

**Міністерство освіти і науки України
Національний університет «Чернігівська політехніка»**

ОСНОВИ ПРОГРАМУВАННЯ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ

Методичні вказівки

до виконання курсового проекту для здобувачів
першого (бакалаврського) рівня вищої освіти спеціальності
175 – Інформаційно-вимірювальні технології

Затверджено
на засіданні кафедри
електричної інженерії та
інформаційно вимірювальних
технологій
протокол №1 від 25.01.2024р.

ЧЕРНІГІВ 2024

Основи програмування та обчислювальної техніки. Методичні вказівки до виконання курсового проекту для здобувачів першого (бакалаврського) рівня вищої освіти спеціальності 175-«Інформаційно-вимірювальні технології». / Укл. Ревко А.С., Безручко В.М. – Чернігів: НУ «Чернігівська політехніка», 2024. – 29с.

Укладачі:

Ревко Анатолій Сергійович, кандидат технічних наук, доцент, доцент кафедри електроніки, автоматики, робототехніки та мехатроніки НУ «Чернігівська політехніка»

Безручко Вячеслав Михайлович, доцент кафедри електричної інженерії та інформаційно вимірювальних технологій, кандидат технічних наук

Відповідальний за випуск:

Приступа А.Л., завідувач кафедри електричної інженерії та інформаційно вимірювальних технологій, кандидат технічних наук, доцент

Рецензент:

Базилевич В.М., к.е.н., завідувач кафедри інформаційних та комп'ютерних систем НУ «Чернігівська політехніка»

ЗМІСТ

ВСТУП.....	4
1 МЕТОДИЧНІ РЕКОМЕНДАЦІЇ.....	6
2 ОФОРМЛЕННЯ КУРСОВОГО ПРОЕКТУ.....	10
2.1 СТРУКТУРА КУРСОВОГО ПРОЕКТУ	10
2.2 ЗМІСТ КУРСОВОГО ПРОЕКТУ	10
3 КРИТЕРІЇ ОЦІНКИ КУРСОВОГО ПРОЕКТУ	14
4 ПОРАДИ ЩОДО ФОРМАТУВАННЯ ТЕКСТУ ЗВІТУ У MICROSOFT WORD	17
4 ДОПОМІЖНИЙ МАТЕРІАЛ	23
ДОДАТОК А - ПРИКЛАД ЗАВДАННЯ НА КУРСОВИЙ ПРОЕКТ	26
ДОДАТОК Б - ПРИКЛАД ТИТУЛЬНОГО ЛИСТА КУРСОВОГО ПРОЕКТУ	27
ДОДАТОК В - ЗВОРОТНА СТОРОНА ТИТУЛЬНОГО ЛИСТА.....	28
РЕКОМЕНДОВАНА ЛІТЕРАТУРА.....	29

ВСТУП

Ці методичні вказівки призначені для студентів другого курсу галузі знань 17 - Електроніка, автоматизація та електронні комунікації спеціальності 175 - Інформаційно-вимірювальні технології денної форми навчання та слугують для полегшення роботи здобувачів вищої освіти над курсовим проектом з дисциплін “Основи програмування та обчислювальної техніки”. Також можуть використовуватися для навчання й студентами інших спеціальностей.

Мета проведення курсового проекту – отримання навичок по програмуванню складних задач та навичок по створенню програм з графічним інтерфейсом, навчитися по заданому завданню самостійно складати алгоритм та писати закінчену програму, а також готувати звіт по програмі з використанням автоматизованих можливостей офісного пакету Microsoft Word в рамках державного стандарту ДСТУ 3008-2015 [1].

Курсовий проект виконується протягом семестру і охоплює основні теми курсу "Основи програмування та обчислювальної техніки" з поглибленим самостійним вивченням студентом роботи у візуальному режимі при створенні програм з графічним інтерфейсом. Завдання на курсовий проект видається викладачем на початку семестру, приклад завдання приведений у додатку А. Теоретичною основою для виконання курсового проекту є курс лекцій, лабораторні роботи та навчальна література.

Здобувач вищої освіти повинен написати порівняно складну програму з використанням графічних можливостей мови програмування (з графічним інтерфейсом). Програма повинна відповідати всім критеріям якості програмного продукту. Для виконання курсового проекту використовується мова програмування C, C++ або інша, за погодженням з викладачем.

Крім написання самої програми здобувач вищої освіти повинен розробити вимоги до комп'ютера, на якому зможе виконуватись його програма.

Протягом семестру здобувач вищої освіти демонструє викладачу поточні результати роботи над проектом. Наприкінці семестру, не пізніше ніж за тиждень до залікової неділі, курсовий проект здається на перевірку. Після перевірки роботи призначається дата захисту. По результатах перевірки курсового проекту і співбесіди зі здобувачем вищої освіти під час захисту,

викладач виставляє оцінку.

Подана на перевірку пояснювальна записка та програма курсового проекту, яка повністю або частково співпадає з роботами, які були виконані іншими здобувачами раніше або які містять значні дослівні або перефразовані з чужої роботи уривки тексту, коду програми, фото, ілюстрацій без вказування першоджерела (лапки, цитування тощо), сприймаються як академічний плагіат. Здобувачі зобов'язані постійно дбати про дотримання академічної доброчесності та у випадку сумнівів звертатися за консультацією до викладача. Пам'ятайте, що академічна недоброчесність – це не лише плагіат, а й списування під час підсумкового контролю та виконання курсового проекту на замовлення. Від усіх здобувачів вищої освіти очікується дотримання академічної доброчесності при виконанні курсового проекту та підсумкового контролю. До здобувачів вищої освіти, у яких було виявлено порушення академічної доброчесності, застосовуються різноманітні дисциплінарні заходи, які залежать від ступеню порушення (від видачі нового завдання до повторного прослуховування курсу).

1 МЕТОДИЧНІ РЕКОМЕНДАЦІЇ

Перш ніж приступити до виконання курсового проекту необхідно повторити пройдений на лекціях та практичних заняттях матеріал, який стосується загальної теорії та практики програмування. У відповідності з наведеними там правилами проектування програмного продукту приступити до виконання курсового проекту. За необхідності скористатися додатковою літературою по програмуванню.

