

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА
ШЕВЧЕНКА**

Факультет радіофізики, електроніки та комп'ютерних систем

Кафедра математики та теоретичної радіофізики

«ЗАТВЕРДЖУЮ»

Заступник декана з навчальної роботи

_____ Олексій НЕЧИПОРУК

«_____» _____ 2022 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Об'єктно-орієнтоване програмування

для студентів

галузь знань
спеціальність
освітній рівень
освітня програма

10 Природничі науки
105 Прикладна фізика та наноматеріали
перший (бакалавр)
Електроніка та інформаційні технології в медицині

вид дисципліни

обов'язкова

Форма навчання	денна
Навчальний рік	2022/2023
Семестр	2
Кількість кредитів ECTS	3
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі:

Світлана ЄФІМЕНКО, кандидат фізико-математичних наук, доцент;
Ростислав ПОГОРЕЛОВ, асистент

Пролонговано:

на 20__/20__ н.р. _____ (_____) «__» _____ 20__ р.
на 20__/20__ н.р. _____ (_____) «__» _____ 20__ р.

КИЇВ - 2022

Розробник:

Світлана Єфіменко, кандидат фізико-математичних наук, доцент, доцент кафедри математики та теоретичної радіофізики

«ЗАТВЕРДЖЕНО»

Зав. кафедри математики та теоретичної радіофізики

_____ Володимир ВИСОЦЬКИЙ

Протокол № __ від «__» _____ 2022 р.

Схвалено науково-методичною комісією факультету радіофізики, електроніки та комп'ютерних систем

Протокол №__ від «__» _____ 2022 р.

Голова науково-методичної комісії

Сергій РАДЧЕНКО

«_____» _____ 2022 р.

ВСТУП

Навчальна дисципліна «Об'єктно-орієнтоване програмування» є складовою освітньої програми підготовки фахівців за освітнім рівнем «бакалавр» галузі знань 10 «Природничі науки» зі спеціальності 105 «Прикладна фізика та наноматеріали» програми «Прикладна фізика та наноматеріали».

Дана дисципліна входить у блок обов'язкові навчальні дисципліни.

Викладається у 2 семестрі (1 року навчання) в обсязі 90 год. (3 кредити ECTS) зокрема: лекції – всього 30 год., лабораторні заняття – 14 год., самостійна робота – 46 год. У курсі передбачено 2 змістових модулі. Підсумковий контроль проводиться у формі іспиту.

1. Мета дисципліни – ознайомлення та оволодіння студентами основами мов програмування C та C++, вивчення можливостей середовища DEV-CPP, оволодіння навичками хорошого стилю програмування, методами проектування та створення програм згідно сучасних технологій програмування.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

- Успішне опанування курсу «Основи програмування»
- Знання основних прийомів програмування, розуміння принципів технології програмування, знання основ логіки та базових відомостей з математичного аналізу, алгебри.

3. Анотація навчальної дисципліни:

Навчальна дисципліна складається з двох частин. Перша частина присвячена знайомству з основами мов програмування C та C++: основні типи та операції, керуючі конструкції, поняття про структури та файли.

Друга частина курсу «Об'єктно-орієнтоване програмування» присвячена оволодінню основними принципами об'єктно-орієнтованого програмування, реалізованими в мові C++. Зокрема, розглядаються принципи створення класів, реалізація спадкування та поліморфізму у вигляді апарату віртуальних функцій.

4. Завдання (навчальні цілі):

- 1) ЗК 9 Здатність вчитися і оволодівати сучасними знаннями.
- 2) ФК9 Здатність використовувати методи і засоби теоретичного дослідження та математичного моделювання в професійній діяльності.

5. Результати навчання за дисципліною:

Результат навчання (1, знати; 2, вміти; 3, комунікація; 4, автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
1	студент повинен знати:	лекційні заняття, лабораторні заняття	письмові тестові модульні контрольні роботи, оцінювання виконання лабораторних завдань та самостійно створених програм	до 45
1.1	основні принципи ООП			
1.2	базовий синтаксис мов C/C++			
1.3	основні керуючі конструкції (оператори) мов C/C++			
1.4	використання адресації та			

