n]{\quad}$ (обчислення кореня n-го степеня).

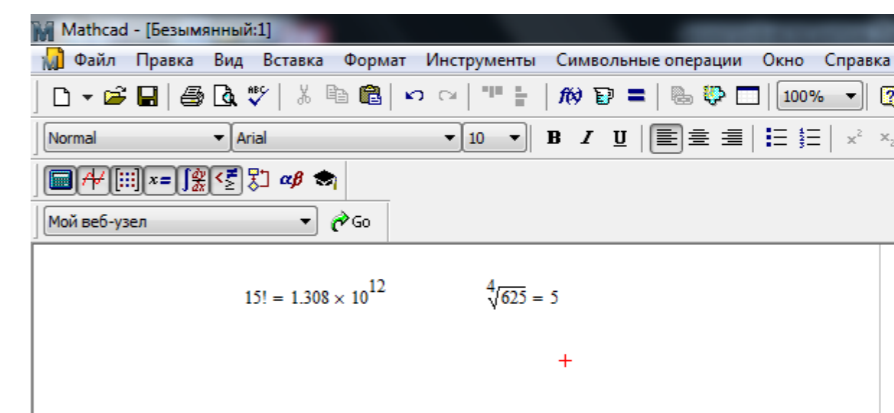



Рис.2.77 – Приклади використання арифметичних операторів

Всі обчислювальні оператори розміщені на панелі інструментів Математическая, яка управляється з панелі інструментів Математика за допомогою кнопки .

Тут розміщені наступні оператори:

- Оператор диференціювання ,

для отримання результату достатньо на панелі натиснути знак \rightarrow :

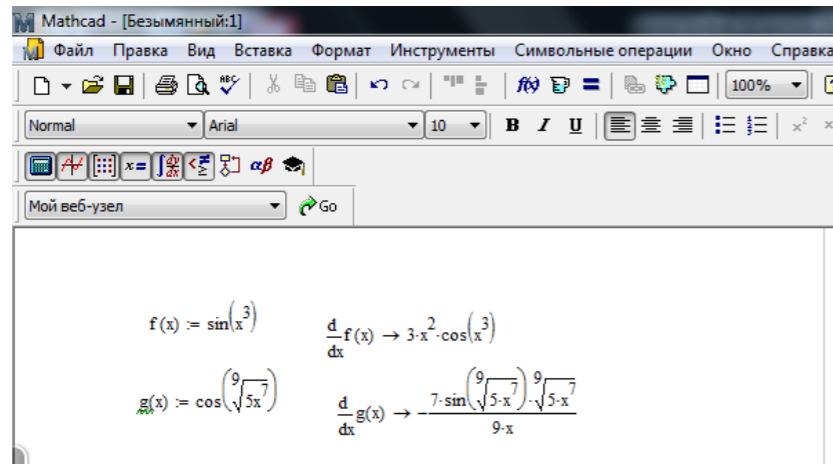




Рис.2.78 – Приклад використання оператора диференціювання

- Оператор знаходження похідної n-го порядку  на рис.35.

$$\frac{d^2}{dx^2} f(x) \rightarrow 6 \cdot x \cdot \cos(x^3) - 9 \cdot x^4 \cdot \sin(x^3)$$

$$\frac{d^3}{dx^3} g(x) \rightarrow \frac{98 \cdot \cos(\sqrt{5 \cdot x^7}) \cdot (\sqrt{5 \cdot x^7})^2}{243 \cdot x^3} - \frac{154 \cdot \sin(\sqrt{5 \cdot x^7}) \cdot \sqrt{5 \cdot x^7}}{729 \cdot x^3} + \frac{343 \cdot \sin(\sqrt{5 \cdot x^7}) \cdot (\sqrt{5 \cdot x^7})^3}{729 \cdot x^3}$$

Рис.2.79 – Приклад використання оператора знаходження похідної n-го порядку

• Оператор визначеного інтеграла  призначений для точного та наближеного обчислення визначеного інтегралу. Для наближеного (чисельного) обчислення після того як з'явиться оператора та введення підінтегрального виразу необхідно натиснути =, а для точного \rightarrow :

$$\int_{\frac{\pi}{6}}^{\frac{5\pi}{6}} \sin(5x) dx = -0.346$$

$$\int_{\frac{\pi}{6}}^{\frac{5\pi}{6}} \sin(5x) dx \rightarrow -\frac{\sqrt{3}}{5}$$

Рис.2.80 – Приклад використання оператора визначеного інтегралу.

сприймаються співробітниками, які не знайомі з цим програмним рішенням, допомагають підтримувати ефективну комунікацію у рамках проектів і передачу технічного знання усередині підприємства.

- Повний пакет додаткових функцій чисельної математики (включаючи новий пакет по плануванню експерименту) дозволяє швидко і якісно вирішити будь-яке обчислювальне завдання, економлячи час і засоби.

- Динамічна перевірка розмірних величин надає повну підтримку розмірних величин і одиниць їх виміру за усіма розрахунками, створеними в Mathcad Prime. Вектори і матриці Mathcad Prime тепер можуть містити величини різної розмірності, що підвищує ефективність процесу розробки продукції, допомагаючи уникати багатьох помилок.

- Інтуїтивно зрозумілий редактор рівнянь, працюючий в режимі повної відповідності (що бачите на екрані, то і отримуєте на папері), дозволяє користувачам описувати умови і рішення в природному математичному виді, зосереджуючись на обчисленнях, а не на роботі з документом.

- Mathcad Prime написаний на абсолютно новій програмній основі, має максимальну продуктивність, а також повну підтримку Windows 7 або 10 і останнього релізу MS Excel.

Слід зазначити, що Mathcad віднесено до пропріетарного програмного забезпечення. Згідно Вікіпедії — вільної енциклопедії: Пропріетарне програмне забезпечення, (від англ. *proprietary software*) — це програмне забезпечення, на яке зберігаються як немайнові, так і майнові авторські права. Отримавши або придбавши таке програмне забезпечення, користувач отримує обмежені права користування ним: може бути заборонено або закрито доступ до коду (вивчення), внесення змін, тиражування, розповсюдження та перепродаж. Крім того, програмне забезпечення вважається *власницьким*, якщо наявне хоча б одне з перелічених обмежень. Як видно, класифікація програмного забезпечення зараз досить складна, тому потребує подальшого дослідження.

2.4. Scilab

Всі символи, що містяться в командному рядку, програма сприймає як текст команди або назву змінних, але якщо перед ними поставити подвійний слеш (//), то наступний за ним текст буде сприйнятий як коментар.

