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

ЗАТВЕРДЖУЮ:
Декан факультету комп'ютерних інформаційних технологій
_____ Ігор ЯКИМЕНКО
"_____"_____2024 р.

ЗАТВЕРДЖУЮ:
Проректор з науково-педагогічної роботи
_____ Віктор ОСТРОВЕРХОВ
"30" 05 2024 р.

РОБОЧА ПРОГРАМА

з дисципліни **«Сучасні парадигми програмування»**

ступінь вищої освіти – **бакалавр**

галузь знань – **12 "Інформаційні технології"**

спеціальність – **122 „Комп'ютерні науки"**

освітньо-професійна програма – **„Штучний інтелект"**

Кафедра інформаційно-обчислювальних систем і управління

Форма навчання	Курс	Семестр	Лекції (год.)	Лабораторні заняття (год.)	ІРС (год.)	Тренінг (год.)	Самост. робота студ. (год.)	Разом (год.)	Залік (сем.)
Денна	1	2	14	30	2	6	35	90	2

30.05.2024 р.

Тернопіль – ЗУНУ
2024

Робоча програма складена на основі освітньо–професійної програми «Штучний інтелект» підготовки бакалавра галузі знань 12 “Інформаційні технології” спеціальності 122 «Комп’ютерні науки», затвердженої Вченою радою ЗУНУ (протокол №11 від 26.06.2024 р.).

Робочу програму склав доцент кафедри ІОСУ, к.т.н., доцент Павло БИКОВИЙ

Робоча програма затверджена на засіданні кафедри інформаційно-обчислювальних систем і управління, протокол № 1 від 27 серпня 2024 р.

В.о. завідувача кафедри Надія ВАСИЛЬКІВ

Розглянуто та схвалено групою забезпечення спеціальності 122 „Комп’ютерні науки”, протокол № 1 від 30 серпня 2024 р.

Голова групи
забезпечення спеціальності,
д-р техн. наук, професор Мирослав КОМАР

Гарант освітньо-професійної
програми "Штучний інтелект",
канд. техн. наук, доцент Василь КОВАЛЬ

СТРУКТУРА РОБОЧОЇ ПРОГРАМИ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ " СУЧАСНІ ПАРАДИГМИ ПРОГРАМУВАННЯ "

1. Опис дисципліни " СУЧАСНІ ПАРАДИГМИ ПРОГРАМУВАННЯ "

Дисципліна «Сучасні парадигми програмування»	**Галузь знань, спеціальність, СВО**	**Характеристика навчальної дисципліни**
Кількість кредитів – 3	Галузь знань – 12 “Інформаційні технології”	**Статус дисципліни:** обов’язкова дисципліна циклу професійної підготовки **Мова навчання:** Українська
Кількість залікових модулів – 4	Спеціальність – 122 «Комп’ютерні науки»	Рік підготовки: 1 Семестр: 2
Кількість змістових модулів – 3	Освітньо-професійна програма «Штучний інтелект»	Лекції: 14 год. Лабораторні заняття: 30 год.
Загальна кількість годин – 90	Ступінь вищої освіти – бакалавр	Самостійна робота: 35 год. Тренінг: 6 год. Індивідуальна робота: 2 год.
Тижневих годин – 6, з них аудиторних – 3 год.		Вид підсумкового контролю – залік

2. Мета і завдання дисципліни "СУЧАСНІ ПАРАДИГМИ ПРОГРАМУВАННЯ"

2.1. Мета вивчення дисципліни

Мета вивчення дисципліни «Сучасні парадигми програмування» – дати студентам систематизовані відомості про основні принципи розробки програмного забезпечення, методи та особливості реалізації програмного коду та провести огляд основних технологій розробки програмного забезпечення.