На початку семестру необхідно отримати завдання на курсовий проект у викладача (приклад завдання приведений у додатку А). Прискіпливо проаналізувати завдання і визначити, що повинна робити готова програма, намітити основні етапи роботи над проектом. Після цього треба проаналізувати літературні джерела для пошуку можливих варіантів вирішення даної задачі та готових алгоритмів, які можна використати у своїй програмі. Далі необхідно обрати один з варіантів вирішення задачі. Необхідно проконсультуватися з викладачем щодо допустимості обраного варіанту вирішення завдання. Після згоди викладача треба докладно описати обраний шлях вирішення задачі, скласти блок схему програми і написати безпосередньо текст програми на обраній мові програмування.

Наступним кроком є відлагодження програми. Її треба протестувати і виправити всі знайдені помилки та неточності. Бажано перевірити програму на працездатність на різних комп'ютерах та з різними версіями операційних систем. Для кращого виявлення прихованих помилок можна домовитися з товаришами про взаємне тестування програм один одного. Після успішного відлагодження програму необхідно продемонструвати викладачу і отримати дозвіл на оформлення курсового проекту або рекомендації щодо дороблення програми. Якщо отримано дозвіл на оформлення, то можна остаточно оформлювати курсовий проект та здавати його на перевірку.

Розроблена програма повинна працювати на комп'ютерах, на яких відсутня використана система розробки програмних продуктів. Тобто треба

створити файл програми, що може бути безпосередньо запущений на виконання (з розширенням `exe` чи `com`) та включити до кінцевого релізу файли, без яких неможлива робота розробленої програми (наприклад бібліотеки `.dll`). Для збірки всіх необхідних файлів можна скористатися інформацією, наведеною в літературі [2].

Під час роботи над курсовим проектом треба слідувати всім вимогам до його змісту і оформлення, які наведені у розділі 2 даних методичних вказівок, не порушувати термін здачі оформленої роботи на перевірку.

Після перевірки курсового проекту викладач допускає до його захисту здобувача вищої освіти та призначає день, час і місце захисту. На передодні захисту необхідно повторити теоретичний матеріал, що стосується роботи та переглянути безпосередньо її зміст. На захисті роботи треба буде відповісти на додаткові запитання та зауваження щодо її змісту, що виникли у викладача під час перевірки курсового проекту. По результатах захисту, у відповідності з критеріями оцінок, що наведені у розділі 3 даних вказівок, виставляється оцінка студенту за його курсовий проект.

Якщо виникають складності з реалізацією якогось етапу роботи треба звертатися за консультацією до викладача згідно з його розкладом консультацій.

Найчастіше автори проекту допускаються таких **помилкок**:

- 1) Нема допомоги по програмі та/або інформації про автора програми.
- 2) У блок-схемі представлені не всі оператори програми. Блок-схеми дуже прості, оформлені не вірно. Неправильно виконаний перенос блок-схем на наступну сторінку. На кожен підпрограму повинна бути окрема блок-схема.
- 3) Відсутні слова "Так" і "Ні" в умовах розгалужень блок-схеми.
- 4) Не описані деякі ідентифікатори в опису тексту програми.
- 5) Не приведені вимоги до комп'ютеру.
- 6) Не вказаний список файлів, що входять до курсового проекту і необхідні для роботи програми, їх розміри та призначення. Або включені до

цього списку файли, без яких програма може працювати (наприклад файли вихідних текстів та проміжні об'єктні файли).

7) На титульному листі неправильно вказана назва кафедри чи дисципліни.

8) Нема вихідного коду програми а є лише зібраний проект або навпаки, є лише вихідний код але нема зібраного проекту, який можна запускати на комп'ютері без встановленої системи проектування програм, що використовувалася для створення проекту. Програма відкомпільована (зібрана) в режимі відлагодження(debug), а не випуск(relis).

9) Невірно написана мета виконання курсового проекту або не написана взагалі.

10) Пояснювальна записка оформлена не у відповідності зі стандартом на оформлення (неправильний міжрядковий інтервал, невірний шрифт чи його розмір, не підписані рисунки та таблиці, неправильні абзацні виступи, не пронумеровані основні розділи або пронумеровані не основні, основні розділи починаються не з нової сторінки чи невірно оформлені їх заголовки, невірно оформлений зміст – не вказані номери сторінок або вказані діапазоном тощо).

11) Відсутні деякі розділи або невірний їх обсяг.

12) Нема перевірки на коректність вхідних даних в алгоритмі та/чи програмі. Не перевіряються заборонені операції (ділення на нуль, корінь з від'ємного числа тощо).

13) Назва програми, вікон залишена за замовчуванням. Нема своїх іконок(ярликів) для вікон та до програми.

14) Використовуються абсолютні шляхи при задаванні константних імен файлів з якими працює програма.

Перед остаточною здачею курсового проекту треба перевірити, його на відсутність вищезазначених помилок.

При виконанні курсового проекту треба дотримуватися кодексу академічної доброчесності університету [3]. Проект буде перевірятися на наявність академічного плагіату у відповідності з порядком проведення

перевірки кваліфікаційних робіт та індивідуальних завдань здобувачів вищої освіти на плагіат в Національному університеті “Чернігівська політехніка” [4].

2 ОФОРМЛЕННЯ КУРСОВОГО ПРОЕКТУ

Курсовий проект повинен бути оформлений у відповідності з вимогами ДСТУ 3008-95 та ЄСКД. Оформлення курсового проекту має вимоги наведені у Методичних рекомендацій щодо виконання та оформлення кваліфікаційних робіт здобувачів вищої освіти освітніх ступенів «бакалавр» і «магістр» Національного університету «Чернігівська політехніка» [5], сторінки 8-11, починаючи з абзацу: «Текст друкують за допомогою комп'ютера....»

2.1 Структура курсового проекту

Пояснювальна записка на курсовий проект повинна включати розділи:

- 1) титульний лист;
- 2) лист мети та завдання;
- 3) зміст;
- 4) вступ;
- 5) опис алгоритму;
- 6) алгоритм у вигляді блок-схеми;
- 7) текст програми;
- 8) опис тексту програми чи коментарі до програми;
- 9) тести та керівництво для користувача програми зі скріншотами;
- 10) висновки;
- 11) використана література.

