

Чернівецький національний університет імені Юрія Федьковича
(повне найменування вищого навчального закладу)

Факультет математики та інформатики

(назва інституту / факультету)

Кафедра прикладної математики та інформаційних технологій

(назва кафедри)



“ЗАТВЕРДЖУЮ”

Деканка факультету математики та інформатики

Мартинюк О.В.

2025 року

РОБОЧА ПРОГРАМА
навчальної дисципліни

Програмування

(назва навчальної дисципліни)

обов'язкова

(вказати: обов'язкова / вибіркова)

Освітньо-професійна програма **Технології програмування та комп'ютерне моделювання**

(назва програми)

Спеціальність **F1 – Прикладна математика**

(вказати: код, назва)

Галузь знань **F – Інформаційні технології**

(вказати: шифр, назва)

Рівень вищої освіти **перший бакалаврський**

(вказати: перший бакалаврський/другий магістерський)

факультет математики та інформатики

(назва факультету/інституту, на якому здійснюється підготовка фахівців за вказаною освітньо-професійною програмою)

Мова навчання **українська**

(вказати: на якій мові читається дисципліна)

Чернівці 2025 рік

Робоча програма навчальної дисципліни **Програмування** складена відповідно до освітньо-професійної програми «Технології програмування та комп'ютерне моделювання».

Розробник: Скутар Ігор Дмитрович, асистент, канд. – фіз. мат. наук.
(П.І.Б. авторів, посада, науковий ступінь, вчене звання)

Викладач, що забезпечує читання даної навчальної дисципліни:
Скутар Ігор Дмитрович, асистент, канд. – фіз. мат. наук.
(П.І.Б. авторів, посада, науковий ступінь, вчене звання)

Погоджено з гарантом ОП  Василь МАЦЕНКО

Затверджено на засіданні кафедри прикладної математики та інформаційних технологій

Протокол № 13 від “24” червня 2025 року

Завідувач кафедри  Ярослав БІГУН

Схвалено методичною радою факультету математики та інформатики
Протокол № 12 від “25” червня 2025 року

Голова методичної ради
факультету математики та інформатики  Віра СІКОРА

Затверджено Вченою радою факультету математики та інформатики
Протокол № 13 від “25” червня 2025 року

Голова Вченої ради
факультету математики та інформатики  Ольга МАРТИНЮК

Мета навчальної дисципліни:

- формування теоретичної бази знань студентів щодо побудови алгоритмів та їх реалізації мовою програмування;
- розвиток уміння розв'язувати алгоритмічні задачі, користуючись відомими теоретичними положеннями, математичним апаратом, літературою та комп'ютерною технікою;
- доведення вивчення курсу до творчого рівня;
- розвиток логічного, аналітичного мислення та основних видів розумової діяльності: уміння використовувати індукцію, дедукцію, аналіз, синтез, робити висновки, узагальнення;
- бачення студентами можливостей використання набутих знань у їх майбутній професії;
- інтеграція курсу з іншими дисциплінами, що викладаються в навчальному закладі.

Результати навчання:

У результаті вивчення дисципліни студенти повинні *знати* основні поняття та твердження з програмного матеріалу даного курсу; *вміти* використовувати вивчений матеріал при розв'язуванні конкретних задач, застосовувати теоретичні знання на практиці.

Знання, які студент отримає в результаті вивчення даної дисципліни, відіграватимуть важливу роль у процесі його професійного формування та зростання, а також вони є необхідною основою для вивчення деяких інших фахових дисциплін.

Під час вивчення дисципліни, відповідно до освітньо-професійної програми, формуються наступні

загальні компетентності:

- ЗК01 - здатність учитися і оволодівати сучасними знаннями;
- ЗК02 - здатність застосовувати знання у практичних ситуаціях;
- ЗК03 - здатність генерувати нові ідеї (креативність);
- ЗК04 - здатність бути критичним і самокритичним;
- ЗК06 - здатність до абстрактного мислення, аналізу та синтезу;
- ЗК07 - здатність до пошуку, оброблення та аналізу інформації з різних джерел;
- ЗК08 - знання та розуміння предметної області та розуміння професійної діяльності;
- ЗК10 - навички у використанні інформаційних і комунікаційних технологій;
- ЗК12 - визначеність і наполегливість щодо поставлених завдань і взятих обов'язків;
- ЗК13 - навички міжособистісної взаємодії;

фахові компетентності:

- ФК02 - здатність виконувати завдання, сформульовані у математичній формі;
- ФК04 - здатність розробляти алгоритми та структури даних, програмні засоби та програмну документацію;

