



Факультет математики та інформатики  
Кафедра диференціальних рівнянь і прикладної математики

**СИЛАБУС НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**  
**ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ**

***Рівень вищої освіти:*** Перший (бакалаврський)

***Освітні програми:*** Комп'ютерне моделювання та  
технології програмування

***Спеціальність:*** F1 Прикладна математика

***Галузь знань:*** F Інформаційні технології

Затверджено на засіданні кафедри  
Протокол № 1 від 26 серпня 2025 р.

## ЗМІСТ

1. [Загальна інформація](#)
2. [Анотація навчальної дисципліни](#)
3. [Мета та цілі навчальної дисципліни](#)
4. [Програмні компетентності](#)
5. [Програмні результати навчання](#)
6. [Структура навчальної дисципліни](#)
7. [Система оцінювання навчальної дисципліни](#)
8. [Політика навчальної дисципліни](#)
9. [Ресурсне забезпечення навчальної дисципліни](#)

## 1. ЗАГАЛЬНА ІНФОРМАЦІЯ

<b>Назва дисципліни</b>	<b>Об'єктно орієнтоване програмування</b> Object-Oriented Programming
<b>Освітні програми</b>	Комп'ютерне моделювання та технології програмування
<b>Спеціальність</b>	F1 Прикладна математика
<b>Галузь знань</b>	F Інформаційні технології
<b>Освітній рівень</b>	перший (бакалаврський)
<b>Статус дисципліни</b>	<b>нормативна</b> (з циклу професійної підготовки)
<b>Рік навчання/Семестр</b>	2 <sup>ий</sup> / 3 <sup>ій</sup>
<b>Обсяг дисципліни</b>	3 кредити ECTS / 90 год.
<b>Розподіл годин за видами занять</b>	лекції – 14 год лабораторні – 30 год самостійна робота – 46 год
<b>Форма контролю</b>	<b>екзамен</b>
<b>Мова викладання</b>	українська
<b>Посилання на сайт дистанційного навчання</b>	<a href="https://d-learn.pnu.edu.ua/oop">https://d-learn.pnu.edu.ua/oop</a> <a href="https://classroom.google.com/oop">https://classroom.google.com/oop</a>
<b>Пререквізити</b>	OK08 Програмування на Python OK13 Програмування на C/C++
<b>Постреквізити</b>	OK24 Курсова робота з програмування OK38 Практикум з моделювання і програмування

<b>Кафедра</b>	диференціальних рівнянь і прикладної математики ауд. 315 ЦК, <a href="https://kdrpm.cnu.edu.ua">https://kdrpm.cnu.edu.ua</a>
<b>Викладач(-и)</b>	<b>Віктор МАЗУРЕНКО</b>
<b>Контактний телефон</b>	(0342)596027
<b>E-mail</b>	<a href="mailto:viktor.mazurenko@cnu.edu.ua">viktor.mazurenko@cnu.edu.ua</a>
<b>Профайл</b>	<a href="https://mazurenko.cnu.edu.ua">https://mazurenko.cnu.edu.ua</a>
<b>Консультації</b>	згідно з розкладом консультацій на сайті кафедри

## 2. АНОТАЦІЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

У цьому курсі здобувачі вищої освіти за спеціальністю «Прикладна математика» мають нагоду заглибитись у світ об'єктно-орієнтованого програмування (де будь-яку програму розглядають як сукупність «об'єктів», які взаємодіють між собою і кожен з яких є екземпляром якогось «класу»), освоїти його ключові концепції та практики. Така парадигма програмування є хоч і не надто простою, але вельми ефективною при розробці складних програмних проєктів. Завдяки таким ключовим концепціям як «абстрагування», «інкапсуляція», «успадкування» і «поліморфізм», програми, написані в об'єктно-орієнтованому стилі, володіють більшою модульністю і гнучкістю, читабельністю і оптимізованістю коду, захищеністю даних і адаптованістю до змін, ніж програми, написані в інших стилях. Розуміння концепцій об'єктно-орієнтованого програмування, його переваг і недоліків разом з практичними навиками написання програм мовою Python/C++ в рамках цієї парадигми дозволить студентів бути успішним в області сучасних технологій програмування.

