

Міністерство освіти і науки України
Національний аерокосмічний університет
«Харківський авіаційний інститут»

Кафедра Інформаційних технологій проєктування (№ 105)

ЗАТВЕРДЖУЮ

Гарант освітньої програми


(підпис)

Олександр КАРАТАНОВ
(ім'я та ПРІЗВИЩЕ)

« 29 » ____ 08 ____ 2025 р.

**СИЛАБУС ОBOB'ЯЗKОВОЇ
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

Об'єктно-орієнтоване програмування

(шифр і назва навчальної дисципліни)

Галузь знань: 12 «Інформаційні технології»
(шифр і найменування галузі знань)

Спеціальність: 122 «Комп'ютерні науки»
(код і найменування спеціальності)

Освітня програма: «Інформаційні технології проєктування»
(найменування освітньої програми)

Рівень вищої освіти: *перший (бакалаврський)*

Силабус введено в дію з 01.09.2025

Харків – 2025 р.

Розробник: доцент кафедри 105 Інформаційних технологій проектування,
(посада, науковий ступінь і вчене звання, ім'я та прізвище)

к.т.н., доцент Віктор ОВСЯННИК


_____ (підпис)

Силабус навчальної дисципліни розглянуто на засіданні кафедри № 105
Інформаційних технологій проектування

(назва кафедри)

Протокол № 1 від 28.08.2025 року

В.о. завідувача кафедри 105


_____ (підпис)

Аліна АРТЬОМОВА
(ім'я та прізвище)

1 Загальна інформація про викладача



ПІБ: Овсяннік Віктор Миколайович

Посада: доцент кафедри кафедри 105

Науковий ступінь: кандидат технічних наук

Вчене звання: доцент

Перелік дисциплін, які викладає:

*Об'єктно-орієнтоване програмування,
Інструментальні засоби візуального
програмування*

Напрями наукових досліджень:

*Основи методології побудови
єдиного інформаційного простору*

Контактна інформація: Електронна пошта:

v.ovsiannik@[khai.edu](mailto:v.ovsiannik@khai.edu)

Телеграм: @victor2166

2 Опис навчальної дисципліни

Форма здобуття освіти	<i>Денна</i>
Семестр	3
Мова викладання	Українська
Тип дисципліни	<i>Обов'язкова</i>
Обсяг дисципліни: кредити ЄКТС/ кількість годин	<i>денна: 5 кредитів ЄКТС / 150 годин (96 аудиторних, з яких: лекції – 48, лабораторні роботи – 48; СРЗ – 54)</i>
Види навчальної діяльності	Лекції, практичні або лабораторні заняття, самостійна робота
Види контролю	Поточний контроль, модульний контроль, семестровий контроль – іспит
Пререквізити	<i>Технологія розробки програм</i>

3 Мета та завдання навчальної дисципліни, переліки компетентностей та очікуваних результатів навчання

Мета – надати здобувачам освіти знання та практичні навички з розробки програмного забезпечення для роботи під керівництвом ОС Windows з широким використанням можливостей концепції об'єктно-орієнтованого програмування (ООП)

Завдання дисципліни:

- ознайомити з основними принципами технології об'єктно-орієнтованого програмування;
- навчити методам проектування, розробки та відлагодження застосунків з використанням середовища Microsoft Visual Studio;
- сформуванати навички використання мови UML;
- розкрити засади забезпечення ефективності, наочності та надійності застосунків Windows;
- ознайомити з сучасними тенденціями розвитку технологій та концепцій програмування

Компетентності, які набуваються:

Інтегральна компетентність: 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.

Загальні компетентності (ЗК)

Після вивчення цієї дисципліни здобувач освіти опанує:

ЗК1. Здатність до абстрактного мислення, аналізу та синтезу

ЗК2. Здатність застосовувати знання у практичних ситуаціях.

ЗК3. Знання та розуміння предметної області та розуміння професійної діяльності.

ЗК6. Здатність вчитися й оволодівати сучасними знаннями.

Спеціальні компетентності (СК)

Після засвоєння цієї дисципліни здобувач освіти опанує:

СК3. 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.

СК8. Здатність проектувати та розробляти програмне забезпечення із застосуванням різних парадигм програмування: узагальненого, об'єктноорієнтованого, функціонального, логічного, з відповідними моделями, методами й алгоритмами обчислень, структурами даних і механізмами управління.

Програмні результати навчання (ПР):

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.

