

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ



ЗАТВЕРДЖУЮ

Декан факультету інформаційних технологій

Тетяна ГОВОРУЩЕНКО

Ім'я, ПРІЗВИЩЕ

09

2025 р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтоване проєктування

Назва дисципліни

Галузь знань – F Інформаційні технології

Спеціальність – F3 Комп'ютерні науки

Рівень вищої освіти – Перший (бакалаврський)

Освітньо-професійна програма – Комп'ютерні науки

Обсяг дисципліни – 6 кредитів ЄКТС

Шифр дисципліни – ОФП.08

Мова навчання – Українська

Статус дисципліни: Обов'язкова (фахової підготовки)

Факультет – Інформаційних технологій

Кафедра – Комп'ютерних наук

Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин						Курсовий проєкт	Курсова робота	Форма семестрового контролю	
					Аудиторні заняття					Самостійна робота, у т.ч. ІРС			Залік	Іспит
			Кредити ЄКТС	Години	Разом	Лекції	Лабораторні роботи	Практичні заняття	Семінарські заняття					
Д	1	2	6	180	66	32	34			114				+
Разом ДФН			6	180	66	32	34			114				2

Робоча програма складена на основі освітньо-професійної програми «Комп'ютерні науки» за спеціальністю F3 «Комп'ютерні науки»

Робоча програма складена

Підпис автора

д-р філософії Марина МОЛЧАНОВА

Науковий ступінь, вчене звання, Ім'я, ПРІЗВИЩЕ автора

канд. техн. наук, доц. Олександр МАЗУРЕЦЬ

Науковий ступінь, вчене звання, Ім'я, ПРІЗВИЩЕ автора

Схвалена на засіданні кафедри Комп'ютерних наук

Протокол від 29.08.2025 № 1.

Зав. кафедри

Підпис

Олександр БАРМАК

Ім'я, ПРІЗВИЩЕ

Робоча програма розглянута та схвалена вченою радою факультету інформаційних технологій

Голова вченої ради факультету

Підпис

Тетяна ГОВОРУЩЕНКО

Ім'я, ПРІЗВИЩЕ

Хмельницький 2025

ЛИСТ ПОГОДЖЕННЯ

Посада	Назва кафедри	Підпис	Ініціали, ПРІЗВИЩЕ
Завідувач кафедри, д-р. техн. наук, проф.	Комп'ютерних наук		Олександр БАРМАК
Гарант освітньо-професійної програми, канд. техн. наук, доц.	Комп'ютерних наук		Олександр МАЗУРЕЦЬ

3. Пояснювальна записка

Дисципліна «Об'єктно-орієнтоване проектування» є однією із дисциплін фахової підготовки і займає провідне місце у підготовці здобувачів першого (бакалаврського) рівня вищої освіти, очної (денної) (далі – денної) форми здобуття вищої освіти, які навчаються за освітньо-професійною програмою «Комп'ютерні науки» в межах спеціальності F3 «Комп'ютерні науки».

Пререквізити – Алгоритмізація та програмування (ОЗП.05)

Кореквізити – Теорія алгоритмів (ОЗП.08), Методи та системи штучного інтелекту (ОФП.01), Методи та системи штучного інтелекту (курсний проєкт) (ОФП.02), Проектування баз даних (ОФП.07), Технології створення програмних продуктів (ОФП.09), Основи програмної інженерії та тестування програмного забезпечення (ОФП.10), Управління ІТ-проєктами (ОФП.15), Проєктно-технологічна практика (ОФП.16), Професійна практика (ОФП.17), Кваліфікаційна робота (ОФП.18).

Відповідно до освітньої програми дисципліна сприяє забезпеченню:

компетентностей: здатність розв'язувати складні спеціалізовані задачі та практичні проблеми у галузі комп'ютерних наук або у процесі навчання, що передбачає застосування теорій та методів інформаційних технологій і характеризується комплексністю та невизначеністю умов (ІК); здатність до абстрактного мислення, аналізу та синтезу (ЗК 01); здатність застосовувати знання у практичних ситуаціях (ЗК 02); знання та розуміння предметної області та розуміння професійної діяльності (ЗК 03); здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, 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 (ФК 03); 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 (ФК 08).

програмних результатів навчання: застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук (ПРН 01); проєктувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій (ПРН 05); 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 (ПРН 09); застосовувати знання методології та CASE-засобів проєктування складних систем, методів структурного аналізу систем, об'єктно-орієнтованої методології проєктування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничотехнічних систем (ПРН 14)

Мета дисципліни. Формування у здобувачів вищої освіти теоретичних знань та практичних навичок з проектування і розроблення програмного забезпечення із застосуванням принципів об'єктно-орієнтованого підходу. У процесі навчання вивчаються основні поняття ООП (класи, об'єкти, наслідування, інкапсуляція, поліморфізм), принципи SOLID, методи візуального моделювання програмних систем (UML-діаграми), інструменти предметно-орієнтованого проектування та сучасні засоби програмування. Це дозволяє студентам опанувати понятійний апарат, навчитися будувати програмні моделі предметних середовищ і розв'язувати практичні завдання у сфері комп'ютерних наук.

Предмет дисципліни. Методи, моделі, технології та програмні засоби об'єктно-орієнтованого проектування.

Завдання дисципліни. формування компетентностей та забезпечення програмних результатів навчання з проектування та розробки програмного забезпечення з використанням об'єктно-орієнтованого підходу, зокрема: проектування та розроблення програмного забезпечення із застосуванням об'єктно-орієнтованого програмування, з відповідними моделями, методами й структурами даних; розроблення програмних моделей предметних середовищ засобами предметно-орієнтованого проектування (DDD) для реалізації методів та алгоритмів розв'язання задач в галузі комп'ютерних наук; використання формальних мов і моделей алгоритмічних обчислень; застосування існуючих та розроблення нових алгоритмів розв'язання задач в галузі комп'ютерних наук; розробка й реалізація проєктів зі створення програмного забезпечення, у тому числі в непередбачуваних умовах, використовуючи засоби та методи проектування та розробки програмного забезпечення із застосуванням об'єктно-орієнтованого програмування; застосування знання об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.

