

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«ДНІПРОВСЬКА ПОЛІТЕХНІКА»



НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ЕЛЕКТРОЕНЕРГЕТИКИ
ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
Кафедра інформаційних технологій та комп'ютерної інженерії

І. М. Гаркуша

КУРСОВА РОБОТА З ПРОГРАМУВАННЯ

Методичні рекомендації

для здобувачів ступеня бакалавра
освітньо-професійної програми «Комп'ютерна інженерія»
спеціальності 123 Комп'ютерна інженерія

Дніпро
НТУ «ДП»
2025

Гаркуша І. М.

Курсова робота з програмування [Електронний ресурс] : методичні рекомендації для здобувачів ступеня бакалавра освітньо-професійної програми «Комп'ютерна інженерія» спеціальності 123 Комп'ютерна інженерія / І. М. Гаркуша ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2025. – 23 с.

Автор

І. М. Гаркуша, канд. техн. наук, доц.

Затверджено науково-методичною комісією спеціальності 123 Комп'ютерна інженерія (протокол № 4 від 17.04.2025) за поданням кафедри Інформаційних технологій та комп'ютерної інженерії (протокол № 13 від 11.04.2025).

Наведено програму курсової роботи з дисципліни «Програмування», а також методичні рекомендації щодо її організації та проведення. Розкриті компоненти, що сприяють формуванню професійних компетентностей, а також підкріпленню комплексу компетентнісних характеристик здобувачів (знань, умінь/навичок, комунікацій, відповідальності і автономії) у процесі виконання курсової роботи.

Методичні рекомендації містять загальні вимоги щодо складання звіту про виконання курсової роботи.

Відповідальний за випуск завідувач кафедри інформаційних технологій та комп'ютерної інженерії В. В. Гнатушенко, д-р техн. наук, проф.

ЗМІСТ

ВСТУП	4
1. МЕТА РОБОТИ, ЗАДАЧІ ТА РЕЗУЛЬТАТИ НАВЧАННЯ	5
2. ОРГАНІЗАЦІЯ ПРОВЕДЕННЯ КУРСОВОЇ РОБОТИ	7
3. СТРУКТУРА ТА ВИМОГИ ДО ОФОРМЛЕННЯ ЗВІТУ	9
4. ВАРІАНТИ ЗАВДАНЬ ТА РЕКОМЕНДАЦІЇ	12
5. КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ	19
ПЕРЕЛІК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ	21

ВСТУП

Сучасна розробка програмного забезпечення (ПЗ) вважається складним процесом, що включає велику кількість різноманітних науково-практичних аспектів, методів, методологій, технологій та інструментів. Існує велика кількість різних мов програмування, що направлені на ефективне рішення певного кола задач. Існують також фактично універсальні мови, використовуючи які можна вирішувати будь-які складні завдання. До однієї такої мови відноситься високорівнева об'єктно-орієнтована мова програмування C++. Протягом свого існування C++ займає верхні рядки серед світових мов програмування та постійно розвивається. Станом на початок 2025 року відомі стандарти ISO/IEC C++, що описують різноманітні діалекти та особливості мови: C++98, C++03, C++11, C++14, C++17, C++20, C++23, C++26 (в процесі підготовки).

Основними цілями курсової роботи є:

- вміння аналізувати предметну область та проводити об'єктно-орієнтований аналіз;
- закріплення знань та навичок, щодо опису алгоритмів за допомогою схем алгоритмів;
- закріплення знань та навичок щодо розробки програм мовою програмування C++, які були набуті під час опанування таких курсів, як “Програмування” та “Об'єктно-орієнтоване програмування”;
- посилення культури документування та висловлення власних думок і творчої складової щодо створеної програмної розробки.

В процесі виконання курсової роботи певна увага приділяється процесу кодування програми, отриманої за варіантом завдання мовою C++.

Тривалість курсової роботи визначається затвердженим навчальним планом за спеціальністю 123 “Комп'ютерна інженерія”. Конкретні терміни проведення курсової роботи встановлюються освітньо-професійною програмою (ОПП) “Комп'ютерна інженерія” та навчальними планами, які затверджуються щорічно керівництвом Національного технічного університету “Дніпровська політехніка”.

1. МЕТА РОБОТИ, ЗАДАЧІ ТА РЕЗУЛЬТАТИ НАВЧАННЯ

Основною метою курсової роботи є формування компетентностей щодо процесу програмування різноманітних програм мовою C++.

Згідно з ОПП підготовки здобувачів спеціальності 123 “Комп’ютерна інженерія”, інтегральна компетентність полягає в здатності “розв’язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в галузі інформаційних технологій або навчання, що передбачає застосування теорій та методів комп’ютерної інженерії і характеризується комплексністю та невизначеністю умов”.