4 Програма навчальної дисципліни

МОДУЛЬ 1

Змістовний модуль 1. Об'єктно-орієнтований аналіз та технологія ООП

Тема 1. Поняття об'єктно-орієнтованого аналізу

Анотація: Структура курсу. Поняття структур, сутності, атрибуту відношення, класу предметної області, простору проблем та простору рішень. Визначення цих понять та їх використання. Об'єкт як абстракція реального світу і понятійна структура, Поведінка об'єкта, його властивості, зв'язки між об'єктами, різних типів Особливості реалізації концепції процедурного програмування як «передвісника» технології ООП

Самостійна робота здобувача освіти: опрацювання матеріалів лекцій та допоміжних методичних посібників.

Тема 2. Об'єктна модель предметної області і принципи її побудови

Анотація: Опис предметного середовища та постановка задачі Принципи побудови програмної моделі, їх використання та приклади застосування

Теми лекцій: «Опис структур, класів, об'єднань».

Самостійна робота здобувача освіти: опрацювання матеріалів лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.

Тема 3. Поняття об'єктів, класів та їх взаємовідносин

Анотація: Поняття класів і об'єктів, формат їх опису та складові частини

Теми лекцій: «Формати опису класів, приклади створення та визначення, використання заголовкових файлів та файлів реалізації».

Самостійна робота здобувача освіти: опрацювання матеріалів лекцій.

Тема 4. Принцип інкапсуляції та регулювання доступу до компонентів класів

Анотація: Розглядаються специфікатори доступу, сфери видимості та правила їх використання для забезпечення приховання даних і контролю доступу до компонентів класів, а також використання псевдо-змінної `this` і визначення методів класу.

Теми лекцій: «специфікатори доступу до компонентів класів, сфери видимості та правила їх використання».

Теми лабораторних/практичних занять: «опис та визначення структур, об'єднань та об'єктів і покажчиків на об'єкти». Псевдо-змінна `this`. Визначення методів класу.

Самостійна робота здобувача освіти: опрацювати матеріали лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.

Змістовний модуль 2. Технологія об'єктно-орієнтованого програмування

Тема 5. Основи об'єктно-орієнтованого проектування мовою UML.

Анотація: Ознайомлення з UML як мовою моделювання та проектування ПЗ, включно зі структурними та поведінковими діаграмами, реалізацією класів і перетворенням діаграм у код C++.

Теми лекцій: Поняття та призначення UML як мови, що призначена для аналізу, візуального моделювання та проектування програмного забезпечення. Реалізація в UML таких загальних функцій як візуалізація, специфікація, конструювання та документування елементів програмних систем. Типи моделей UML та їх використання, приклади реалізації. Структурні діаграми, діаграми поведінки та взаємодії. Позначення в термінах нотації Г.Буча. Переведення діаграм класів в C++.

Тема 6. Основи об'єктно-орієнтованої мови програмування.

Анотація: Вивчається ООП як метод програмування на основі взаємодії об'єктів і класів, розглядаються три ключові концепції: інкапсуляція, успадкування та поліморфізм.

Теми лекцій: Три основні концепції ООП: інкапсуляція, успадкування та поліморфізм.

Теми лабораторних/практичних занять: Розробка класу у відповідності з індивідуальним завданням. Розробка класу вектор, що надає можливості зручної роботи з масивами: додавання та видалення елементів по одному або по декілька. Наслідування класів. Абстрактні класи та віртуальні функції

Тема 7. Абстрагування даних та інкапсуляція.

Анотація: Висвітлюється структура класу, приховання даних та реалізація аксесорів, що дозволяють забезпечити контрольований доступ до внутрішніх компонентів об'єктів.

Теми лекцій: Клас як сфера видимості член-даних та член-функцій скриття член-даних та аксесори

Теми лабораторних/практичних занять: Розробка класу контейнера.

Тема 8. Розмежування доступу до компонентів класів і їх ініціалізація

Анотація: Розглядаються конструктори і деструктори класів, їх виклики, особливості розміщення об'єктів у динамічній пам'яті та використання покажчика `this`.

Теми лекцій: Призначення і формат опису конструкторів і деструкторів. Особливості виклику конструкторів і деструкторів класів. Розміщення екземплярів класів у «купі(heap)». Огляд `CRect` як прикладу класу, що має декілька конструкторів, а також інші компоненти. Використання покажчика `this`.

