



## ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

### Робоча програма навчальної дисципліни (Силабус)

#### Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>122 Комп'ютерні науки</i>
Освітня програма	<i>Комп'ютерний моніторинг та геометричне моделювання процесів і систем</i>
Статус дисципліни	<i>Нормативна</i>
Форма навчання	<i>Очна (денна)</i>
Рік підготовки, семестр	<i>2 курс, осінній семестр</i>
Обсяг дисципліни	<i>135 годин / 4,5 кредити ЄКТС (лекції – 36 год., практичні заняття – 18 год., СРС – 81 год.)</i>
Семестровий контроль/ контрольні заходи	<i>Екзамен, МКР</i>
Розклад занять	<i><a href="http://rozklad.kpi.ua/">http://rozklad.kpi.ua/</a></i>
Мова викладання	<i>Українська</i>
Інформація про керівника курсу / викладачів	<i>Лектор: доцент, к.ф.-м.н., доц. Карпенко Станіслав Григорович, E-mail: <a href="mailto:stangmp@gmail.com">stangmp@gmail.com</a> Лабораторні заняття: доцент, к.ф.-м.н., доц. Карпенко Станіслав Григорович, E-mail: <a href="mailto:stangmp@gmail.com">stangmp@gmail.com</a></i>
Розміщення курсу	<i><a href="https://campus.kpi.ua">https://campus.kpi.ua</a></i>

#### Програма навчальної дисципліни

### 1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

**Метою навчальної дисципліни** «Об'єктно-орієнтоване програмування» є навчання студентів теоретичним основам та практичним навичкам використання парадигми об'єктно-орієнтоване програмування, апаратного та програмного захисту.

Завданнями вивчення навчальної дисципліни є:

- оволодіння та ефективне використання основних засад об'єктно-орієнтоване програмування – інкапсуляцією, поліморфізмом та успадкуванням;
- вивчення та застосування характерних особливостей статичного та динамічного поліморфізму;
- набуття практичних навичок побудови та використання систем віртуальних функцій та віртуальних базових класів, шаблонів функцій, шаблонів класів.

Згідно з вимогами освітньо-професійної програми студенти після засвоєння навчальної дисципліни мають продемонструвати такі результати навчання:

**знання:** новітніх технологій в сфері об'єктно-орієнтованого програмування; принципів створення та використання програмного забезпечення на мові C++, сучасних середовищ розробки програмного забезпечення,

**вміння:** володіння принципами мови та бібліотеками об'єктно-орієнтованої мови C++; володіння практичними навичками роботи з сучасними середовищами розробки.

**досвід:**

- використання особливостей парадигми об'єктно-орієнтоване програмування – інкапсуляція, поліморфізм, успадкування;
- застосування засобів побудови синтаксичних конструкцій для об'єктів створених класів;
- ефективного використання віртуальних функцій, віртуальних базових класів;
- практичного застосування принципів створення та реалізації шаблонів функцій та шаблонів класів.

## 2. Пререквізити та постреквізити дисципліни

Вивчення дисципліни спирається на знання, отримані за програмою першого року навчання за спеціальністю 122 «Комп'ютерні науки».

У структурно-логічній схемі навчання зазначений кредитний модуль розміщується тоді, коли студенти вже прослухали такі дисципліни, як "Основи програмування" та "Алгоритми та структури даних", і є логічним продовженням навчального процесу набуття знань та навичок сучасних систем програмування.

З іншого боку, викладений матеріал може бути використаний при вивченні дисциплін "Комп'ютерна графіка", "Основи розробки трансляторів", "Основи крос-платформного програмування", які подаються в наступних семестрах.

## 3. Зміст навчальної дисципліни

В дисципліні вивчаються такі теми.

### Розділ 1. Базові поняття мови ООП

- Тема 1.1. Парадигми програмування. Структури.
- Тема 1.2. Класи і об'єкти. Конструктори та деструктори.
- Тема 1.3. Перевантаження функцій. Перевантаження операцій.
- Тема 1.4. Шаблони функцій.
- Тема 1.5. Успадкування. Ієрархія класів. Абстрактні класи.