Програмування надає певні інструменти, які використовуються для розв’язання широкого кола задач як для потреб науки, так і виробництва. Безпосередньо курсова робота готує здобувачів для розробки певної частини кваліфікаційної випускової роботи бакалавра, є базою для інших дисциплін плану навчання, а також закріплює знання та уміння здобувачів з дисциплін “Програмування” та “Об’єктно-орієнтоване програмування”.

Курсова робота повинна закріпити знання здобувачів щодо використання стандартної бібліотеки C++, а також уміння будувати та реалізовувати алгоритми, використовувати інструменти збірки програм мовою C++ для виконання різноманітних задач, використовувати сторонні бібліотеки.

До спеціальних, фахових компетентностей в ОПП, які пов’язані з цією курсовою роботою, віднесені:

Z3. Здатність застосовувати знання у практичних ситуаціях.

P2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.

P3. Здатність створювати системне та прикладне програмне забезпечення комп’ютерних систем та мереж.

Результатами навчання з курсової роботи за ОПП є:

N10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати, типове для спеціальності обладнання.

N22. Вміти обґрунтовано обирати структуру та обладнання комп’ютерних або кіберфізичних систем та розробляти для них програмне забезпечення.

Згідно з ОПП та навчальним планом, курсова робота з програмування проводиться у здобувачів у 6-й чверті 2-го курсу навчання.

Основними задачами курсової роботи з програмування в рамках спеціальності 123 “Комп’ютерна інженерія”, враховуючи ОПП, є наступні.

1. Закріпити практичні навички щодо програмування мовою C++ для вирішення певних завдань.

2. Виконати опис програми, яку потрібно розробити та ключових алгоритмів, що реалізує програма.

3. Представити програмний код.

4. Запропонувати тести для перевірки працездатності розробки та відповідності виконання поставленому завданню.

2. ОРГАНІЗАЦІЯ ПРОВЕДЕННЯ КУРСОВОЇ РОБОТИ

Курсова робота виконуються здобувачами як на базі обладнання комп'ютерних класів випускової кафедри, так і на власному обладнанні за бажанням здобувача. Зокрема, використовуються комп'ютери на базі сучасних процесорів з об'ємами оперативної пам'яті 4 Гб або більше.

Для виконання поставлених задач курсової роботи здобувач використовує версії ПЗ:

– операційна система (ОС) MS Windows 10/11 (або новіша) або GNU/Linux-сумісна ОС (наприклад: Ubuntu 24.04 чи Debian 12.10 (або новіша) чи подібна), або різновид ОС Apple macOS;

– середовище розробки MS Visual Studio Community edition 2022 (або новіше) або Code::Blocks 25.03 (або новіше), або Xcode 16 (або новіше у випадку Apple-платформ), або Eclipse 2025-03 (або новіше) з компіляторами GNU GCC/G++, MS Visual C++ Compiler, Clang, які йдуть або із середовищами розробки, або встановлюються окремо;

– програма-браузер, наприклад, Mozilla FireFox або Google Chrome, або подібна.

У разі дистанційного навчання, здобувачі виконують завдання курсової роботи у віддаленому режимі за місцем перебування. При цьому, керівник курсової роботи періодично спілкується із здобувачами за допомогою різноманітних програм, наприклад таких, як Microsoft Teams або електронна пошта.

До кінця терміну виконання курсової роботи, здобувач зобов'язаний скласти звіт, який містить всі етапи та результати створення програми згідно обраного варіанту завдання та захистити його перед викладачем. Викладач, згідно з критеріями оцінювання, зараховує здобувачу відповідні бали. Допускається комплексне виконання курсової роботи парою здобувачів.

Здобувачі зобов'язані:

1) пройти індивідуальний інструктаж з техніки безпеки на робочому місці в комп'ютерному класі кафедри;

2) дотримуватися інструкцій з охорони праці для працівників своєї професії;

3) при зміні робочого місця пройти додатковий інструктаж;

4) виконувати всі рекомендації з охорони праці, які надає викладач;

5) під час дії карантинних заходів, здобувач повинен дотримуватися всіх правил поведінки та безпеки, встановлених або рекомендованих Міністерством охорони здоров'я України, а також внутрішніми розпорядженнями та наказами керівництва НТУ “Дніпровська політехніка”.

Оскільки основна робота здобувача зосереджена за робочим місцем, яке обладнане комп'ютером, що під'єднаний до електричної мережі, то перш за все потрібно виконувати всі норми та правила з техніки безпеки і охорони праці, які прописані у вимогах щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями.

У разі багаторазового порушення здобувачем правил техніки безпеки та охорони праці, питання про подальше використання техніки комп'ютерного класу та перебування в класі цього здобувача розглядається керівництвом кафедри.

3. СТРУКТУРА ТА ВИМОГИ ДО ОФОРМЛЕННЯ ЗВІТУ

Звіт оформлюється у вигляді окремого документу на аркушах паперу формату А4 державною мовою згідно загальних правил оформлення звітної документації, що встановлені Національним стандартом України ДСТУ 3008:2015. Текст звіту пишеться від третьої особи та присвячений опису виконання певного варіанту завдання.

