

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЗАХІДНОУКРАЇНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

ЗАТВЕРДЖУЮ
Декан факультету комп'ютерних
інформаційних технологій
_____ Ігор ЯКИМЕНКО
" ____ " _____ 2025 р.

ЗАТВЕРДЖУЮ
Проректор
з науково-педагогічної роботи
_____ Віктор ОСТРОВЕРХОВ
" ____ " _____ 2025 р.

ЗАТВЕРДЖУЮ
Директор навчально наукового
інституту новітніх освітніх
технологій
_____ Святослав ПИТЕЛЬ
" ____ " _____ 2025 р.

РОБОЧА ПРОГРАМА
з дисципліни «Об'єктно-орієнтоване програмування»

Ступінь вищої освіти: бакалавр
Галузь знань – F Інформаційні технології
Спеціальність – F4 Системний аналіз
Освітньо-професійна програма – «Системний аналіз»

Кафедра комп'ютерних наук

Форма навчання	Курс	Семест р	Лекції (год.)	Лаб. (год.)	ІРС (год.)	Тренінг (год.)	СРС (год.)	Разом (год.)	Залік (сем.)
Денна	1	2	30	14	2	6	38	90	2
Заочна	1	2	8	4	-	-	78	90	2

Тернопіль – ЗУНУ
2025

Робоча програма складена на основі освітньо-професійної програми підготовки бакалаврів галузі знань F Інформаційні технології спеціальності F4 Системний аналіз, затвердженої Вченою радою ЗУНУ №8 від 26 червня 2025 року.

Робоча програма розроблена професором кафедри комп'ютерних наук, д.т.н. Манжулою Володимиром Івановичем

Робоча програма затверджена на засіданні кафедри економічної кібернетики та інформатики, протокол №1 від 29 серпня 2025 р.

Завідувач кафедри
д.е.н, професор

Леся БУЯК

Гарант ОП
д.т.н, професор

Роман ПАСІЧНИК

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ “Об’єктно-орієнтоване програмування”

1. Опис дисципліни “Об’єктно-орієнтоване програмування”

Дисципліна – «Об’єктно-орієнтоване програмування»	Галузь знань, спеціальність, ступінь вищої освіти	Характеристика навчальної дисципліни
Кількість кредитів – 3	Галузь знань: F Інформаційні технології	Статус дисципліни: нормативна Мова навчання: Українська
Кількість залікових модулів: 4	Спеціальність: F4 Системний аналіз	Рік підготовки: Денна – 1 Заочна – 1. Семестр: Денна – 2. Заочна – 2.
Кількість змістових модулів – 3	Ступінь вищої освіти: бакалавр	Лекції: Денна – 30 год. Заочна – 8 год. Лабораторні заняття: Денна – 14 год. Заочна – 4 год.
Загальна кількість годин – 90		Самостійна робота: Денна – 40 год. Тренінг – 6 Заочна – 78 год. Індивідуальна робота: Денна – 2 год.
Тижневих годин: 6 год., з них аудиторних: 3		Вид підсумкового контролю: Денна – залік, 2 Заочна – залік, 2.

2. Мета й завдання вивчення дисципліни «Об’єктно-орієнтоване програмування»

2.1. Мета вивчення дисципліни

Мета викладання дисципліни “Об’єктно-орієнтоване програмування” полягає у наданні теоретичних знань та практичних навиків програмування з використанням об’єктно-орієнтованих мов програмування.

Мета проведення лекцій полягає у тому, щоб ознайомити студентів із головними методологічними та методичними питаннями об’єктно-орієнтованої парадигми програмування, та інструментарієм для створення об’єктно-орієнтованих програм від моделювання взаємозв’язків між об’єктами до створення програмного коду засобами об’єктно-орієнтованих мов програмування

Мета проведення лабораторних занять полягає у тому, щоб виробити у студентів практичні навички побудови об’єктно-орієнтованих моделей та їх програмної реалізації засобами об’єктно-орієнтованої мови програмування C++.