Результати навчання. Студент, який успішно завершив вивчення дисципліни, повинен: вміло використовувати понятійний апарат та основоположні поняття об'єктно-орієнтованого проектування: абстракція даних, спадкування, інкапсуляція, поліморфізм; володіти принципами побудови об'єктно-орієнтованої структури програм; знати поняття об'єктно-орієнтованого проектування: клас, об'єкт, метод, властивість; вміти писати програми мовою C# застосовуючи GUI, вміти реалізовувати принципи та алгоритми об'єктно-орієнтованого проектування, SOLID-принципи при написанні програмних продуктів. 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. Проєктувати програмне забезпечення з використанням методів та засобів візуального проектування (діаграми Use Case, Statechart, Activity, Sequence). Застосовувати знання об'єктно-орієнтованої методології проектування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.

4. Структура залікових кредитів дисципліни

Назва розділу	Кількість годин, відведених на:		
	лекції	лабораторні заняття	СРС
<i>Розділ 1</i> Основи парадигм програмування та об'єктно-орієнтованого підходу	6	8	22
<i>Розділ 2</i> Базові конструкції та елементи ООП у C#	8	8	22
<i>Розділ 3</i> Візуальні та інструментальні засоби моделювання	8	12	28
<i>Розділ 4</i> Принципи якісного проєктування програмних систем	6	6	22
<i>Розділ 5</i> Розширення об'єктно-орієнтованого підходу	4		20
Разом за семестр:	32	34	114

5. Програма навчальної дисципліни

5.1 Зміст лекційного курсу

Номер лекції	Перелік тем лекцій, їх анотації	Кількість годин
	2-ий семестр	
	Розділ 1 Основи парадигм програмування та об'єктно-орієнтованого підходу	6
1	Лекція 1. Парадигми програмування. Об'єктно-орієнтований підхід до програмування. Парадигми програмування для проєктування та розробки програмного забезпечення (узагальнене, функціональне, процедурне, об'єктно-орієнтоване та логічне програмування). Об'єктно-орієнтований підхід до програмування. Основні елементи об'єктно-орієнтованого підходу (об'єкти, класи, методи). Діаграми класів. Літ.: [1] с. 8-10, [2] с. 19-21, [3] с. 29-34, [4] с. 10-13	2
2	Лекція 2. Принципи об'єктно-орієнтованого підходу до програмування. Основні принципи об'єктно-орієнтованого підходу (спадкування, інкапсуляція, поліморфізм). Перехід з процедурної розробки на об'єктно-орієнтовану. Реалізація принципів об'єктно-орієнтованого програмування засобами розробки (платформа .NET, середовище розробки Visual Studio, мова програмування C#). Літ.: [1] с. 10-15, [2] с. 8-22, [4] с. 5-9	2
3	Лекція 3. Прикладне застосування об'єктно-орієнтованого підходу. Інтерфейси та реалізації. Проєктування класів. Проєктування інтерфейсів користувача та поведінка об'єктів. Літ.: [1] с. 18-20, [2] с. 94-109, [4] с. 27	2
	Розділ 2 Базові конструкції та елементи ООП у C#	8
4	Лекція 4. Абстрактні класи. Структури та перерахування. Абстрактні класи. Поняття розгорнутого і типу-посилання. Структури - реалізація розгорнутих класів. Синтаксис структур. Порівняння структур і класів. Вбудовані структури. Перерахування – окремий випадок класу. Особливості перерахувань. Літ.: [1] с. 20 - 23, [2] с. 88-90, [4] с. 39-47	2
5	Лекція 5. Конструктори класів. Об'єктно-орієнтовані концепції. Конструктор класу. Конструктор за замовчуванням. Множинні конструктори. Перехоплення виключень в об'єктно-орієнтованих програмах. Агрегація, асоціація і композиція. Літ.: [1] с. 34-40, [2] с. 111-125, [4] с. 13-15	2
6	Лекція 6. Делегати та події. Функціональний тип. Функції вищих порядків. Два способи взаємодії частин при побудові складних систем. Клас Delegate. Властивості делегатів Класу Delegate. Операції над делегатами. Комбінування делегатів. Список викликів. Lambda вирази. Події – загальні поняття. Події для багатоадресної передачі. Літ.: [1] с. 43-49, [2] с. 193-197, [4] с. 59-96	2
7	Лекція 7. Використання віртуальних методів. Віртуальні методи і їх властивості. Реалізація віртуальних методів в C#. Літ.: [1] с. 25-26, [2] с. 79-87, [4] с. 29-33	2
	Розділ 3 Візуальні та інструментальні засоби моделювання	8
8	Лекція 8. Візуальні засоби об'єктно-орієнтованого проєктування. Діаграми варіантів використання (Use case diagram). Уніфікована мова моделювання UML. Елементи діаграми варіантів використання. Виділення групи дійових осіб. Опис прецедентів. Побудова діаграм варіантів використання в середовищі Software Ideas Modeler. Літ.: [1] с. 51-52, 55-59, [3] с. 17-29	2
9	Лекція 9. Діаграма станів (Statechart diagram). Елементи діаграм станів. Умовні позначення. Побудова діаграм станів в середовищі Software Ideas Modeler. Літ.: [1] с. 52-55, 59-60	2
10	Лекція 10. Діаграми активності (Activity diagram). Елементи діаграм активності. Особливості використання діаграми активності. Побудова діаграм активності в середовищі Software Ideas Modeler. Літ.: [1] с. 61-66	2
11	Лекція 11 Діаграми послідовності (Sequence diagram). Елементи діаграм послідовності. Особливості використання діаграми послідовності. Побудова діаграм послідовності в середовищі Software Ideas Modeler. Літ.: [1] с. 67-70, [3] с. 43-45	2
	Розділ 4 Принципи якісного проєктування програмних систем	6
12	Лекція 12. SOLID-принципи. S-принцип, О-принцип. Базові принципи об'єктно-орієнтованого дизайну (OOD). SOLID-принципи в розрізі загальної концепції ООП. Принцип єдиного обов'язку (Single responsibility principle). Принцип відкритості/закритості (Open-closed principle).	2