## 3. МЕТА І ЦІЛІ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Сформувати у здобувачів вищої освіти фахових компетентностей з об'єктно-орієнтованого програмування для створення програмного забезпечення, яке є більш розширюваним, підтримуваним і зрозумілим як для розробників, так і користувачів.

Ознайомити студентів з основними поняттями і концепціями об'єктно-орієнтованого програмування. Сформувати вміння писати програми мовою Python в об'єктно-орієнтованому стилі для автоматизації задач різної складності. Сприяти розвитку навиків командної роботи та спільної програмної розробки.

## 4. ПРОГРАМНІ КОМПЕТЕНТНОСТІ

Відповідно до освітньо-професійної програми «Комп'ютерне моделювання та технології програмування» для першого (бакалаврського) рівня вищої освіти:

**ІК.** Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми прикладної математики у професійній діяльності або у процесі навчання, що передбачає застосування математичних теорій та методів і характеризується комплексністю та невизначеністю умов.

**ФК04.** Здатність розробляти алгоритми та структури даних, програмні засоби та програмну документацію.

**ФК05.** Здатність проектувати бази даних, інформаційні системи та ресурси.

**ФК08.** Здатність використовувати сучасні технології програмування та тестування програмного забезпечення.

## 5. ПРОГРАМНІ РЕЗУЛЬТАТИ НАВЧАННЯ

**РН11.** Вміти застосовувати сучасні технології програмування та розроблення програмного забезпечення, програмної реалізації чисельних і символічних алгоритмів.

## 6. СТРУКТУРА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

ТЕМА, ПЛАН	КІЛЬКІСТЬ ГОДИН		
	ЛЕКЦІЇ	ЛАБОРАТОРНІ	САМОСТІЙНА РОБОТА
<b>ФУНКЦІЇ</b> - поняття підпрограми і функції, визначення і виклик функцій, повернення кількох значень - аргументи (параметри) функції: формальні і фактичні, передача аргументів: позиційні та іменовані (ключові) аргументи; типові (за замовчуванням) значення аргументів - функції з довільною кількістю аргументів, області видимості: локальні і глобальні змінні - поняття про функційний тип даних, анонімні (lambda-функції) - рекурсія і рекурсивні функції, генератор-функції - декоратори для функцій	<b>2</b>	<b>4</b>	<b>4</b>
<b>КЛАСИ І ОБ'ЄКТИ, АТРИБУТИ І МЕТОДИ</b> - основні парадигми програмування та об'єктно-орієнтований стиль програмування - класи і об'єкти (екземпляри) класу, атрибути і методи класу/об'єкта, абстрагування і абстракція - створення класу, посилання self на об'єкт класу, конструктор __init__ (ініціалізація) і деструктор __del__ (знищення) об'єкта - створення і знищення об'єктів - звернення до атрибутів і методів об'єкта, атрибути об'єкта і атрибути класу, додавання і вилучення атрибутів і методів - методи і функції	<b>2</b>	<b>4</b>	<b>4</b>
<b>КОНЦЕПЦІЯ ІНКАПСУЛЯЦІЇ</b> - інкапсуляція - статичні атрибути і методи - вступ до UML: діаграма класів, видимість атрибутів і методів, відношення між класами	<b>2</b>	<b>4</b>	<b>4</b>
<b>КОНЦЕПЦІЯ УСПАДКУВАННЯ</b> - успадкування (наслідування) - перевизначення атрибутів і методів - особливості роботи з методом-конструктором, виклик методів базового класу, суперклас object - успадкування на діаграмах класів - множинне і багатократне успадкування - проблеми, пов'язані з множинним успадкуванням - виклик методів базових класів	<b>2</b>	<b>4</b>	<b>4</b>