2.2. Завдання вивчення дисципліни

Завданням дисципліни є набуття студентами теоретичних знань та практичних навичок принципів об'єктно-орієнтованої декомпозиції складних об'єктів, розробки проектів та написання на їх основі ефективних програм з використанням всіх переваг об'єктно-орієнтованої парадигми програмування.

У результаті вивчення дисципліни студенти повинні знати:

- фундаментальні концепції ООП;
- принципи програмування та реалізації ООП.
- мову об'єктно-орієнтованого програмування C++;
- перевантаження операторів;
- наслідування;
- поліморфізм
- шаблони;
- методи обробки виняткових ситуацій;
- ієрархію класів;
- організацію стандартної бібліотеки і контейнерів;

У результаті вивчення дисципліни студенти повинні вміти:

- визначати класи та об'єкти при застосуванні об'єктно-орієнтованих технологій програмування;
- здійснювати об'єктно-орієнтовану декомпозицію предметної області в умовах програмування складних об'єктів і систем за допомогою процедур об'єктно-орієнтованого аналізу, об'єктно-орієнтованого програмування, використовуючи визначення класів, поведінки об'єктів, структури даних та їх взаємозв'язки;
- розробляти об'єктно-орієнтовану модель предметної галузі за допомогою мов об'єктно-орієнтованого моделювання в умовах програмування складних об'єктів та систем, використовуючи об'єктно-орієнтовану нотацію складних систем.

2.3. Найменування та опис компетентностей, формування котрих забезпечує вивчення дисципліни:

- Здатність до абстрактного мислення, аналізу та синтезу;
- Здатність застосовувати знання у практичних ситуаціях;
- Знання та розуміння предметної області та розуміння професійної діяльності;
- Здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, оптимізації, обробки інформації, інтелектуального аналізу даних;
- Здатність використовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей та прогнозування поведінки конкретних систем а саме: об'єктно-орієнтований підхід при проектуванні складних систем різної природи, прикладні математичні пакети, застосування баз даних і знань;

- Здатність організовувати роботу з аналізу та проектування складних систем, створення відповідних інформаційних технологій та програмного забезпечення.

2.4. Передумови для вивчення дисципліни.

Дисципліни, які повинні бути вивчені попередньо:

- «Основи програмування»;

2.5. Результати навчання

У результаті вивчення курсу «Об'єктно-орієнтоване програмування» студенти повинні:

- Володіти сучасними методами розробки програм і програмних комплексів та прийняття оптимальних рішень щодо складу програмного забезпечення, алгоритмів процедур і операцій;

- Вміти створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень;

- Знати архітектуру сучасних обчислювальних систем і комп'ютерних мереж;

- Знати і вміти застосовувати на практиці системи управління базами даних і знань та інформаційні системи;

- Проектувати, реалізовувати, тестувати, впроваджувати, супроводжувати, експлуатувати програмні засоби роботи з даними і знаннями в комп'ютерних системах і мережах.

3. Зміст дисципліни “Об'єктно-орієнтоване програмування”

Змістовий модуль 1. Основні парадигми ООП.

Тема 1. Вступ в об'єктно-орієнтоване програмування.

Історія об'єктно-орієнтованого програмування. Основні поняття. Абстракція даних. Наслідування, інкапсуляція та поліморфізм.

Література: 1,2,10.

Тема 2. Об'єкти та класи.

Поняття об'єкта в ООП. Стан та поведінка об'єкта. Життєвий цикл об'єкта. Відношення між об'єктами. Поняття класу. Типи класів. Відношення між класами. Інстанціювання.

Література: 1,2,9,10.

Тема 3. Об'єктно-орієнтований аналіз і декомпозиція складних об'єктів.

Об'єктно-орієнтований аналіз і декомпозиція складних об'єктів. Об'єктно-орієнтований синтез інформаційних моделей. Використання мови об'єктно-орієнтованого моделювання UML.