	Літ.: [1] с. 73-75	
13	<i>Лекція 13. SOLID-принципи. L-принцип, I-принцип та D-принцип.</i> Принцип підстановки (Liskov Substitution principle). Принцип розділення інтерфейсу (Interface segregation principle). Принцип інверсії залежностей (Dependency inversion principle) Літ.: [1] с. 75-79	2
14	<i>Лекція 14. Патерни проєктування GRASP. Засоби аналізу та забезпечення якості проєктування програмного забезпечення. Антипатерни.</i> Інформаційний експерт (Information expert), Творець (Creator), Низька зв'язаність (Low Coupling) і Високе зачеплення (High cohesion), Контролер (Controller), Перенаправлення (Indirection), Поліморфізм (Polymorphism), Стійкий до змін (Protected variations), Чиста вигадка (Pure fabrication) . Аналіз якості проєктування програмного забезпечення (Code Smells). Процес оптимізації коду (Refactoring). Антипатерни (Anti-patterns) та їх вирішення. Літ.: [1] с. 79-81	2
	Розділ 5 Розширення об'єктно-орієнтованого підходу	4
15	<i>Лекція 15. Введення в бази даних.</i> Основні поняття баз даних (БД). Області застосування БД. Приклади використання БД застосунками на мові С#. Літ.: [1] с. 23-25, [3] с.34-36	2
16	<i>Лекція 16. Сучасні підходи до проєктування програмного забезпечення. Предметно-орієнтоване проєктування (DDD). Застосування DDD для розробки програмних моделей предметних середовищ.</i> Інноваційні парадигми проєктування програмного забезпечення. Задачі DDD – предметно-орієнтованого проєктування (ПОП). Відмінності між ООП та ПОП. Елементи ПОП (обмежений контекст, сутності, об'єкт-значення, агрегат, служби предметних областей). Приклади практичного застосування ПОП. Літ.: [1] с. 82-84	2
	Разом за семестр:	32

5.2 Зміст лабораторних занять

№ п/п	Тема лабораторного заняття	Кількість годин
1	Застосування об'єктно-орієнтованого підходу для консольних застосунків. Літ.: [1] с. 7-17	4
2	Створення Windows Forms ООП-застосунку «Візуальний калькулятор» на мові С# в Visual Studio .NET. Літ.: [1] с. 18-34	4
3	Розширення функціональності об'єктно-орієнтованого проєкту. Літ.: [1] с. 34-42	4
4	Використання делегатів та подій. Створення Windows Forms застосунку «Банкомат» на мові С# в Visual Studio .NET із використанням делегатів та подій. Літ.: [1] с. 43-51	4
5	Проектування UML-діаграм: діаграми варіантів використання (Use case diagram) та діаграми станів (Statechart diagram). Літ.: [1] с. 52-63	4
6	Проектування UML-діаграм: діаграми активності (Activity diagram). Літ.: [1] с. 64-70	4
7	Побудова UML діаграм: діаграми послідовності (Sequence diagram). Літ.: [1] с. 71-76	4
8	Удосконалення об'єктно-орієнтованого коду за SOLID-принципами. Аналіз програмного коду та виявлення порушень SOLID-принципів. Виправлення програмного коду. Літ.: [1] с. 77-97	6
	Разом:	34

5.3 Зміст самостійної (у т.ч. індивідуальної) роботи здобувача вищої освіти

Самостійна робота студентів усіх форм здобуття освіти полягає у систематичному опрацюванні програмного матеріалу з відповідних джерел інформації, підготовці до лабораторних занять, контрольних робіт, виконанні індивідуальних завдань. Крім цього до послуг студентів сторінка навчальної дисципліни у Модульному середовищі для навчання, де розміщені Робоча програма дисципліни та необхідні документи з її навчально-методичного забезпечення.