<b>КОНЦЕПЦІЯ ПОЛІМОРФІЗМУ</b> - поліморфізм - віртуальні методи	<b>2</b>	<b>4</b>	<b>4</b>
<b>СПЕЦІАЛЬНІ (МАГІЧНІ) АТРИБУТИ І МЕТОДИ</b> - спеціальні атрибути - спеціальні методи - спеціальні методи для колекцій - перевантаження операторів (арифметичні, присвоєння, порівняння, унарні) - ітератори і генератори	<b>2</b>	<b>4</b>	<b>4</b>
<b>КЛАСИ ВИКЛЮЧЕНЬ І АБСТРАКТНІ КЛАСИ</b> - класи виключень - абстрактні класи - інтерфейси та шаблони програмування - домішки	<b>2</b>	<b>4</b>	<b>4</b>
<b>МОДУЛЬНИЙ КОНТРОЛЬ</b>	–	<b>2</b>	<b>6</b>
<b>ПІДСУМКОВИЙ КОНТРОЛЬ</b>	–	–	<b>12</b>
<b>ВСЬОГО</b>	<b>14</b>	<b>30</b>	<b>46</b>

## 7. СИСТЕМА ОЦІНЮВАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

<b>Загальна система оцінювання</b>	Підсумкова оцінка з навчальної дисципліни є сумою оцінок з таких видів навчальних робіт: лабораторні роботи (виконання і захист), модульний контроль (тест), самостійна робота (за окремими темами) та підсумковий контроль (екзамен). Підсумкова оцінка визначається відповідно до поданої нижче таблиці оцінювання за різними шкалами (100-бальна, ECTS, національна).
<b>Лабораторні роботи</b>	Максимальна оцінка за вчасно виконані і захищені лабораторні роботи (сім з яких є індивідуальними, а одна – командною) становить 30 балів.
<b>Модульний контроль</b>	Тест з об'єктно-орієнтованого програмування містить 15 завдань закритого/відкритого типу на знання і розуміння базових концепцій (абстрагування, інкапсуляція, успадкування, поліморфізм) об'єктно-орієнтованого програмування, на вміння створювати класи і об'єкти, працювати з атрибутами і методами, на тестування програмного коду і обробку помилок. Максимальна оцінка за тест становить 15 балів.
<b>Самостійна робота</b>	Максимальна оцінка за активну і змістовну аудиторну роботу та самостійну роботу за окремими темами становить 5 балів.
<b>Умови допуску до підсумкового контролю</b>	Загальна кількість балів за всі види робіт впродовж семестру становить не менше 25 балів.

<b>Підсумковий контроль</b>	Екзамен складається з 2-х частин: тест на знання теоретичних основ об'єктно-орієнтованого програмування та написання програми в об'єктно-орієнтованому стилі для вирішення практичної задачі. Максимальна оцінка за екзамен становить 50 балів.
-----------------------------	---

Накопичування балів впродовж вивчення дисципліни

Види навчальної роботи	Максимальна кількість балів
Лабораторні роботи	30
Модульний контроль	15
Самостійна робота	5
Підсумковий контроль	50
<b>Підсумкова оцінка</b>	<b>100</b>

### ШКАЛА ОЦІНЮВАННЯ: НАЦІОНАЛЬНА ТА ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проєкту (роботи), практики	для заліку
90 – 100	<b>A</b>	відмінно	зараховано
80 – 89	<b>B</b>	добре	
70 – 79	<b>C</b>		
60 – 69	<b>D</b>	задовільно	
50 – 59	<b>E</b>		
26 – 49	<b>FX</b>	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-25	<b>F</b>	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

### КРИТЕРІЇ ОЦІНЮВАННЯ НАВЧАЛЬНИХ ДОСЯГНЕНЬ

(відповідно до [Положення](#) про організацію освітнього процесу)

**«відмінно»** – здобувач освіти міцно засвоїв теоретичний матеріал, глибоко і всебічно знає зміст навчальної дисципліни, основні положення рекомендованої літератури, логічно мислить і буде відповідь, вільно використовує набуті теоретичні знання при аналізі практичного матеріалу, висловлює своє ставлення до тих чи інших проблем, демонструє високий рівень засвоєння практичних навичок;

**«добре»** – здобувач освіти добре засвоїв теоретичний матеріал, володіє основними аспектами рекомендованої літератури, аргументовано викладає його; має практичні навички, висловлює свої міркування з приводу тих чи інших проблем, але припускається певних неточностей і похибок у логіці викладу теоретичного матеріалу або при аналізі практичного матеріалу;

**«задовільно»** – здобувач освіти в основному опанував теоретичними знаннями навчальної дисципліни, орієнтується в першоджерелах та рекомендованій літературі, але непереконливо відповідає, плутає поняття, додаткові питання викликають невпевненість або відсутність стабільних знань; відповідаючи на запитання практичного характеру, виявляє неточності у знаннях, не вміє оцінювати факти та явища, пов'язувати їх із майбутньою діяльністю;

**«незадовільно»** – здобувач освіти не опанував навчальний матеріал дисципліни, не знає наукових фактів, визначень, майже не орієнтується в рекомендованій літературі, відсутнє наукове мислення, практичні навички не сформовані.