2.2 Зміст курсового проекту

Титульний лист

На титульному листі вказується назва навчального закладу, назва кафедри, назва дисципліни, тема курсового проекту, хто виконав та перевірів проект, місто і рік. У додатку Б приведений приклад титульного листа а його зворотна сторона. На зворотній стороні титульного листа необхідно своїм підписом підтвердити відсутність плагіату у роботі, приклад тексту для підтвердження наведений в додатку В.

Лист мети та завдання

На цьому листі вказується мета курсового проекту (зазначена у вступі цих методичних вказівок) та приводиться завдання, отримане студентом від викладача, на основі якого виконується курсовий проект. Необхідно вклеїти саме той лист завдання, що отриманий від викладача з його підписом.

Зміст

Приводиться зміст курсового проекту, який повинен включати в себе всі розділи, крім титульного листа і листа мети та завдання, з вказуванням сторінок. На титульному листі номер сторінки не вказується, але в нумерації сторінок він враховується.

Вступ

Проводиться аналіз мети та завдання, можливі варіанти його вирішення (не менше трьох варіантів), обирається найкращий варіант вирішення завдання з його обґрунтуванням. Орієнтований обсяг 3-8 сторінок.

Опис алгоритму

Описується більш докладно обраний шлях вирішення завдання, наводяться математичні описи (формули) та перераховуються стандартні (запозичені) алгоритми, які використовуються у курсовому проекті з обов'язковим посиланням на джерело (джерело повинне бути вказане у списку літератури). Орієнтований обсяг 4-8 сторінок.

Алгоритм у вигляді блок-схеми

Безпосереднє зображення алгоритму у вигляді блок-схеми з використанням стандартних блоків. Не слід наводити детальний алгоритм, до окремого оператора. В підписах блоків треба вказувати дію, яку вони виконують, а не приводити відповідні оператори мови програмування. Також не слід і значно спрощувати блок-схему. Орієнтуватися необхідно на те, щоб можна було повністю зрозуміти нюанси алгоритму, авторські нововведення та здобутки. В той-же час не переобтяжувати блок-схему описом загальновідомих і широко розповсюджених алгоритмів які необхідно вказати скорочено

(наприклад в одному блоці). Слід пам'ятати, що абсолютно вся програма повинна бути представлена блок-схемою. Не припустимо наявність у тексті програми операторів, дія яких ніяким чином не відображена в алгоритмі. Орієнтований обсяг 5-10 сторінок.

Текст програми

Приводиться безпосередньо текст (листинг, роздруківка) програми. Необхідно роздрукувати вміст всіх файлів з вихідними текстами, що входять до проекту. Якщо текст програми великого обсягу, то його частково або повністю можна приводити в додатках меншим шрифтом, але щоб можна було прочитати не напружуючи зір. Якщо весь чи частина тексту програми розміщена у додатках, то в розділі необхідно зробити посилання на відповідні додатки.

Опис тексту програми чи коментарі до програми

Наводяться всі ідентифікатори (імена констант, міток, типів даних, змінних, підпрограм, введених автором проекту), які використовуються у програмі з обов'язковим зазначенням де і для чого вони використовуються, бажано привести їх в алфавітному порядку. Стандартні ідентифікатори мови програмування приводити не треба. Якщо один і той-же ідентифікатор застосовується в різних місцях з різною метою, про це обов'язково зазначити і вказати всі призначення відповідного ідентифікатора. Дозволяється цей розділ не виносити окремо, а розмістити його безпосередню в тексті програми у вигляді коментарю до кожного ідентифікатора, розміщеного там де він вперше зустрічається у тексті програми.

Тести та керівництво для користувача програми

Описується призначення програми, наводиться докладна інструкція по роботі з програмою, в якій описується склад програмного продукту (імена всіх файлів та каталогів, з яких складається програма з зазначенням їх розміру і призначення, розташування на диску), інтерфейс (зовнішній вигляд, кнопки керування та їх призначення), розроблюються та наводяться тести, виконання

яких дозволяє пересвідчитись у правильності роботи програми. Орієнтований обсяг 2-6 сторінок.

Висновки

Аналізується виконана робота, наскільки вона відповідає меті та завданню курсового проекту, які виникли складнощі, під час роботи над проектом. Якщо не вдалося в повному обсязі виконати завдання, вказати причину цього, якщо завдання було перевиконане, також це відзначити. Бажано вказати можливі шляхи удосконалення програми.

Використана література

Перераховується література, яка використовувалась під час написання курсового проекту. В використаній літературі треба наводити лише ту літературу яку використовувалась. В тому числі і джерела з Інтернету. На кожен літературу зі списку літератури в тексті пояснювальної записки повинно бути посилання там де з цієї літератури щось використовується. Посилання наводиться у вигляді цифр у квадратних дужках.

Список використаної літератури необхідно оформлювати у згідно відповідних правил оформлення списку літератури. Приклади можна подивитися в [5]. Приклади можливих варіантів правильного оформлення посилань на джерела в інтернеті наведені нижче.

1. Шкляр відмовився від Шевченківської премії [Електронний ресурс] // Українська правда. – 2011. – Режим доступу до ресурсу: <https://life.pravda.com.ua/culture/2011/03/4/74229/>.

2. Калькулятор кольорового маркування резисторів [Електронний ресурс] – Режим доступу до ресурсу: <https://radiotechlab.com.ua/kalkulyatori/>

3. Бот кафедри комп'ютерних наук та інформаційних технологій [Електронний ресурс] // ХАІ. – 2018. – Режим доступу до ресурсу: http://k302.khai.edu/index.php/telegrambot_ius/

Обов'язково зазначаються джерела з яких було запозичено той чи інший текст чи код програми (програми в інтернеті, курсові проекти інших студентів, тощо). Робота обов'язково буде перевірятися на плагіат і якщо буде знайдене запозичення без посилання на нього, робота буде забракована і потрібно буде брати нове завдання і писати нову програму. При безпосередньому написанні коду програми можна скористатися літературою [6-11].

3 КРИТЕРІЇ ОЦІНКИ КУРСОВОГО ПРОЕКТУ

Курсовий проект оцінюється максимум у 100 балів, оцінка складається з якості виконання проекту (відповідність завданню та повнота вирішення завдання до 40 балів), якості оформлення пояснювальної записки та своєчасності здачі (до 40 балів), захисту (до 20 балів).

