

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



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

## **НАВЧАЛЬНА ПРАКТИКА**

### **Методичні рекомендації**

для здобувачів ступеня бакалавра освітньо-професійної програми  
«Інформаційні системи та технології»  
спеціальності 126 Інформаційні системи та технології

Дніпро  
НТУ «ДПУ»  
2024

**Навчальна практика [Електронний ресурс] : методичні рекомендації для здобувачів ступеня бакалавра освітньо-професійної програми «Інформаційні системи та технології» спеціальності 126 Інформаційні системи та технології / уклад.: І.М. Гаркуша, В.Ю. Каштан, Д.В. Іванов, К.Л. Сергєєва ; М-во освіти і науки України, НТУ «Дніпровська політехніка». – Дніпро : НТУ «ДП», 2024. – 22 с.**

Укладачі:

І.М. Гаркуша – канд. техн. наук, доц.;

В.Ю. Каштан – канд. техн. наук, доц.;

Д.В. Іванов – канд. техн. наук, доц.;

К.Л. Сергєєва – канд. техн. наук, доц.

Затверджено науково-методичною комісією спеціальності 126 Інформаційні системи та технології (протокол №8 від 29.08.2024) за поданням кафедри інформаційних технологій та комп'ютерної інженерії (протокол №1 від 27.08.2024).

Методичні рекомендації призначені для надання допомоги здобувачам першого (бакалаврського) рівня вищої освіти спеціальності 126 Інформаційні системи та технології під час проходження навчальної практики та підготовки до захисту звіту.

Подано основні вимоги щодо організації та проведення навчальної практики, змісту і структури звіту, його оформлення й захисту, критерії та порядок оцінювання.

Методичні рекомендації орієнтовано на активізацію виконавчого етапу навчальної діяльності здобувачів при проходженні навчальної практики.

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

## ЗМІСТ

ВСТУП.....	4
1. Мета та завдання навчальної практики.....	5
2. Організація навчальної практики.....	8
3. Варіанти індивідуальних завдань навчальної практики.....	10
4. Вимоги до оформлення звіту з навчальної практики.....	15
5. Критерії оцінювання.....	18
Перелік рекомендованих джерел.....	20

## ВСТУП

Практика здобувачів вищої освіти є невід'ємною складовою процесу підготовки фахівців різних освітніх рівнів у Національному технічному університеті «Дніпровська політехніка». Вона є важливою та обов'язковою ланкою освітнього процесу і дає змогу забезпечити набуття фахових компетентностей здобувачами вищої освіти.

Навчальним планом підготовки здобувачів ступеня бакалавра освітньо-професійної програми «Інформаційні системи та технології» спеціальності 126 Інформаційні системи та технології передбачено проходження навчальної практики на першому курсі в обсязі 6 кредитів ЄКТС (Європейська кредитна трансферно-накопичувальна система) – 180 годин. Ця практика є одним з важливих компонентів в структурі навчальних заходів, що спрямовані, включно з іншими різновидами практик, на формування всебічно підготовленого висококваліфікованого фахівця.

Під час навчальної практики здобувачі закріплюють набуті протягом навчання на першому курсі знання та практичні навички з таких обов'язкових освітніх компонент, як “Програмування”, “Алгоритми та структури даних” і “Об'єктно-орієнтоване програмування”.

Під час виконання завдання практики здобувач використовує об'єктно-орієнтовану мову програмування C++.

## 1. Мета та завдання навчальної практики

У процесі проходження навчальної практики, здобувачі набувають компетентностей, що являють собою спеціальну структуровану сукупність знань, умінь, навичок і ставлень, які визначені освітньо-професійною програмою «Інформаційні системи та технології» для першого (бакалаврського) рівня вищої освіти спеціальності 126 Інформаційні системи та технології.

Ця освітньо-професійна програма закріплює за навчальною практикою такі результати навчання:

– ПР 2 – застосовувати знання фундаментальних і природничих наук, системного аналізу та технологій моделювання, стандартних алгоритмів та дискретного аналізу при розв’язанні задач проектування і використання інформаційних систем та технологій;

– ПР 6 – демонструвати знання сучасного рівня технологій інформаційних систем, практичні навички програмування та використання прикладних і спеціалізованих комп’ютерних систем та середовищ з метою їх запровадження у професійній діяльності.