- ФК06 - здатність розв'язувати професійні задачі за допомогою комп'ютерної техніки, комп'ютерних мереж та Інтернету, в середовищі сучасних операційних систем, з використанням стандартних офісних додатків;
- ФК07 - здатність експлуатувати та обслуговувати програмне забезпечення автоматизованих та інформаційних систем різного призначення;
- ФК08 - здатність використовувати сучасні технології програмування та тестування програмного забезпечення;
- ФК10 - здатність створення документів встановленої звітності, використання нормативно-правових документів;
- ФК17 - здатність до використання новітніх інформаційно-комунікаційних технологій;
- ФК19 - здатність застосовувати сучасні інформаційні технології для комп'ютерної реалізації математичних моделей і алгоритми комп'ютерної графіки для розробки графічного програмного забезпечення;
- та отримуються наступні **програмні результати навчання**:
- ПРН11 - вміти застосовувати сучасні технології програмування та розроблення програмного забезпечення, програмної реалізації чисельних і символьних алгоритмів;
- ПРН14 - виявляти здатність до самонавчання та продовження професійного розвитку;
- ПРН15 - уміти організувати власну діяльність та одержувати результат у рамках обмеженого часу;
- ПРН18 - ефективно спілкуватися з питань інформації, ідей, проблем та рішень зі спеціалістами та суспільством загалом.

Опис навчальної дисципліни Загальна інформація

Форма навчання	Рік підготовки	Семестр	Кількість		Кількість годин						Вид підсумкового контролю
			кредитів	годин	лекції	практичні	семінарські	лабораторні	самостійна робота	індивідуальні завдання	
Денна	1	1	7	210	30	30	0	30	120	0	залік

Структура змісту навчальної дисципліни

Назви змістових модулів і тем	Кількість годин												
	денна форма							Заочна форма					
	усього	у тому числі					усього	у тому числі					
		л	п	лаб	інд	с.р.		л	п	лаб	інд	с.р.	
1	2	3	4	5	6	7	8	9	10	11	12		
Змістовий модуль 1.													
Тема 1. Основні поняття алгоритмізації та програмування.		2	0	0		8							
Тема 2. Мова програмування Python.		2	0	0		8							
Тема 3. Логічні оператори і вирази.		2	4	4		8							
Тема 4. Циклічні конструкції.		2	4	4		8							
Тема 5. Одновимірні масиви. Найпростіші алгоритми роботи з одновимірними масивами.		2	2	4		8							

Тема 6. Упорядкування масивів.		2	4	4		8						
Тема 7. Двовимірні масиви.		2	2	2		8						
Тема 8. Базові алгоритми для обробки елементів двовимірних масивів.		2	2	2		8						
Тема 9. Словники.		2	2	2		8						
Разом за змістовим модулем 1		18	20	22		72						
Змістовий модуль 2.												
Тема 1. Рядки і символні масиви. Стандартні функції для роботи з рядками.		2	2	2		8						
Тема 2. Найпростіші алгоритми роботи із символними та рядковими величинами.		2	4	2		8						
Тема 3. Функції користувача.		2	4	2		8						
Тема 4. Модулі.		2	0	2		8						
Тема 5. Робота із файлами.		2	0	0		8						
Тема 6. Математичні пакети.		2	0	0		8						
Разом за змістовим модулем 2		12	10	8		48						
Усього годин за семестр		30	30	30		120						

Теми семінарських занять

Семінарські заняття не передбачені навчальним планом.

Теми практичних занять

№ з/п	Назва теми	Кількість годин
1	Логічні оператори і вирази.	4
2	Циклічні конструкції.	4
3	Одновимірні масиви. Найпростіші алгоритми роботи з одновимірними масивами. Упорядкування масивів.	6
4	Двовимірні масиви. Алгоритми для обробки елементів двовимірних масивів.	4

5	Словники	2
6	Символьні рядки	6
7	Створення і використання функцій	4
	Разом	30

Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Лінійне програмування та умовний оператор	4
2	Циклічні програми	4
3	Алгоритми обробки масивів. Одновимірні масиви	4
4	Алгоритми обробки масивів. Багатовимірні масиви	4
5	Упорядкування і пошук даних	4
6	Словники	2
7	Символьні рядки	4
8	Створення і використання функцій	4
	Разом	30

Самостійна робота студентів

№ з/п	Назва теми	Кількість годин
1	Основні поняття алгоритмізації та програмування.	8
2	Мова програмування Python.	8
3	Логічні оператори і вирази.	8
4	Циклічні конструкції.	8
5	Одновимірні масиви. Найпростіші алгоритми роботи з одновимірними масивами.	8
6	Упорядкування масивів.	8
7	Двовимірні масиви.	8
8	Базові алгоритми для обробки елементів двовимірних масивів.	8
9	Словники.	8
10	Рядки і символьні масиви. Стандартні функції для роботи з рядками.	8
11	Найпростіші алгоритми роботи із символьними та рядковими величинами.	8
12	Функції користувача.	8
13	Модулі.	8
14	Робота із файлами.	8
15	Математичні пакети.	8

