КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ТАРАСА ШЕВЧЕНКА

Факультет радіофізики, електроніки та комп'ютерних систем

Кафедра комп'ютерної інженерії

«ЗАТВЕРДЖУЮ»
Заступник декана з навчальної роботи

_____ Світлана ЄФІМЕНКО

« ____ » _____ 2025 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Програмування

для студентів

галузь знань	F Інформаційні технології
спеціальність	F7 Комп'ютерна інженерія
освітній ступінь	бакалавр
освітня програма	Інженерія комп'ютерних систем і мереж
вид дисципліни	обов'язкова

Форма здобуття освіти	Денна
Навчальний рік	2025/2026
Семестр	2
Кількість кредитів ECTS	5
Мова викладання	українська
Форма заключного контролю	іспит

Викладачі:
Єфіменко Світлана Володимирівна, к.ф.-м.н, доцент кафедри математики та теоретичної радіофізики

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

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

Київ – 2025

Розробник:
Світлана ЄФІМЕНКО, к.ф.-м.н, доцент кафедри математики та теоретичної радіофізики

«ЗАТВЕРДЖУЮ»

Завідувач кафедри математики та теоретичної радіофізики

_____ Володимир ВИСОЦЬКИЙ

Протокол № ___ від « ___ » _____ 2025 р.

Завідувач кафедри комп'ютерної інженерії

_____ Юрій БОЙКО

Протокол № ___ від « ____» _____ 2025 р.

Схвалено науково-методичною комісією факультету радіофізики, електроніки та комп'ютерних систем
Протокол №___ від «____» _____ 2025 р.

Голова науково-методичної комісії Сергій РАДЧЕНКО

« ___ » _____ 2025 року.

ВСТУП

1. Метою і завданням навчальної дисципліни «Програмування» є отримання знань, вмінь та практичних навичок у програмуванні, зокрема, в області об’єктно-орієнтованого програмування, вивчення студентами основ мов програмування С та С++, оволодіння навичками хорошого стилю програмування, методами проектування та створення програм згідно сучасних технологій програмування.

Дана дисципліна входить у блок обов’язкові навчальні дисципліни.

Викладається у 2 семестрі (1 року навчання) в обсязі 150 год. (5 кредитів ECTS) зокрема: лекції – всього 28 год., лабораторні заняття – 42 год., самостійна робота – 90 год. У курсі передбачено 2 змістових модулі та 2 модульні тестові контрольні роботи. Підсумковий контроль проводиться у формі іспиту.

2. Передумови для вивчення навчальної дисципліни:

До вивчення дисципліни «Програмування (другий семестр)» необхідні знання з інформатики, основ програмування та основ вищої математики. Має бути успішно зданий залік із першої частини дисципліни «Програмування», яка вивчається у першому семестрі.

3. Анотація навчальної дисципліни:

В курсі розглядаються принципи створення, налагодження та тестування програм, створених мовами програмування високого рівня С/С++. Вивчаються основні елементи програмування мовами С/С++: типи змінних, операції, вирази, розгалуження, цикли, структури, робота з файлами, функції та класи. Розглядається об'єктно-орієнтована парадигма програмування: класи, успадкування, поліморфізм методів, шаблони функцій та класів. Вивчаються деякі методи програмування, пов’язані із динамічними структурами даних, такими як зв'язані списки, а також обробка винятків. На лабораторних заняттях вивчаються деякі методи налагодження та тестування власних програм.

4. Завдання навчальної дисципліни (навчальні цілі):

1. Надати базові знання з об'єктно-орієнтованого програмування. Закріпити навички підходів, що будуть використані у подальшому навчанні.
2. Розвинути у студентів вміння та навички, пов’язанні із створенням кодів програм, їх налагодженням та тестуванням.
3. Надати студентам знання та вміння, необхідні їм у їх професійній роботі по завершенні навчання в університеті.
4. Дисципліна спрямована на формування таких загальних та фахових компетентностей:
   ЗК1. Здатність до абстрактного мислення, аналізу і синтезу.
   ЗК 2. Здатність вчитися і оволодівати сучасними знаннями.
   ЗК 3. Здатність застосовувати знання у практичних ситуаціях.
   СК2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення.
   СК3. Здатність створювати системне та прикладне програмне забезпечення комп’ютерних систем та мереж.
   СК11. Здатність оформляти отримані робочі результати у вигляді презентацій, науково-технічних звітів.

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	до 45
1.1	основні принципи ООП			
1.2	базовий синтаксис мов С/С++			
1.3	основні керуючі конструкції (оператори) мов С/С++			
1.4	використання адресації та вказівників в мовах С/С++			
1.5	правила створення методів та передачі параметрів методам та з методів в мовах С/С++			
1.6	правила створення класів в мові С++			
1.7	правила спадкування в мові С++ та використання об’єктів в програмах			
1.8	поліморфізм методів в С++			
2	студент повинен **вміти**:	лекційні заняття, лабораторні заняття	письмові тестові модульні контрольні роботи, оцінювання виконання лабораторних завдань	до 45
2.1	продумати алгоритм створення програми			
2.2	реалізувати задачу у вигляді програмного коду 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 принципи спадкування та поліморфізм методів			
2.6	Використовувати сучасні прикладні системи програмування			
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) / Програмні результати навчання (назва)	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	2.6	3.1	4.1
ПРН3. Знати новітні технології в галузі комп’ютерної інженерії.	+	+	+	+	+	+	+	+								
ПРН 6 Вміти застосовувати знання для ідентифікації, формулювання і розв’язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.													+	+		
ПРН10. Вміти розробляти програмне забезпечення для вбудованих і розподілених застосувань, мобільних і гібридних систем, розраховувати, експлуатувати типове для спеціальності обладнання.									+	+	+	+				
ПРН 11 Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп’ютерної інженерії.									+		+					
ПРН14. Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів.															+	
ПРН18. Використовувати інформаційні технології для ефективного спілкування на професійному та соціальному рівнях														+		
ПРН19. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.																+