2.2. Завдання вивчення дисципліни

В результаті вивчення дисципліни студент повинен:

1. розуміти процеси та технології створення програмного забезпечення, сервісів, та інтерфейсів з інтеграцією зовнішніх даних і програмних продуктів;
2. аналізувати та аргументувати рішення щодо методів розробки веб-застосувань, моделей їх архітектур, засобів інтеграції програмного забезпечення із зовнішніми даними та програмними продуктами, методів забезпечення безпеки коду та даних тощо;
3. здійснювати синтез архітектури і ефективних алгоритмів функціонування програмного забезпечення в процесі, моделювання, проектування та програмування програмного забезпечення складних об’єктів і систем професійної діяльності.
4. проводити емпіричні дослідження, оцінювати ефективність рішень та якість програмного забезпечення, визначати критерії вибору платформи для розробки програмного забезпечення.

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

СК3. Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проєктування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв’язності та нерозв’язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем.

СК8. Здатність проєктувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об’єктно-орієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

2.4. Передумови для вивчення дисципліни

Вивчення курсу „Сучасні парадигми програмування” передбачає наявність систематичних та ґрунтовних знань із дисципліни «Основи програмування».

2.5. Результати навчання

ПР5. Проектувати, розробляти та аналізувати алгоритми розв’язання обчислювальних та логічних задач, оцінювати ефективність та складність

алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.

ПР9. Розробляти програмні моделі предметних середовищ, вибирати парадигму програмування з позицій зручності та якості застосування для реалізації методів та алгоритмів розв’язання задач в галузі комп’ютерних наук.

ПР14. Застосовувати знання методології та CASE-засобів проектування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем..

3. Програма навчальної дисципліни «СУЧАСНІ ПАРАДИГМИ ПРОГРАМУВАННЯ» (Частина 1)

Змістовий модуль 1. Базові парадигми програмування та алгоритми

Тема 1. Основи парадигм програмування та алгоритми

Парадигми і мови програмування. Узагальнене, об’єктно-орієнтоване, функціональне і логічне програмування. Програмні середовища. Алгоритми розв’язання обчислювальних та логічних задач. Оцінка ефективності та складності алгоритмів.

Тема 2. Функції та рекурсія

Оголошення функцій, передача параметрів, рекурсія, шаблони функцій, вказівники на функції.

Змістовий модуль 2. Об'єктно-орієнтоване програмування та моделювання

Тема 3. Основи об'єктно-орієнтованого програмування

Принципи ООП. Абстрагування, модульність, доступом до даних, конструктори, деструктори та статичні члени класу.

Тема 4. Успадкування, поліморфізм та перевантаження операцій

Механізм успадкування, поліморфізм, віртуальні функції та перевантаження операцій.

Тема 5. Уніфікована мова моделювання UML

Загальна характеристика. Діаграми. Зображення класу. Асоціації між класами. Агрегація та композиція. Узагальнення та залежності.

Тема 6. Шаблони та стандартна бібліотека шаблонів

Параметризовані функції та класи. Робота з бібліотеками C++ (STL, Boost, OpenCV, Dlib, TensorFlow), які забезпечують додаткову функціональність, зокрема комп'ютерний зір та машинне навчання.

Змістовий модуль 3. Взаємодія програмних компонентів та мережеве програмування

Тема 7. Взаємодія між програмними компонентами: сокети та мережеві протоколи

Розробка програмного забезпечення для Windows. Процеси та потоки в ОС Windows. Робота з мережевими протоколами та програмним інтерфейсом протоколів за допомогою сокетів API та Boost.Asio.