Структура звіту з виконаної курсової роботи:

Титульний аркуш

Зміст

Вступ

Розділ 1. Огляд рішення завдання

Розділ 2. Реалізація програмного рішення

Висновки

Перелік використаних джерел

Додаток А. Текст коду програми

Сторінки звіту повинні мати поля: ліве – 30 мм, праве – 15 мм, верхнє – 20 мм, нижнє – 20 мм.

Властивості звичайного тексту звіту:

- шрифт: Times New Roman, кегль 14 (для заголовків першого рівня – 16, для номерів сторінок – 12);
- міжрядковий інтервал: 1,5;
- орієнтовна кількість рядків на сторінку: ~28-30;
- орієнтовна кількість знаків у рядку: до ~65;
- абзацний відступ: 1,25 см;
- вирівнювання тексту: по ширині сторінки;
- вирівнювання рисунків, таблиць, математичних виразів: по центру сторінки.

Властивості тексту кодів програми:

- один із будь-яких шрифтів: Source Code Pro, Monospace, Consolas, DejaVu Sans Mono, Courier New, Monospaced, Menlo, SF Mono або подібні моноширинні шрифти, кегль 9 або 10;
- міжрядковий інтервал: 1;
- абзацний відступ: 1,25 см;

– вирівнювання тексту на сторінці: ліворуч.

Звіт починається з титульного аркуша, приклад якого представлений на рис. 3.1. У разі виконання комплексного завдання парою здобувачів, замість рядку КУРСОВА РОБОТА пишуть рядок КОМПЛЕКСНА КУРСОВА РОБОТА. У цьому випадку кожний із здобувачів робить особистий звіт, в якому потрібно чітко описувати власно розроблену комплексну складову роботи за варіантом завдання.

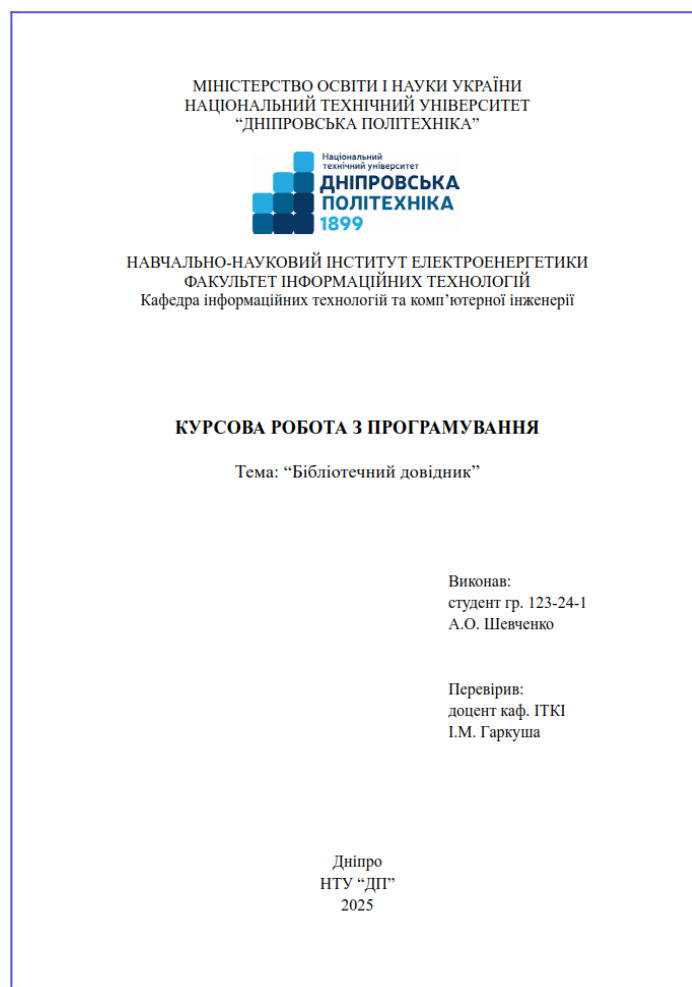


Рис. 3.1. Зразок титульного аркуша звіту виконаної курсової роботи

У вступі здобувач надає короткий опис переваги використання мови програмування C++, її стандартної бібліотеки та інших сторонніх бібліотек, останніх стандартів мови C++, нововведень, інструментів та технологій розробки мовою C++, дає опис завдання, а також розписує задачі, які на його думку потрібно виконати для створення заданої розробки.

Обсяг вступу не більше трьох сторінок.

У першому розділі, в окремих підрозділах, повинен бути розміщений літературний огляд рішення обраного завдання, подається огляд існуючих аналогів розробки.