### Розділ 2. Властивості класів

- Тема 2.1. Множинне успадкування.
- Тема 2.2. Потоківі класи. Функції керування потоками.
- Тема 2.3. Шаблони класів.

### Розділ 3. Файлові потоки. Винятки. Стандартні шаблони

- Тема 3.1. Класи файлових потоків.
- Тема 3.2. Обробка виняткових ситуацій.
- Тема 3.3. Контейнерні класи.

## 4. Навчальні матеріали та ресурси

### Базова література

1. Аспектно-орієнтоване програмування. URL: [https://uk.wikipedia.org/wiki/Аспектно-орієнтоване\\_програмування](https://uk.wikipedia.org/wiki/Аспектно-орієнтоване_програмування).
2. Дейтел Х.М., Дейтел П.Дж. Как программировать на Си++. М.: ЗАО БИНОМ, 1999, 1000 с.
3. Карпенко С. Г. Практикум з програмування. Мова С: Початковий курс: Навч. посіб.-/ С. Г. Карпенко, Шпортюк Г. А. К. – К.: МАУП, 2006. – 152с. – Бібліогр.: с. 149.
4. Компонентне програмування. URL: [https://uk.wikipedia.org/wiki/Компонентне\\_програмування](https://uk.wikipedia.org/wiki/Компонентне_програмування).
5. Лаптев В. В. С++. Объектно-ориентированное программирование: Учебное пособие. — СПб.: Питер, 2008. —464 с.: ил.
6. Лафоре Р. Объектно-ориентированное программирование в С++. Классика Computer Science. 4 изд.. – СПб.: Питер, 2003.
7. Липпман С., Лажойе Ж. Язык программирования С++. Полное руководство, 3-е изд./Пер. с англ. – СПб.: «Невский диалект», М.: ДМК Пресс. – 1104 с.
8. Модульне програмування. URL: [https://uk.wikipedia.org/wiki/Модульне\\_програмування](https://uk.wikipedia.org/wiki/Модульне_програмування).
9. Об'єктно-орієнтоване програмування. URL: [https://uk.wikipedia.org/wiki/Об'єктно-орієнтоване\\_програмування](https://uk.wikipedia.org/wiki/Об'єктно-орієнтоване_програмування).

10. Павловская Т.А., Щупак Ю.А. С++. Структурное программирование: Практикум. – СПб.: Питер, 2003 -240 с.
11. Павловская Т.А. С/С++ Программирование на языке высокого уровня. – СПб.: Питер, 2007 -461 с.
12. Подійно-орієнтоване програмування. URL: [https://uk.wikipedia.org/wiki/Подійно-орієнтоване\\_програмування](https://uk.wikipedia.org/wiki/Подійно-орієнтоване_програмування).
13. Прата, Стивен. Язык программирования С++. Лекции и упражнения, 5-е изд. : Пер. с англ. – М.: ООО "И.Д. Вильямс", 2007. – 1184 с.
14. Процедурне програмування. URL: [https://uk.wikipedia.org/wiki/Процедурне\\_програмування](https://uk.wikipedia.org/wiki/Процедурне_програмування).
15. Шилдт Г. Полный справочник по С++, 4-е издание.: Пер. с англ. – М.: Издательский дом "Вильямс", 2006. – 800 с.