Оцінка відмінно (90-100 балів) виставляється, якщо проект виконано в повному обсязі згідно завдання. Мають місце:

- 1) обґрунтований опис алгоритму, алгоритм у вигляді блок-схеми, спроектований оптимально, приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана оптимально, текст програми має необхідний опис чи коментарі;
- 3) є повний опис роботи програми та роботи з програмою для користувача;
- 4) пояснювальна записка оформлена згідно з вимогами стандартів та акуратно;
- 5) при захисті курсового проекту студент вільно орієнтується в матеріалі, який відноситься до теми курсового проекту та своїй записці, відповідає на запитання.

Можлива присутність деяких недоліків у вигляді описок та дрібних помилок при відповідях на запитання.

Оцінка добре (75-99 балів) – виставляється, якщо курсовий проект виконаний в повному обсязі згідно завдання. Мають місце:

- 1) обґрунтований опис алгоритму, сам алгоритм у вигляді блок-схеми, спроектований переважно оптимально, приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана переважно оптимально, текст програми має необхідний опис чи коментарі;
- 3) є повний опис роботи програми та роботи з програмою для користувача;
- 4) пояснювальна записка оформлена згідно з вимогами стандартів та в основному акуратно;

- 5) при захисті курсового проекту студент в основному орієнтується в матеріалі, який відноситься до теми курсового проекту та своїй записці, відповідає на більшу частину запитань.

Можлива присутність недоліків у вигляді описок, помилок, які не впливають на працездатність програми та помилок при відповідях на запитання.

Оцінка задовільно (60-74 бали) – виставляється, якщо робота виконана не в повному обсязі, але згідно завдання. Мають місце:

- 1) необґрунтований опис алгоритму, алгоритм у вигляді блок-схеми, спроектований не оптимально, не приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана не оптимально, текст програми не має необхідний опис чи коментарі;
- 3) є неповний опис роботи програми та роботи з програмою для користувача;
- 4) пояснювальна записка оформлена в основному згідно вимог стандартів але з помилками та неакуратно;
- 5) при захисті курсового проекту студент погано орієнтується в матеріалі який відноситься до теми курсового проекту та в своїй записці, відповідає на меншу частину запитань.

Можлива присутність помилок, які впливають на працездатність програми, або відсутність частини алгоритму, студент допускає грубі помилки при відповідях на запитання.

Оцінка незадовільно (0-59 балів) – виставляється, якщо робота виконана в неповному обсязі згідно технічного завдання, не згідно технічного завдання або зовсім не виконано. Мають місце:

- 1) відсутній опис алгоритму, сам алгоритм у вигляді блок-схеми або спроектований повністю не оптимально, не приведений, там де потрібно, математичний опис алгоритму;
- 2) програма написана не оптимально або взагалі неправильно, текст програми не має необхідний опис чи коментарі;
- 3) не має опису роботи програми та роботи з програмою для користувача;

- 4) при захисті курсового проекту студент не орієнтується в матеріалі, який відноситься до теми курсового проекту та в своїй записці, не відповідає на запитання.

4 ПОРАДИ ЩОДО ФОРМАТУВАННЯ ТЕКСТУ ЗВІТУ У MICROSOFT WORD

Вказані нижче поради розраховані лише на нагадування особливостей роботи текстового редактора Microsoft Word та не надають можливостей його вивчення.

На початку роботи над документом необхідно налаштувати поля, орієнтацію та розміри сторінки документа. У Word для цього слід користуватися вкладкою «Размітка сторінки». Згідно ДСТУ 3008-95 поля документа повинні бути: ліве, верхні та нижнє – не менш 2 см, праве – не менш 1 см, сторінка книжкова формату А4. Зовнішній вигляд вікон для налагодження параметрів сторінки представлений на рисунку 4.1.

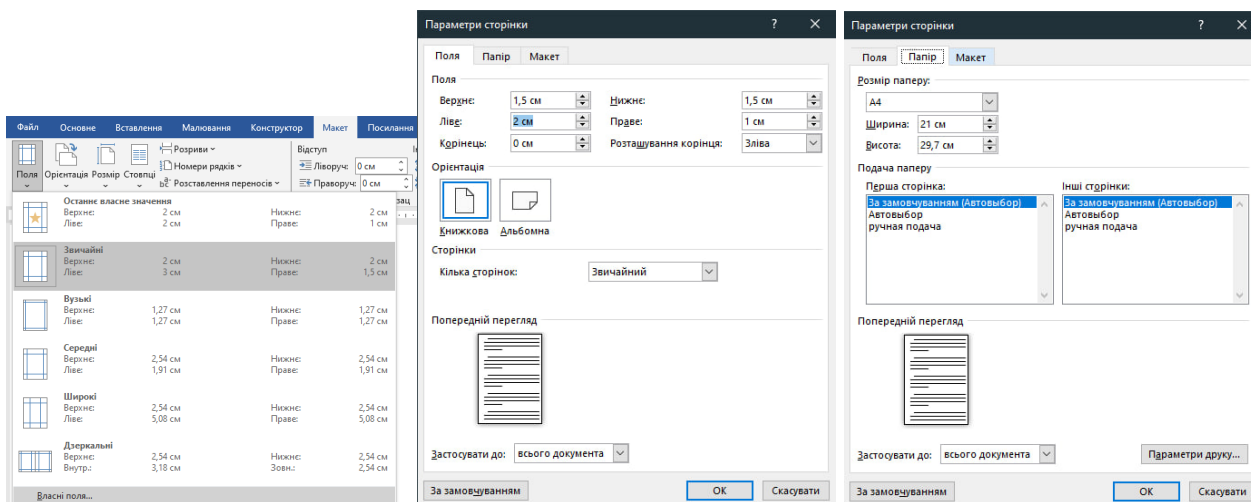


Рисунок 4.1 – Параметри сторінки

Текст документа повинен формуватися з використанням стилів. В Microsoft Word будь-який текст, формула або картинка мають стиль. Звичайний текст документа має стиль «Звичайний», заголовки – «Заголовок 1», «Заголовок 2», «Заголовок 3». Для інших елементів документа стилі варто створювати самостійно. У Word скорочений перелік стилів розташовано у вкладці «Гловна», для відкриття повного переліку стилів слід натиснути на кнопку 1 (див. рис.4.2).