Зміст самостійної роботи студентів денної форми навчання

Номер тижня	Вид самостійної роботи	Кількість годин
1–3	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Виконання завдання для самостійної роботи №1. (За одержаним варіантом, створити консольний застосунок мовою C# з використанням принципів ООП. Оформити звіт з виконання ІЗ). Літ: [1] с. 94-96	18
4-5	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Продовження виконання завдання для самостійної роботи №1 (За одержаним варіантом, доповнити застосування з завдання 1, перетворивши застосування у Windows Forms. Оформити звіт з виконання ІЗ). Літ: [1] с. 94-96	12
6-7	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Підготовка до проміжного контрольного заходу. Продовження виконання завдання для самостійної роботи №1 (За одержаним варіантом, розширити функціональність застосування із завдання 2, додавши обробку виключень та підходи ООП такі як: композицію, агрегацію, асоціацію, конструктори. Або письмово привести аргументацію, що вони не доцільні відповідно до індивідуального варіанту завдання. Оформити звіт з виконання ІЗ). Літ: [1] с. 94-96	14
8-9	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Виконання завдання для самостійної роботи №1 (За одержаним варіантом, доповнити проєкт, замінивши ряд методів на Лямбда-вирази. Додати події та делегати. Оформити звіт з виконання ІЗ.) Літ: [1] с. 94-96	12
10-11	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Виконання завдання для самостійної роботи №2 (За одержаним варіантом, до розроблюваного проєкту створити UML діаграми станів та варіантів використання у середовищі Software Ideas Modeler. Оформити звіт з виконання ІЗ). Літ: [1] с. 94-96	12
12-13	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Виконання завдання для самостійної роботи №2 (За одержаним варіантом, до розроблюваного проєкту створити UML діаграми активності у середовищі Software Ideas Modeler. Оформити звіт з виконання ІЗ). Літ: [1] с. 94-96	12
14-16	Опрацювання лекційного матеріалу. Підготовка до лабораторної роботи. Підготовка до захисту лабораторної роботи. Підготовка до проміжного контрольного заходу. Виконання завдання для самостійної роботи №2 (За одержаним варіантом, до розроблюваного проєкту створити UML діаграми послідовності у середовищі Software Ideas Modeler. Оформити звіт з виконання ІЗ). Підготовка до підсумкового контрольного заходу. Презентація результатів виконання індивідуальних завдань. Літ: [1] с. 94-96	34
Всього		114

Примітки: КР – контрольна робота, ІЗ – індивідуальне завдання.

На самостійне опрацювання студентів виносяться визначені у методичних рекомендаціях до лабораторних занять та самостійної роботи питання з відповідних тем. Керівництво самостійною роботою та контроль за виконанням індивідуального завдання здійснюється викладачем згідно з розкладом консультацій у позаурочний час.

Вимоги до виконання індивідуального домашнього завдання викладені в Модульному середовищі для навчання на сторінці навчальної дисципліни.

6. Технології та методи навчання

Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: лекції (з використанням методів візуалізації, проблемного й інтерактивного навчання, мотиваційних прийомів, інформаційно-комунікаційних технологій); лабораторні заняття (з використанням інструктування, демонстрування, розв'язування типових і прикладних задач, аналізу кейсів, ситуаційних завдань, елементів дискусії тощо); самостійна робота (опрацювання теоретичного матеріалу, підготовка до виконання лабораторних робіт, поточного та підсумкового контролю, виконання індивідуальних завдань), з використанням інформаційно-комп'ютерних технологій та технологій дистанційного навчання.

7. Методи контролю

Поточний контроль здійснюється під час аудиторних лабораторних занять, а також у дні проведення контрольних заходів, встановлених робочою програмою і графіком освітнього процесу, в т.ч. з використанням Модульного середовища для навчання. При цьому використовуються такі методи поточного контролю:

- оцінювання результатів виконання індивідуального завдання;
- оцінювання контрольних робіт;
- оцінювання результатів захисту лабораторних робіт.

При виведенні підсумкової семестрової оцінки враховуються результати як поточного контролю, так і підсумкового контролю, який проводиться з усього матеріалу дисципліни за білетами, попередньо розробленими і затвердженими на засіданні кафедри. Здобувач вищої освіти, який набрав з будь-якого виду навчальної роботи, суму балів нижчу за 60 відсотків від максимального балу, **не допускається** до семестрового контролю, поки не виконає обсяг роботи, передбачений Робочою програмою. Здобувач вищої освіти, який набрав позитивний середньозважений бал (60 відсотків і більше від максимального балу) з усіх видів поточного контролю і не склав іспит, вважається таким, який **має** академічну заборгованість. Ліквідація академічної заборгованості із семестрового контролю здійснюється у період екзаменаційної сесії або за графіком, встановленим деканатом відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ».

8. Політика дисципліни

Політика навчальної дисципліни загалом визначається системою вимог до здобувача вищої освіти, що передбачені чинними положеннями Університету про організацію і навчально-методичне забезпечення освітнього процесу. Зокрема, проходження інструктажу з техніки безпеки; відвідування занять з дисципліни є обов'язковим. За об'єктивних причин (підтверджених документально) теоретичне навчання за погодженням із лектором може відбуватись в он-лайн режимі. Успішне опанування дисципліни і формування фахових компетентностей і програмних результатів навчання передбачає необхідність підготовки до лабораторного заняття (вивчення теоретичного матеріалу з теми роботи, попередню підготовку протоколу роботи (наведені у Методичних рекомендаціях до лабораторних занять)), активно працювати на занятті, якісно підготувати звіт, захистити результати виконаної роботи, брати участь у дискусіях щодо прийнятих конструктивних рішень при виконанні здобувачами лабораторних робіт тощо.

Здобувачі вищої освіти мають дотримуватися встановлених термінів виконання всіх видів навчальної роботи відповідно до робочої програми навчальної дисципліни. Термін захисту лабораторної роботи вважається своєчасним, якщо студент захистив її на наступному після виконання роботи занятті. Пропущене лабораторне заняття студент зобов'язаний відпрацювати у встановлений викладачем термін, але не пізніше, ніж за два тижні до кінця теоретичних занять у семестрі.

Засвоєння студентом теоретичного матеріалу з дисципліни оцінюється за результатами опитування під час лабораторних занять, виконання індивідуального завдання та контрольних робіт. Виконання індивідуального завдання завершується його здачею на перевірку у терміни, встановлені графіком самостійної роботи.

Здобувач вищої освіти, виконуючи самостійну роботу або індивідуальну роботу з дисципліни, має дотримуватися політики доброчесності (заборонені списування, плагіат (в т.ч. із використанням мобільних девайсів)). У разі виявлення порушення політики академічної доброчесності в будь-яких видах навчальної роботи здобувач вищої освіти отримує незадовільну оцінку і має повторно виконати завдання з відповідної теми (виду роботи), що передбачені робочою програмою. Будь-які форми порушення академічної доброчесності **не допускаються**.