Поки не натиснуто Enter, текст в командному рядку можна редагувати стандартним способом. Після натискання Enter команда переміщується в область перегляду доступ до її редагування закритий. Якщо є необхідність повернутися до цієї чи іншої даної команди, то за допомогою клавіші управління курсором \uparrow раніше виконані команди можуть бути повернуті в командний рядок. Після цього їх можна редагувати і виконувати повторно. При натисканні клавіші \downarrow команди з'являються в командному рядку в зворотному порядку.

2.4.1. Арифметичні обчислення

Він має наступну конструкцію

вираз1 on error вираз 2

де *вираз 2* дія, яка повинна виконатись у даному рядку програмного модуля, але якщо при цьому виникне помилка, то замість нього буде виконуватись *вираз1*.

2.3.25. Переваги використання Mathcad для студентів

- Не обмежені можливості для використання у навчальному процесі.
 - Автоматизація лабораторних робіт.
 - Курсове проектування.
 - Виконання РГР.
 - Створення власних творчих проектів.
 - Економія часу при виконанні завдань навчального плану.
 - Можливість зберігання розрахунків та власних коментарів на одному аркуші, потайні області забезпечать печатання необхідної частини документа.
 - Зберігання та документування своїх розрахунків, створення звітів. Все у одній програмі.
 - Перспективи на майбутнє – випускник, який володіє Mathcad більш конкурентоспроможний при працевлаштуванні.
- За допомогою Mathcad студенти можуть експериментувати, досліджувати та моделювати та одночасно документувати результати роботи.

2.3.26. Переваги використання Mathcad для викладачів

- Застосування у дослідницькій діяльності для підготовки та документування навчальних посібників, методичних розробок у форматах xps, pdf, html.
- Використання на лекціях для наочної презентації, дослідження конструкцій.
- Використання у навчальному процесі для проведення лабораторних робіт та аналізу експериментальних даних.

2.3.27. Mathcad Prime

Mathcad Prime є останнім релізом компанії РТС в програмному забезпеченні для інженерних розрахунків. Основні відмінності нового Mathcad Prime:

- Змінений інтерфейс користувача, який тепер виконаний в стилі останніх версій MS Office.
- Середовище обчислень, орієнтоване на роботу з документами, дозволяє користувачам швидко і просто створювати деталізовані технічні документи, які включають складні обчислення, використовуючи «живі» математичні примітки з текстом, зображеннями і діаграмами. Подібні документи легко читаються і

- Оператор невизначеного інтеграла \int дає можливість обчислювати невизначений інтеграл, для цього після введення підінтегрального виразу достатньо натиснути \rightarrow :

$$\int \frac{x^2 - 5x}{3x^2} dx \rightarrow \frac{x}{3} - \frac{5 \cdot \ln(x)}{3}$$

Рис.2.81 – Приклад використання оператора невизначеного інтеграла

- Оператор знаходження суми $\sum_{n=1}^m$;
- Оператор знаходження суми за ранжируваною змінною \sum_n ;
- Оператор добутку $\prod_{n=1}^m$;
- Оператор знаходження добутку за ранжируваною змінною \prod_n ;
- Оператор знаходження границі $\lim_{\rightarrow a}$.

$$\lim_{x \rightarrow 5} (5x^2 - 4x + 12) \rightarrow 117$$

$$\lim_{n \rightarrow \infty} \left(\frac{n^5 - 4n^4 + 5n^2 - 9}{2n^5 - 7n^3 + 2} \right) \rightarrow \frac{1}{2}$$

Рис.2.82 – Приклад застосування оператора границі


- Оператор знаходження границі праворуч $\lim_{\rightarrow a^+}$;
- Оператор знаходження границі ліворуч $\lim_{\rightarrow a^-}$,

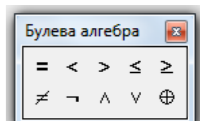
$$\lim_{x \rightarrow 0^+} \operatorname{atan}\left(\frac{1}{x}\right) \rightarrow \frac{\pi}{2}$$

$$\lim_{x \rightarrow 0^-} \operatorname{atan}\left(\frac{1}{x}\right) \rightarrow -\frac{\pi}{2}$$

Рис.2.83 Приклади використання оператора знаходження границі праворуч та ліворуч.

- Оператор градієнта $\nabla_x f$

На панелі **Булева алгебра**, яка керується кнопкою  математичної палітри, розміщено логічні оператори та оператори відношення:



- дорівнює, менше, більше, менше або дорівнює, більше або дорівнює, не дорівнює $\{ = < > \leq \geq \neq \}$;
- ні, і, або, виключне або $\{ \neg \wedge \vee \oplus \}$.

2.3.17. Алгебраїчні рівняння та системи

Всяке рівняння $P(x)=0$ – многочлен відмінний від нульового, називається алгебраїчним рівнянням (поліномом) відносно змінної x . Алгебраїчні рівняння в MathCad розв’язуються за допомогою вбудованих функцій:

- $root(P(x),x)$ – повертає з заданою точністю значення змінної x , при якому вираз $P(x)=0$ дорівнює нулю. Функція реалізує обчислення ітераційним методом, та перед її застосуванням необхідно задати початкові значення змінної x , що належить проміжку ізоляції кореня;

- $polyroots(v)$ – дозволяє знайти всі корені полінома (як дійсні та і комплексні), при цьому матриця-вектор задається від вільного члена до вищого степеня).

А) $x := 1$
 $root(x^2 - 2 \cdot x + 1, x) = 1$

Б) $x := 7$
 $root(x^3 - 4x^2 + 2x - 7, x) = 3.943$

В)
 $polyroots \left(\begin{pmatrix} 8 \\ 2 \\ 3 \end{pmatrix} \right) = \begin{pmatrix} -0.333 + 1.599i \\ -0.333 - 1.599i \end{pmatrix}$

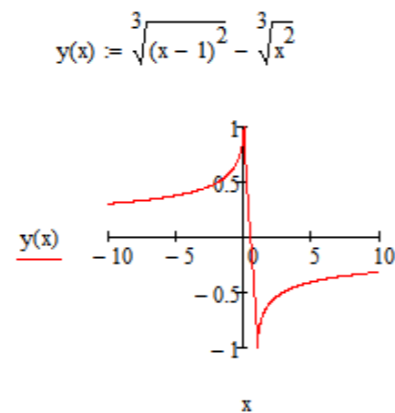
Г) $3x^4 + 5x^3 + 2x - 7 = 0$

$$polyroots \left(\begin{pmatrix} -7 \\ 2 \\ 0 \\ 5 \\ 3 \end{pmatrix} \right) = \begin{pmatrix} -2.08 \\ -0.234 - 1.104i \\ -0.234 + 1.104i \\ 0.882 \end{pmatrix}$$

Трансцендентне рівняння можна розв’язати за допомогою функції $root(P(x),x)$, або графічно:

1. $\sqrt[3]{(x-1)^2} - \sqrt[3]{x^2} = 0$

Розв’язок графічним методом:



```
SUM(m,n) :=
  S ← 0
  i ← m
  while i ≤ n
    S ← S + i
    i ← i + 1
  S
```

SUM(11,24) = 245

Рис. 2.103 – Застосування умовного циклічного оператора

Це означає, що оператор буде виконуватись у циклі до тих пір, поки істинна умова.