Література: 1,2,9,10.

Змістовий модуль 2. Реалізація принципів ООП мовою C++.

Тема 4. Класи та об'єкти в C++.

Класи і члени. Ініціалізація. Друзі класу. Інтерфейс та реалізація. Конструктори і деструктори. Конструювання і знищення об'єктів. Статичні члени класу. Вказівник this. Вказівники на члени класу. Статичні змінні класу. Статичні методи класу. Поняття дружності.

Література: 2,3,6,7.

Тема 5. Наслідування.

Похідні класи. Конструктори похідних класів. Виклик конструкторів похідних класів. Виклик конструкторів ієрархічних класів. Множинне наслідування. Виклик конструкторів при множинному наслідуванні. Передача параметрів конструктору базового класу.

Література: 2,3,6,7.

Тема 6. Віртуальні функції та поліформізм.

Віртуальні функції та поліформізм. Чисто віртуальні функції та абстрактні класи. Механізм пізнього скріплення.

Література: 2,3,6,7.

Тема 7. Перевантаження операторів.

Перевантаження операторів. Створення операторної функції члена. Створення префіксної і постфіксної форм операторів інкрементації і декрементації. Перевантаження скорочених операторів присвоєння. Перевантаження операторів new і delete. Перевантаження операторів «[]», «()», «->», «,».

Література: 2,3,6,7.

Тема 8. Шаблони.

Простий шаблон. Шаблони-списки. Параметризовані функції. Правила ототожнення перевантажених параметризованих функцій. Наслідування і шаблони. Узагальнені класи. Приклад використання двох узагальнених типів даних. Застосування аргументів за умовчанням в шаблонних класах. Явні спеціалізації класів.

Література: 2,3,6,7.

Тема 9. Обробка виняткових ситуацій.

Обробка помилок. Розпізнавання ситуацій. Класифікація ситуацій. Використання ресурсів. Генерація виняткових ситуацій. Повторне генерування виняткової ситуації. Функції terminate() і unexpected(). Класи exception і bad_exception. Перехоплення класів виняткових ситуацій. Застосування декількох операторів catch.

Література: 2,3,6,7.

Змістовий модуль 3. Бібліотека стандартних шаблонів STL.

Тема 10. Вступ в STL.

Склад STL. Предикати та функції-об'єкти. Алокатори. Ітератори. Поняття ітератора.

Література: 2,3,6,7.

Тема 11. Контейнерні класи.

Поняття контейнера. Типи контейнерів. Контейнер vector. Контейнер deque. Асоціативні контейнери. Контейнер Set. Контейнер Map.

Література: 2,3,6,7.

Тема 12. Алгоритми STL.

Модифікуючі та немодифікуючі операції. Алгоритми сортування. Алгоритми роботи з множинами.

Література: 2,3,6,7.

Тема 13. Оптимізація обчислень в STL.

Шаблон класу valarray. Перетин масивів.

Література: 2,3,6,7.

4. Структура залікового кредиту з дисципліни “Об’єктно-орієнтоване програмування”

(Денна форма навчання)