У межах вивчення навчальної дисципліни допускається визнання та зарахування результатів навчання, здобутих шляхом неформальної освіти на відкритих освітніх платформах. Такі результати можуть враховуватися за умови їх відповідності компетентностям і програмним результатам навчання, визначеним робочою програмою дисципліни, або у разі забезпечення вивчення окремих тем і видів робіт, що передбачені програмою. (детальніше у Положенні про порядок визнання та зарахування результатів навчання здобувачів вищої освіти у ХНУ).

9. Оцінювання результатів навчання студентів у семестрі

Оцінювання академічних досягнень здобувача вищої освіти здійснюється відповідно до «Положення про контроль і оцінювання результатів навчання здобувачів вищої освіти у ХНУ». При поточному оцінюванні виконаної здобувачем роботи з кожної структурної одиниці і отриманих ним результатів викладач виставляє йому певну кількість балів із встановлених Робочою програмою для цього виду роботи. При цьому кожна структурна одиниця навчальної роботи може бути зарахована, якщо здобувач набрав не менше 60 відсотків (мінімальний рівень для позитивної оцінки) від максимально можливої суми балів, призначеної структурній одиниці.

При оцінюванні результатів навчання здобувачів вищої освіти з будь-якого виду навчальної роботи (структурної одиниці) рекомендується використовувати наведені нижче узагальнені критерії:

Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти

Оцінка та рівень досягнення здобувачем запланованих ПРН та сформованих компетентностей	Узагальнений зміст критерія оцінювання
--	--

Відмінно (високий)	Здобувач вищої освіти глибоко і у повному обсязі опанував зміст навчального матеріалу, легко в ньому орієнтується і вміло використовує понятійний апарат; уміє пов'язувати теорію з практикою, вирішувати практичні завдання, впевнено висловлювати і обґрунтовувати свої судження. Відмінна оцінка передбачає логічний виклад відповіді мовою викладання (в усній або у письмовій формі), демонструє якісне оформлення роботи і володіння спеціальними приладами та інструментами, прикладними програмами. Здобувач не вагається при видозміні запитання, вміє робити детальні та узагальнюючі висновки, демонструє практичні навички з вирішення фахових завдань. При відповіді допустив дві–три несуттєві <i>помилки</i> .
Добре (середній)	Здобувач вищої освіти виявив повне засвоєння навчального матеріалу, володіє понятійним апаратом, орієнтується у вивченому матеріалі; свідомо використовує теоретичні знання для вирішення практичних задач; виклад відповіді грамотний, але у змісті і формі відповіді можуть мати місце окремі неточності, нечіткі формулювання правил, закономірностей тощо. Відповідь здобувача вищої освіти будується на основі самостійного мислення. Здобувач вищої освіти у відповіді допустив дві–три <i>несуттєві помилки</i> .
Задовільно (достатній)	Здобувач вищої освіти виявив знання основного програмного матеріалу в обсязі, необхідному для подальшого навчання та практичної діяльності за професією, справляється з виконанням практичних завдань, передбачених програмою. Як правило, відповідь здобувача вищої освіти будується на рівні репродуктивного мислення, здобувач вищої освіти має слабкі знання структури навчальної дисципліни, допускає неточності і <i>суттєві помилки</i> у відповіді, вагається при відповіді на видозмінене запитання. Разом з тим, набув навичок, необхідних для виконання нескладних практичних завдань, які відповідають мінімальним критеріям оцінювання і володіє знаннями, що дозволяють йому під керівництвом викладача усунути неточності у відповіді.
Незадовільно (недостатній)	Здобувач вищої освіти виявив розрізнені, безсистемні знання, не вміє виділяти головне і другорядне, допускається помилок у визначенні понять, перекидає їх зміст, хаотично і невпевнено викладає матеріал, не може використовувати знання при вирішенні практичних завдань. Як правило, оцінка «незадовільно» виставляється здобувачеві вищої освіти, який не може продовжити навчання без додаткової роботи з вивчення навчальної дисципліни.

Структурування дисципліни за видами навчальної роботи і оцінювання результатів навчання студентів денної форми здобуття освіти у семестрі

Аудиторна робота								Контрольні заходи		Самостійна робота		Семестровий контроль	Разом
Сьомий семестр													
Лабораторні роботи №:								Контрольні роботи №:		ІЗ №:		Іспит	Сума балів
1	2	3	4	5	6	7	8	1	2	1	2		
Кількість балів за вид навчальної роботи (мінімум-максимум)													
3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	3-5	24-40	60-100
24-40								6-10		6-10		24-40	

Примітки: *За набрану з будь-якого виду навчальної роботи з дисципліни кількість балів, нижче встановленого мінімуму, здобувач отримує незадовільну оцінку і має її перездати у встановлений викладачем (деканом) термін. Інституційна оцінка встановлюється відповідно до таблиці «Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС».

Оцінювання результатів захисту лабораторної роботи

Виконана й оформлена відповідно до встановлених Методичними рекомендаціями вимог лабораторна робота комплексно оцінюється викладачем при її захисті з урахуванням таких критеріїв: самостійність та правильність виконання; повнота відповіді та знання.

Результат виконання і захисту здобувачем вищої освіти кожної лабораторної роботи оцінюється відповідно до таблиці Критеріїв оцінювання навчальних досягнень здобувача вищої освіти.

У випадку виявлення здобувачем рівня знань, нижчого ніж 60 відсотків від максимального балу, встановленого Робочою програмою для кожної структурної одиниці, лабораторна робота йому *не зараховується* і для її захисту він має детальніше опрацювати матеріал з теми роботи, методику її виконання, виправити грубі помилки та повторно вийти на її захист у призначений для цього викладачем час.