- Оператори переривання та продовження. Оператор переривання *break* використовується для дострокового завершення циклу. На рис. 59 показано як визначити першого нульового елемента у масиві чисел далі. Як тільки такий елемент знайдений, то непотрібно перевіряти масив. За допомогою оператора переривання управління передається у такому випадку на кінець циклу.

Оператор *continue* – це також оператор переривання, але на відміну від оператора *break*, який перериває цикл, він зупиняє виконання тільки поточної ітерації.

```
s(v,x) :=
  j ← 0
  for i ∈ v
    break if i = x
    a_i ← i
    j ← j + 1
  a

g(n) :=
  x ← 0
  for i ∈ 0..n
    for j ∈ 0..n
      continue if mod(i+j,2) = 0
      x ← x + 1
      a_{i,j} ← x
  a
```

v := (1 2 3 0 5)

s(v,0)^T = (0 1 2 3)

g(2) = $\begin{pmatrix} 0 & 1 & 0 \\ 2 & 0 & 3 \\ 0 & 4 & 0 \end{pmatrix}$

Рис.2.104 – Використання операторів break та continue

- Оператор переривання програмного модуля *return* перериває виконання програмного модуля улюбій точці та повертає значення виразу або змінної, або текстове повідомлення, що стоїть слідом за ним.

- Оператор обробки помилок. Якщо при складанні програмного модуля припускається, що якась команда може визвати помилку (наприклад, ділення на нуль), то можна використати оператор *on error* для перехоплення цієї помилки.

Тобто, якщо умова істина то виконується *оператор1*, якщо хибна – то *оператор2*.

•Цикл з відомою кількістю повторень. Оператор *for* дає можливість побудувати цикл за деякою змінною, що змінюється у деякому діапазоні. Він має вигляд

for змінна \in початкове значення, крок .. кінцеве значення
оператор

Це означає, що оператор виконається для значень змінної, яка змінюється у діапазоні від початкового значення до кінцевого, з заданим кроком (рис.2.101).

Сума непарних чисел від 5 до 17

```
SUM := | S ← 0
      | for i ∈ 5,7..17
      | S ← S + i
SUM = 77
```

Рис.2.101 – Використання оператора циклу з відомою кількістю повторень

Рис. 2.102 мстить програмний модуль з застосуванням циклу, у якому виконується два оператора. Результатом роботи функції є вектор, перший елемент якого – сума чисел від 1 до n, другий – добуток цих чисел.

```
M(n) := | S ← 0           присвоювання початкових значень
      | M ← 1
      | for i ∈ 1..n
      | | S ← S + i       обчислюється значення суми і добутку
      | | M ← M * i       чисел від 1 до n
      | N0 ← S
      | N1 ← M           формується вектор із значень
      | N                суми та добутку
M(8) = ( 36
        4.032 × 104 )   результат роботи функції
```

Рис.2.102 – Створення циклу з застосуванням циклічного оператора

Границі діапазону можуть бути задані не тільки конкретним значенням, а й змінними, а діапазон мати вигляд вектора.

•Цикл з умовою. За допомогою оператора *while* створюється цикл, кількість повторень якого невідома, але передбачений вихід з нього за деякою логічною умовою. Цикл має вигляд

while умова
оператор

За допомогою функції *root(P(x),x)*:

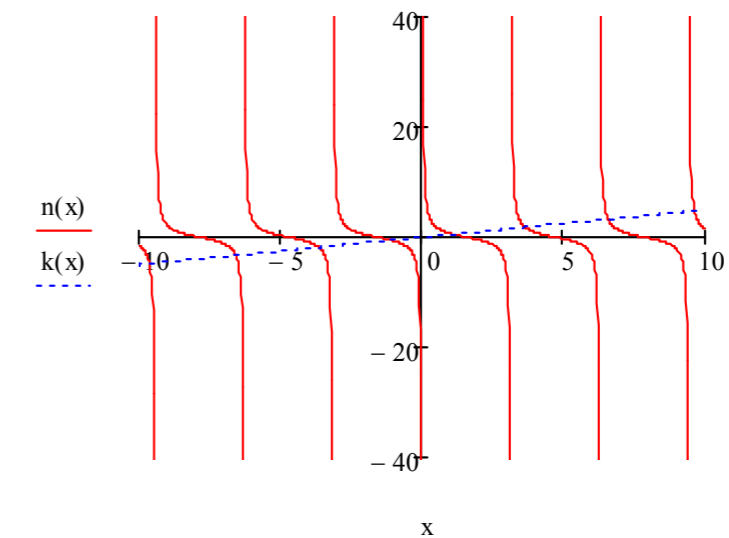
$x := 6$

$$\text{root}\left[\left[\sqrt[3]{(x-1)^2} - \sqrt[3]{x^2}\right], x\right] = 7.153 \times 10^8$$

один дійсний корінь

$$2. \frac{1}{\text{tg}x} - \frac{x}{2} = 0$$

$$n(x) := \frac{1}{\tan(x)} \quad k(x) := \frac{x}{2}$$



Для розв'язання систем рівнянь в MathCad існує розв'язувальний блок, який відкривається ключовим словом *Given* та має відповідну структуру:

- початкові умови – визначають початкові значення змінних, їх задають звичайним присвоюванням;
- ключове слово *Given*;
- рівняння – задаються з використанням жирного знака дорівнює;
- обмеження записуються у вигляді нерівностей або рівностей, які повинні виконуватись при розв'язуванні систем рівнянь;
- вираз, що містить функцію для розв'язування системи – може бути представлено однією з функцій *Find*, *Minerr*, *Maximize*, *Minimize*, *Odesolve*, *Pdesolve*.

Функція *Find* використовується для знаходження точного розв'язку системи, а *Minerr* – для наближеного розв'язку системи з мінімальною похибкою. Різниця між ними полягає в тому, що перша використовується коли розв'язок реально існує, а друга – коли відповідна функція максимально наближена до нуля. Таким чином, якщо функція *Find* видає повідомлення про відсутність розв'язку, то функція *Minerr* видає значення змінних, але з меншою точністю.