У великих документах виконувати операції по форматуванню тексту шляхом його виділення з наступним обранням шрифту, розміру шрифту і т.д. для кожного абзацу не рекомендується. Слід наголосити, що форматування

тексту повинне проводитися з використанням стилів, тому на початку роботи з документом необхідно налаштувати стилі.

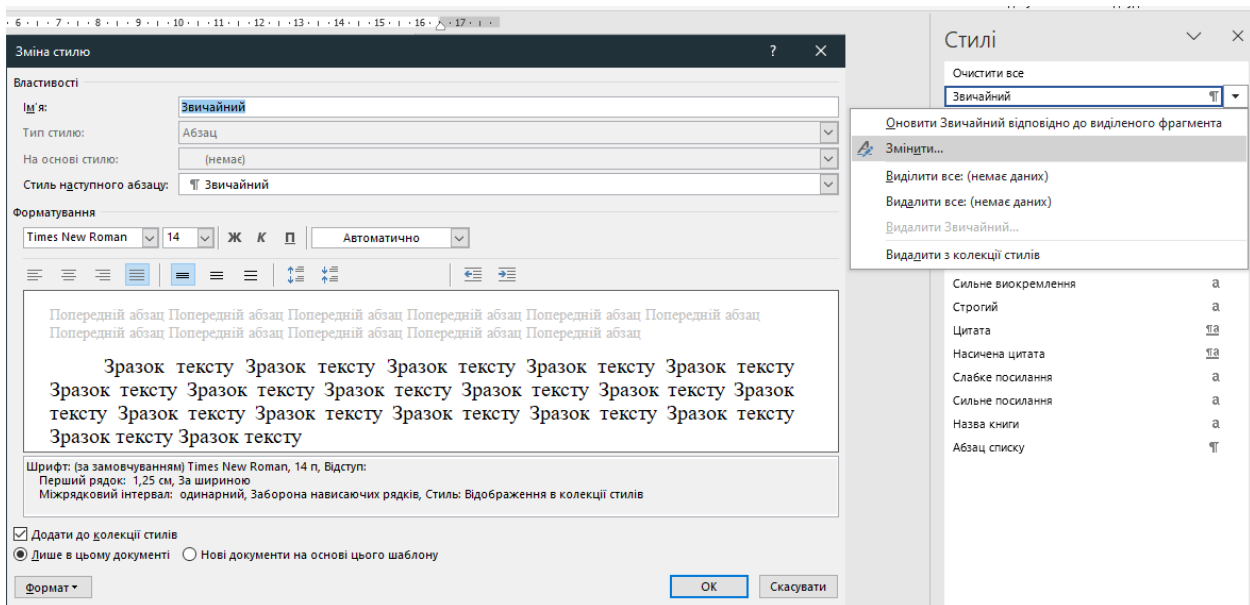


Рисунок 4.2 – Налаштування стилю «Звичайний» у Word

Налаштуємо стиль «Звичайний». Для цього обираємо його в списку та у меню, що з'явилося, обираємо «Змінити...» (рис. 4.2)

У вікні, що з'явиться, слід налаштувати стиль основного тексту документа так, щоб він відповідав вищезгаданому стандарту. Відповідно до стандарту основний текст документа повинен мати наступний стиль: шрифт – «Times New Roman», розмір – 14, накреслення – звичайний, видозміна – немає; налаштування абзацу: відступи ліворуч і праворуч – 0 см, перший рядок – відступ 1,25 см (допускається від 1 см до 1,5 см, але однаковий у всьому документі), інтервал перед – 0 пт і після – 0 пт, міжрядковий – полуторний. Вікно для налаштування шрифту викликається по натисканню кнопки «Формат / Шрифт», а вікно для форматування абзацу викликається по натисканню кнопки «Формат / Абзац», відповідні вікна зображені на рисунку 4.3.

На цьому налаштування стилю «Обычный» завершено. Далі аналогічно налаштовуються стилі «Заголовок 1», «Заголовок 2», «Заголовок 3» та «Заголовок 4».

Стиль «Заголовок 1» повинен мати налаштування шрифту: шрифт – «Times New Roman», розмір – 14, накреслення – напівжирний, видозміна – «всі

прописні»; налаштування абзацу: відступи ліворуч і праворуч – 0 см, перший рядок – 0 см, інтервал перед і після – 28 пт, міжрядковий – полуторний.

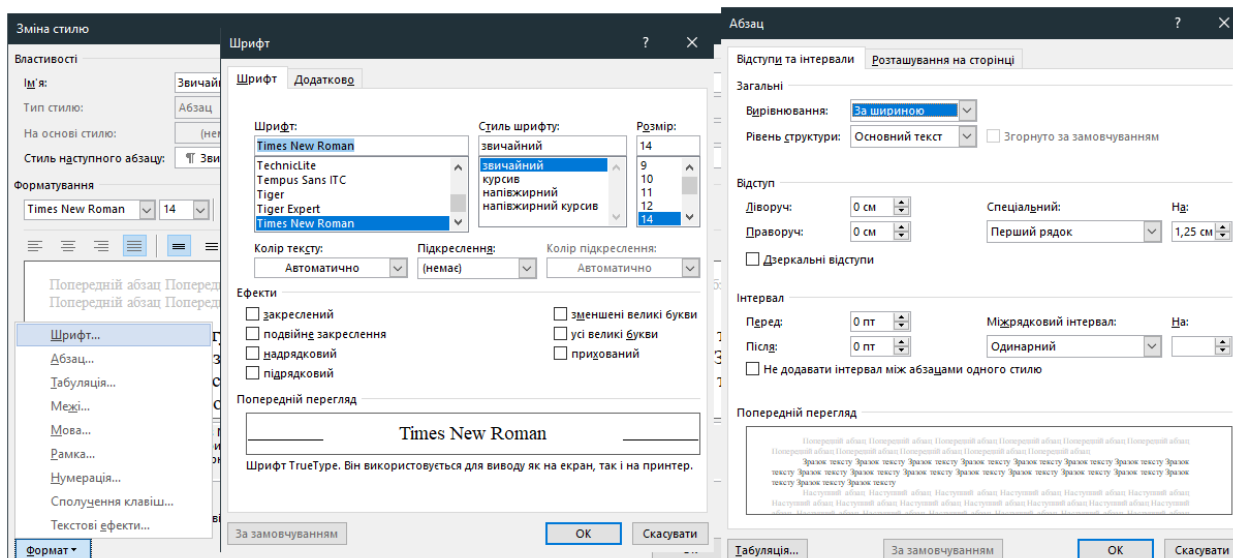


Рисунок 4.3 – Налаштування стилю «Звичайний»

Стиль «Заголовок 2» повинен мати налаштування шрифту: шрифт – «Times New Roman», розмір – 14, накреслення – напівжирний, видозміна – немає; налаштування абзацу: відступи ліворуч і праворуч – 0 см, перший рядок – відступ 1,25 см, інтервал перед – 14 пт і після – 0 пт, міжрядковий – полуторний.