Тема 9. Варіанти розміщення екземплярів класу у пам'яті

Анотація: Аналізуються способи розміщення об'єктів у глобальній пам'яті, стеку та динамічній пам'яті, а також вплив цього розташування на роботу конструкторів і деструкторів.

Теми лекцій: «Розміщення екземплярів класу у глобальній пам'яті, стеку програми та у динамічній пам'яті». особливості викликів конструкторів та деструкторів в залежності від місця розташування об'єктів класу

Самостійна робота здобувача освіти: опрацювання матеріалів лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.

Модульний контроль 1.

МОДУЛЬ 2. Успадкування класів і поліморфізм

Змістовний модуль 3. Успадкування класів

Тема 10. Визначення похідного класу

Анотація: Вивчається формат опису похідного класу, специфікатори доступу до компонентів батьківського класу та вплив ключа `class` на успадкування.

Теми лекцій: Сутність успадкування класів формат опису та визначення похідного класу специфікатори доступу до компонентів батьківського класу Особливості доступу похідного класу до компонентів батьківського класу Вплив ключа класу на успадкування

Тема 11. Сумісність об'єктів ієрархії класів

Анотація: Розглядається сумісність об'єктів у межах ієрархії, передача об'єктів як параметрів функцій та типи параметрів.

Теми лекцій: *Поняття сумісності об'єктів Передача об'єктів як параметрів глобальних функцій. Тип параметрів глобальних функцій*

Тема 12. Видимість компонентів класів ієрархії

Анотація: *Досліджується вплив специфікаторів доступу на доступність компонентів батьківського класу у похідному класі.*

Теми лекцій: *Вплив специфікаторів доступу на доступність компонентів батьківського класу у похідному*

Самостійна робота здобувача освіти: *опрацювати матеріали лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.*

Тема13. Віртуальні функції

Анотація: *Описується призначення віртуальних функцій, раннє та пізнє зв'язування (early/late binding) та таблиці віртуальних функцій на прикладі графічних об'єктів.*

Теми лекцій: *Приклад ієрархії класів графічних фігур Поняття та призначення віртуальних функцій формат опису віртуальних функцій, раннє(early binding) та пізнє (late binding) зв'язування функцій Таблиця віртуальних функцій*

Самостійна робота здобувача освіти: *опрацювати матеріали лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.*

Тема 14. Віртуальні функції конструктори і деструктори

Анотація: *Розглядаються конструктори і деструктори у класах з віртуальними функціями, їх опис та виклик у ієрархії класів.*

Теми лекцій: *Конструктори і деструктори в ієрархії класів. Специфіка опису та виклику конструкторів в ієрархії класів з віртуальними функціями*

Самостійна робота здобувача освіти: *Опрацювати матеріали лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.*

Тема 15. Абстрактні класи

Анотація: *Вивчається формат опису абстрактних класів, роль абстрактного класу як базового та застосування віртуальних функцій.*

Теми лекцій: *абстрактні віртуальні функції поняття та формат опису абстрактного класу роль абстрактного класу як базового класу ієрархії з віртуальними функціями*

Самостійна робота здобувача освіти: *опрацювання матеріалів лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.*

Тема 16. Контейнерні класи

Анотація: *Розглядається реалізація контейнерних класів із внутрішніми класами та покажчиками на них, а також приклади практичного використання.*

Теми лекцій: *Контейнер як зовнішній клас. Контейнер із внутрішнім класом член-данім. Контейнер з покажчиком на внутрішній клас.*

Теми лабораторних/практичних занять: *Розробка контейнерного класу.*

Самостійна робота здобувача освіти: *опрацювання матеріалів лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.*

Тема 17. Дружні функції й класи

Анотація: *Висвітлюється призначення дружніх функцій та класів, а також правила їх опису й визначення для забезпечення спеціального доступу до закритих компонентів.*

Теми лекцій: *Призначення дружніх функцій і дружніх класів; формати опису та визначення дружніх функцій і дружніх класів.*

Самостійна робота здобувача освіти: *опрацювання матеріалів лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.*

Тема 18. Статичні компоненти класів.

Анотація: Розглядаються статичні атрибути і методи класів, формат їх опису та особливості ініціалізації для забезпечення спільного доступу між об'єктами.

Теми лекцій: Призначення статичних компонентів класів, Формат опису та особливості ініціалізації статичних атрибутів класів

Самостійна робота здобувача освіти: опрацювання матеріалів лекцій, виконання індивідуального завдання, підготовка до захисту лабораторних робіт.

Модульний контроль 2.