В окремому підрозділі першого розділу подається опис бачення побудови програми, а саме наводиться структурне уявлення програми (можливо її умовної архітектури), подається огляд ключових алгоритмів програми, описаних за допомогою схем алгоритмів за вимогами Національного стандарту України ДСТУ ISO 5807:2016 (ISO 5807:1985, IDT), подаються обґрунтування використання сторонніх технологій (якщо такі є) окрім стандартної бібліотеки мови C++.

Загальний обсяг першого розділу не більше 7-ми сторінок.

У розділі реалізації програмного рішення здобувач надає опис інтерфейсу користувача програми та інструкцію по використанню програми. В окремому підрозділі описується методика тестування програми та результати тестування.

Загальний обсяг другого розділу не більше 25-ти сторінок.

Структурований за модулями програмний код реалізації програми вноситься окремо в Додаток А звіту курсової роботи.

У певних частинах звіту є доречним посилання на потрібні номери рядків коду програми з додатку А.

При описі програми у розділах 1 та 2 рекомендується користуватися матеріалами лекційних курсів “Програмування” та “Об’єктно-орієнтоване програмування”. Доречним є також використання додаткових літературних джерел.

У висновках здобувач надає короткий опис результатів роботи розробленої програми та її переваги перед існуючими (якщо такі є) аналогами.

Обов’язковим є надання переліку використаних джерел, на які здобувач робив посилання у тексті звіту. Бібліографічні посилання роблять згідно вимог Національного стандарту України ДСТУ 8302:2015.

Правила оформлення рисунків, таблиць, математичних виразів та посилань на них в тексті, а також правила нумерації розділів, підрозділів, пунктів чітко описані в Національному стандарті України ДСТУ 3008:2015.

4. ВАРІАНТИ ЗАВДАНЬ ТА РЕКОМЕНДАЦІЇ

Здобувач обирає один із варіантів завдань, які представлені у таблиці 4.1. Рекомендації по виконанню варіантів завдань надані в таблиці 4.2. У разі обрання складної теми, можливо виконання роботи в парі (варіант комплексної курсової роботи). В цьому випадку потрібно у звіті чітко розписати, яку з частин роботи було виконано певним здобувачем. Кожний здобувач пари робить окремий звіт за своєю частиною виконаної роботи.

Таблиця 4.1

Варіанти тем курсової роботи з програмування

<i>№ з/п</i>	<i>Назва теми</i>	<i>Завдання</i>
1.	Бібліотечний довідник	Розробити мовою C++ консольну програму, що створює та обробляє базу даних книжок бібліотеки.
2.	Каталог автозапчастин	Розробити мовою C++ консольну програму, яка дозволить звертатися до каталогу певних запчастин та повертати інформацію про їх характеристики.
3.	База даних автомобілістів	Розробити мовою C++ консольну програму, яка дозволить працювати з базою даних автомобілів та їх власників.
4.	Обробка файлу закладок браузера	Розробити мовою C++ консольну програму, що дозволяє проводити опрацювання історії закладок браузера Mozilla Firefox, що розміщені у файлі bookmarks.html.
5.	Web-орієнтована пошукова система	Розробити мовою C++ програму, що реалізує CGI-інтерфейс Web-серверу для пошуку певних товарів з локальної бази даних.
6.	Web-орієнтована програма малювання геометричних фігур	Розробити мовою C++ програму, що реалізує CGI-інтерфейс Web-серверу для малювання різноманітних геометричних фігур.
7.	Програма малювання геометричних фігур	Розробити мовою C++ програму з GUI, що виконує малювання різноманітних геометричних фігур у головному вікні.
8.	Інформація про BMP-файл	Розробити мовою C++ консольну програму дешифрування бінарних файлів формату

		Windows Bitmap.
9.	Програма моделювання звукової азбуки Морзе	Розробити мовою C++ консольну програму навчання звуковій азбуці Морзе (символи української абетки).
10.	Кулінарна книга	Розробити мовою C++ консольну програму надання кулінарних рецептів.
11.	Web-орієнтований каталог автозапчастин	Розробити мовою C++ програму, що реалізує CGI-інтерфейс Web-серверу для звернення до каталогу певних автозапчастин та повернення інформації про них.
12.	Web-орієнтований бібліотечний довідник	Розробити мовою C++ програму, що реалізує CGI-інтерфейс Web-серверу для створення та обробки бази даних книжок бібліотеки.
13.	Калькулятор	Розробити мовою C++ консольну програму-калькулятор. Програма повинна виконувати операції та функції: +, -, *, /, \log_2 , \log_{10} , sqrt, pow над двома векторами або над вектором чисел з плаваючою комою (у випадку функцій), які передаються програмі як аргументи через командний рядок. Результат виводиться у вікно терміналу консолі.
14.	Файловий калькулятор	Розробити мовою C++ консольну програму-калькулятор, який виконує операції та функції: +, -, *, /, \log_2 , \log_{10} , sqrt, pow над двома векторами або над вектором чисел з плаваючою комою (у випадку функцій), які задані у двох бінарних файлах. Результат обчислення виводиться у текстовий файл.
15.	Гра “хрестики-нулики”	Розробити мовою C++ гру “хрестики-нулики” із графічним інтерфейсом користувача.
16.	Розбір результату API-запиту	Розробити мовою C++ консольну програму, яка надасть основну інформацію з результату запиту за URL: https://restcountries.com/v3.1/capital/Kyiv
17.	Анімація за періодичною функцією	Розробити мовою C++ програму з GUI, яка виводить у головне вікно графік періодичної функції \sin та запускає поверх графіку геометричний об’єкт із траєкторією руху за цією функцією. В програмі передбачити керування швидкістю переміщення об’єкту по