4. Структура залікового кредиту з дисципліни «СУЧАСНІ ПАРАДИГМИ ПРОГРАМУВАННЯ» (Частина 1)

Денна форма навчання

Тема	Кількість годин					
	Лекції	**Лабора-торні заняття**	**Індивідуа-льна робота**	**Тренінг**	**Самостійна робота**	**Контрольні заходи**
Змістовий модуль 1. Базові парадигми програмування та алгоритми						
Тема 1. Основи парадигм програмування та алгоритми	2	2	0,5	2	5	Опитування під час заняття
Тема 2. Функції та рекурсія	2	2			5	Опитування під час заняття
Змістовий модуль 2. Об'єктно-орієнтоване програмування та моделювання						
Тема 3. Основи об'єктно-орієнтованого програмування	2	4	0,5	2	5	Опитування під час заняття
Тема 4. Успадкування, поліморфізм та перевантаження операцій	2	2			5	Опитування під час заняття
Тема 5. Уніфікована мова моделювання UML	1	4			5	Опитування під час заняття
Тема 6. Шаблони та стандартна бібліотека шаблонів	2	8			5	Опитування під час заняття
Змістовий модуль 3. Взаємодія програмних компонентів та мережеве програмування						
Тема 7. Взаємодія між програмними компонентами: сокети та мережеві протоколи	3	8	1	2	5	Опитування під час заняття
Разом	**14**	**30**	**2**	**6**	**35**	

5. Тематика лабораторних занять

.

1. Керуючі структури в C++.
2. Функції та шаблони функцій.
3. Робота з класами, інкапсуляція, конструктори та деструктори.
4. Реалізація успадкування та поліморфізму
5. Аналіз і проектування програмних систем на основі UML.
6. Шаблони класів та STL
7. Робота з бібліотеками C++: STL, Boost, OpenCV, Dlib, TensorFlow
8. Створення процесів і потоків в ОС Windows
9. Розробка мережевих застосунків в ОС Windows

6. Самостійна робота

Студенти проводять дослідження на обрану з поданого нижче переліку тему та готують результати дослідження у вигляді реферату та виконують одне із практичних індивідуальних завдань.

Перелік тем:

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

Перелік практичних індивідуальних завдань:

1. Створити вектор, що містить об’єкти класу “учасник”: прізвище, телефон, адреса. Продемонструвати основні операції з контейнером.
2. Створити список, що містить об’єкти класу “фігура”: координати на шахівниці, колір. Продемонструвати основні операції з контейнером.
3. Створити чергу, що містить об’єкти класу “прямокутник”: координати верхнього лівого і нижнього правого кутів, порядковий номер. Продемонструвати основні операції з контейнером.
4. Створити стек, що містить об’єкти класу “нота”: назва, октава, тривалість звучання. Продемонструвати основні операції з контейнером.
5. Створити вектор, що містить об’єкти класу “іграшка”: ціна, назва, кількість на складі. Продемонструвати основні операції з контейнером.
6. Створити список, що містить об’єкти класу “годинник”: стиль відображення (24 чи 12), години, хвилини, секунди. Продемонструвати основні операції з контейнером.
7. Створити чергу, що містить об’єкти класу “Товар”: назва, порядковий номер, постачальник, ціна, кількість одиниць. Продемонструвати основні операції з контейнером.
8. Створити список, що містить об’єкти класу “Точка на площині”: координати. Продемонструвати основні операції з контейнером.
9. Створити список, що містить об’єкти класу “учасник змагань”: країна, вид спорту, назва учасника. Продемонструвати основні операції з контейнером.
10. Створити стек, що містить об’єкти класу “довге число”: кількість знаків, основа системи числення. Продемонструвати основні операції з контейнером.
11. Створити вектор, що містить об’єкти класу “РадіоСтанція”: координати_частота, назва. Продемонструвати основні операції з контейнером.
12. Створити вектор, що містить об’єкти класу “ істота ”: координати, вік, назва. Продемонструвати основні операції з контейнером.
13. Створити чергу, що містить об’єкти класу “кліматичні умови”: температура. освітленість. вологість, кислотність гранта . Продемонструвати основні операції з контейнером.
14. Створити чергу, що містить об’єкти класу “давач”: поточне значення, максимально і мінімально допустимі, тип (t, p, i, pH), сигнал тривоги. Продемонструвати основні операції з контейнером.
15. Створити список, що містить об’єкти класу “товар на складі”: назва, кількість, місце розташування. Продемонструвати основні операції з контейнером.
16. Створити дек, що містить об’єкти класу “фраза”: (кількість слів, кількість символів, кількість різних символів. Продемонструвати основні операції з контейнером.
17. Створити вектор, що містить об’єкти класу “коло”: x, y, R. Продемонструвати основні операції з контейнером.