Стиль «Заголовок 3» й «Заголовок 4» повинен мати налаштування шрифту: шрифт – «Times New Roman», розмір – 14, накреслення – напівжирний, видозміна – немає; налаштування абзацу: відступи ліворуч і праворуч – 0 см, перший рядок – відступ 1,25 см, інтервал перед – 0 пт і після – 0 пт, міжрядковий – полуторний.

Імена стилів заголовків можуть бути змінені відповідно до термінів, які використовуються в ДСТУ 3008-95, на «Розділ», «Підрозділ», «Пункт» і «Підпункт».

Для форматування рисунків необхідно створити новий стиль. Нехай він називається «Рисунки». Для того, щоб його створити у Word необхідно натиснути кнопку «Створити стиль...» під списком стилів.

Нагадаємо, що рисунки необхідно розміщувати посередині аркуша, під рисунком повинна бути напис «Рисунок» із номером рисунку назвою за

необхідністю, що вказується після тире. Після та до рисунку повинне бути залишена відстань, яка дорівнює одному рядку.

Для створеного стилю слід:

- вказати ім'я стилю «Рисунки»,
- вказати стиль, на якому він буде заснований «Звичайний» (це означає що всі налаштування стилю «Звичайний» будуть діяти, крім тих, які будуть зазначені),
- обрати вирівнювання тексту по центру,
- налаштувати форматування абзацу, див. рис. **Ошибка! Источник ссылки не найден.** (відступи ліворуч і праворуч – 0 см, перший рядок – (немає), міжрядковий – полуторний, інтервал перед – 21 пт і після – 21 пт (14пт·1.5), розбивка на сторінки – не розривний абзац).

На рисунку 4.5 приведено зовнішній вигляд вікон Word при налаштуванні стилю.

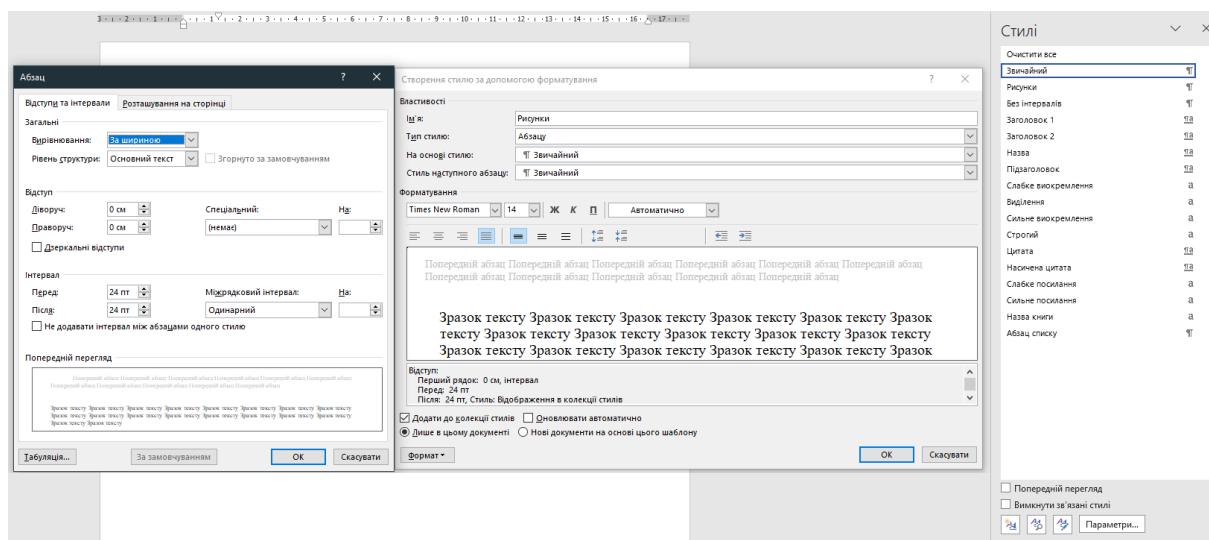


Рисунок 4.5 – Створення стилю «Рисунки» у Word

Рисунок і його назва повинні розділятися не натисканням «Enter», а натисканням «Shift+Enter». У цьому випадку обидві рядки вважаються одним абзацом.

Для форматування формул необхідно створити новий стиль. Нехай він називається «Формули».

Слід зазначити, що за стандартом формули необхідно розташовувати посередині аркушу, праворуч у дужках повинен вказуватися номер формули, на

наступному рядку після слова «де» повинні розшифруватися прийняті позначення у формулі, позначення вказуються в порядку їхнього згадування у формулі.

Аналогічно рисунку 4.5 створюємо новий стиль, вводимо ім'я стилю «Формули», обираємо вирівнювання тексту по лівому краю, налаштовуємо форматування абзацу (відступи: ліворуч, праворуч – 0 см, перший рядок – немає; інтервали: перед – 21пт, після – 21пт), налаштовуємо табуляцію (див. рис. 4.6).

Для налаштування табуляції слід натиснути «Формат / Табуляція...».