		синусоїді.
18.	Гра “морський бій”	Розробити мовою С++ гру “морський бій” із графічним інтерфейсом користувача.
19.	Телефонний довідник	Розробити мовою С++ консольну програму, що створює та дозволяє працювати із базою даних телефонів та адрес звичайних людей (П.І.Б. людини, адреса, телефон, додаткова інформація), а також підтримує інформацію по відділам організації (назва відділу, розміщення відділу в будівлі, телефон, контактна особа). За бажанням виконавця програма може мати GUI.
20.	Книжковий магазин	Розробити мовою С++ консольну програму, що створює та дозволяє працювати із базою даних книжок (П.І.Б. автора, назва книги, ключові слова, видання, рік видання, посилання на сторінку сайту, додаткова інформація). За бажанням виконавця програма може мати GUI.

Таблиця 4.2

Рекомендації до виконання завдань курсової роботи

<i>№ з/п</i>	<i>Рекомендації</i>
1.	Розробити структури даних, що містять інформацію про автора (одна структура) та інформацію про певну книжку автора (друга структура). Розробити клас для управління даними, що містяться у вказаних структурах, розробити методи читанням/запису даних з цих структур. Для спрощення можна організувати БД у бінарних файлах власного формату. Організувати невеличку консольну програму, яка опрацьовує роботу з БД на основі описаних структур даних та класу (класів). Залучити динамічні структури даних.
2.	Розробити структуру даних, що містять інформацію про запчастини автомобіля. Розробити клас для управління даними, що містяться у вказаній структурі. Для спрощення можна організувати БД каталогу запчастин у бінарних файлах власного формату. Організувати невеличку консольну програму, яка опрацьовує роботу з БД на основі описаної структури даних та класу (класів). Залучити динамічні структури даних.

3.	<p>Розробити структури даних, що містять інформацію про власника (одна структура) та інформацію про автомобілі: марка, рік випуску, реєстраційний номер (друга структура). Розробити клас для управління даними, що містяться у вказаних структурах, розробити методи читанням/запису даних з цих структур. Для спрощення можна організувати БД у бінарних файлах власного формату. Організувати невеличку консольну програму, яка опрацьовує роботу з БД на основі описаних структур даних та класу (класів). Залучити динамічні структури даних.</p>
4.	<p>Розробити консольну програму, яка буде парсити теги <A> у файлі bookmarks.html, який можна створити через інтерфейс браузера Mozilla FireFox (вікно керування закладками – пункт експортування в html-файл) та перезберігає у вказаний форматований текстовий файл назви закладок та їх URL. Для спрощення процесу парсингу можна використовувати можливості сторонніх бібліотек. Наприклад, libxml2, libxml++ та ін.</p>
5.	<p>Розробити структуру даних, що містить інформацію про товар (назва товару, код товару, виробник, опис товару та т.ін.). Розробити клас, який спростить роботу з подібною структурою та буде надавати можливість створення, читання, запису та пошуку інформації у таблиці БД SQLite. Для спрощення роботи з БД SQLite рекомендовано використовувати SQLite Database Browser (https://sqlitebrowser.org/). Для опрацювання з БД можна використовувати, наприклад, бібліотеку: https://github.com/SRombauts/SQLiteCpp</p> <p>В якості локального Web-серверу можна використати сервер проекту Apache. Для програмування CGI можна скористатися порадами з ресурсу: https://www.tutorialspoint.com/cplusplus/cpp_web_programming.htm</p> <p>Програма опрацьовує CGI-запити та звертається до БД для формування динамічної сторінки відповіді користувачу.</p>
6.	<p>Побудувати ієрархію класів для малювання геометричних примітивів та складних геометричних фігур (точка, лінія, полілінія, прямокутник, квадрат, полігон, трикутник, коло, еліпс). Малювання повинно проходити на HTML-сторінці.</p> <p>Про малювання на Web-сторінках можна прочитати на ресурсі: https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial</p> <p>В якості локального Web-серверу можна використати сервер проекту Apache. Для програмування CGI можна скористатися порадами з ресурсу: https://www.tutorialspoint.com/cplusplus/cpp_web_programming.htm</p> <p>Програма опрацьовує CGI-запити користувача та звертається до об'єктів класів малювання, які формують динамічної сторінки відповіді.</p>