Основними базовими обов’язковими освітніми компонентами для навчальної практики за освітньо-професійною програмою «Інформаційні системи та технології» є:

- програмування;
- алгоритми та структури даних;
- об’єктно-орієнтоване програмування.

**Метою навчальної практики** є формування компетентностей щодо:

- закріплення опису алгоритмів за допомогою схем алгоритмів;
- використання діаграм для подання складу модулів програм, що розробляються;

– використання об'єктно-орієнтованого програмування мовою C++ для вирішення поставлених прикладних задач;

– розробки програм під операційну систему MS Windows в професійних середовищах розробки із залученням різних бібліотек та фреймворків.

Основою для виконання завдань навчальної практики є досягнуті дисциплінарні результати навчання за підсумками вивчення дисциплін першого курсу.

#### 1. Програмування:

– використовуючи отримані знання щодо побудови алгоритмів, представляти їх словесними, формульно-словесними способами та за допомогою схем алгоритмів при розв'язанні певних обчислювальних задач;

– мати базові знання щодо архітектури обчислювальної техніки, історії її розвитку, систем числення, одиниць вимірювання та представлення даних в пам'яті комп'ютера;

– розробляти простіші консольні програми на базі отриманих знань щодо будування алгоритмів та навичок програмування на мові C;

– реалізовувати обчислення при розробці консольних програм в операційному середовищі MS Windows, роблячи обґрунтований вибір певних структур даних та алгоритмів обробки;

– програмувати алгоритми, що реалізовані через консольні програми в операційному середовищі MS Windows, використовуючи мову програмування C;

– програмувати динамічні структури даних, користувацькі типи, а також вміти програмувати операції файлового введення/виведення.

#### 2. Алгоритми та структури даних:

– вміти проектувати, будувати та реалізовувати базові алгоритми обробки різноманітних цифрових даних при створенні новітніх інформаційних технологій в структурі інформаційних систем;

– вміти конструювати та реалізовувати відповідні алгоритми обробки цифрових даних засобами сучасних мов програмування;

– вміти формалізувати алгоритмічні рішення на всіх етапах збору, передачі та обробки цифрових даних в структурі інформаційних систем та відповідних технологій;

– ефективно проводити адміністрування, автоматизувати процеси створення програмних продуктів.

### 3. Об'єктно-орієнтоване програмування:

– мати базові знання щодо основних положень об'єктно-орієнтованого програмування, визначення та вміст базових понять "інкапсуляція", "успадкування", "поліморфізм", а також принципи організації взаємодії об'єктів

– знати засоби подання даних простими (базовими) та складними (похідними) типами, методики об'єктно-орієнтованого програмування в сучасних інтегрованих середовищах розробки програм;

– вміти реалізовувати прототипи архітектури програмного забезпечення, ефективно проводити адміністрування, автоматизувати процеси створення програмних продуктів;

– вміти розробляти ієрархію класів та об'єктів, створювати шаблонні класи.

Результати проходження навчальної практики повинні показати ступінь підготовленості здобувача на даному етапі навчання до самостійної діяльності при використанні інструментів опису схем алгоритмів та розробки комп'ютерних програм.

## 2. Організація навчальної практики

Навчальна практика у здобувачів 1-го курсу освітньо-професійної програми «Інформаційні системи та технології» проводиться на базі комп'ютерних класів випускової кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка» згідно затвердженого графіка навчального процесу. Форма контролю результатів практики – диференційований залік.

Для виконання поставлених задач навчальної практики здобувач використовує програмне забезпечення (ПЗ):

- операційна система (ОС) MS Windows 10/11 (або новіша);
- середовище розробки MS Visual Studio Community edition 2022 (або новіше) або Code::Blocks 20.03 (або новіше) з компіляторами GNU GCC/G++, MS Visual C++ Compiler, або 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 навчальної практики у віддаленому режимі за місцем перебування. При цьому, керівник практики спілкується із здобувачами за допомогою різноманітних програм, наприклад таких, як Microsoft Teams або через електронну пошту.

До кінця терміну перебування на практиці здобувач повинен завершити складання звіту. Звіт про практику студент здає на перевірку керівнику практики у встановлені терміни.

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.

Після закінчення терміну навчальної практики, здобувачі вищої освіти звітують про виконання завдання практики. Форма звітності здобувача вищої освіти про проходження практики – письмовий звіт державною мовою.

Керівник практики приймає залік у здобувачів протягом встановленого терміну.