Оцінювання контрольної роботи

Контрольна робота передбачає виконання п'яти завдань. При оцінюванні контрольної роботи враховуються: повнота відповіді та якість виконання. Кожне завдання оцінюється 1 балом, загальна сума балів на позитивну оцінку становить від 3 до 5.

Розподіл балів при оцінюванні завдань контрольної роботи

Кількість правильних відповідей	1	2	3	4	5
Відсоток правильних відповідей	0-59		60	80	100
Кількість отриманих балів	0		3	4	5

При отриманні негативної оцінки контрольну роботу слід перездати до терміну *наступного* контролю.

Оцінювання результатів виконання індивідуального завдання

Виконане та оформлене відповідно до вимог, визначених Методичними рекомендаціями, індивідуальне завдання (ІЗ) комплексно оцінюється викладачем з урахуванням таких критеріїв: самостійність виконання; правильність розв'язання поставлених задач; обґрунтованість вибору методів розв'язання; повнота пояснень та аргументованість відповідей; якість оформлення та дотримання вимог до структури і змісту роботи.

Результат виконання здобувачем вищої освіти кожного ІЗ оцінюється відповідно до таблиці **Критеріїв оцінювання навчальних досягнень здобувача вищої освіти** з урахуванням рівня досягнення запланованих програмних результатів навчання та сформованих компетентностей. За підсумками захисту присвоюється відповідна сума балів (мінімальний позитивний бал – 3 бали, максимальний – 5 балів).

У разі, якщо здобувач вищої освіти виявив рівень знань і виконання ІЗ, що нижчий ніж 60 відсотків від максимальної кількості балів, встановленої Робочою програмою для цієї структурної одиниці, завдання не зараховується. У такому випадку студент має повторно опрацювати зміст завдання, усунути помилки та здати на перевірку доопрацьоване ІЗ у терміни, погоджені з викладачем.

Оцінювання результатів підсумкового семестрового контролю (іспит)

Освітня програма передбачає підсумковий семестровий контроль з дисципліни у формі іспиту, завданням якого є системне й об'єктивне оцінювання як теоретичної, так і практичної підготовки здобувача з навчальної дисципліни. Складання іспиту відбувається за попередньо розробленими і затвердженими на засіданні кафедри білетами. Відповідно до цього в екзаменаційному білеті пропонується поєднання питань як теоретичного (в т.ч. у тестовій формі), так і практичного характеру.

Таблиця – Оцінювання результатів підсумкового семестрового контролю здобувачів денної форми навчання (40 балів для підсумкового контролю)

Види завдань	Для кожного окремого виду завдань		
	Мінімальний (достатній) бал (задовільно)	Потенційні позитивні бали* (середній бал) (добре)	Максимальний (високий) бал (відмінно)
Завдання № 1	6	8	10
Завдання № 2	6	8	10
Завдання № 3	6	8	10
Завдання № 4	6	8	10
Разом:	24		40

Примітка. *Позитивний бал за іспит, відмінний від мінімального (24 бали) та максимального (40 балів), знаходиться в межах 25-39 балів та розраховується як сума балів за усі структурні елементи (завдання) іспиту.

Для кожного окремого виду завдань підсумкового семестрового контролю застосовуються критерії оцінювання навчальних досягнень здобувача вищої освіти, наведені вище (**Таблиця – Критерії оцінювання навчальних досягнень здобувача вищої освіти**).

Підсумкова семестрова оцінка за інституційною шкалою і шкалою ЄКТС визначається в автоматизованому режимі після внесення викладачем результатів оцінювання у балах з усіх видів навчальної роботи до електронного журналу. Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС наведені нижче у таблиці «Співвідношення».

Семестровий іспит виставляється, якщо загальна сума балів, яку набрав студент з дисципліни за результатами поточного контролю, знаходиться у межах від 60 до 100 балів. При цьому за інституційною шкалою ставиться оцінка «відмінно/добре/задовільно», а за шкалою ЄКТС – буквене позначення оцінки, що відповідає набраній студентом сумі балів відповідно до таблиці Співвідношення.

Таблиця – Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Рейтингова шкала балів	Інституційна оцінка (рівень досягнення здобувачем вищої освіти запланованих результатів навчання з навчальної дисципліни)	
		Залік	Іспит/диференційований залік
A	90-100	Зараховано	<i>Відмінно/Excellent</i> – високий рівень досягнення запланованих результатів навчання з навчальної дисципліни, що свідчить про безумовну готовність здобувача до подальшого навчання та/або професійної діяльності за фахом
B	83-89		<i>Добре/Good</i> – середній (максимально достатній) рівень досягнення запланованих результатів навчання з навчальної дисципліни та готовності до подальшого навчання та/або професійної діяльності за фахом
C	73-82		
D	66-72		<i>Задовільно/Satisfactory</i> – Наявні мінімально достатні для подальшого

E	60-65		навчання та/або професійної діяльності за фахом результати навчання з навчальної дисципліни
FX	40-59	Незараховано	Незадовільно/Fail – Низка запланованих результатів навчання з навчальної дисципліни відсутня. Рівень набутих результатів навчання є недостатнім для подальшого навчання та/або професійної діяльності за фахом
F	0-39		Незадовільно/Fail – Результати навчання відсутні