7.	<p>Для побудови програми в середовищі MS Windows рекомендовано використання, наприклад, базових можливостей бібліотеки MFC. Швидко про головні засади відображення графіки на прикладі можна подивитися, наприклад, на ресурсах:</p> <p>https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture https://learn.microsoft.com/en-us/cpp/mfc/using-views https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view</p>
8.	<p>В ході дешифрування BMP-файлу використовувати довідникову інформацію про формат.</p> <p>Наприклад: https://en.wikipedia.org/wiki/BMP_file_format</p> <p>При розробці використовувати можливості стандартної бібліотеки C++. Для виведення інформації використовувати консоль або перенаправляти інформацію у текстовий файл.</p>
9.	<p>При розробці програми використовувати можливості Windows API або сторонніх бібліотек (наприклад: PortAudio, SFML, SDL). Можливо використовувати ресурси:</p> <p>https://learn.microsoft.com/en-us/windows/win32/api/utilapiset/nf-utilapiset-beep https://www.portaudio.com/ та https://github.com/notfed/beep ,</p> <p>Додаткові ресурси: https://www.sfml-dev.org/ , https://www.libsdl.org/</p>
10.	<p>Розробити окремо клас, який надає можливість управління інформацією по рецептам та їх збереженням, а також окремо консольну програму з простим інтерфейсом, який надасть доступ до рецептів. При розробці використовувати можливості стандартної бібліотеки C++.</p>
11.	<p>Розробити структуру даних, що містять інформацію про запчастини автомобіля. Данні можна зберігати в БД SQLite. Для спрощення роботи з БД SQLite рекомендовано використовувати SQLite Database Browser (https://sqlitebrowser.org/). Для опрацювання з SQLite БД можна використовувати, наприклад, бібліотеку https://github.com/SRombauts/SQLiteCpp</p> <p>В якості локального Web-серверу можна використати сервер проекту Apache. Для програмування CGI можна скористатися порадами з ресурсу: https://www.tutorialspoint.com/cplusplus/cpp_web_programming.htm</p> <p>Програма опрацьовує CGI-запити та звертається до БД для формування динамічної сторінки відповіді користувачу.</p>
12.	<p>Розробити структури даних, що містять інформацію про автора (одна структура) та інформацію про певну книжку автора (друга структура). Данні повинні зберігатися в БД SQLite. Для спрощення роботи з БД</p>

	<p>SQLite рекомендовано використовувати SQLite Database Browser (https://sqlitebrowser.org/). Для опрацювання з SQLite БД можна використовувати, наприклад, бібліотеку https://github.com/SRombauts/SQLiteCpp</p> <p>В якості локального Web-серверу можна використати сервер проекту Apache. Для програмування CGI можна скористатися порадами з ресурсу: https://www.tutorialspoint.com/cplusplus/cpp_web_programming.htm</p> <p>Програма опрацьовує CGI-запити та звертається до БД для формування динамічної сторінки відповіді користувачу.</p>
13.	При розробці використовувати можливості стандартної бібліотеки C++.
14.	При розробці використовувати можливості стандартної бібліотеки C++.
15.	<p>При розробці гри в MS Windows використати можливості бібліотеки MFC. Швидко про головні засади відображення графіки на прикладі можна подивитися, наприклад, на ресурсах: https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture</p> <p>https://learn.microsoft.com/en-us/cpp/mfc/using-views</p> <p>https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view</p>
16.	<p>В програмі запропонувати клас C++, який використовує основні засади виконання Get-запиту HTTP до Web-серверу на основі бібліотеки libcurl. При виконанні роботи є доречним використання можливостей стандартної бібліотеки C++, а також інформації з ресурсів: https://medium.com/@GeneHFang/json-in-c-9190fbe2d07e</p> <p>https://everything.curl.dev/examples/get.html</p> <p>https://everything.curl.dev/transfers/callbacks/index.html</p> <p>https://github.com/curl/curl/blob/master/docs/examples/href_extractor.c</p> <p>https://curl.se/libcurl/</p> <p>https://everything.curl.dev/</p> <p>https://github.com/open-source-parsers/jsoncpp</p> <p>https://open-source-parsers.github.io/jsoncpp-docs/doxygen/index.html</p> <p>Додаткові ресурси:</p> <p>https://github.com/DaveGamble/cJSON</p> <p>https://github.com/vincenthz/libjson</p> <p>https://github.com/json-c/json-c</p> <p>https://nlohmann.github.io/json/</p>
17.	<p>Для об'єкта переміщення створити окремий клас, що описує такий об'єкт й містить відповідні методи по його керуванню та малюванню. При розробці програми в MS Windows використати можливості бібліотеки MFC або Win32 API.</p> <p>Швидко про головні засади відображення графіки можна подивитися на ресурсах:</p>