### 3. Варіанти індивідуальних завдань навчальної практики

В якості індивідуального завдання навчальної практики, керівник видає здобувачу один з варіантів тем. Приклади тем представлені в таблиці 3.1. В таблиці 3.2 наводяться основні рекомендації щодо виконання кожного з варіантів.

Таблиця 3.1

#### Приклади тем індивідуальних завдань навчальної практики

<i>№ з/п</i>	<i>Назва теми</i>	<i>Завдання</i>
1.	Бібліотечний довідник	Розробити мовою С++ консольну програму, що створює та обробляє базу даних книжок бібліотеки.
2.	Каталог автозапчастин	Розробити мовою С++ консольну програму, яка дозволить звертатися до каталогу певних запчастин та повертати інформацію про їх характеристики.
3.	База даних автомобілістів	Розробити мовою С++ консольну програму, яка дозволить працювати з базою даних автомобілів та їх власників.
4.	Обробка файлу закладок браузера	Розробити мовою С++ консольну програму, що дозволяє проводити опрацювання історії закладок браузера Mozilla Firefox, що розміщені у файлі bookmarks.html.
5.	Програма малювання геометричних фігур	Розробити мовою С++ програму з GUI, що виконує малювання різноманітних геометричних фігур у головному вікні.
6.	Інформація про BMP-файл	Розробити мовою С++ консольну програму дешифрування бінарних файлів формату Windows Bitmap.
7.	Програма моделювання звукової азбуки Морзе	Розробити мовою С++ консольну програму навчання звуковій азбуці Морзе (символи української абетки).

8.	Кулінарна книга	Розробити мовою С++ консольну програму надання кулінарних рецептів.
9.	Калькулятор	Розробити мовою С++ 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.
10.	Файловий калькулятор	Розробити мовою С++ 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.
11.	Гра “хрестики-нулики”	Розробити мовою С++ гру “хрестики-нулики” із графічним інтерфейсом користувача.
12.	Анімація за періодичною функцією	Розробити мовою С++ програму з GUI, яка виводить у головне вікно графік періодичної функції <i>sin</i> та запускає поверх графіку геометричний об’єкт із траєкторією руху за цією функцією. В програмі передбачити керування швидкістю переміщення об’єкту по синусоїді.
13.	Гра “морський бій”	Розробити мовою С++ гру “морський бій” із графічним інтерфейсом користувача.
14.	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.
15.	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 3.2

Рекомендації до виконання індивідуальних завдань навчальної практики

<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.	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.
4.	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 перезберігас у вказаний форматований текстовий файл назви закладок та їх URL. Для спрощення процесу парсингу можна використовувати можливості сторонніх бібліотек. Наприклад, libxml2, libxml++ та ін.

5.	<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><a href="https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture">https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture</a>  <a href="https://learn.microsoft.com/en-us/cpp/mfc/using-views">https://learn.microsoft.com/en-us/cpp/mfc/using-views</a>  <a href="https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view">https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view</a></p>
6.	<p>В ході дешифрування BMP-файлу використовувати довідникову інформацію про формат.</p> <p>Наприклад: <a href="https://en.wikipedia.org/wiki/BMP_file_format">https://en.wikipedia.org/wiki/BMP_file_format</a></p> <p>При розробці використовувати можливості стандартної бібліотеки C++. Для виведення інформації використовувати консоль або перенаправляти інформацію у текстовий файл.</p>
7.	<p>При розробці програми використовувати можливості Windows API або сторонніх бібліотек (наприклад: PortAudio, SFML, SDL). Можливо використовувати ресурси:</p> <p><a href="https://learn.microsoft.com/en-us/windows/win32/api/utilapiset/nf-utilapiset-beep">https://learn.microsoft.com/en-us/windows/win32/api/utilapiset/nf-utilapiset-beep</a>  <a href="https://www.portaudio.com/">https://www.portaudio.com/</a> та <a href="https://github.com/notfed/beep">https://github.com/notfed/beep</a> ,</p> <p>Додаткові ресурси:  <a href="https://www.sfml-dev.org/">https://www.sfml-dev.org/</a> ,  <a href="https://www.libsdl.org/">https://www.libsdl.org/</a></p>
8.	<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>
9.	<p>При розробці використовувати можливості стандартної бібліотеки C++.</p>
10.	<p>При розробці використовувати можливості стандартної бібліотеки C++.</p>
11.	<p>При розробці гри в MS Windows використати можливості бібліотеки MFC. Швидко про головні засади відображення графіки на прикладі можна подивитися, наприклад, на ресурсах:</p> <p><a href="https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture">https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture</a>  <a href="https://learn.microsoft.com/en-us/cpp/mfc/using-views">https://learn.microsoft.com/en-us/cpp/mfc/using-views</a>  <a href="https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view">https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view</a></p>
12.	<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>