## 8. ПОЛІТИКА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

<b>Академічна доброчесність</b>	Важливим є самостійне виконання навчальних завдань та посилення на джерела інформації у разі використання ідей, розробок, тверджень, відомостей. Плагіат та інші види академічної недоброчесності не принесуть користі, тому є недоречними та контролюються відповідно до <a href="#">Положення</a>
<b>Пропуски занять (відпрацювання)</b>	Наслідком періодичних пропусків є самостійне опрацювання навчального матеріалу з можливим консультуванням у викладача відповідно до графіка консультацій. Наслідком систематичних пропусків є додаткові види самостійної роботи, які контролюються на передбачених у графіку навчального процесу тижнях контролю самостійної роботи
<b>Виконання завдань пізніше встановленого терміну</b>	Приводить до втрати частини балів, запланованих у системі оцінюванні навчальної дисципліни (усі види навчальної роботи важливо виконувати належним чином і вчасно, щоб зберігати загальний темп курсу, котрий сприяє ефективному засвоєнню матеріалу без шкоди здоров'ю
<b>Додаткові бали</b>	До 5 балів за активність, комунікативність, креативність, наполегливість, самостійність при вивченні дисципліни
<b>Неформальна освіта</b>	Можливість повного або часткового зарахування результатів неформальної освіти відповідно до <a href="#">Положення</a> . Рекомендовані платформи: <a href="https://coursera.org">coursera.org</a> , <a href="https://prometheus.org.ua">prometheus.org.ua</a> , <a href="https://udemy.com">udemy.com</a> , <a href="https://netacad.com">netacad.com</a> , <a href="https://edx.org">edx.org</a>

## 9. РЕСУРСНЕ ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

<b>Матеріально-технічне забезпечення</b>	Лекційна аудиторія, мультимедіа, комп'ютерна лабораторія з програмним забезпеченням
<b>Навчально-методичне забезпечення</b>	Навчальний контент на освітніх платформах <a href="https://d-learn.pnu.edu.ua/PwP">https://d-learn.pnu.edu.ua/PwP</a> <a href="https://classroom.google.com/PwP">https://classroom.google.com/PwP</a>
<b>Інформаційне забезпечення</b>	<a href="#">Наукова бібліотека КНУВС</a> , Рекомендована література

### РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Будаї А. Дизайн-патерни - просто як двері. – Ел. вид., 2012. – Режим доступу: [url](#)
2. Васильєв О.М. Програмування мовою Python. – Тернопіль: Навчальна книга – Богдан, 2019. – Режим доступу: [url](#)
3. Висоцька В.А., Оборська О.В. Python: алгоритмізація та програмування: навчальний посібник. – Львів: Новий Світ – 2000, 2021. – Режим доступу: [url](#)

4. Кренивч А.П. Python у прикладах і задачах. Частина 2. Об'єктно-орієнтоване програмування: Навчальний посібник. – К.: ВПЦ "Київський університет", 2020. – Режим доступу: [url](#)
5. Мізюк О. Путівник мовою програмування Python. – Ел. вид., 2020. – Режим доступу: [url](#)
6. Швець О. Занурення в патерни проектування. – Ел. вид., 2021. – Режим доступу: [url](#)
7. Gamma E., Helm R., Johnson R., Vlissides J. Design Patterns. Elements of Reusable Object-Oriented Software. - Addison-Wesley Professional, 1994.
8. PEP 8 – Style Guide for Python Code. – Access mode: [url](#)
9. The Python Tutorial. - Access mode: [url](#)

Викладач В.Мазуренко