7. Схема формування оцінки

7.1. Форми оцінювання

Семестрове оцінювання: контроль здійснюється за таким принципом. Навчальний семестр має два змістові модулі: у змістовий модуль 1 (ЗМ1) входять теми 1-7, у змістовий модуль 2 (ЗМ2) входять теми 9-15. Протягом модуля студенти виконують 10 лабораторних робіт, що мають індивідуальні завдання. Для визначення рівня досягнення результатів навчання завдання в лабораторних роботах перевіряють уміння створювати програми під конкретні задачі. Обов'язковим для допуску до іспиту є успішна здача всіх 10-ми лабораторних робіт не менше ніж на 36 балів.

Підсумкове оцінювання (у формі іспиту): форма іспиту – письмово-усна. Екзаменаційний білет іспиту складається із двох питань, кожне питання оцінюється від 0 до 10 балів, та задачі на створення програми, яка оцінюється від 0 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.

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 має виконати додаткове завдання надане викладачем.

7.2. Організація оцінювання;

Критерієм успішного проходження здобувачем освіти підсумкового оцінювання є досягнення ним мінімальних порогових рівнів умінь (створення програм) і знань (опанування теоретичного матеріалу)

Оцінювання за формами контролю:

Семестрова робота	Кількість балів	
	Min. – 12	Max. – 20
Модульна контрольна робота 1 (Тест)	2	5
Модульна контрольна робота 2 (Тест)	2	5
Виконання лабораторних робіт	32	50

Орієнтований графік оцінювання:

Форма оцінювання	Орієнтовний період для здійснення відповідної форми оцінювання
Модульна контрольна робота 1	лютий
Модульна контрольна робота 2	травень
Виконання лабораторних робіт	лютий - травень
Добір балів/додаткова контрольна робота	червень
Іспит	червень

Розрахунок балів, які отримують при успішній здачі іспиту:

Значення	Протягом семестру	Іспит	Підсумкова оцінка
Мінімум	36	24	60
Максимум	60	40	100

8. Структура навчальної дисципліни. Тематичний план лекційних занять

№ п/п	Назва теми	У тому числі		
		Лекції	Лабораторні заняття	Самостійна робота
Змістовний модуль №1 : Основні конструкції мов програмування С та С++.				
1	Основні елементи мови С/С++.	2	2	4
2	Основні керуючі конструкції мови С/С++ (розгалуження та цикли).	2	4	6
3	Адресація, вказівники та масиви в мові С/С++.	2	4	8
4	Введення та виведення інформації в С/С++.	2	2	4
5	Функції в мовах С/С++.	2	4	8
6-7	Структури та об'єднання в мовах С/С++. Динамічні структури даних – однозв'язні списки.	4	4	6
8	Робота з файлами в мовах С/С++.	2	2	4
	Всього	**14**	**22**	**40**
Змістовий модуль №2 : Реалізація принципів об'єктно-орієнтованого програмування в мові С++				
9	Деякі особливості функцій мови С++ – вбудовані функції, перевантаження функцій, аргументи функцій за замовчуванням.	2	2	6
10	Принципи об'єктно-орієнтованого програмування та класи в мові С++.	2	4	6
11	Конструктори та деструктор класу	2	2	4
12	Перевантаження операцій в класі.	2	4	6
13	Спадкування, віртуальні функції, поліморфізм.	2	4	8
14	Шаблони функцій та класів.	2	2	6
15	Виключення. Бібліотеки класів та шаблонів.	2	2	4
	Всього	**14**	**20**	**40**
	Загалом за курсом	**28**	**42**	**80**

Загальний обсяг занять **150** *год.,* в тому числі:
Лекцій **–28** *год.*
Лабораторних робіт **– 42** *год.*
Самостійна робота – **80** *год.*

9. СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

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. О. Васильев. Програмування С++ в прикладах і задачах. “Ліра”. 2017
4. С.В.Єфіменко. Методичний посібник з курсу «Об’єктно-орієнтоване програмування. Мови С/С++» . – К.: 2021 https://matphys.rpd.univ.kiev.ua/wp/wp-content/uploads/2022/01/OOP.pdf
5. В.О.Грязнова, С.В. Єфіменко. Основи методології програмування. - К.: ВПЦ "Київський університет", 2005 р.
6. С.В. Єфіменко, О.В. Сугакова. Програмування: мови С і С++. - К.: ВПЦ "Київський університет", 2006 р.

Додаткові і джерела:

7. Stanley B. Lippman, Josée Lajoie, Barbara E. Moo. C++ Primer (5th Edition). 2012.Addison-Wesley Professional
8. Т.П. Караванова. Основи алгоритмізації та програмування. 750 задач з рекомендаціями та прикладами. – К.: Форум, 2002.
9. Niklaus Wirth. Algorithms + Data Structures = Programs (Prentice-Hall Series in Automatic Computation)
10. Адіт’я Бхаргава Грокаємо алгоритми. Ілюстрований посібник для програмістів і допитливих – К: ArtHuss, 2023, 256 с.