	Кількість годин					
	Лекції	Лабораторні заняття	Самостійна робота	ІРС	Тренінг	Контрольні заходи
Змістовий модуль 1. Основні парадигми ООП.						
Тема 1. Вступ в об’єктно–орієнтоване програмування	2		2		2	Усне опитування та тестування
Тема 2. Об’єкти та класи	2	1	2			Усне опитування та тестування
Тема 3. Об’єктно-орієнтований аналіз і декомпозиція складних об’єктів	2	1	2			Усне опитування та тестування
Змістовий модуль 2. Реалізація принципів ООП мовою C++.						
Тема 4. Класи та об’єкти в C++	2	2	2	1	2	Усне опитування та тестування
Тема 5. Наслідування	2	2	2			Усне опитування та тестування
Тема 6. Віртуальні функції та поліформізм	2	2	2			Усне опитування та тестування
Тема 7. Перевантаження операторів	2	2	2			Усне опитування та тестування
Тема 8. Шаблони	2	1	4			Усне опитування та тестування
Тема 9. Обробка виняткових ситуацій	2	1	4			Усне опитування та тестування
Змістовий модуль 3. Бібліотека стандартних шаблонів STL.						
Тема 10. Вступ в STL	2		4	1	2	Усне опитування та тестування
Тема 11. Контейнерні класи	4	1	4			Усне опитування та тестування
Тема 12. Алгоритми STL	4	1	4			Усне опитування та тестування
Тема 13. Оптимізація обчислень в STL	2		4			Усне опитування та тестування
Разом	30	14	38	2	6	

(Заочна форма навчання)

	Кількість годин		
	Лекції	Лабораторні заняття	Самостійна робота
Змістовий модуль 1. Основні парадигми ООП.			
Тема 1. Вступ в об'єктно-орієнтоване програмування	1		6
Тема 2. Об'єкти та класи			6
Тема 3. Об'єктно-орієнтований аналіз і декомпозиція складних об'єктів	1	1	6
Змістовий модуль 2. Реалізація принципів ООП мовою C++.			
Тема 4. Класи та об'єкти в C++	1	1	6
Тема 5. Наслідування	1	1	6
Тема 6. Віртуальні функції та поліморфізм	1	1	6
Тема 7. Перевантаження операторів			6
Тема 8. Шаблони	1		6
Тема 9. Обробка виняткових ситуацій			6
Змістовий модуль 3. Бібліотека стандартних шаблонів STL.			
Тема 10. Вступ в STL			6
Тема 11. Контейнерні класи	1		6
Тема 12. Алгоритми STL	1		6
Тема 13. Оптимізація обчислень в STL			6
Разом	8	4	78

5. Тематика лабораторних занять

Лабораторна робота № 1

Тема: Об'єктно-орієнтований аналіз і декомпозиція складних об'єктів.

Мета: Навчитись будувати моделі складних об'єктів з використанням мови об'єктно-орієнтованого моделювання UML.

Література: 1, 2.

Лабораторна робота № 2

Тема: Конструювання об'єктів конкретних класів.

Мета: Засвоєння методики створення конкретних класів з використанням мови програмування C++.

Література: 2, 4, 12.

Лабораторна робота № 3

Тема: Реалізація наслідування.

Мета: Засвоєння методики реалізації наслідування з використанням мови програмування C++.

Література: 2, 10, 12.

Лабораторна робота № 4

Тема: Поліморфізм.

Мета: Засвоєння методики реалізації поліморфізму з використанням мови програмування C++.

Література: 2, 10, 12.

Лабораторна робота № 5

Тема: Перевантаження операторів.

Мета: Засвоєння методики перевантаження унарних та бінарних операторів для класів користувача.

Література: 2, 6, 11.

Лабораторна робота № 6

Тема: Розробка шаблонів.

Мета: Навчитись використовувати можливості параметричного поліморфізму.

Література: 2, 10, 12.

Лабораторна робота № 7

Тема: Обробка виняткових ситуацій.

Мета: Засвоєння методики відлагодження програм та обробки програм на етапі виконання засобами об'єктно-орієнтованого програмування.

Література: 4, 10, 12.

Лабораторна робота № 8

Тема: Використання STL.

Мета: Засвоєння принципів роботи з STL.

Література: 7, 10, 12.

6. Організація та проведення тренінгу з курсу

Тема: реалізація консольного програмного додатку.

Порядок проведення:

- здійснити об'єктно-орієнтований аналіз предметної області;
- реалізувати об'єктно-орієнтовану модель на основі методів об'єктно-орієнтованого проектування;
- реалізувати додаток на мові об'єктно-орієнтованого програмування;
- провести тестування програми;
- представити результати роботи у вигляді короткого звіту: діаграми варіантів використання та класів, реалізація, результати тестування.