	<a href="https://learn.microsoft.com/en-us/windows/win32/gdiplus/-gdiplus-drawing-a-line-use">https://learn.microsoft.com/en-us/windows/win32/gdiplus/-gdiplus-drawing-a-line-use</a> <a href="https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture">https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture</a> <a href="https://learn.microsoft.com/en-us/cpp/mfc/using-views">https://learn.microsoft.com/en-us/cpp/mfc/using-views</a> <a href="https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view">https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view</a>
13.	<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><a href="https://learn.microsoft.com/en-us/windows/win32/gdiplus/-gdiplus-drawing-a-line-use">https://learn.microsoft.com/en-us/windows/win32/gdiplus/-gdiplus-drawing-a-line-use</a>  <a href="https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture">https://learn.microsoft.com/en-us/cpp/mfc/document-view-architecture</a>  <a href="https://learn.microsoft.com/en-us/cpp/mfc/using-views">https://learn.microsoft.com/en-us/cpp/mfc/using-views</a>  <a href="https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view">https://learn.microsoft.com/en-us/cpp/mfc/drawing-in-a-view</a></p>
14.	<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>
15.	<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>

#### 4. Вимоги до оформлення звіту з навчальної практики

Звіт оформлюється на аркушах формату А4 та включає титульний аркуш (зразок оформлення представлений на рис. 4.1) та опис роботи за представленим нижче зразком.

<p><b>Міністерство освіти і науки України</b> <b>Національний технічний університет</b> <b>«Дніпровська політехніка»</b> <b>Навчально-науковий інститут електроенергетики</b> <b>Факультет інформаційних технологій</b> <b>Кафедра інформаційних технологій та комп'ютерної інженерії</b></p>
<p><b>Звіт</b> <b>з навчальної практики</b></p>
<p>Тема: « _____ »</p>
<p>Виконав: здобувач гр. шифр групи <i>Ініціали та прізвище</i> здобувача</p>
<p>Перевірив: посада каф. ІТКІ <i>Ініціали та прізвище</i> керівника практики</p>
<p>Дніпро <i>Поточний рік складання звіту</i></p>

Рис. 4.1. Зразок шаблону титульного аркуша звіту

Звіт має відображати результати виконаного в процесі практики індивідуального завдання та складається з титульного аркуша та текстової частини з таких розділів:

Зміст

1. Лист завдання.
2. Загальний опис розробки.
3. Схеми алгоритмів роботи програми.
4. Код програми з розбиттям за модулями та коментарями.
5. Висновки.
6. Перелік використаних джерел.

Звіт перевіряє й оцінює керівник практики.

У разі захисту звіту з виконання завдання практики при аудиторній формі навчання, звіт друкується та надається викладачу на розгляд.

Для захисту виконаного завдання у дистанційній (online) формі навчання, звіт подається до захисту в електронному форматі PDF (Portable Document Format) та надсилається керівнику практики через засоби online-спілкування для розгляду.

Звіт про проходження навчальної практики має бути виконано з дотриманням вимог державних стандартів щодо оформлення текстових документів, зокрема звітів у сфері науки і техніки (ДСТУ 3008-2015. Інформація та документація. Звіти у сфері науки і техніки. Структура та правила оформлювання).

Шрифтом з розміром 10 оформлюють текст коду розробленої програми. При цьому бажано використовувати один з моноширинних шрифтів: Source Code Pro, Monospace, Consolas, DejaVu Sans Mono, Courier New, Monospaced, Menlo, SF Mono або подібні.

Вирівнювання тексту в звіті – по ширині сторінки.

Вирівнювання рисунків та підписів до них – по центру сторінки.

Вирівнювання таблиць (якщо такі будуть) – по центру сторінки.

Якщо код програми не є великим за обсягом, то його можна подати у вигляді знімку з екрану, причому фон повинен бути білого кольору, а текст коду програми повинен добре виділятися. Такий рисунок потрібно розмістити по центру сторінки звіту та надати по центру під ним підпис у формі, наприклад: Рисунок 3.1 – Схема алгоритму головної функції програми.