Разом	120
-------	-----

Методи навчання

- **Пояснювально-ілюстративний метод:** викладач пояснює теоретичний матеріал, супроводжуючи пояснення прикладами коду та наочними ілюстраціями. Це допомагає студентам краще зрозуміти абстрактні концепції.
- **Репродуктивний метод:** студенти повторюють типові задачі та алгоритми, які розглядаються на заняттях. Це дозволяє їм закріпити базові навички програмування.
- **Проблемний метод:** викладач пропонує задачі, які потребують творчого підходу та аналізу. Студенти вчаться шукати оптимальні рішення, експериментуючи з різними підходами.
- **Метод проєктів:** студенти працюють над створенням власного проєкту на Python, що може включати розробку веб-додатку, автоматизацію процесів або аналіз даних.

Система контролю та оцінювання

Формами поточного контролю є усна (захист лабораторних робіт) чи письмова (тестування, практичні завдання) відповіді студента тощо.

Формами підсумкового контролю є залік.

Поточний контроль: контрольне тестування після модуля 1, практична контрольна робота, вибіркове опитування на лекційних заняттях.

Підсумковий контроль – залік.

Критерії оцінювання результатів навчання з навчальної дисципліни

Залік. Підсумковий модуль-контроль відбувається у вигляді тестування. Тест (30 балів) складається з 30-х теоретичних та практичних питань по 1 балу за кожну правильну відповідь. У сумі з модульними контролями (70 балів) це загалом становитиме 100 балів.

Підсумкова оцінка. Підсумкова оцінка виставляється за загальною сумою набраних студентом балів згідно із наступною таблицею:

Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота		Підсумковий тест (залік)	Сума
Змістовий модуль 1	Змістовий модуль 2	30	100
35	35		

Бали у змістових модулях включають:

1. Лабораторні роботи – 40 балів

2. Модуль-контроль (тести) – 10 балів
3. Контрольна робота (практична) – 10 балів
4. Практичні заняття – 10 балів

Шкала оцінювання: національна та ЄКТС

Оцінка за національною шкалою	Оцінка за шкалою ЄКТС	
	Оцінка (бали)	Пояснення за розширеною шкалою
Відмінно	A (90-100)	відмінно
Добре	B (80-89)	дуже добре
	C (70-79)	добре
Задовільно	D (60-69)	задовільно
	E (50-59)	достатньо
Незадовільно	FX (35-49)	(незадовільно) з можливістю повторного складання
	F (1-34)	(незадовільно) з обов'язковим самостійним опрацюванням освітнього компоненту до перескладання

Засоби оцінювання

1. Практичні контрольні роботи.
2. Теоретичні тестові завдання.
3. Вибіркове опитування на лекційних та практичних заняттях.
4. Лабораторні роботи.

Рекомендована література

основна:

1. Караванова Т.П. Інформатика. Основи алгоритмізації та програмування (процедурне програмування). Базовий курс. Навч. посіб. Доп. та випр. – Шепетівка: Аспект, 2005. –250 с
2. Караванова Т.П. Основи алгоритмізації та програмування: 777 задач з рекомендаціями та прикладами: Навч. посіб. Доп. та випр. – К.: Генеза, 2006. – 288 с.: іл.
3. Бріггс Джейсон Р. PYTHON для дітей. Веселий вступ до програмування. Видавництва Старого Лева. 2017р. 400 ст.
4. Маттес Ерік. Пришвидшений курс Python. Видавництво Старого Лева. 2021р. 600 ст.
5. Васильєв Олексій. Програмування мовою Python. – Львів: Видавництво «Навчальна книга — Богдан», 2019. – 504 с.

допоміжна:

6. Mark Lutz. Learning Python, 5th Edition (2013). 1600p.
7. Michael Dawson. Python Programming for the Absolute Beginner (2013)

Інформаційні ресурси

1. Python official community resource [Electronic Resource]. – Mode of access: URL: <https://www.python.org/>
2. Python 3: Курс Молодого Бійця [Електронний ресурс]. – Режим доступу: URL: <http://www.vitaliy podoba.com/tutorials/python3-beginners-course/>

Політика академічної доброчесності

Дотримання політики щодо академічної доброчесності учасниками освітнього процесу при вивченні навчальної дисципліни регламентовано такими документами:

- «Етичний кодекс Чернівецького національного університету імені Юрія Федьковича»
<https://www.chnu.edu.ua/media/jxdfs0zb/etychnyi-kodeks-chernivetskoho-natsionalnoho-universytetu.pdf>
- «Положенням про виявлення та запобігання академічного плагіату у Чернівецькому національному університету імені Юрія Федьковича»
<https://www.chnu.edu.ua/media/n5nbzwgb/polozhennia-chnu-pro-plahiat-2023plUSDodatky-31102023.pdf>