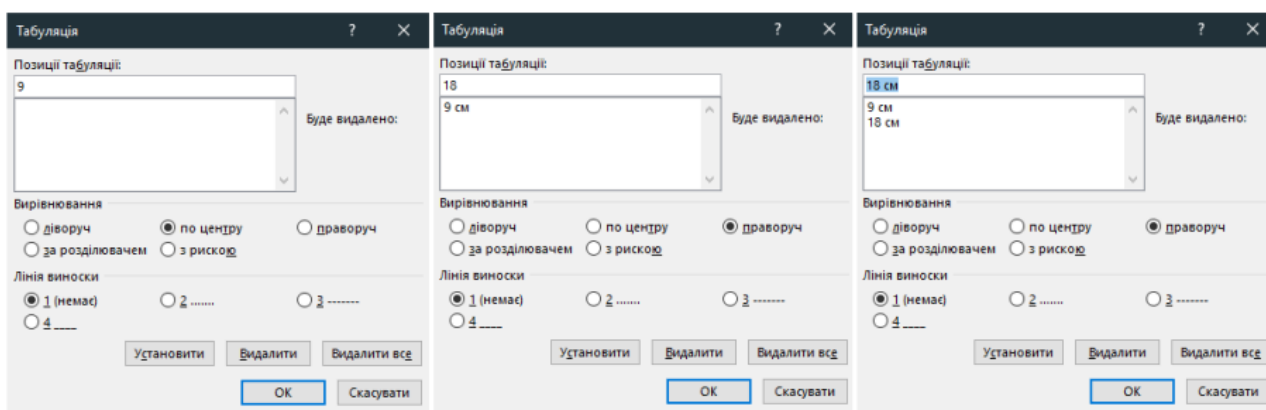


Рисунок 4.6 – Налаштування табуляції

Перший табулятор виставляється на відстань 9 см від краю (центр сторінки, розрахований за формулою (1.1)) з вирівнюванням – «по центру». Він служить для вирівнювання формули по центрі сторінки. Позиція центра сторінки розраховується за формулою

$$\rightarrow X = \frac{W_{A4} - L - R}{2}, \rightarrow (1.1)$$

де X – позиція табулятора формули;

W_{A4} – ширина сторінки;

L – розмір лівого поля;

R – розмір правого поля.

Другий табулятор виставляється на відстань 18 см від краю (відстань від лівого поля до правого) з вирівнюванням «по правому краю», він призначений для вирівнювання номера формули. Табулятори вставляються в текст

натисканням клавіші «TAB» на клавіатурі. Стрілки біля формули (1.1) вказують на необхідність розташування в цьому місці табулятора. Окремі рядки відокремлюються натисканням «Shift + Enter».

Для шифрування прийняті позначення у формулі слід створити додатковий стиль. Аналогічно пункту **Ошибка! Источник ссылки не найден.** створюємо новий стиль, уводимо ім'я стилю «де», обираємо вирівнювання тексту по правому краю, налаштовуємо форматування абзацу (відступи: ліворуч, праворуч – 0 см, перший рядок: виступ – 1 см; інтервали: перед – 0пт, після – 0пт), налаштовуємо табуляцію (один табулятор: позиція – 1 см, вирівнювання – по правому краю, заповнювач – нема).

5 ДОПОМІЖНИЙ МАТЕРІАЛ

У таблиці 5.1 приведені кодування символів по стандарту ASCII для російської кодової таблиці з вказуванням символу та його десяткового і шістнадцяткового кодів, а в таблиці 5.2 у компактному вигляді для шістнадцяткового коду. Так як у таблицях символи будуть відображатися тільки у віконному режимі роботи програми під керуванням операційної системи Windows. У DOS, а також у повноекранному режимі Windows деякі символи будуть іншими.

Таблиця 5.1– Коди символів ASCII

Chr	Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex
	0	00		32	20	@	64	40	`	96	60
☺	1	01	!	33	21	A	65	41	a	97	61
☹	2	02	"	34	22	B	66	42	b	98	62
♥	3	03	#	35	23	C	67	43	c	99	63
♦	4	04	\$	36	24	D	68	44	d	100	64
♣	5	05	%	37	25	E	69	45	e	101	65
♠	6	06	&	38	26	F	70	46	f	102	66
•	7	07	'	39	27	G	71	47	g	103	67
◼	8	08	(40	28	H	72	48	h	104	68
○	9	09)	41	29	I	73	49	i	105	69
◻	10	0a	*	42	2a	J	74	4a	j	106	6a
♂	11	0b	+	43	2b	K	75	4b	k	107	6b
♀	12	0c	,	44	2c	L	76	4c	l	108	6c
♪	13	0d	-	45	2d	M	77	4d	m	109	6d
♫	14	0e	.	46	2e	N	78	4e	n	110	6e
♪	15	0f	/	47	2f	O	79	4f	o	111	6f
▶	16	10	0	48	30	P	80	50	p	112	70
▲	17	11	1	49	31	Q	81	51	q	113	71
↕	18	12	2	50	32	R	82	52	r	114	72
↕	19	13	3	51	33	S	83	53	s	115	73
↕	20	14	4	52	34	T	84	54	t	116	74
§	21	15	5	53	35	U	85	55	u	117	75
┃	22	16	6	54	36	V	86	56	v	118	76
↕	23	17	7	55	37	W	87	57	w	119	77
↕	24	18	8	56	38	X	88	58	x	120	78
↕	25	19	9	57	39	Y	89	59	y	121	79
→	26	1a	:	58	3a	Z	90	5a	z	122	7a