Приклад зображено на рис.2.84.

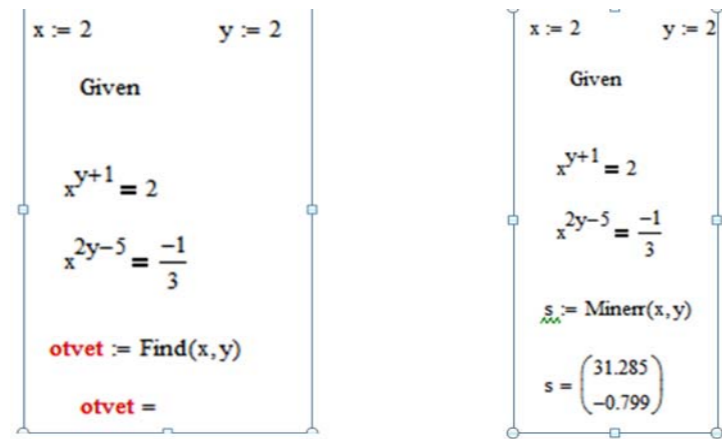


Рис.2.84 – Використання функцій Find, Minerr

Для знаходження всіх розв'язків системи можна використати символний розв'язок:

$$\begin{aligned} &\text{Given} \\ &x^2 + y^2 = 5 \\ &x \cdot y = 2 \\ &\text{Find}(x,y) \rightarrow \begin{pmatrix} 2 & 1 & -2 & -1 \\ 1 & 2 & -1 & -2 \end{pmatrix} \end{aligned}$$

Якщо задати початкові значення, то отримаємо не всі корені (треба задавати початкові значенні змінних в кожній чверті)

$$\begin{aligned} &x := -5 \quad y := -4 \\ &\text{Given} \\ &x^2 + y^2 = 5 \\ &x \cdot y = 2 \\ &\text{otvet} := \text{Find}(x,y) \\ &\text{otvet} = \begin{pmatrix} -1 \\ -2 \end{pmatrix} \end{aligned}$$

Для знаходження розв'язку заданої системи рівнянь графічним методом, достатньо виразити y через x в кожному рівнянні та побудувати графіки отриманих функцій у декартовій системі координат.

Точки перетину кола та гіперболи і є розв'язками системи рівнянь. (рис.2.85).

2.3.23. Створення програмного модуля

Оператор *Add Line* з панелі Программирование дає можливість вставити до робочого документу конструкцію. Блок обмежений жирною лінією, праворуч від якої розміщено поля введення команд програмного модуля. Для розширення програмного блоку шляхом додавання нових полів введення, необхідно повторно визивати оператор *Add Line*, але вже усередині програмного модуля. Причому можна не тільки зробити блок довше, але й створити різні вкладені структури, наприклад такі, як на рис. 2.100.

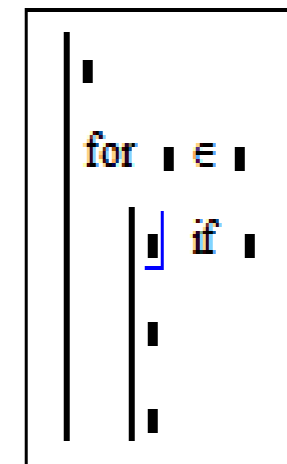


Рис.2.100 – Програмний модуль деревовидної структури

Треба звернути увагу, якщо необхідно додати нове поле введення поза умовного оператора, то потрібно виділити його повністю, а потім натиснути кнопку *Add Line* (Додати лінію).

2.3.24. Оператори програмування

•Оператор присвоювання

Оператор на панелі Программирование, який позначено \leftarrow , виконує операцію присвоювання. Змінній, яка розміщена зліва від оператора присвоюється значення, що вказане праворуч. Причому це значення може бути вже задане, або його треба обчислити за допомогою виразу. Особливість цього оператора полягає в тому, що присвоювання виконується локально, тобто усередині програмного модуля.

•Умовний оператор застосовується, якщо в залежності від деякої умови потрібно виконати одну або іншу дію. Він має структуру:

оператор if умова,

де оператор – дія, що виконується, у випадку коли умова істина. Коли умова хибна, управління передається наступному за *if* оператору.

•Оператор альтернативного вибору дозволяє запрограмувати умовну конструкцію вигляду:

оператор1 if умова
оператор2 otherwise

Використавши вбудовану функцію Minimize знаходимо мінімальне значення функції при відповідних значеннях змінних x_1, \dots, x_{12} .

$$\text{Minimize}(F, x) = \begin{array}{|c|c|} \hline & 1 \\ \hline 1 & 0 \\ \hline 2 & 100 \\ \hline 3 & 0 \\ \hline 4 & 0 \\ \hline 5 & 80 \\ \hline 6 & 10 \\ \hline 7 & 80 \\ \hline 8 & 30 \\ \hline 9 & 120 \\ \hline 10 & 0 \\ \hline 11 & 0 \\ \hline 12 & 0 \\ \hline \end{array}$$


Обчислюємо мінімальну вартість перевезення

$$4 \cdot 0 + 2 \cdot 100 + 6 \cdot 0 + 0 \cdot 0 + 7 \cdot 80 + 5 \cdot 10 + 3 \cdot 80 + 0 \cdot 30 + 1 \cdot 120 + 7 \cdot 0 + 6 \cdot 0 + 0 \cdot 0 = 1.17 \times 10^3$$

Як бачимо від оформлення розв'язування задачі розв'язок не змінюється.

2.3.22. Програмування у MathCad

З точки зору програмування MathCad-програма представляє собою підпрограму-функцію, яка повертає у якості результату число, вектор або матрицю. Ця програма складається у вигляді програмного модуля – самостійного блоку, який вирізняє жирна вертикальна пряма. Для створення програмного модуля треба скористатися панеллю інструментів

Программування, яка визивається кнопкою , що розташована на математичній панелі.

Ця панель містить наступні оператори:

- *Add Line* – створення блоку для введення команд;
- ← - присвоювання.
- *if* - умовний;
- *otherwise* - альтернативного вибору (використовується разом з умовним);
- *for* – цикл з відомою кількістю повторень;
- *while* – умовного циклу;
- *break* – переривання;
- *continue* – продовження;
- *return* – повернення;
- *on error* – обробки помилок.