10. Питання для самоконтролю результатів навчання

1. Загальні поняття об'єктно-орієнтованого програмування: абстракція даних, наслідування, інкапсуляція, поліморфізм.
2. Каркас середовища розробки Framework .Net. Поняття простору імен.
3. Поняття: об'єкт, клас, метод. Синтаксис визначення класу. Типи доступу.
4. Створення об'єкту класу. Конструктор, деструктор, статичні поля.
5. Метод, процедура. Статичний метод. Вказівник this.
6. Концепція наслідування. Відношення між класами. Класи без потомків.
7. Наслідування. Базовий, похідний клас. Інтерфейс.
8. Реалізація наслідування в C#. Віртуальні методи. Модифікатор override.
9. Конструктори батьків та нащадків. Ключове слово base.
10. Перевантаження методу.
11. Укривання методу.
12. Поняття абстрактного класу.
13. Діаграми класів.
14. Діаграми варіантів використання.
15. Діаграми послідовності.
16. Діаграми активності.
17. Діаграми станів.
18. Розвернуті та посилочні типи.
19. Поняття структури.
20. Перерахування.
21. Інтерфейси. Способи реалізації.
22. Обгортання та кастинг.
23. Перетворення до класу інтерфейсу.
24. Проблеми множинного наслідування. Колізія імен.
25. Наслідування від загального предка.
26. Вбудовані інтерфейси. Клонування.
27. Поняття делегата.
28. Операції над делегатами. Клас Delegate.
29. Конструкція try-catch.
30. Загальні поняття подій.
31. Засоби доступу до подій. Add, remove.
32. Опрацювання подій в середовищі .NET Framework.
33. SOLID принципи.
34. Предметно-орієнтоване проектування.
35. Code Smells.
36. Refactoring.
37. Anti-patterns.
38. Патерни проектування GRASP.

11. Навчально-методичне забезпечення

Освітній процес з дисципліни «Об'єктно-орієнтоване проектування» повністю і в достатній кількості забезпечений необхідною навчально-методичною літературою. Зокрема, викладачами кафедри підготовлені і видані такі роботи:

1. Молчанова М. О., Собко О. В., Бармак О. В., Мазурець О. В., Скрипник Т. К. Об'єктно-орієнтоване проектування. Методичні рекомендації до лабораторних робіт. ХНУ, 2023. 97 с. URL: http://lib.khnu.km.ua/EL_LIBRARY/vidavn/metod/mtd2023_1e/91%D0%B523/index.pdf.

12. Матеріально-технічне та програмне забезпечення дисципліни (за потреби)

Інформаційна та комп'ютерна підтримка: ПК, планшет, смартфон або інший мобільний пристрій, проектор. Програмне забезпечення: програми Microsoft Office або аналогічні, доступ до мережі Інтернет, робота з презентаціями.

Вивчення навчальної дисципліни потребує використання спеціального програмного прикладного забезпечення, крім загальнонавчаних програм і операційних систем - інтегроване середовище розробки MS Visual Studio. Усі зазначені програмні продукти є безкоштовними або надаються студентам за академічними ліцензіями.

13. Рекомендована література:

Основна

2. Об'єктно-орієнтоване програмування [Електронний ресурс] : електрон. метод. посіб. до лекц. та лаб. занять для студентів ф-ту математики, фізики та інформ. технологій першого (бакалавр.) рівня освіти спец. 122 Комп'ютерні науки / уклад.: А. Л. Рачинська, О. А. Недева, К. С. Палій. – Одеса : Одес. нац. ун-т ім. І. І. Мечникова, 2024. – 338 с. URL: <https://dspace.onu.edu.ua/bitstreams/68babdee-36dd-42e9-aeaa-ecb0a23abd2b/download>

3. Технології розроблення програмного забезпечення. Лабораторний практикум [Електронний ресурс] : навч. посіб. / КПІ ім. Ігоря Сікорського; уклад.: О. А. Амонс, М. Ю. Мякий. – Електронні текстові дані (1 файл: 2,02 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2025. – 113 с. URL: <https://ela.kpi.ua/server/api/core/bitstreams/504c3de0-158e-4d25-8dbf-35752dbcdc9f/content>

4. Сорокати Р.В., Пасічник О.А., Скрипник Т.К. Основи об'єктно-орієнтованого програмування. Навчальний посібник. Хмельницький, ХНУ. 2019. 175с. URL: http://lib.khmnu.edu.ua/EL_LIBRARY/vidavn/metod/mtd2019_1/7/index.pdf

Додаткова

5. Основи об'єктно-орієнтованого програмування : навч. посібник / Гришанович Т. О., Глинчук Л. Я.; ВНУ імені Лесі Українки. Луцьк : ВНУ імені Лесі Українки, 2022. – 120 с. URL: <https://evnuir.vnu.edu.ua/bitstream/123456789/20320/1/oop.pdf>

6. C# language documentation. URL: <https://learn.microsoft.com/en-us/dotnet/csharp>

7. Visual Studio documentation. URL: <https://learn.microsoft.com/en-gb/visualstudio/windows/?view=vs-2022>

14. Інформаційні ресурси

1. Модульне середовище для навчання. URL : <https://msn.khmnu.edu.ua>

2. Електронна бібліотека університету. URL: <http://library.khmnu.edu.ua/>

3. Репозитарій ХНУ. URL : <https://elar.khmnu.edu.ua/home>

ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОЄКТУВАННЯ

Тип дисципліни	Обов'язкова
Рівень вищої освіти	Перший (бакалаврський)
Мова викладання	Українська
Семестр	Другий
Кількість призначених кредитів ЄКТС	6,0
Форми здобуття освіти, для яких викладається дисципліна	Очна (денна)

Результати навчання. Студент, який успішно завершив вивчення дисципліни, повинен: вміло використовувати понятійний апарат та основоположні поняття об'єктно-орієнтованого проєктування: абстракція даних, спадкування, інкапсуляція, поліморфізм; володіти принципами побудови об'єктно-орієнтованої структури програм; знати поняття об'єктно-орієнтованого проєктування: клас, об'єкт, метод, властивість; вміти писати програми мовою C# застосовуючи GUI, вміти реалізовувати принципи та алгоритми об'єктно-орієнтованого проєктування, SOLID-принципи при написанні програмних продуктів. 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. Проєктувати програмне забезпечення з використанням методів та засобів візуального проєктування (діаграми Use Case, Statechart, Activity, Sequence). Застосовувати знання об'єктно-орієнтованої методології проєктування при розробці і дослідженні функціональних моделей організаційно-економічних і виробничо-технічних систем.