Література: 2, 10, 12.

Варіанти завдань для тренінгу з дисципліни “Об'єктно-орієнтоване програмування”

1. Реалізація об'єктно-орієнтованої програми для імітації роботи ліфта.
2. Реалізація об'єктно-орієнтованої програми для імітації роботи музичного центру.

3. Реалізація об'єктно-орієнтованої програми для імітації роботи холодильника.
4. Реалізація об'єктно-орієнтованої програми для імітації роботи пральної машина.
5. Реалізація об'єктно-орієнтованої програми для імітації роботи охоронної системи приміщення (в тому числі - протипожежна).
6. Реалізація об'єктно-орієнтованої програми для імітації роботи телефонного апарату.
7. Реалізація об'єктно-орієнтованої програми для імітації роботи системи забезпечення безпеки руху.
8. Реалізація об'єктно-орієнтованої програми для імітації роботи автоматичної системи контролю стану систем літака.
9. Реалізація об'єктно-орієнтованої програми для імітації роботи автоматичної системи контролю стану систем автомобіля.
10. Реалізація об'єктно-орієнтованої програми для імітації роботи банкомата.
11. Реалізація об'єктно-орієнтованої програми для імітації роботи автомата по продажу цукерок.
12. Реалізація об'єктно-орієнтованої програми для імітації роботи телевізора.
13. Реалізація об'єктно-орієнтованої програми для імітації роботи системи опрацювання текстів.
14. Реалізація об'єктно-орієнтованої програми для імітації роботи мікрохвильової печі.
15. Реалізація об'єктно-орієнтованої програми для імітації роботи турнікета метрополітену.
16. Реалізація об'єктно-орієнтованої програми для імітації процесу вирощування рослини.
17. Реалізація об'єктно-орієнтованої програми для імітації роботи системи зрошування газону.
18. Реалізація об'єктно-орієнтованої програми для імітації роботи системи кухонного комбайну.
19. Реалізація об'єктно-орієнтованої програми для імітації роботи системи багатофункціонального дистанційного пульта управління (tv, cd, vcd, sat,...).
20. Реалізація об'єктно-орієнтованої програми для імітації роботи тестера (амперметр, омметр, вольтметр).
21. Реалізація об'єктно-орієнтованої програми для імітації роботи інженерного калькулятора (dec, hex, bin, ...).
22. Реалізація об'єктно-орієнтованої програми для імітації роботи системи відеоспостереження.
23. Реалізація об'єктно-орієнтованої програми для імітації роботи системи опалення приміщень (газ. котел, насос, ...).
24. Реалізація об'єктно-орієнтованої програми для опису функціонування факультету ВНЗ.

7. Самостійна робота

Для успішного вивчення і засвоєння дисципліни «Об'єктно-орієнтоване програмування» студенти повинні володіти значним обсягом інформації, частину якої вони отримують і опрацьовують шляхом самостійної роботи. Самостійна робота полягає в опрацюванні сучасної навчальної і наукової фахової літератури.

Результати самостійної роботи студента з дисципліни «Об'єктно-орієнтоване програмування» оцінюються за 100-бальною шкалою у вигляді тестування.