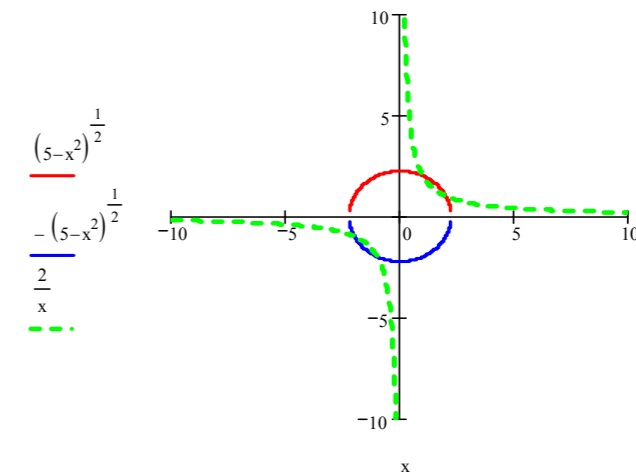


Рис.2.85 – Розв'язок системи рівнянь графічним методом

Якщо систему лінійних рівнянь задати в матричній формі то для знаходження її розв'язку можна використати функцію *lsolve* (рис.2.86).

$$\begin{aligned} x + 4y + 3z &= 2 \\ 2x - 5y + 4z &= 7 \\ 8x + 3y + 2z &= 9 \end{aligned}$$

$$V := \begin{pmatrix} 1 & 4 & 3 \\ 2 & -5 & 4 \\ 8 & 3 & 2 \end{pmatrix} \quad M := \begin{pmatrix} 2 \\ 7 \\ 9 \end{pmatrix}$$


$$\text{lsolve}(V, M) \rightarrow \begin{pmatrix} \frac{121}{114} \\ \frac{20}{57} \\ \frac{89}{114} \end{pmatrix} = \begin{pmatrix} 1.061 \\ -0.351 \\ 0.781 \end{pmatrix}$$

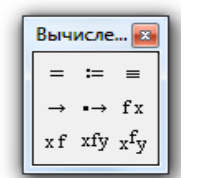
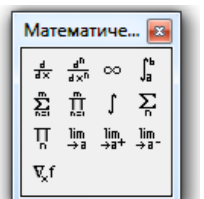
Рис.2.86 – Використання функції *lsolve* для знаходження розв'язку системи рівнянь

Основні інструменти для розв'язування задач математичного аналізу знаходяться на панелі інструментів .

За допомогою цієї панелі можна:

- обчислити похідну;
- невизначений та визначений інтеграли;
- обчислити суму;
- знайти добуток;
- також визначити різні границі.

Результатом роботи оператора може бути як чисельний, та і символний розв'язок. Для отримання символного розв'язку призначена кнопка  з панелі інструментів .



2.3.18. Інтегрування

Чисельне інтегрування в Mathcad реалізовано у вигляді оператора визначеного інтеграла, який дозволяє обчислювати інтеграли від скалярних функцій у скалярних границях інтегрування. Границі інтегрування повинні бути дійсні, а підінтегральна функція може мати і комплексні значення, тому і значення інтегралу може бути комплексним. Якщо границі інтегрування мають розмірність, то вона повинна бути однакою для обох границь.

Підінтегральна функція може залежати від будь-якої кількості змінних. Для того щоб вказати, за якою змінною обчислюється інтеграл, необхідно ввести її ім'я у відповідний місцезаповнювач. Для чисельного інтегрування по одній із змінних попередньо треба задати значення решти змінних, від яких залежить підінтегральна функція (рис.2.87).

$$m := 3 \quad x := 1$$

$$\int_{\frac{\pi}{4}}^{\frac{5\pi}{4}} m \cdot \cos(x) \, dx = -4.243$$

$$\int_2^7 m \cdot \cos(x) \, dm = 12.157$$

Рис.2.87 – Інтегрування функції двох змінних за різними змінними

Оператор інтегрування може використовуватися для обчислення функцій від ранжированих змінних (рис.2.88).

$$f(m) := \int_{\frac{\pi}{4}}^{\frac{5\pi}{4}} m \cdot \cos(x) \, dx \quad i := 1..5$$

f(i) =
-1.414
-2.828
-4.243
-5.657
-7.071

Рис.2.88 – Використання оператора інтегрування від ранжированої змінної

Можна обчислювати інтеграли з однією або обома нескінченими границями: в місце відповідної границі треба ввести символ нескінченності. Для введення знаку мінус нескінченність достатньо додати знак мінус до символу нескінченності, як до звичайного числа. Для отримання результату потрібно ввести знак дорівнює.

$$\int_2^{\infty} \frac{1}{\sqrt{n}} \cdot \ln\left(\frac{n+1}{n-1}\right) \, dn = 2.88$$

Для обчислення невизначених інтегралів достатньо використати оператор невизначеного інтеграла, вказавши підінтегральну функцію та змінну інтегрування.

Для розв'язання транспортної задачі потрібно, щоб вона була збалансованою. Тому так як у нашій задачі запас вантажу складає 420 т, а потреба 390 т, вводимо додатковий пункт призначення з потребою у вантажі 420-390=30 т, та нульовими тарифами. Позначимо через $x_{ij}(t)$ – кількість перевезеного вантажу з i -го пункту відправлення до j -го пункту призначення.

Розв'язок задачі засобами MathCad на рис.2.99.

$$i := 1..3 \quad j := 1..4 \quad x_{i,j} := 12$$

$$F(x) := 4x_{1,1} + 2x_{1,2} + 6x_{1,3} + 0x_{1,4} + 7x_{2,1} + 5x_{2,2} + 3x_{2,3} + 0x_{2,4} + 1x_{3,1} + 7x_{3,2} + 6x_{3,3} + 0x_{3,4}$$

Given

$$\begin{aligned} x_{1,1} + x_{1,2} + x_{1,3} + x_{1,4} &= 100 & x_{1,1} &\geq 0 & x_{1,2} &\geq 0 & x_{1,3} &\geq 0 \\ x_{2,1} + x_{2,2} + x_{2,3} + x_{2,4} &= 200 & x_{1,4} &\geq 0 & x_{2,1} &\geq 0 & x_{2,2} &\geq 0 \\ x_{3,1} + x_{3,2} + x_{3,3} + x_{3,4} &= 120 & x_{2,3} &\geq 0 & x_{2,4} &\geq 0 & x_{3,1} &\geq 0 \\ x_{1,1} + x_{2,1} + x_{3,1} &= 200 & x_{3,2} &\geq 0 & x_{3,3} &\geq 0 & x_{3,4} &\geq 0 \\ x_{1,2} + x_{2,2} + x_{3,2} &= 110 & & & & & & \\ x_{1,3} + x_{2,3} + x_{3,3} &= 80 & & & & & & \\ x_{1,4} + x_{2,4} + x_{3,4} &= 30 & & & & & & \end{aligned}$$

$$\text{Minimize}(F, x) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 \\ 0 & 80 & 10 & 80 & 30 \\ 0 & 120 & 0 & 0 & 0 \end{pmatrix} \quad M := F \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 100 & 0 & 0 \\ 0 & 80 & 10 & 80 & 30 \\ 0 & 120 & 0 & 0 & 0 \end{pmatrix}$$

$$M = 1.17 \times 10^3$$

Рис.2.99 – Розв'язок транспортної задачі

Можна оформити розв'язок задачі таким чином, якщо вводити не матричний індекс змінних :