Зміст навчальної дисципліни. Парадигми програмування. Об'єктно-орієнтований підхід до програмування. Принципи об'єктно-орієнтованого підходу до програмування. Прикладне застосування об'єктно-орієнтованого підходу. Абстрактні класи. Структури та перерахування. Конструктори класів. Об'єктно-орієнтовані концепції. Делегати та події. Візуальні засоби об'єктно-орієнтованого проєктування. Діаграми варіантів використання (Use case diagram). Діаграма станів (Statechart diagram). Діаграми активності (Activity diagram). Діаграми послідовності (Sequence diagram). Використання віртуальних методів. SOLID-принципи. Патерни проєктування GRASP. Засоби аналізу та забезпечення якості проєктування програмного забезпечення. Антипатерни. Введення в бази даних. Сучасні підходи до проєктування програмного забезпечення. Предметно-орієнтоване проєктування (DDD). Застосування DDD для розробки програмних моделей предметних середовищ.

Пререквізити – Алгоритмізація та програмування (ОЗП.05)

Кореквізити – Теорія алгоритмів (ОЗП.08), Методи та системи штучного інтелекту (ОФП.01), Методи та системи штучного інтелекту (курсний проєкт) (ОФП.02), Проєктування баз даних (ОФП.07), Технології створення програмних продуктів (ОФП.09), Основи програмної інженерії та тестування програмного забезпечення (ОФП.10), Управління ІТ-проєктами (ОФП.15), Проєктно-технологічна практика (ОФП.16), Професійна практика (ОФП.17), Кваліфікаційна робота (ОФП.18)..

Запланована навчальна діяльність: Мінімальний обсяг навчальних занять в одному кредиті ЄКТС навчальної дисципліни для *першого* (бакалаврського) рівня вищої освіти за денною формою здобуття освіти становить 10 годин на 1 кредит ЄКТС.

Форми (методи) навчання: Процес навчання з дисципліни ґрунтується на використанні традиційних та сучасних технологій та методів навчання, зокрема: лекції (з використанням методів візуалізації, проблемного й інтерактивного навчання, мотиваційних прийомів, інформаційно-комунікаційних технологій); лабораторні заняття (з використанням інструктування, демонстрування, розв'язування типових і прикладних задач, аналізу кейсів, ситуаційних завдань, елементів дискусії тощо); самостійна робота (опрацювання теоретичного матеріалу, підготовка до виконання лабораторних робіт, поточного та підсумкового контролю, виконання індивідуальних завдань), з використанням інформаційно-комп'ютерних технологій та технологій дистанційного навчання.

Форми оцінювання результатів навчання: оцінювання лабораторних та контрольних робіт; виконання індивідуального завдання; підсумковий контрольний захід.

Вид семестрового контролю: іспит – 2 семестр.

Навчальні ресурси:

1. Молчанова М. О., Собко О. В., Бармак О. В., Мазурець О. В., Скрипник Т. К. Об'єктно-орієнтоване проєктування. Методичні рекомендації до лабораторних робіт. ХНУ, 2023. 97 с. URL: http://lib.khnu.km.ua/EL_LIBRARY/vidavn/metod/mtd2023_1e/91%D0%B523/index.pdf.

2. Об'єктно-орієнтоване програмування [Електронний ресурс] : електрон. метод. посіб. до лекц. та лаб. занять для студентів ф-ту математики, фізики та інформ. технологій першого (бакалавр.) рівня освіти спец. 122 Комп'ютерні

науки / уклад.: А. Л. Рачинська, О. А. Недєва, К. С. Палій. – Одеса : Одес. нац. ун-т ім. І. І. Мечникова, 2024. – 338 с. URL: <https://dspace.onu.edu.ua/bitstreams/68babdee-36dd-42e9-aeaa-ecb0a23abd2b/download>

3. Технології розроблення програмного забезпечення. Лабораторний практикум [Електронний ресурс] : навч. посіб. / КПІ ім. Ігоря Сікорського; уклад.: О. А. Амонс, М. Ю. Мякий. – Електронні текстові дані (1 файл: 2,02 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2025. – 113 с. URL: <https://ela.kpi.ua/server/api/core/bitstreams/504c3de0-158e-4d25-8dbf-35752dbcdc9f/content>

4. Сорокати Р.В., Пасічник О.А., Скрипник Т.К. Основи об'єктно-орієнтованого програмування. Навчальний посібник. Хмельницький, ХНУ. 2019. 175с. URL: http://lib.khmnu.edu.ua/EL_LIBRARY/vidavn/metod/mtd2019_1/7/index.pdf

5. Основи об'єктно-орієнтованого програмування : навч. посібник / Гришанович Т. О., Глинчук Л. Я.; ВНУ імені Лесі Українки. Луцьк : ВНУ імені Лесі Українки, 2022. – 120 с. URL: <https://evnuir.vnu.edu.ua/bitstream/123456789/20320/1/oop.pdf>

6. C# language documentation. URL: <https://learn.microsoft.com/en-us/dotnet/csharp>

7. Visual Studio documentation. URL: <https://learn.microsoft.com/en-gb/visualstudio/windows/?view=vs-2022>

8. Модульне середовище для навчання. Доступ до ресурсу: <https://msn.khmnu.edu.ua>

9. Електронна бібліотека університету. Доступ до ресурсу: <http://library.khmnu.edu.ua/>

Викладачі: д-р філософії, ст.викл., Молчанова М.О.
к.т.н., доцент, Мазурець О.В.