## Додаткова література

1. Буч, Гради, Максимчук, Роберт А., Энгл, Майкл У, Янг, Бобби Дж., Коналлен, Джим, Хьюстон, Келли А. Объектно-ориентированный анализ и проектирование с примерами приложений, 3-е изд.: Пер. с англ. - М.: ООО "И.Д. Вильямс", 2008. - 720 с.:
2. Пахомов Б.И. С/С++ и MS Visual С++ 2012 для начинающих. – СПб.: БХВ-Петербург, 2013. – 512с.: ил.
3. Подбельский В.В. Язык Си++: Учеб. пособие. – 5-е изд. – М.: Финансы и статистика, 2003. – 560 с.: ил.
4. Войтенко В.В., Морозов А.В. С/С++ : Теория та практика. Навч.-мет. посібник. – Житомир: ЖДТУ, 2004. – 324 с.
5. Лаптев В.В. С++. Экспресс-курс. – СПб.: БХВ-Петербург, 2004. – 512 с.: ил.

## Навчальний контент

### 5. Методика опанування навчальної дисципліни (освітнього компонента)

#### Лекційні заняття

№ з/п	Назва теми лекції та перелік основних питань (перелік дидактичних засобів, посилання на літературу та завдання на СРС)	К-сть ауд.год.
<b>Розділ 1 Базові поняття мови ООП</b>		
Тема 1.1. Парадигми програмування. Структури.		
1.	<i>ЛЕКЦІЯ 1.</i> Мови програмування. Поняття про парадигми програмування. Структури, поля структур, ініціалізація структур. Оголошення структур.	2
Тема 1.2. Класи і об'єкти. Конструктори та деструктори.		
2.	<i>ЛЕКЦІЯ 2.</i> Класи. Опис класу. Властивості полів класу. Опис об'єктів. Вказівник this. Конструктори. Типи конструкторів. Деструктори. Особливості створення та ініціалізації об'єктів з динамічними полями.	2
Тема 1.3. Перевантаження функцій. Перевантаження операцій.		
3.	<i>ЛЕКЦІЯ 3.</i> Перевантаження функцій. Перевантаження операцій. Застосування дружніх функцій.	2
Тема 1.4. Шаблони функцій.		
4.	<i>ЛЕКЦІЯ 4.</i> Створення шаблонів функцій. Особливості використання шаблонів.	2
Тема 1.5. Успадкування. Ієрархія класів. Абстрактні класи.		
5.	<i>ЛЕКЦІЯ 5.</i> Принцип успадкування. Ключі доступу. Просте успадкування. Ієрархія класів. Конструктори в похідних класах.	2
6.	<i>ЛЕКЦІЯ 6.</i> Доступ до об'єктів в ієрархії класів. Вказівники на об'єкти похідних і базових класів. Статичне і динамічне зв'язування.	2
<b>Розділ 2. Властивості класів</b>		
Тема 2.1. Множинне успадкування.		
7.	<i>ЛЕКЦІЯ 7.</i> Множинне успадкування. Конструктори і деструктори в ієрархії класів. Віртуальні базові класи.	2
Тема 2.2. Поточкові класи. Функції керування потоками.		
8.	<i>ЛЕКЦІЯ 8.</i> Поточкові класи. Класи та об'єкти стандартних потоків. Форматоване та неформатоване введення/виведення.	2
9.	<i>ЛЕКЦІЯ 9.</i> Функції керування процесами введення/виведення. Використання маніпуляторів.	2

10.	ЛЕКЦІЯ 10. Прапорці керування потоками. Обробка помилок потоків.	2
Тема 2.3. Шаблони класів.		
11.	ЛЕКЦІЯ 11. Шаблони класів, синтаксис визначення шаблонів класів. побудова та використання шаблонів класів. Визначення об'єктів шаблонів класів.	2
12.	ЛЕКЦІЯ 12. Зберігання в шаблонах об'єктів класів. Створення шаблонів класів зв'язних списків.	2
<b>Розділ 3. Файлові потоки. Винятки. Стандартні шаблони.</b>		
Тема 3.1. Класи файлових потоків.		
13.	ЛЕКЦІЯ 13. Класи файлових потоків. Режими відкриття файлових потоків.	2
14.	ЛЕКЦІЯ 14. Довільний доступ до файлів. Вказівники зчитування та запису. Аналіз помилок файлового введення/виведення. Використання функції typeid().	2
Тема 3.2. Обробка виняткових ситуацій.		
15.	ЛЕКЦІЯ 15. Винятки. Синтаксис винятків. Перехоплення винятків.	2
16.	ЛЕКЦІЯ 16. Ієрархія винятків. Специфікації винятків.	2
Тема 3.3. Контейнерні класи.		
17.	ЛЕКЦІЯ 17. Бібліотека стандартних шаблонів. Елементи бібліотеки стандартних шаблонів. Ітератори. Вектори.	2
18.	ЛЕКЦІЯ 18. Елементи бібліотеки стандартних шаблонів. Списки. Асоціативні контейнери. Рядки символів, клас string.	2
	<b>Всього годин</b>	<b>36</b>