- 1 Об'єктний підхід в об'єктно-орієнтованому проектуванні.
- 2 Класифікація в об'єктно-орієнтованому аналізі.
- 3 Об'єктно-орієнтоване проектування. Система позначень.
- 4 Діаграми класів.
- 5 Діаграми переходів станів.
- 6 Діаграми об'єктів.
- 7 Часові діаграми.
- 8 Модульні діаграми.
- 9 Діаграми процесів.
- 10 Ідентифікація зв'язків між класами та об'єктами.
- 11 Об'єктно-орієнтоване програмування в життєвому циклі розробки програми.
- 12 Переваги та недоліки об'єктно-орієнтованого програмування.
- 13 Відношення між об'єктами.
- 14 Відношення між класами.
- 15 Взаємозв'язки класів та об'єктів.
- 16 Перевантаження операторів.
- 17 Наслідування.
- 18 Віртуальні функції.
- 19 Абстрактні класи.
- 20 Контроль доступу до членів класу.
- 21 Шаблони класів.
- 22 Параметризовані функції.
- 23 Обробка виняткових ситуацій.
- 24 Множинне наслідування. Керування доступом.
- 25 Інформація про тип на етапі виконання.
- 26 Стандартні контейнери.
- 27 Визначення нового контейнера.
- 28 Алгоритми стандартної бібліотеки.
- 29 Ітератори і розподіл пам'яті.
- 30 Потоки вводу/виводу.
- 31 Стандартні бібліотеки чисельних методів.

8. Методи навчання

У навчальному процесі застосовуються: лекції, в тому числі з використання мультимедійного проектора та інших ТЗН; лабораторні роботи, індивідуальні заняття; робота в Інтернет.

9. Засоби оцінювання та методи демонстрування результатів.

В процесі вивчення дисципліни “Об’єктно-орієнтоване програмування” використовуються наступні методи оцінювання навчальної роботи студента:

- поточне тестування та опитування;
- підсумкове тестування за кожним змістовним модулем;
- оцінювання виконання лабораторних робіт;
- оцінювання тренінгів;
- оцінювання результатів самостійної роботи;.

10. Політика оцінювання

– *Політика щодо дедлайнів і перескладання.* Для виконання усіх видів завдань

здобувачами освіти і проведення контрольних заходів встановлюються конкретні терміни. Перескладання модулів проводиться в установленому порядку.

– *Політика щодо академічної доброчесності.* Списування під час проведення контрольних заходів заборонені. Під час контрольного заходу учасник може користуватися лише дозволеними допоміжними матеріалами або засобами, йому забороняється в будь-якій формі обмінюватися інформацією з іншими учасниками, використовувати, розповсюджувати, збирати варіанти контрольних завдань.

– *Політика щодо відвідування.* За об’єктивних причин (наприклад, карантин, воєнний стан, хвороба, закордонне стажування) навчання може відбуватись в дистанційній формі за погодженням із керівником курсу з дозволу дирекції факультету.

11. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни “Об’єктно-орієнтоване програмування” визначається як середньозважена величина, в залежності від питомої ваги кожної складової залікового кредиту:

Модуль 1		Модуль 2	Модуль 3
40%	40%	5%	15%
Поточне оцінювання	Модульний контроль	Тренінги	Самостійна робота
Оцінка за даний модуль визначається як середнє арифметичне з оцінок, отриманих за лабораторні роботи №1-8.	Підсумкове модульне тестування за темами №1-13.	Оцінка за виконання одного вибраного завдання тренінгу	Підсумкове тестування за результатами опрацювання теоретичного матеріалу.(25 питань по 4 бали - max 100 балів)

Виконання лабораторних робіт:

90-100 балів (Відмінно) - здобувач освіти самостійно, без помилок, виконав усі кроки в рамках лабораторної роботи, правильно задокументував етапи та вільно оперує поняттями та принципами, що відносяться до теми дисципліни.

75-89 балів (Добре) - здобувач освіти виконав завдання лабораторної роботи, проте в процесі виконання допустив кілька дрібних помилок, які не вплинули на кінцевий результат (наприклад, неточна послідовність дій), в процесі роботи виникали додаткові запитання.

60-74 балів (Задовільно) - здобувач освіти виконав завдання лабораторної роботи, але з суттєвими помилками, наприклад, не з першого разу чи не до кінця. Розуміння поставлених у лабораторній роботі завдань є поверхневим та неповним.

1-59 балів (Незадовільно) - здобувач освіти не зміг виконати завдання або результати були повністю невірними. Не продемонстрував базових навичок роботи з програмним забезпеченням.