	вказівників в мовах C/C++			
1.5	правила створення методів та передачі параметрів методам та з методів в мовах C/C++			
1.6	правила створення класів в мові C++			
1.7	правила спадкування в мові C++ та використання об'єктів, зокрема фізичної природи, в програмах			
1.8	поліморфізм методів в C++			
2	студент повинен вміти:	лекційні заняття, лабораторні заняття	письмові тестові модульні контрольні роботи, оцінювання виконання лабораторних завдань та самостійно створених програм	до 45
2.1	продумати алгоритм створення програми			
2.2	створювати програму у відповідності до поставленої задачі			
2.3	аналізувати код програми, вміти налагодити його			
2.4	тестувати програму та відшукувати і виправляти в ній помилки			
2.5	використовувати спадкування та поліморфізм методів			
3	комунікація	лекційні заняття, лабораторні заняття		до 5
3.1	здатність грамотно будувати комунікацію, виходячи з мети і ситуації спілкування – найкраще засвоєння матеріалу відбувається саме при спільному аналізі своїх програм			
4	автономність та відповідальність	лекційні заняття, лабораторні заняття	письмові тестові модульні контрольні роботи, оцінювання виконання лабораторних завдань та самостійно створених програм	до 5
4.1	продемонструвати розуміння особистої/персональної відповідальності за професійні та/або управлінські рішення, які базуються на використанні математичних методів			

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни (код)	1.1	1.2	1.3	1.4	1.5	1.6	1.7	1.8	2.1	2.2	2.3	2.4	2.5	3.1	4.1
Програмні результати навчання (назва)															
ПРН 1 Знати і розуміти сучасну фізику на рівні, достатньому для розв'язання складних спеціалізованих задач і практичних проблем прикладної фізики ПРН 1	+	+	+	+	+	+	+	+							
ПРН 2 Застосовувати ефективні технології, інструменти та методи експериментального дослідження властивостей речовин і матеріалів, включаючи наноматеріали, при розв'язанні практичних проблем прикладної фізики. ПРН 2									+	+	+	+	+		
ПРН 7 Застосовувати фізичні, математичні та комп'ютерні моделі для дослідження фізичних явищ, розробки приладів і наукоємних технологій ПРН 7														+	+
ПРН 11 Планувати й організувати результативну професійну діяльність індивідуально і як член команди при розробці та реалізації наукових і прикладних проєктів ПРН 11														+	+

7. Схема формування оцінки

7.1. Форми оцінювання

Рівень досягнення всіх запланованих результатів навчання визначається за результатами виконання завдань лабораторних робіт та тестових контрольних робіт. Внесок результатів навчання у підсумкову оцінку, за умови їх опанування на належному рівні:

1. результати навчання 1.1 – 1.7 [знання] – до 45 %;
2. результат навчання 2.1 – 2.5 [вміння] – до 45%;
3. результат навчання 3.1-3.2 [комунікація] – до 5%;
4. результат навчання 4.1 [автономність та відповідальність] – до 5%;

Форми оцінювання:

1. **семестрове оцінювання:** Навчальний семестр має два змістовні модулі. Після завершення лекцій №6 та №13 проводяться письмові модульні контрольні роботи. Обов'язковим для допуску до іспиту є: написання модульних контрольних робіт з кількістю балів не менше 6.
2. **підсумкове оцінювання (у формі іспиту):** форма іспиту – письмово-усна. Екзаменаційний білет складається з 2 питань, питання оцінюються по 10 балів та завдання на складання програми, яке оцінюється до 20 балів. Всього за іспит можна отримати від 0 до 40 балів. Умовою досягнення позитивної оцінки за дисципліну є отримання не менш ніж 60 балів, оцінка за іспит не може бути меншою **24 бали**.
3. **умови допуску до підсумкового іспиту:** умовою допуску до іспиту є отримання студентом протягом семестру (сумарно) не менше, ніж 36 балів. Студенти, які протягом семестру набрали сумарно меншу кількість балів, для одержання допуску до іспиту повинні відпрацювати і захистити невиконані лабораторні завдання на мінімально необхідну кількість балів.