### Лабораторні роботи

N	Назва лабораторних робіт	К-сть ауд.год.
1.	Створення на базі структур програми з динамічними масивами та випадковими числами	4
2.	Побудова класів для конкретної предметної області з конструкторами, деструктором та з певним полем класу у вигляді динамічного масиву	4
3.	Створення ієрархії класів з конструкторами, деструктором та перевантаженими операціями останнього похідного класу	4
4.	Побудова перевантаженого оператора індексування та перевантаженого оператора виведення полів класу на екран у вигляді таблиці	2
5.	Використання віртуальних методів, створення шаблону функції для підрахунку статистичних характеристик	2
6.	Запис та зчитування об'єктів похідного класу в текстовому та двійковому режимах	2
	<b>Всього годин</b>	<b>18</b>

### 6. Самостійна робота

№ з/п	Назва теми, що виноситься на самостійне опрацювання	К-сть годин СРС
<b>Розділ 1. Базові поняття мови ООП</b>		
Парадигми програмування. Структури. Класи і об'єкти. Конструктори та деструктори. Перевантаження функцій. Перевантаження операцій. Дружні функції. Шаблони функцій. Успадкування.		
1.	Відмінності між мовами С та С++. Присвоєння та ініціалізація структур. Вкладені структури.	3
2.	Структури як параметри функцій. Структури та класи.	6
3.	Перевантаження операцій присвоювання, приведення типів. Перевантаження оператора індексування.	8

4.	Приклади використання шаблонів. Явна спеціалізація шаблонів.	6
5.	Приклад альтернативної ієрархії класів.	4
6.	Захищене успадкування.	4
<b>Розділ 2. Властивості класів</b>		
Множинне успадкування. Потокові класи. Функції керування потоками. Шаблони класів.		
7.	Невизначеність при множинному успадкуванні.	6
8.	Функції-елементи <code>get</code> та <code>getline</code> .	4
9.	Функції-елементи класу <code>istream</code> ( <code>peek</code> , <code>putback</code> , <code>ignore</code> ).	4
10.	Перевантаження операцій виведення та введення для типів, визначених користувачем	4
11.	Конкретизація шаблонів, аргументи шаблону класу для параметрів-констант.	4
12.	Діаграми UML і шаблони.	4
<b>Розділ 3. Файлові потоки. Винятки. Стандартні шаблони.</b>		
Класи файлових потоків. Обробка виняткових ситуацій. Бібліотека стандартних шаблонів.		
13.	Ієрархія класів файлових потоків.	4
14.	Функції керування вказівниками зчитування та запису, призначення функції <code>typeid()</code> .	4
15.	Конструктори та функціональні <code>try</code> -блоки.	4
16.	Ієрархія класів винятків в стандартній бібліотеці C++.	4
17.	Розподільники пам'яті. Предикати.	4
18.	Алгоритми. Функції класу <code>string</code> .	4
	<b>Всього годин</b>	<b>81</b>

## Політика та контроль

### 7. Політика навчальної дисципліни (освітнього компоненту)

- Відвідування лекцій, а також відсутність на них, не оцінюється. Відвідування лабораторних занять є обов'язковою складовою вивчення матеріалу;
- При захисті лабораторних робіт студент має продемонструвати розроблений програмний код та результати його виконання на тестах, як заздалегідь підготованих, так і запропонованих викладачем. У випадку дистанційної форми навчання захист відбувається на відповідній конференції шляхом демонстрації екрана.
- Політика та принципи академічної доброчесності визначені у розділі 3 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.
- Норми етичної поведінки Норми етичної поведінки студентів і працівників визначені у розділі 2 Кодексу честі Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Детальніше: <https://kpi.ua/code>.