18. Створити список, що містить об’єкти класу “книга”: код УДК, назва, автор, рік видання, кількість сторінок. Продемонструвати основні операції з контейнером.
19. Створити вектор, що містить об’єкти класу “обладнання”: назва, вартість, дата виготовлення, група (5, 15, 25 – амортизація) і клас “модернізація обладнання” (дата ремонту, вартість ремонту). Продемонструвати основні операції з контейнером.
20. Створити дек, що містить об’єкти класу “товар”: назва, виробник. Продемонструвати основні операції з контейнером.
21. Створити вектор, що містить об’єкти класу “банківський рахунок”: назва банку, номер рахунку, МФО. Продемонструвати основні операції з контейнером.
22. Створити чергу з приорітетом, що містить об’єкти класу “підключення”: назва мережі, тариф, наявність пільги. Продемонструвати основні операції з контейнером.
23. Створити вектор, що містить об’єкти класу “іграшка”: ціна, назва, кількість на складі. Продемонструвати основні операції з контейнером.
24. Створити дек, що містить об’єкти класу “учасник змагань”: країна, вид спорту, назва учасника. Продемонструвати основні операції з контейнером.
25. Створити дек, що містить об’єкти класу “товар”: назва, виробник. Продемонструвати основні операції з контейнером.

7. Тренінг з дисципліни

№ з/п	Вид роботи	Порядок проведення тренінгу
1.	Вступна частина	Ознайомлення студентів з темою тренінгового заняття «Робота з бібліотеками C++ для комп’ютерного зору» і видача завдання
2.	Практична частина	1. Ознайомитися з основами OpenCV та Dlib. 2. Виконати завдання з обробки зображень за допомогою OpenCV (наприклад, виявлення країв або фільтрація). 3. Використати Dlib для розпізнавання облич або трекінгу об’єктів. 4. Оформити звіт із кодом, результатами та описом виконання завдань.
3.	Підведення підсумків	Обговорення результатів, аналіз помилок та рекомендації для подальшого розвитку в напрямку комп’ютерного зору.

8. Засоби оцінювання та методи демонстрування результатів навчання

У процесі вивчення дисципліни «Сучасні парадигми програмування» використовуються наступні засоби оцінювання та методи демонстрування результатів навчання:

- поточне тестування та опитування;
- оцінювання результатів лабораторних робіт;
- оцінювання виконання завдань тренінгу;
- оцінювання виконання завдань самостійної роботи.

9. Критерії, форми поточного та підсумкового контролю

Підсумковий бал (за 100-бальною шкалою) з дисципліни «Сучасні парадигми програмування» визначається як середньозважена величина, залежно від питомої ваги кожної складової залікового кредиту:

Модуль 1		Модуль 2		Модуль 3	Модуль 4
20 %	20 %	20 %	20 %	5 %	15%
Поточне оцінювання	Модульний контроль 1	Поточне оцінювання	Модульний контроль 2	Тренінг	Самостійна робота
Оцінка визначається як середнє арифметичне з оцінок, отриманих за виконання і захист 4 лабораторних робіт	Модульна контрольна робота (15 тестових завдань та 1 практичне)	Оцінка визначається як середнє арифметичне з оцінок, отриманих за виконання і захист 5 лабораторних робіт	Модульна контрольна робота (15 тестових завдань та 1 практичне)	Виконання завдань під час тренінгу (4 завдання)	Виконання завдання для самостійної роботи

Оцінка за “Поточне оцінювання” визначається як середнє арифметичне з оцінок, отриманих під час занять (тестові завдання під час лекцій та оцінки лабораторних робіт).