Підсумкове модульне тестування - вид контролю, при якому засвоєний здобувачем освіти теоретичний та практичний матеріал оцінюється у форматі тестування. Тестування містить 50 запитань кожна правильна відповідь дає 2 бали, максимум 100 балів.

Тренінг:

90-100 балів (Відмінно) - здобувач освіти самостійно, без помилок, виконав усі етапи завдання, правильно задокументував усі етапи роботи, та вільно оперує поняттями та принципами дисципліни.

75-89 балів (Добре) - здобувач освіти виконав завдання, але з кількома дрібними помилками, які не вплинули на кінцевий результат, в процесі роботи виникали додаткові запитання.

60-74 балів (Задовільно) - здобувач освіти виконав завдання, але з суттєвими помилками, наприклад, не з першого разу. Розуміння поставлених у тренінгу завдань є поверхневим.

1-59 балів (Незадовільно) - здобувач освіти не зміг виконати завдання або результати були повністю невірними. Не продемонстрував достатній рівень навичок роботи з апаратним та програмним забезпеченням.

Самостійна робота: оцінюється у вигляді тестування за результатами опрацювання теоретичного матеріалу. Теоретичні питання: 25 питань по 4 бали, Максимальна оцінка 100 балів.

Шкала оцінювання:

За шкалою Університету	За національною шкалою	За шкалою ECTS
90-100	відмінно	A (відмінно)
85-89	Добре	B (дуже добре)
75-84		C (добре)
65-74	задовільно	D (задовільно)
60-64		E (достатньо)

35-59	незадовільно	FX (незадовільно з можливістю повторного складання)
1-34		F (незадовільно з обов'язковим повторним курсом)

12. Інструменти, обладнання та програмне забезпечення, використання яких передбачає навчальна дисципліна

№	Найменування	Номер теми
1.	Мультимедійний проектор	1-24
2.	Проекційний екран	1-24
3.	Комунікаційне програмне забезпечення (Internet Explorer, Google Chrome, Firefox)	1-24
4.	Операційна система Windows, наявність доступу до мережі Internet	1-24
5.	Персональні комп'ютери	1-24
6.	Комунікаційне програмне забезпечення (Zoom) для проведення занять у режимі онлайн (за необхідності)	1-24
7.	Комунікаційна навчальна платформа (Moodle) для організації дистанційного навчання (за необхідності)	1-24
8.	Базове програмне забезпечення Microsoft Office	1-24
9.	Спеціалізоване програмне забезпечення: - Середовище розробки Microsoft Visual Studio	3-24

Рекомендовані джерела інформації

1. C/C++ language and standard libraries reference: [Електронний ресурс]. – Режим доступу: <https://msdn.microsoft.com/en-us/library/hh875057.aspx>.
2. C++ Crash Course: A Fast-Paced Introduction. / Lospinoso Josh. ISBN 1593278885. – 2019. – 792с.
3. <http://cppstudio.com> – Сайт з ресурсами по вивченню C++
4. <http://www.cplusplus.com/> – Сайт з найновішою довідкою по C++
5. <https://msdn.microsoft.com/> – керівництво по продуктах Microsoft
6. International Standard ISO/IEC 14882:2014(E) – Programming Language C++, ISBN13: 978-0321563842: [Електронний ресурс]. – Режим доступу: <https://isocpp.org/std/the-standard>.
7. Jospeh Albahari. C# 8.0 Pocket Reference. – 1st edition, O'Reilly LCF Publishing. C#. – 1st edition, Learn Coding Fast
8. Mark J. Price C# 8.0 and .NET Core 3.0. – 4th edition. Packt Publishing
9. Andrew Stellman. Head First C#. – 4th edition, O'Reilly Media, Inc. – 2020.
10. Mikael Olsson. C# 10 Quick Syntax Reference. – 4th Ed. Apress. – 2023