	https://learn.microsoft.com/en-us/windows/win32/gdiplus/-gdiplus-drawing-a-line-use https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture https://learn.microsoft.com/en-us/cpp/mfc/using-views https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view
18.	<p>Створити окремо клас, який буде відповідати за стратегію гри. Створити окремо класи, які описують певні графічні об'єкти (моделі різних суден) – їх відображення (звичайний стан або об'єкт зруйновано). Створити окремо клас, який буде відповідати за малювання об'єктів гри. Можлива й інша структуризація.</p> <p>При розробці гри в MS Windows використати можливості бібліотеки MFC або Win32 API. Швидко про головні засади відображення графіки можна подивитися на ресурсах:</p> <p>https://learn.microsoft.com/en-us/windows/win32/gdiplus/-gdiplus-drawing-a-line-use https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture https://learn.microsoft.com/en-us/cpp/mfc/using-views https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view</p>
19.	<p>Розробити клас, що містить та управляє інформацією про людину та клас, який містить та управляє інформацією про відділ підрозділу. Розробити клас для управління динамічним списком об'єктів даних, що є об'єктами вказаних класів, розробити методи створення, читання, запису, пошуку, оновлення даних. Для спрощення можна організувати БД у бінарних файлах власного формату. У разі побудови програми із GUI, використовувати певну бібліотеку для побудови GUI-інтерфейсів користувача. Наприклад, MFC або FLTK.</p>
20.	<p>Розробити клас, що містить та управляє інформацією про книжки магазину. Розробити клас для управління динамічним списком об'єктів даних, що є об'єктами створеного класу про книжку, розробити методи створення, читання, запису, пошуку, оновлення даних. Для спрощення можна організувати БД у бінарних файлах власного формату. У разі побудови програми із GUI, використовувати певну бібліотеку для побудови GUI-інтерфейсів користувача. Наприклад, MFC або FLTK.</p>

5. КРИТЕРІЇ ОЦІНЮВАННЯ РЕЗУЛЬТАТІВ

Оцінювання результатів виконання курсової роботи здобувачів здійснюється за допомогою прозорих процедур, що ґрунтуються на об'єктивних критеріях відповідно до Положення університету «Про оцінювання результатів навчання здобувачів вищої освіти».

Досягнутий рівень компетентностей відносно очікуваних, що ідентифікований під час контрольних заходів, відображає реальний результат навчання здобувача.

Оцінювання навчальних досягнень здобувачів НТУ «Дніпровська політехніка» здійснюється за рейтинговою (100-бальною) та інституційною шкалами (табл. 5.1). Остання необхідна для конвертації (переведення) оцінок мобільних здобувачів.

Таблиця 5.1

Шкали оцінювання навчальних досягнень здобувачів
НТУ «Дніпровська політехніка»

<i>Рейтингова</i>	<i>Інституційна</i>
90...100	відмінно / Excellent
74...89	добре / Good
60...73	задовільно / Satisfactory
0...59	незадовільно / Fail

Курсова робота зараховується, якщо здобувач отримав підсумкову оцінку не менше 60-ти балів. Нижча оцінка вважається академічною заборгованістю, що підлягає ліквідації відповідно до Положення про організацію освітнього процесу НТУ «Дніпровська політехніка».

Здобувач на контрольних заходах має надати звіт про виконання курсової роботи та продемонструвати працездатність розробленої програми та її вірні результати відповідно до завдання обраної теми.

Основні критерії оцінювання результату захисту звіту курсової роботи представлені в таблиці 5.2.

Таблиця 5.2

Критерії оцінювання результатів виконання курсової роботи

№ з/п	Результат виконання	Рейтингова кількість балів	Інституційна оцінка
1.	- звіт містить текст з детальним описом розробки в розділах 1 та 2, а також висновки; - програма виконує всі операції за завданням;	90...100	відмінно / Excellent
2.	- звіт містить незначні відхилення від правил оформлення; - здобувач при захисті дав невпевнені відповіді на запитання викладача або помилявся; - програма виконує всі операції за завданням.	74...89	добре / Good
3.	- звіт містить незначні відхилення від правил оформлення; - здобувач при захисті погано володіє описом створеної розробки; - програма не виконує всі операції за завданням, але працює; - звіт поданий до захисту несвоєчасно.	60...73	задовільно / Satisfactory
4.	- звіт про виконання створений не за вимогами оформлення чи не за вимогами Національного стандарту України; - програма не працює; - програмний код або якась з його частин відсутні у звіті (тобто звіт містить не весь опис роботи або створеної програми); - звіт про виконання відсутній; - здобувач не з'явився на залік.	0...59	незадовільно / Fail

Кінцева кількість балів із визначених в таблиці 5.2 діапазонів встановлюється з врахуванням якості захисту виконаного звіту та вірних відповідей на поставлені викладачем запитання, відповідності сформованим компетентностям та результатам навчання.

