

Київський національний університет будівництва і архітектури Кафедра <u>інформаційних технологій</u>	Шифр спеціальності	Назва спеціальності, освітньої програми	Освітній рівень
	122	Комп'ютерні науки, Інформаційні управляючі системи і технології	Бакалавр

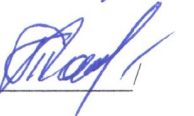
«Затверджую»

Завідувачка кафедри  
Тетяна ГОНЧАРЕНКО



Розробник силабусу

Тетяна ГОНЧАРЕНКО




## СИЛАБУС

### ОК10 «ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»

(назва, шифр освітньої компоненти (дисципліни))

<b>1) Статус освітньої компоненти:</b> обов'язкова	
<b>2) Контактні дані викладача:</b> д.т.н, доцент Гончаренко Т.А., <a href="mailto:goncharenko.ta@knuba.edu.ua">goncharenko.ta@knuba.edu.ua</a> , +380 (93) 022 82 01, <a href="https://www.knuba.edu.ua/elementor-161543/">https://www.knuba.edu.ua/elementor-161543/</a>	
<b>3) Пререквізити:</b> «Програмування та алгоритмічні мови»; «Офісні інформаційні технології»; «Інструментальні засоби програмування».	
<b>4) Коротка анотація дисципліни</b> Метою освітньої компоненти є формування результатів навчання, що охоплюють основні поняття, парадигми та принципи розробки програм із застосуванням об'єктно-орієнтованої методології. Здобувачі опановують знання, навички та вміння, які дозволяють автоматизувати бізнес-процеси будівництва, обробку інформації та керування об'єктами, створеними на основі стандартних і користувацьких класів. Компонента передбачає засвоєння сутності та особливостей об'єктно-орієнтованого підходу, а також набуття теоретичних знань і практичних навичок об'єктно-орієнтованого програмування на мові C#.	
<b>5) Структура курсу:</b> лекції, лабораторні роботи, самостійні роботи, розрахунково-графічна робота, курсова робота, залік, екзамен	
<b>Загальна кількість кредитів ECTS</b>	7,5
<b>Сума годин:</b>	225
<b>Вид індивідуального завдання</b>	РГР, КР
<b>Форма контролю</b>	Залік, Екзамен
<b>б) Зміст курсу:</b> <b>Змістовий модуль 1. Основні складові ООП</b> <i>Лекція 1.</i> Введення в ООП. Процедурні мови. Недоліки структурного програмування. <i>Лекція 2.</i> Об'єктно-орієнтований підхід. Основні концепції об'єктно-орієнтованого програмування. <i>Лекція 3.</i> Абстрагування даних та інкапсуляція через специфікатори доступу. <i>Лекція 4.</i> Оброблення виняткових ситуацій. Розробка графічних інтерфейсів користувача. <i>Лекція 5.</i> Конструктори класів. Використання конструкторів з аргументами за замовчуванням. Деструктори.	

## **Змістовний модуль 2. Розробка ООП Windows-Forms застосувань**

*Лекція 1.* Побудова UML-діаграм: прецедентів, класів, послідовностей, компонентів.

*Лекція 2.* Основні принципи перевантаження операцій. Функції-операції як елементи класу і як дружні функції.

*Лекція 3.* Успадкування. Доступ до членів класу.

*Лекція 4.* Організація захищеного доступу та приховування імен.

*Лекція 5.* Поліморфізм. Нові класи і динамічне зв'язування. Віртуальні методи.

*Лекція 6.* Абстрактні класи і методи. Перевизначення методів.

*Лекція 7.* Робота з файлами. Потокowe введення-виведення даних. Стандартні класи.

## **Змістовний модуль 3. Додаткові можливості ООП у C#**

*Лекція 1.* Структури, делегати, події, лямбда-вирази.

*Лекція 2.* Рефлексія. Серіалізація, десеріалізація.

*Лекція 3.* Примітивні контейнери для зберігання посилань на об'єкти: масиви, колекції, хеш-таблиці.

*Лекція 4.* Параметризовані класи.

*Лекція 5.* Архітектура ієрархій об'єктів і класів.