Given

$$\begin{aligned} x_1 + x_2 + x_3 + x_4 &= 100 & x_1 &\geq 0 & x_2 &\geq 0 & x_3 &\geq 0 & x_4 &> 0 \\ x_5 + x_6 + x_7 + x_8 &= 200 & x_5 &\geq 0 & x_6 &\geq 0 & x_7 &\geq 0 & x_8 &\geq 0 \\ x_9 + x_{10} + x_{11} + x_{12} &= 120 & x_9 &\geq 0 & x_{10} &\geq 0 & x_{11} &\geq 0 & x_{12} &\geq 0 \\ x_1 + x_5 + x_9 &= 200 & & & & & & & & \\ x_2 + x_6 + x_{10} &= 110 & & & & & & & & \\ x_3 + x_7 + x_{11} &= 80 & & & & & & & & \\ x_4 + x_8 + x_{12} &= 30 & & & & & & & & \end{aligned}$$

• завершити розв'язувальний блок зверненням до функції Maximize (Minimize) (рис.2.98).

$$f(x) := 45x_0 + 48x_1 + 45x_2 + 50x_3$$

$$M := \begin{pmatrix} 6 & 3 & 4 & 5 \\ 2 & 0 & 3 & 5 \\ 15 & 15 & 10 & 0 \end{pmatrix} \quad V := \begin{pmatrix} 3500 \\ 4200 \\ 10000 \end{pmatrix}$$

$$x_0 := 1 \quad x_1 := 1 \quad x_2 := 1 \quad x_3 := 1$$

Given

$$M \cdot x \leq V \quad x \geq 0$$

$$\text{Maximize}(f, x) = \begin{pmatrix} 0 \\ 666.667 \\ 0 \\ 300 \end{pmatrix}$$

$$f \left(\begin{pmatrix} 0 \\ 666.667 \\ 0 \\ 300 \end{pmatrix} \right) = 4.7 \times 10^4$$

Рис.2.98 – Використання функцій Maximize (Minimize)

Отже для отримання максимального прибутку 4.7×10^4 грн. достатньо випустити 666.667 кг цукерок «Півник» та 300 кг цукерок «Білочка».

2. Розглянемо таку транспортну задачу: є три пункти відправлення з запасом вантажу у кількості 100, 200 та 120 т, та три пункти призначення з потребою у вантажі відповідно 200, 110, 80 т. Вартість перевезення вантажу з пунктів відправлення до пунктів призначення характеризується матрицею:

$$C = \begin{pmatrix} 4 & 2 & 6 \\ 7 & 5 & 3 \\ 1 & 7 & 6 \end{pmatrix}$$

Необхідно перевезти вантаж з пунктів відправлення до пунктів призначення за мінімальних витрат.

Розрізняють три типи транспортних задач:

- збалансована – кількість запасів дорівнює кількості потреб;
- транспортна задача з умовою перевиробництва для зведення її до збалансованої вводиться фіктивний пункт споживання, вартість перевезення до якого дорівнює 0;
- транспортна задача з умовою дефіциту – для зведення її до збалансованої вводиться фіктивний пункт постачання, вартість перевезення до якого дорівнює 0.

Застосування символічної операції призводить до вивода аналітичного виразу інтеграла. Приклади обчислення невизначеного інтеграла на рис.2.89.

$$\int \frac{x+2}{x^2-4x+3} dx \rightarrow \frac{5 \cdot \ln(x-3)}{2} - \frac{3 \cdot \ln(x-1)}{2}$$

$$\int (\sin(3x))^3 dx \rightarrow \frac{\cos(9-x)}{36} - \frac{\cos(3-x)}{4}$$

$$\int e^{5x} \cdot \cos(7x) dx \rightarrow \frac{e^{5x} \cdot (5 \cdot \cos(7-x) + 7 \cdot \sin(7-x))}{74}$$

Рис.2.89 – Обчислення невизначеного інтегралу

В символічному вигляді можна також обчислювати визначені інтеграли. Для цього використовується оператор \int , який використовується для чисельного інтегрування але замість знака дорівнює використовується стрілка.

$$\int_0^{\pi/6} \frac{\cos(2x)}{\cos(x)} dx \rightarrow \ln(\sqrt{3}-1) - \ln(3-\sqrt{3}) + 1$$

Рис.2.90 – Обчислення інтегралу в символічній формі

2.3.19. Диференціювання

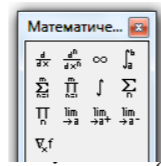
Обчислювальний процесор Mathcad забезпечує велику точність чисельного диференціювання. Але більш за все користувач зможе оцінити можливості символічного процесора, який дозволяє з легкістю здійснити рутинну роботу обчислення похідних громіздких функцій, бо символічне диференціювання виконується успішно для більшості аналітично заданих функцій.

В Mathcad можна обчислювати похідні скалярних функцій будь-якої кількості аргументів до п'ятого порядку. Функції та аргументи можуть бути дійсними та комплексними. Неможливо диференціювання функцій тільки поблизу їхньої сингулярності. Наприклад при знаходженні похідної функції $f(x)=1/x$ в околі сингулярної точки $x=0$, буде видано повідомлення про помилку ділення на нуль. Для чисельного диференціювання Mathcad застосовує доволі складний алгоритм, який обчислює похідну з точністю до 7-8-го знака після коми.

Для диференціювання функції у деякій точці:

- необхідно визначити точку, у якій буде обчислюватися похідна;

- ввести оператор диференціювання $\frac{d}{dx}$ лі;
- у місце заповнювач ввести функцію та аргумент;
- для отримання відповіді клацнути мишкою на знак дорівнює (рис.2.91).