Модуль “Тренінг” визначається, як середнє арифметичне з оцінок, отриманих під час тренінгу.

Модуль “Самостійна робота” визначається, як середнє арифметичне з оцінок, отриманих під час виконання завдання самостійної роботи, а саме оцінка за теоретичні знання, представлення результатів та проведених досліджень.

Шкала оцінювання:

За шкалою ЗУНУ	**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)

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

№	Найменування	Номер теми
1	Редактор Visual Studio Code	1-4, 6, 7
2	Середовище MS Visual Studio	1-4, 6, 7
3	Редактор UML-діаграм: PlantUML	5

11. Рекомендовані джерела інформації

Основна література

1. Tony Gaddis. Starting Out with C++ from Control Structures to Objects. 9th edition, Pearson, 2022, 1311 p.
2. Bjarne Stroustrup. Tour of C++, A (C++ In-Depth Series), 3rd Edition. Addison-Wesley Professional, 2022, 320 p.
3. Електронний курс з дисципліни «Сучасні парадигми програмування» для студентів галузі знань 12 «Інформаційні технології» спеціальності 122 «Комп'ютерні науки» на платформі Moodle ЗУНУ - Тернопіль, 2021. https://moodle.wunu.edu.ua.
4. Биковий П.Є. Опорний конспект лекцій з дисципліни «Сучасні парадигми програмування» для студентів спеціальності 122 «Комп'ютерні науки». - Тернопіль: ЗУНУ, 2021. - 80 с.
5. Биковий П.Є. Методичні вказівки до виконання лабораторних робіт з дисципліни «Сучасні парадигми програмування» для студентів освітньо-професійної програми «Штучний інтелект» спеціальності 122 «Комп'ютерні науки» першого (бакалаврського) рівня вищої освіти. - Тернопіль: ЗУНУ, 2024. - 52 с.
6. Thadchanamoorthy, Jathurshan & Ahamed, Faamie. (2024). UML Class Diagram. http://dx.doi.org/10.13140/RG.2.2.27945.51049

Додаткова література

1. Wang, C., Wang, B., Liang, P. and Liang, J., 2024. Assessing UML Models by ChatGPT: Implications for Education. https://doi.org/10.48550/arXiv.2412.17200, 17 p.
2. Abhishek Jain. Face detection using opencv and dlib library in C++ and how to make CMakeLists.txt to include opencv and dlib, 2024. Електронний ресурс: https://medium.com/@abhishekjainindore24/face-detection-using-opencv-and-dlib-library-in-c-592d9bf269ff

3. Беркунський Є. Ю., Павленко А. Ю. Алгоритмізація та програмування мовами Kotlin, C/C++: навчальний посібник. Миколаїв : НУК, 2022. 256 с. URL: https://rep.nuos.edu.ua/server/api/core/bitstreams/59c1a3c1-d372-4a36-983b-a150db560d24/content.

4. Д. Загородня, П. Биковий, Х. Ліп'яніна-Гончаренко, В. Дорош, І. Кіт, А. Каньовський, за редакцією проф. Саченка Анатолія Олексійовича. Методи та засоби ідентифікації та класифікації об'єктів за характерними точками їх контурів. - Тернопіль : Економічна думка ЗУНУ, 2020. - 164 с.

5. D. Zahorodnia, P. Bykovyy, A. Sachenko V. Krylov, G.Shcherbakova, I. Kit, A.Kaniovskyi, M. Dacko. Analysis of Objects Classification Approaches Using Vectors of Inflection Points. Lecture Notes in Computational Intelligence and Decision Making. Springer Nature Switzerland AG. V.Lytvynenko et al.(Eds.) (ISSN 2194-5357):, 2020. - P.148-157.