Закінчення таблиці 4.1

Chr	Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex	Chr	Dec	Hex
←	27	1b	;	59	3b	[91	5b	{	123	7b
L	28	1c	<	60	3c	\	92	5c		124	7c
↔	29	1d	=	61	3d]	93	5d	}	125	7d
▲	30	1e	>	62	3e	^	94	5e	~	126	7e
▼	31	1f	?	63	3f	_	95	5f	△	127	7f
А	128	80	а	160	a0	␣	192	c0	⋮	176	b0
Б	129	81	б	161	a1	␣	193	c1	⋮	177	b1
В	130	82	в	162	a2	␣	194	c2	⋮	178	b2
Г	131	83	г	163	a3	␣	195	c3	␣	179	b3
Д	132	84	д	164	a4	␣	196	c4	␣	180	b4
Е	133	85	е	165	a5	␣	197	c5	␣	181	b5
Ж	134	86	ж	166	a6	␣	198	c6	␣	182	b6
З	135	87	з	167	a7	␣	199	c7	␣	183	b7
И	136	88	и	168	a8	␣	200	c8	␣	184	b8
Й	137	89	й	169	a9	␣	201	c9	␣	185	b9
К	138	8a	к	170	aa	␣	202	ca	␣	186	ba
Л	139	8b	л	171	ab	␣	203	cb	␣	187	bb
М	140	8c	м	172	ac	␣	204	cc	␣	188	bc
Н	141	8d	н	173	ad	␣	205	cd	␣	189	bd
О	142	8e	о	174	ae	␣	206	ce	␣	190	be
П	143	8f	п	175	af	␣	207	cf	␣	191	bf
Р	144	90	р	224	e0	␣	208	d0	␣	240	f0
С	145	91	с	225	e1	␣	209	d1	␣	241	f1
Т	146	92	т	226	e2	␣	210	d2	␣	242	f2
У	147	93	у	227	e3	␣	211	d3	␣	243	f3
Ф	148	94	ф	228	e4	␣	212	d4	␣	244	f4
Х	149	95	х	229	e5	␣	213	d5	␣	245	f5
Ц	150	96	ц	230	e6	␣	214	d6	␣	246	f6
Ч	151	97	ч	231	e7	␣	215	d7	␣	247	f7
Ш	152	98	ш	232	e8	␣	216	d8	␣	248	f8
Щ	153	99	щ	233	e9	␣	217	d9	␣	249	f9
Ъ	154	9a	ъ	234	ea	␣	218	da	␣	250	fa
Ы	155	9b	ы	235	eb	■	219	db	␣	251	fb
Ь	156	9c	ь	236	ec	■	220	dc	№	252	fc
Э	157	9d	э	237	ed	■	221	dd	α	253	fd
Ю	158	9e	ю	238	ee	■	222	de	■	254	fe
Я	159	9f	я	239	ef	■	223	df	■	255	ff

Додаток А - Приклад завдання на курсовий проект**ЗАВДАННЯ**

на курсовий проект

Студента групи

ВТ–2ХХ

Іванов В.В.

По курсу “Основи програмування та обчислювальної техніки”

**Написати програму емуляції на комп’ютері механічного
годинника з будильником**

Програма повинна містити:

1. Годинникову, хвилину, секундну стрілки, стрілку будильника та відповідні позначки для зчитування поточного часу.
2. Після запуску програми встановлювати на циферблаті поточний час з операційної системи.
3. Мати можливість встановити новий час на циферблаті зі зміною його в операційній системі, якщо остання це дозволяє.
4. Мати можливість встановлювати час спрацювання будильника.
5. Мати кнопку вимикання будильника та обмежувати час його звучання, якщо не була натиснута кнопка вимикання будильника.
6. Мати можливість вибору декількох мелодій (звуків) для будильника.

Додаток Б - Приклад титульного листа курсового проекту

Міністерство освіти і науки України
Національний університет «Чернігівська політехніка»

Кафедра електроніки, автоматики, робототехніки та мехатроніки

КУРСОВИЙ ПРОЕКТ

з дисципліни

”Основи програмування та обчислювальної техніки”

на тему: ”*(назва теми)*”

Виконав:

ЗВО групи (*шифр групи*)

(*П.І.Б. ЗВО*)

Перевірив:

(*П.І.Б. викладача*)

Чернігів (*рік*)

Додаток В - зворотна сторона титульного листа

Я, _____, підтверджую, що дана робота є моєю власною письмовою роботою, оформленою з дотриманням цінностей та принципів етики і академічної доброчесності відповідно до Кодексу академічної доброчесності Національного університету «Чернігівська політехніка». Я не використовував/ла жодних джерел, крім процитованих, на які надано посилання в роботі.

(Дата)

(Підпис)

РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. ДСТУ 3008:2015 Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання
2. Qt for Windows – Deployment. – [Електронний ресурс]. – Режим доступу: <https://doc.qt.io/qt-5/windows-deployment.html>.
3. Кодекс академічної доброчесності Національного університету «Чернігівська політехніка» (нова редакція). Затверджено Вченою радою Національного університету «Чернігівська політехніка» 31 травня 2021 року, протокол № 5. [Електронний ресурс]. – Режим доступу: <https://stu.cn.ua/wp-content/uploads/2021/06/kodeks-akademichnoyi-dobrochesnosti-nova-redakciya.pdf>.
4. Порядок проведення перевірки кваліфікаційних робіт та індивідуальних завдань здобувачів вищої освіти на плагіат в Національному університеті «Чернігівська політехніка» (нова редакція). Затверджено Вченою радою Національного університету «Чернігівська політехніка» 31 травня 2021 року, протокол № 5. – [Електронний ресурс]. – Режим доступу: <https://stu.cn.ua/wp-content/uploads/2021/06/poryadok-provedennya-perevirky-kvalifikacijnyh-robot-ta-indyvidualnyh-zavdannova-redakciya.pdf>.
5. Методичні рекомендації щодо виконання та оформлення кваліфікаційних робіт здобувачів вищої освіти освітніх ступенів «бакалавр» і «магістр» Національного університету «Чернігівська політехніка» – [Електронний ресурс]. – Режим доступу: <https://stu.cn.ua/wp-content/uploads/2021/03/p-mr-kvalrob.pdf>.
6. Ковалюк Т.В. Алгоритмізація та програмування. Підручник. — Львів: «Магнолія 2006», 2013. – 400 с.
7. Засоби відображення вимірювальної інформації у середовищі програмування QT CREATOR / А. Озгович, І. Ліхновський, О. Тищенко, А. Кузій // Вимірювальна техніка та метрологія - Львів : Львівська політехніка, - 2014. - №75. - С.92-97. 2.
8. Козак, Л.І. Основи програмування : навч. посіб. : рек. МОН України / Л.І. Козак, І.В. Костюк, С.Л. Стасевич. - Львів. : Новий Світ - 2019. - 325 С.
9. Грицюк Ю.І., Рак Т.Є. Програмування мовою С++ Навч. пос. – Львів: Вид-во Львівського ДУ БЖД, 2011. – 292 с.
10. Jasmin Blanchette, Mark Summerfield. С++ GUI Programming with Qt 4. Second Edition. – Prentice Hall, 2008. – 752 p.
11. Qt – [Електронний ресурс]. – Режим доступу : <https://www.qt.io/>