Якщо функція вводиться явно до оператора диференціювання, то необхідно попередньо визначити точку, у якій здійснюється чисельне диференціювання, інакше буде видано повідомлення про помилку, у якому вказується що змінна або функція, що входить до виразу, раніш не визначена. Можна попередньо визначити функцію в окремому виразі, а потім обчислити її похідну у заданій точці.

$$x := 0.025$$

$$\frac{d}{dx}(\tan(2x) \cdot \log(x)) = -2.343$$

Рис.2.91 – Чисельне диференціювання

Для обчислення похідної до 3-го порядку достатньо з панелі вибрати відповідне позначення (рис.2.92).

$$x := 0.025$$

$$\frac{d^3}{dx^3}(\tan(2x) \cdot \log(x)) = -1.403 \times 10^3$$

Рис.2.92 – Чисельне диференціювання 3-го порядку

За допомогою Mathcad можна розв'язувати задачі з використанням похідної. Наприклад, для знаходження рівняння дотичної до графіка функції

$$f(x) = \frac{2x^2 - 4}{3x + 2} \text{ в точці } x_0 = 1 \text{ потрібно:}$$

- задати функцію;
- задати точку дотику;
- знайти похідну заданої функції;
- обчислити значення похідної у точці дотику;
- записати рівняння дотичної.

наростаючими інформаційними течіями, які надходять до економічних й управлінських органів. Різко зростає кількість операцій щодо переробки інформації, необхідної для пошуку найкращих (оптимальних) варіантів розвитку виробництва й прийняття рішень.

Таким чином, пошук найкращого (оптимального) плану (варіанта) простим перебором і порівняння всіх можливих планів стає вкрай непосильною задачею, при цьому не враховується той факт, що на складання одного варіанта плану також витрачається дуже багато часу.

Програма MathCad дає можливість знайти розв'язок багатьох задач оптимізації. Розглянемо декілька з них.

1. Фабрика випускає цукерки чотирьох найменувань. Кожний вид цукерок містить наступні компоненти: цукор, какао, наповнювач. Вміст компонентів у цукерках кг/порц., їхній запас та вартість 1 кг цукерок кожного типу розміщено у таблиці:

	Червоний мак	Півник	Золотий колос	Білочка	Запас (кг)
Цукор	6	3	4	5	3500
Какао	2	0	3	5	4200
Наповнювач	15	15	10	0	10000
Вартість 1 кг (грн.)	45	48	45	50	

Знайти оптимальний план випуску продукції, при якому прибуток від реалізації був би максимальний.

Складемо математичну модель задачі: позначимо через x_1, x_2, x_3 (кг) - кількість цукерок кожного найменування, тоді ціль задачі - отримання максимального прибутку від реалізації цукерок буде мати вигляд $F_{max} = 45x_1 + 48x_2 + 45x_3 + 50x_4$. Система обмежень описує умови досягнення цілі - вміст цукру, какао та наповнювача у кожному виді цукерок.

$$\begin{cases} 6x_1 + 3x_2 + 4x_3 + 5x_4 \leq 3500 \\ 2x_1 + 3x_3 + 5x_4 \leq 4200 \\ 15x_1 + 15x_2 + 10x_3 \leq 10000 \\ x_1, x_2, x_3 \geq 0 \end{cases}$$

Знайдемо розв'язок цієї задачі за допомогою MathCad, для цього:

- визначимо початкові наближення змінних;
- задамо функцію цілі;
- почати розв'язувальний блок службовим словом Given;
- у розв'язувальному блоці вводимо обмеження, враховуючи умови невід'ємності змінних;

2.3.21. Застосування MathCad для розв'язання задач оптимізації

Апарат лінійного програмування можна застосовувати до доволі широкого кола практичних задач, причому діапазон цих задач зростає. Задачі, які можна розв'язати методом лінійного програмування незалежно від їх конкретного змісту мають такі особливості:

- із багатьох допустимих розв'язків вибирають найкращий або оптимальний;
- розв'язками є набір кінцевої кількості змінних величин;
- на вибір допустимих розв'язків накладаються обмеження, що обумовлюються специфікою задачі. Математичною моделлю цих обмежень є система лінійних рівнянь або нерівностей;
- критерієм якості розв'язку є значення цільової функції, яка є лінійною функцією від змінних, оптимальне значення яких шукаємо.

Відповідно до цього формальний опис практичної задачі, яку можна розв'язати методом лінійного програмування складається з етапів:

- позначення змінних величин плану задачі;
- математичний опис обмежень на вибір значення змінних (умова практичних, допустимих значень);
- побудова цільової функції, значення якої є критерієм якості розв'язання;
- остаточне формулювання математичної моделі задачі, як задачі на знаходження екстремуму цільової функції на множині допустимих значень змінних - планів задачі.

З великого різноманіття практичних задач лінійного програмування можна виділити декілька груп економічно споріднених задач. Всі останні задачі можна звести до однієї з виділених груп.

Такими класифікуючими групами можуть бути:

- станочні задачі (з обладнанням), де треба знайти оптимальне розподілення асортименту виробів між окремими взаємопов'язаними машинами. Критерієм оптимальності цієї групи задач може бути прибуток, собівартість, кількість продукції, продуктивність праці тощо;
- задачі на суміші, де необхідно знайти оптимальну рецептуру різних сумішей. До цієї групи належить одна з перших задач лінійного програмування, яка називається "задача про кормовий раціон або дієту". Критерієм оптимальності може бути вартість, деякі технологічні показники суміші тощо;
- задачі оптимального розкрою матеріалів. Критерієм оптимальності у більшості випадків є досягнення найменших відходів при розкроюванні стандартних зразків;
- задачі транспортного типу. Для класичних транспортних задач критерієм оптимальності є вартість перевезень або час, який витрачається.

Сучасні умови виробництва продукції в різних галузях на рівні окремих підприємств, а також на вищому макроекономічному рівні супроводжуються

$$f(x) := \frac{2x^2 - 4}{3x + 2}$$

$$x_0 := 1$$

$$y(x) := \frac{d}{dx} f(x) \rightarrow \frac{4-x}{3-x+2} - \frac{3 \cdot (2x^2 - 4)}{(3-x+2)^2}$$

$$f(x_0) = -0.4$$

$$k := y(x_0) = 1.04$$

$$g(x) := f(1) + k \cdot (x - 1) \rightarrow 1.04 \cdot x - 1.44$$

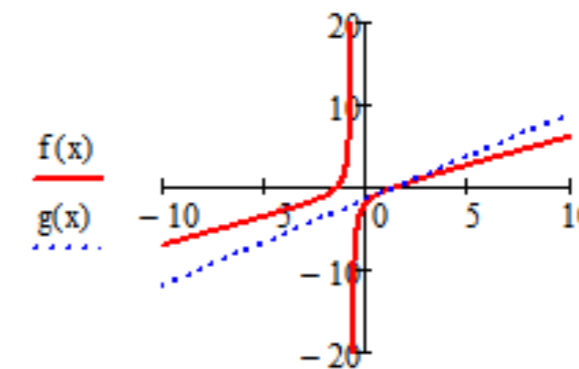


Рис.2.93 – Розв'язок задачі, графік функції та дотичної.