### 8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

- Рейтинг студента з освітнього компонента розраховується зі 100 балів, з них 50 балів складає стартова

шкала. Стартовий рейтинг (протягом семестру) складається з балів, що студент отримує за виконання лабораторних робіт (6 робіт) та 2 МКР;

- Критерії нарахування балів за виконання лабораторних робіт. Ваговий бал за виконання завдань лабораторних робіт складає 5 балів. Максимальна кількість балів за всі завдання дорівнює  
 $5 \text{ балів} \times 6 = 30 \text{ балів}$ .

Виконання кожного завдання оцінюється за наступними критеріями:

1. правильність виконання завдань лабораторної роботи – від 1 до 4 балів;
2. правильність відповідей на додаткові питання стосовно лабораторної роботи – від 1 до 3 балів.

Мінімальна кількість для зарахування лабораторної роботи складає 3 бали ( 60%).

Максимальна кількість балів за контрольну роботу дорівнює 5 балів. На модульну контрольну роботу вноситься два теоретичних питання та одне практичне.

Контрольна робота оцінюється наступним чином:

1. правильність написання кожного теоретичного питання - 1 бали;
2. надання прикладу на вказані завдання – 0.3 бали;
3. правильність розв'язання практичного завдання - 2 бали;
4. правильність написання програмного коду – 0.7 балів.

Умови допуску до екзамену: зарахування всіх лабораторних робіт. Мінімальна кількість набраних балів – 30 ( 60%).

На екзамені студенти виконують письмову контрольну роботу. Екзаменаційний білет складається з двох теоретичних питань та одного практичного завдання. Ваговий бал кожного теоретичного питання – 15.

Ваговий бал практичного завдання – 20.

Максимальна кількість балів за складання екзамену дорівнює

$$15 \text{ балів} \times 2 + 20 \text{ балів} = 50 \text{ балів}$$

Теоретична частина оцінюється наступним чином:

1. «відмінно» , правильна чітко викладена, повна відповідь – (не менше 90% потрібної інформації) – 14-15 балів;
2. «добре», достатньо повна відповідь (не менше 75% потрібної інформації) – 11-13 балів;
3. «задовільно», неповна відповідь (не менше 60% потрібної інформації) – 9-10 балів;
4. «незадовільно», незадовільна відповідь - 0 балів.

Практичне завдання оцінюється наступним чином:

- «відмінно» , повне, безпомилкове розв'язування завдання– 18-20 балів;
- «добре», повне, розв'язування завдання із несуттєвими неточностями – 15-17 балів;
- «задовільно», завдання виконане з певними недоліками – 12-14 балів;
- «незадовільно» завдання невиконано.

Сума стартових балів і балів за екзаменаційну контрольну роботу переводиться за освітній компонент згідно з таблицею.

Бали: практичні роботи + екзаменаційна контрольна робота	Оцінка
100 – 95	Відмінно
94 – 85	Дуже добре
84 – 75	Добре
74 – 65	Задовільно
64 – 60	Достатньо
< 60	Незадовільно
Наявність не зарахованих лабораторних робіт	Не допущено

**Робочу програму навчальної дисципліни (силабус):**

**Складено** доцентом, к.ф.-м.н., доц. Карпенком Станіславом Григоровичем

**Ухвалено** кафедрою АПЕПС (протокол № 16 від 18.06.21)

**Погоджено** Методичною комісією ТЕФ КПІ ім. Ігоря Сікорського (протокол № 11 від 24.06.21)