## **Змістовний модуль 4. Класи середовищ розробника програм**

*Лекція 1.* Поняття шаблону в C# (template). Мета, переваги та недоліки, приклади.

*Лекція 2.* Популярні шаблони проектування ООП.

*Лекція 3.* Розробка веб-додатків як одна з сфер застосування C#.Net.

*Лекція 4.* Підхід MVC, платформа ASP.NET MVC.

*Лекція 5.* Стандартні бібліотеки класів середовищ розробника програм.

## **Теми лабораторних занять**

### **Змістовний модуль 1. Основні складові ООП.**

*Лабораторна робота №1.* Оброблення виняткових ситуацій.

*Лабораторна робота №2.* Розробка графічних інтерфейсів користувача на основі інструментів Toolbox.

*Лабораторна робота №3.* Абстрагування даних та інкапсуляція через специфікатори доступу.

*Лабораторна робота №4.* Конструктори класів. Використання конструкторів з аргументами за замовчуванням. Деструктори.

### **Змістовний модуль 2. Розробка ООП Windows-Forms застосувань**

*Лабораторна робота №5.* Основні принципи перевантаження операцій.

*Лабораторна робота №6.* Успадкування. Доступ до членів класу

*Лабораторна робота №7.* Організація захищеного доступу та приховування імен

*Лабораторна робота №8.* Поліморфізм. Віртуальні функції. Абстрактні класи та інтерфейси

### **Змістовний модуль 3. Додаткові можливості ООП у C#**

*Лабораторна робота №1.* Структури, делегати, події, лямбда-вирази

*Лабораторна робота №2.* Рефлексія. Серіалізація, десеріалізація

*Лабораторна робота №3.* Примітивні контейнери для зберігання посилань на об'єкти: масиви, колекції, хеш-таблиці

*Лабораторна робота №4.* Параметризовані класи.

### **Змістовний модуль 4. Класи середовищ розробника програм**

*Лабораторна робота №5.* Побудова UML-діаграм: прецедентів, класів, послідовностей, компонентів

*Лабораторна робота №6.* Розробка веб-додатків на основі застосування C#.Net.

*Лабораторна робота №7.* Розробка веб-додатків за принципом MVC

## **Лабораторна робота №8.** Використання Postman для тестування CRUD-операцій

### **Самостійна робота:**

#### **Змістовий модуль 1.**

1. Сучасні інструментальні засоби візуального програмування
2. Абстрагування та інкапсуляція через специфікатори доступу
3. Оброблення виняткових ситуацій.
4. Члени класу – конструктори, властивості, індексатори, методи

#### **Змістовий модуль 2.**

5. Основні принципи перевантаження операцій.
6. Успадкування. Доступ до членів класу
7. Організація захищеного доступу та приховування імен
8. Абстрактні класи та інтерфейси
9. Виконання РГР за варіантом
10. Підготовка до заліку

#### **Змістовий модуль 3.**

11. Структури, делегати, події, лямбда-вирази.
12. Рефлексія. Серіалізація, десеріалізація.
13. Контейнери для зберігання посилань на об'єкти
14. Архітектура ієрархій об'єктів і класів

#### **Змістовий модуль 4.**

15. Побудова UML-діаграм: прецедентів, класів, послідовностей, компонентів
16. Робота з JSON. Серіалізація в JSON. JsonSerializer
17. Розробка веб-додатків за принципом MVC
18. Використання Postman для тестування CRUD-операцій
19. Виконання та захист курсової роботи
20. Підготовка до екзамену

**Індивідуальна робота:** Виконання РГР згідно індивідуального варіанту для закріплення теоретичних знань і практичних навичок з ООП у 3 семестрі.

Курсова робота - закріплення теоретичних знань і практичних навичок з технології об'єктно-орієнтованого програмування та поглиблене оволодіння сучасними можливостями програмування на С# з використанням стандартних і користувацьких класів, інструментів успадкування та поліморфізму у 4 семестрі.

**7) Посилання на сторінку електронного навчально-методичного комплексу дисципліни:**  
<https://org2.knuba.edu.ua/course/view.php?id=285>