Знайдемо розв'язок задачі на визначення точок максимуму та мінімуму функції (рис.2.94).

Для цього необхідно:

- знайти похідну функції;
- знайти критичні точки – прирівняти похідну до нуля та розв'язати отримане рівняння;
- визначити знак похідної зліва та справа від критичної точки;
- якщо знак похідної змінюється з «+» на «-» то це точка max, а з «-» на «+» точка min;
- обчислити значення функції в точках екстремуму.

+++++

Рис.2.94 – Знаходження максимального та мінімального значень функції

Таким чином максимум досягається у точці (-0.667), а мінімум у точці (0.4).

2.3.20. Розв'язування диференціальних рівнянь

Для розв'язування звичайних диференціальних рівнянь в системі MathCad призначені наступні функції:

• `odesolve ((vector),t,b,(n))` – повертає функцію (скалярну для рівняння та вектор-функцію для системи рівнянь), яка є розв'язком диференційного рівняння. `Vector` – необов'язковий параметр (застосовується тільки при розв'язуванні систем), містить імена функцій, `t` - ім'я незалежної змінної, `b` - кінцева точка інтегрування, `n` - кількість кроків, на які розбивається інтервал інтегрування диференційного рівняння або системи, ця функція завершує розв'язувальний блок `Given`. Між словом та функцією потрібно ввести рівняння та початкові умови (рис.2.95).

$$\frac{dx}{dt} + x = \sin(x \cdot t)$$

$$x(0) = 1.5$$

Given

$$\frac{d}{dt} x(t) + x(t) = \sin(x(t) \cdot t)$$

$$x(0) = 1.5$$

$$x := \text{Odesolve}(t, 35)$$

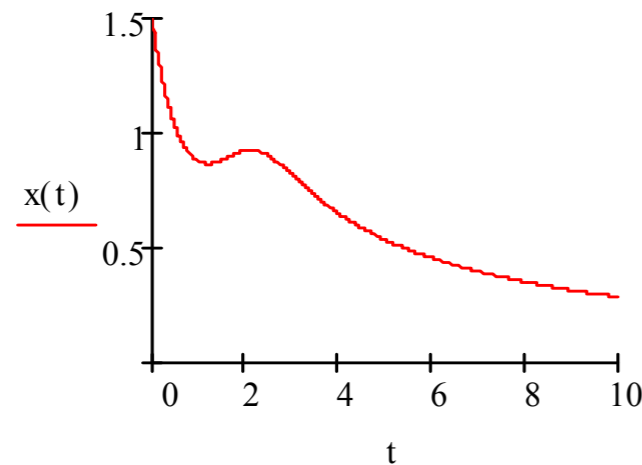


Рис.2.95 – Розв'язок диференціального рівняння за допомогою функції `odesolve`

• `Rkfixed(init,t1,t2,npoints,D)` – розв'язує диференціальне рівняння першого порядку або систему з n таких рівнянь (рівняння або систему потрібно привести до вигляду $x' = F(t,x)$) методом Рунге-Кутта четвертого порядку з фіксованим кроком; `init` - вектор початкових умов, $[t_1, t_2]$ - інтервал інтегрування, `points` - кількість проміжків, на які розбивається інтервал $[t_1, t_2]$, `D` - вектор правих частин системи диференціальних рівнянь. Функція `Rkfixed` видає матрицю, нульовий стовпчик якої містить всі точки інтервалу інтегрування $[t_1, t_2]$, а всі решта стовпчики – значення знайдених функцій у вузлових точках (рис.2.96).

	0	1
0	0	1.5
1	0.175	1.279
2	0.35	1.123
3	0.525	1.015
4	0.7	0.942
5	0.875	0.897
6	1.05	0.874
7	1.225	0.868
8	1.4	0.875
9	1.575	0.889
10	1.75	0.906
11	1.925	0.92
12	2.1	0.926
13	2.275	0.921
14	2.45	0.907
15	2.625	...

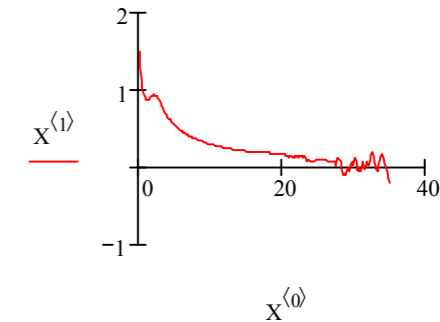


Рис.2.96 – Розв'язок диференціального рівняння з використанням функції `Rkfixed`

• `Rkadapt(init,t1,t2,npoints,D)` – розв'язує диференціальні рівняння першого порядку або систему з n таких рівнянь методом Рунге-Кутта з автоматичним вибором кроку; `init` - вектор початкових умов, $[t_1, t_2]$ - інтервал інтегрування, `points` - кількість проміжків, на які розбивається інтервал $[t_1, t_2]$, `D` - вектор правих частин системи диференціальних рівнянь. Функція `Rkadapt` видає матрицю, нульовий стовпчик якої містить всі точки інтервалу інтегрування $[t_1, t_2]$. А всі решта стовпчиків – значення знайдених функцій у вузлових точках (рис.2.97).

$$x' + x^2 + x^3 = \cos(5t)$$

$$D(t,x) := -x^3 - x^2 + \cos(5t)$$

$$X := \text{Rkadapt}(3, 0, 5, 50, D)$$

	0	1
0	0	3
1	0.1	1.612
2	0.2	1.217
3	0.3	0.993
4	0.4	0.819
5	0.5	0.661
6	0.6	0.515
7	0.7	0.387
8	0.8	0.29
9	0.9	0.238
10	1	0.235
11	1.1	0.277
12	1.2	0.35
13	1.3	0.427
14	1.4	0.485
15	1.5	...

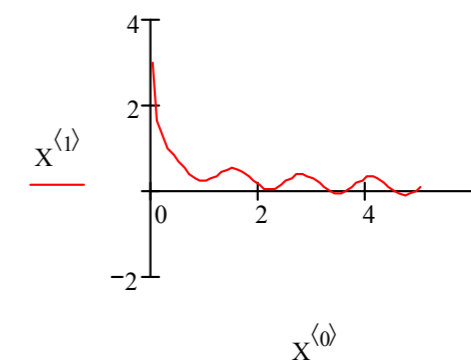


Рис.2.97 – Розв'язок диференціального рівняння з використанням функції `Rkadapt`.