5 Індивідуальні завдання

Індивідуальне завдання у якості розрахунково-графічної роботи за курсом передбачає розробку програми на мові С++ і оформлення пояснювальної записки, У пояснювальній записці студент формулює технічне завдання, виконує проектування програми, розробляє та описує алгоритми роботи програми. До складу пояснювальної записки повинні входити також керівництво користувача програми та керівництво програміста, перелік посилань.

6 Методи навчання

Лекції проводяться з використанням конспекту лекцій в електронній формі, елементів мультимедійної підтримки курсу (презентації), демонстрацій окремих прийомів роботи з інструментальним середовищем та створенням програмних застосунків.

Лабораторні роботи виконуються з використанням навчальних (демонстраційних) та ліцензованих робочих версій функціональних середовищ Microsoft Visual Studio або QT СУБД.

Самостійна робота включає підготовку до лабораторних/практичних робіт, модульного контролю та іспиту, виконання розрахунково-графічної роботи з використанням навчально-методичної літератури та документації до програмного забезпечення і вивчення вказаних вище тем за конспектом, літературними джерелами та документацією з мови програмування та інтегрованих середовищ розробки програмного забезпечення (IDE)

7 Методи контролю

Контроль здійснюється згідно з «Положення про рейтингове оцінювання досягнень студентів».

Поточний контроль здійснюється відповідно до повноти, своєчасності та якості виконання лабораторних робіт та домашнього завдання (розрахунково-графічної роботи). **Проміжні (модульні) контролю** – письмова контрольна робота на 8-му 14-му тижнях. **Підсумковий контроль** – письмовий іспит та усні відповіді на запитання викладача за курсом.

8 Критерії оцінювання та розподіл балів, які отримують здобувачі освіти

Таблиця 8.1 – Розподіл балів, які отримують здобувачі освіти

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Модульні контрольні роботи	0–20	2	0–40
Лабораторні/практичні роботи	0–5	8	0–40
Розрахунково-графічна робота	0–20	1	0–20
Усього за семестр			0–100

Семестровий контроль (іспит/залік) проводиться у разі відмови студента від балів поточного тестування та за наявності допуску до іспиту/заліку. При складанні семестрового іспиту/заліку студент має можливість отримати максимум 100 балів.

Білет для іспиту/заліку складається з 5 запитань: 3 теоретичних та 2 практичних, по 20 балів за кожне.

Таблиця 8.2 – Критерії оцінювання роботи студента протягом семестру

Сума балів з дисципліни	Оцінка за проміжної атестації	Характеристика рівня освоєння дисципліни
від 90 до 100	«зараховано» / «відмінно»	Здобувач володіє знаннями, уміннями й навичками, що дають змогу самостійно та обґрунтовано виконувати проектування та розробку класів, призначених для вирішення різноманітних задач. Виконує всі лабораторні/практичні та модульні роботи на високому рівні. Уміє пропонувати комплексні рішення для підвищення продуктивності та ефективності застосунків Windows.
від 75 до 89	«зараховано» / «добре»	Здобувач має ґрунтовні знання, уміння й навички для забезпечення програмних результатів навчання. Виконує та здає лабораторні/практичні роботи, та модульні завдання. Додатково до вимог оцінки «задовільно»: вміє самостійно обирати відповідні програмні засоби для вирішення простих задач, проводити аналіз структури вхідних даних програми, та обирати оптимальні алгоритми їх обробки.
від 60 до 74	«зараховано» / «задовільно»	Здобувач має базові знання та уміння для забезпечення програмних результатів навчання. Виконує та здає практичні роботи і лабораторні завдання. Знає основні принципи побудови та функціонування баз даних, послідовність проектування класів та їх ієрархій.
від 41 до 59	«не зараховано» / «незадовільно»	Студент демонструє сформованість дисциплінарних компетенцій на рівні нижче базового, проявляється недостатність знань, умінь, навичок.
від 0 до 40	«не зараховано» / «незадовільно»	Дисциплінарні компетенції не формуються. Виявляється повне або практично повна відсутність знань, умінь, навичок.

Таблиця 8.3 – Шкали оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

9 Політика навчального курсу

Відвідування занять. Регуляція пропусків. Інтерактивний характер курсу передбачає обов'язкове відвідування практичних занять. Здобувачі освіти, які за певних обставин не можуть відвідувати практичні заняття регулярно, повинні протягом тижня узгодити із викладачем графік індивідуального відпрацювання пропущених занять. Окремі пропущені заняття мають бути відпрацьовані на найближчій консультації протягом тижня після їх пропуску. Відпрацювання занять здійснюється усно у формі співбесіди за питаннями, визначеними планом заняття. В окремих випадках дозволяється письмове відпрацювання пропущених занять шляхом виконання індивідуального письмового завдання.