ПЕРЕЛІК РЕКОМЕНДОВАНИХ ДЖЕРЕЛ

1. Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями. Затв. Наказом Міністерства соціальної політики України 14.02.2018. Офіційний вісник України від 18.05.2018 – 2018 р., No 38, стор. 121, стаття 1352, код акта 90123/2018. URL: <https://zakon.rada.gov.ua/laws/show/z0508-18> (дата звернення: 25.02.2025).

2. ДСТУ 3008:2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання. Київ : ДП «УкрНДНЦ», 2016. 32 с.

3. ДСТУ ISO 5807:2016 (ISO 5807:1985, IDT). Обробляння інформації. Символи та угоди щодо документації стосовно даних, програм та системних блок-схем, схем мережевих програм та схем системних ресурсів. Київ : ДП «УкрНДНЦ», 2016. 30 с.

4. ДСТУ 8302:2015. Інформація та документація. Бібліографічне посилання. Загальні положення та правила складання. Київ : ДП «УкрНДНЦ», 2016. 21 с.

5. Положення про організацію освітнього процесу Національного технічного університету «Дніпровська політехніка» (зі змінами та доповненнями від 28.05.2020, 07.03.2023, 27.06.2024 та 12.12.2024, затвердженими Вченою радою університету) / М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». Дніпро : НТУ «ДП», 2024. 46 с.

6. Положення про систему запобігання та виявлення плагіату Національного технічного університету «Дніпровська політехніка» (із змінами та доповненнями, затвердженими Вченою радою університету від 26.03.2019) / М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». Дніпро: НТУ «ДП», 2019. 12 с.

7. Положення про оцінювання результатів навчання здобувачів вищої освіти Національного технічного університету «Дніпровська політехніка» (зі змінами та доповненнями від 18.09.2018, 11.12.2018, 08.12.2021, 27.06.2024 та 12.12.2024, затвердженими Вченою радою університету) / М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». Дніпро : НТУ «ДП», 2024. 33 с.

8. Bjarne Stroustrup. Programming: Principles and Practice Using C++ (C++ In-depth), 3rd Edition. Addison-Wesley Professional, 2024. 656 p.

9. C++ reference. URL: <https://en.cppreference.com/w/cpp> (дата звернення: 25.02.2025).
10. Torsten T. Will. C++: The Comprehensive Guide to Mastering Modern C++ from Basics to Advanced Concepts with Hands-on Examples, and Best Practices for Writing Efficient, Secure, and Scalable Code. Rheinwerk Computing, 2024. 1120 p.
11. Gareth Morgan Thomas. The Complete C++ Programming Guide: Master C++ Syntax, Data Structures, OOP, Templates, STL, Multithreading, and Design Patterns. Independently published, 2024. 454 p.
12. Jacob Ricco. Object-Oriented Programming with C++: A Fundamental Guide to Building Robust, Scalable Applications with Real-World Examples. Independently published, 2024. 209 p.
13. Bill Weinman. C++20 STL Cookbook: Leverage the latest features of the STL to solve real-world problems. Packt Publishing, 2022. 450 p.
14. Jacek Galowicz. C++17 STL Cookbook: Discover the latest enhancements to functional programming and lambda expressions. Packt Publishing, 2017. 532 p.
15. MFC Desktop Applications. Article 08/03/2021. URL: <https://learn.microsoft.com/en-us/cpp/mfc/mfc-desktop-applications> (дата звернення: 25.02.2025).
16. Pekka K. MFC C++ How-To : Creating a SDI based Project Video 3 | MFC Basics. URL: https://www.youtube.com/watch?v=yZwsCYr__wo (дата звернення: 25.02.2025).
17. Pekka K. MFC C++ How-To : Graphics GDI / GDI+ Drawing Video 16 | MFC Basics. URL: https://www.youtube.com/watch?v=_Xr-TyJsbHU (дата звернення: 25.02.2025).
18. Nicholas L. Pappas Ph.D. Programming with MFC & Visual C++ (Computer Science Design Series). CreateSpace Independent Publishing Platform, 2017. 149 p.
19. Tutorials for SFML 3.0. URL: <https://www.sfml-dev.org/tutorials/3.0/> (дата звернення: 25.02.2025).
20. Djoni Darmawikarta. Learning SQL in SQLite for Beginners. Independently published, 2022. 102 p.

Навчальне видання

Гаркуша Ігор Миколайович

КУРСОВА РОБОТА З ПРОГРАМУВАННЯ

Методичні рекомендації

для здобувачів ступеня бакалавра
освітньо-професійної програми «Комп'ютерна інженерія»
спеціальності 123 Комп'ютерна інженерія

Видано в авторській редакції.

Електронний ресурс
Підписано до видання 28.04.2025. Авт. арк. 1,7.

Національний технічний університет «Дніпровська політехніка».
49005, м. Дніпро, просп. Дмитра Яворницького, 19.