У випадку відсутності студента з поважних причин відпрацювання та перездачі модульних контрольних робіт здійснюються у відповідності до „Положення про організацію освітнього процесу у Київському національному університеті”

7.2. Організація оцінювання;

Оцінювання за формами контролю:

Семестрова робота	Кількість балів	
	Min. – 3	Max. – 5
Тестова модульна контрольна робота 1	3	5
Тестова модульна контрольна робота 2	3	5

Орієнтований графік оцінювання:

Форма оцінювання	Орієнтовний період для здійснення відповідної форми оцінювання
Тестова модульна контрольна робота 1	березень
Тестова модульна контрольна робота 2	травень
Добір балів/додаткова лабораторні завдання	травень
Іспит	червень

Розрахунок балів, які отримують при успішній здачі іспиту:

Значення	Змістовні модулі	Іспит	Підсумкова оцінка
Мінімум	36	24	60
Максимум	60	40	100

7.3. Шкала відповідності оцінок

Оцінка (за національною шкалою) / National grade	Рівень досягнень, % / Marks, %
Відмінно / Excellent	90-100%
Добре / Good	75-89%
Задовільно / Satisfactory	60-74%
Незадовільно / Fail	0-59%

8. Структура навчальної дисципліни. Тематичний план лекційних занять

№ з/п	Назва теми	Кількість годин		
		Лекції	Лабораторні роботи	Самостійна робота
Змістовний модуль №1 : Основні конструкції мов програмування C та C++.				
1	Основні елементи мови C/C++.	2		3
2	Основні керуючі конструкції мови C/C++ (розгалуження та цикли).	1	2	3
3	Адресація, вказівники та масиви в мові C/C++.	2	2	3
4	Введення та виведення інформації в C/C++.	3	1	3
5-6	Функції в мовах C/C++.	4	1	3
7	Структури та об'єднання в мовах C/C++.	2	1	4
8	Робота з файлами в мовах C/C++.	2	1	4
Змістовний модуль №2 : Реалізація принципів об'єктно-орієнтованого програмування в мові C++.				
9	Принципи об'єктно-орієнтованого програмування, деякі особливості мови C++ – вбудовані функції, перевантаження функцій, аргументи за замовчуванням.	2	1	2
10	Принципи об'єктно-орієнтованого програмування та класи в мові C++.	2	1	4
11	Конструктори та деструктор класу	2	1	2
12	Перевантаження операцій в класі.	2	2	4
13	Спадкування, віртуальні функції, поліморфізм.	2	1	3
14	Шаблони функцій та класів.	2		4
15	Бібліотеки класів та шаблонів.	2		4
Всього		30	14	46

Загальний обсяг **90** год., в тому числі:

Лекції **30** год.

Лабораторні роботи **14** год.

Консультація **1** год.

Самостійна робота **46** год.

Приклади лабораторних завдань:

- 1) Обчислити час вільного падіння деякого тіла з відомої висоти та з заданою початковою швидкістю.

- 2) Обчислити час, через який зустрінуться два тіла, що рухаються рівноприскорено назустріч одне одному, якщо відомі їх початкові швидкості, прискорення та відстань між ними.
- 3) Визначити клас, який реалізує масив. Клас має містити кількість елементів масиву та вказівник на його місце розташування. Передбачити конструктор, який створює масив, метод для ініціалізації елементів, друку його довжини та елементів, деструктор для знищення масиву; метод пошуку елемента масиву з певним значенням, пошуку максимального та мінімального елементів.
- 4) Визначити структуру для збереження інформації про час (години, хвилини, секунди). Написати функцію, яка повертає час на секунду більший від заданого (врахувати перехід через хвилину, годину, на наступний день).

9. Рекомендована література:

Основні джерела:

1. Brian W. Kernigan, Dennis M. Ritchie The ANSI C Programming Language Prentice Hall (1988)
2. Bjarne Stroustrup. The C++ Programming Language (4th Edition) Addison-Wesley ISBN 978-0321563842. May 2013.
3. С.В.Єфіменко. Методичний посібник з курсу «Об'єктно-орієнтоване програмування. Мови C/C++» . – К.: 2021
<https://matphys.rpd.univ.kiev.ua/wp/wp-content/uploads/2022/01/OOP.pdf>
4. В.О.Грязнова, С.В. Єфіменко. Основи методології програмування. - К.: ВПЦ "Київський університет", 2005 р.
5. С.В. Єфіменко, О.В. Сугакова. Програмування: мови C і C++. - К.: ВПЦ "Київський університет", 2006 р.

Додаткові і джерела:

6. Т.П. Караванова. Основи алгоритмізації та програмування. 750 задач з рекомендаціями та прикладами. – К.: Форум, 2002.
7. Niklaus Wirth. Algorithms + Data Structures = Programs (Prentice-Hall Series in Automatic Computation)