Дотримання вимог академічної доброчесності здобувачами освіти під час вивчення навчальної дисципліни. Під час вивчення навчальної дисципліни здобувачі освіти мають дотримуватися загальноприйнятих морально-етичних норм і правил поведінки, вимог академічної доброчесності, передбачених Положенням про академічну доброчесність Національного аерокосмічного університету «Харківський авіаційний інститут» (<https://khai.edu/assets/files/polozhennya/polozhennya-pro-akademichnu-dobrochesnist.pdf>).

Очікується, що роботи здобувачів освіти будуть їх оригінальними дослідженнями або міркуваннями. Відсутність посилань на використані джерела, фабрикування джерел, списування, втручання в роботу інших здобувачів освіти становлять, але не обмежують, приклади можливої академічної недоброчесності. Виявлення ознак академічної недоброчесності в письмовій роботі здобувача освіти є підставою для її незарахування викладачем незалежно від масштабів плагіату чи обману.

Вирішення конфліктів. Порядок і процедури врегулювання конфліктів, пов'язаних із корупційними діями, зіткненням інтересів, різними формами дискримінації, сексуальними домаганнями, міжособистісними стосунками та іншими ситуаціями, що можуть виникнути під час навчання, а також правила етичної поведінки регламентуються Кодексом етичної поведінки в Національному аерокосмічному університеті «Харківський авіаційний інститут» (<https://khai.edu/ua/university/normativna-baza/ustanovchi-dokumenti/kodeks-etichnoi-povedinki/>).

10 Методичне забезпечення

Навчально-методичний комплекс дисципліни у електронному вигляді знаходиться на сайті дистанційної освіти ХАІ «Ментор»:

- силабус дисципліни;
- конспект лекцій, підручники (навчальні посібники), в тому числі в електронному вигляді, які за змістом повністю відповідають робочій програмі дисципліни;
- методичні вказівки та рекомендації для виконання курсових робіт та проектів, розрахункових та розрахунково-графічних робіт, лабораторних та практичних робіт, а також рекомендації для самостійної підготовки;
- тематики індивідуальних завдань;
- приклади розв'язування типових задач чи виконання типових завдань;
- питання, тести для контрольних заходів;
- каталоги інформаційних ресурсів.

11 Рекомендована література

11.1 Базова

1. Овсяннік, В. М. Мова С++ не для чайників [Електронний ресурс] : навч. посіб./ В. М. Овсяннік, О. К. Погудіна. – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2020. – 130 с.

2. Основи програмування [Текст] : навч. посіб. / О. К. Погудіна, В. М. Овсяннік, В. І. Калашнікова, А. В. Погудін. – Харків: Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. Ін-т», 2021. - 116 с.

3. Основи програмування [Електронний ресурс] : Методичні рекомендації до виконання лабораторних робіт О. К. Погудіна, В. М. Овсяннік, М. О. Бичок, А. В. Погудін – Харків : Нац. аерокосм. ун-т ім. М. Є. Жуковського «Харків. авіац. ін-т», 2021. – 73 с.

4. В.В. Бублик Об'єктно-орієнтоване програмування: [Підручник] / В.В. Бублик. – К.: ІТ-книга, 2015. – 624 с.: іл.

11.2 Допоміжна

1. Vjarne Stroustrup C++Programming Language Fourth Edition 2013. 1366с.

2. Веклич Р. А. Вступ до програмування мовою С++. Структури даних: навч. посіб. / Р. А. Веклич, Т. О. Карнаух, А. Б. Ставровський – К. : ВПЦ "Київський університет", 2018. – 99 с
3. ДСТУ 2938-94. Системи оброблення інформації. Основні поняття. Терміни та визначення. – К. : Держстандарт України, 1995. – 32 с.
4. Learning C++plusplus. Pdf844.pdf
5. Кравець П.О. Об'єктно-орієнтоване програмування: навч. посібник/ П.О. Кравець. – Львів: Видавництво Львівської політехніки, 2012. – 624с.

12 Інформаційні ресурси

1. C++/CLI Programing [Електронний ресурс]. — Режим доступу: <http://www.functionx.com/cppcli>