Якщо в тексті звіту наводиться рисунок, то на нього обов'язково повинно бути посилання з тексту та пояснення. Якщо поданий текст коду програми, то при описі вказують номер рядку, коментуючи дії, які виконані в рядку під цим номером.

При складанні тексту висновків, потрібно вказати на результати (зокрема тестування) щодо досягнення поставленого завдання практики. Вказати особливості виконання роботи та складнощі.

## 5. Критерії оцінювання

Оцінювання результатів практики здобувачів проводиться за 100-бальною шкалою з обов'язковим переведенням бальних оцінок до інституційної шкали (табл. 5.1). Оцінка за практику вноситься до залікової відомості за підписом керівника практики.

Таблиця 5.1

### Шкала оцінювання навчальних досягнень здобувачів НТУ «ДП»

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

Оцінювання результатів проходження навчальної практики здійснюється за критеріями, наведеними у табл. 5.2:

Таблиця 5.2

### Розподіл балів за критеріями оцінювання навчальної практики

№ з/п	Результат виконання	Рейтингова кількість балів	Інституційна оцінка
1.	- звіт містить текст з детальним описом коду програми, схемами алгоритмів, 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 керівника практики або помилявся;	74...89	добре / Good

	- програма виконує всі операції за завданням.		
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 виконаного завдання 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. Положення про проведення практики здобувачів вищої освіти Національного технічного університету «Дніпровська політехніка», затверджене Вченою радою 11.12.2018, протокол №15 / М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». Дніпро: НТУ «ДП», 2018. 21 с.

2. Положення про організацію освітнього процесу Національного технічного університету «Дніпровська політехніка», затверджене Вченою радою університету від 25.10.2019, протокол № 20 (зі змінами та доповненнями) / Мін-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка» – Дніпро : НТУ «ДП», 2024. – 46 с.

3. Положення про оцінювання результатів навчання здобувачів вищої освіти Національного технічного університету «Дніпровська політехніка» (зі змінами та доповненнями) / М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». Дніпро : НТУ «ДП», 2024. 33 с.

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

5. 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.

6. 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.

7. Jacob Ricco. Object-Oriented Programming with C++: A Fundamental Guide to Building Robust, Scalable Applications with Real-World Examples. Independently published, 2024. 209 p.

8. Bill Weinman. C++20 STL Cookbook: Leverage the latest features of the STL to solve real-world problems. Packt Publishing, 2022. 450 p.

9. Jacek Galowicz. C++17 STL Cookbook: Discover the latest enhancements to functional programming and lambda expressions. Packt Publishing, 2017. 532 p.

10. Nicholas L. Pappas Ph.D. Programming with MFC & Visual C++ (Computer Science Design Series). CreateSpace Independent Publishing Platform, 2017. 149 p.

11. Соколова Н.О. Методичні рекомендації до виконання курсової роботи з дисципліни «Об'єктно-орієнтоване програмування» для здобувачів ступеня бакалавра спеціальності 126 Інформаційні системи та технології / Н.О. Соколова ; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». Дніпро: НТУ «ДП», 2023. 34с.

12. Бобков, В. Б. Програмування - 2. Об'єктно-орієнтоване програмування. Навчальний посібник [Електронний ресурс] : навчальний посібник для здобувачів ступеня бакалавра за освітньою програмою «Автоматизація та комп'ютерно – інтегровані технології кібер - енергетичних систем» спеціальності 151 Автоматизація та комп'ютерно – інтегровані технології / В. Б. Бобков, Ю. Є. Грудзинський, К. В. Крилов ; КПІ ім. Ігоря Сікорського. - Електронні текстові дані (1 файл: 653.01 Кбайт). - Київ : КПІ ім. Ігоря Сікорського, 2023. - 77 с. - Назва з екрана.

Навчальне видання

**Гаркуша Ігор Миколайович**  
**Каштан Віта Юріївна**  
**Іванов Денис Валерійович**  
**Сергєєва Катерина Леонідівна**

## **НАВЧАЛЬНА ПРАКТИКА**

### **Методичні рекомендації**

для здобувачів ступеня бакалавра освітньо-професійної програми  
«Інформаційні системи та технології»  
спеціальності 126 Інформаційні системи та технології

Видано в редакції авторів

Електронний ресурс.  
Підписано до видання 02.09.2025. Авт. арк. 0,67.

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