Чернівецький національний університет імені Юрія Федьковича

(повне найменування головного закладу вищої освіти)

Відокремлений структурний підрозділ «Фаховий коледж

Чернівецького національного університету імені Юрія Федьковича»

(повна назва відокремленого структурного підрозділу)

Циклова комісія *комп'ютерна інженерія*

(назва циклової/предметної комісії)

"ПОГОДЖЕНО"
Завідувач
Природничого відділення
_____ *Христина МЕЛЬНИЧУК*
(підпис) (ім'я та прізвище)
"27" 08 20 26 року

"ЗАТВЕРДЖУЮ"
Заступник директора
з навчально-методичної роботи
_____ *Микола ДЕРЕВ'ЯНЧУК*
(підпис) (ім'я та прізвище)
"28" 08 20 26 року

ПРОГРАМА

навчальної дисципліни

"Алгоритмізація та програмування"

(вкажіть назву навчальної дисципліни (іноземною, якщо дисципліна викладається іноземною мовою))

обов'язкова

(вказати: обов'язкова/вибіркова)

Освітньо-професійна програма

"Комп'ютерна інженерія"

(назва освітньо-професійної програми)

Спеціальність

F7 Комп'ютерна інженерія

(код і назва спеціальності)

Галузь знань

F Інформаційні технології

(код і назва галузі знань)

Освітньо-професійний ступінь

фаховий молодший бакалавр

(назва освітньо-професійного ступеня)

Мова навчання

українська

(вказати: на якій мові читається предмет)

Чернівці, 2026

Програма навчальної дисципліни

"Алгоритмізація та програмування"

(назва навчальної дисципліни)

складена відповідно до освітньо-професійної програми

"Комп'ютерна інженерія"

(назва освітньо-професійної програми)

затвердженої Вченою радою Чернівецького національного університету імені Юрія Федьковича (Протокол № 4 від «28» березня 2025 року) та введеної в дію наказом ректора №315 від «28» серпня 2026 року.

Розробники: (вказати авторів, їхні посади, наукові ступені та вчені (педагогічні) звання)

викладач вищої кваліфікаційної категорії, В.В. Коропецький, викладач першої кваліфікаційної категорії, Ю.В. Луцюк, викладач вищої кваліфікаційної категорії, Т.І.Крошка

Профайли викладачів https://cs-college.chnu.edu.ua/tsyklova-komisiia/sklad-tsyklovoi-komisii/

E-mail v.koropetskiy@chnu.edu.ua, y.lutsiuk@chnu.edu.ua, t.kroshka@chnu.edu.ua

Сторінка курсу в Classroom/Moodle https://moodle.chnu.edu.ua/course/view.php?id=6312 https://moodle.chnu.edu.ua/course/view.php?id=7945

Робоча програма обговорена та узгоджена на засіданні циклової комісії

комп'ютерна інженерія

Протокол № 1 від "27" 08 2026 року

Голова циклової комісії _____ *Олександр ТАЩУК*

(підпис) (ім'я та прізвище)

Схвалена Методичною радою

Протокол № 1 від "27" 08 2026 року

Голова методичної ради _____ *Ольга БІЛОКРИЛА*

(підпис) (ім'я та прізвище)

ПЕРЕЗАТВЕРДЖЕНО

Протокол № ____ від _____, 20 __ р. _____ _____

(підпис) (ініціали та прізвище голови ЦК)

Протокол № ____ від _____, 20 __ р. _____ _____

(підпис) (ініціали та прізвище голови ЦК)

Протокол № ____ від _____, 20 __ р. _____ _____

(підпис) (ініціали та прізвище голови ЦК)

1. Мета та завдання навчальної дисципліни

Анотація: Дисципліна "Програмування Ч.1. Алгоритмізація та програмування" є нормативною для студентів 2 курсу спеціальності "Комп'ютерні науки". Предметом вивчення дисципліни є імперативне та процедурне програмування мовами C/C++, робота зі структурованими типами даних, вказівниками, динамічною пам’яттю, функціями, рекурсивними алгоритмами, динамічними структурами даних та файлами. Матеріал курсу спрямований на формування практичних навичок створення, налагодження та тестування програм засобами мов C/C++.

Матеріал курсу є початковим етапом у циклі дисциплін, пов’язаних з вивченням мов програмування та реалізацією системних та прикладних програм. Важливою складовою частиною вивчення дисципліни є ознайомлення та засвоєння набору елементарних базових алгоритмів для розв’язування математичних і кібернетичних задач та їх застосування в прикладних та системних програмах.

Мета навчальної дисципліни:

- ознайомлення студентів з інтегрованими середовищами розробки програмних продуктів, з їх функціональними можливостями та етапами реалізації програм;

вивчення синтаксичних конструкцій мов програмування C/C++ та вимог імперативної процедурної парадигми програмування;

- засвоєння студентами основних стереотипних математичних та структурних алгоритмів, можливості їх модифікації та комбінування для розв’язування задач та реалізації системних та прикладних задач.

Алгоритми, техніка програмування, засоби розробки програм та технологій, які вивчаються у цьому курсі є основою для подальшої професійної підготовки , вивчення ряду професійно-орієнтованих дисциплін.

Пререквізити: Для підвищення ефективності вивчення навчальної дисципліни здобувач освіти повинен до початку курсу мати знання зі шкільного курсу інформатики та базові знання з курсів вищої математики.

Перелік предметних компетентностей здобувача вищої освіти:

Загальні:

ЗК.03. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК.04. Здатність застосовувати знання у практичних ситуаціях.

ЗК.05. Здатність спілкуватися державною мовою як усно, так і письмово.

ЗК.07. Здатність працювати в команді.

Спеціальні:

СК.01. Здатність застосовувати законодавчу та нормативно правову базу, а також державні та міжнародні вимоги, практики і стандарти з метою здійснення професійної діяльності в галузі інформаційних технологій.

СК.02. Здатність застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування апаратних, програмних та інструментальних засобів комп’ютерної інженерії.

СК.08. Здатність здійснювати організацію робочих місць з урахуванням вимог охорони праці, їх технічне оснащення, розміщення комп'ютерного устаткування, використання організаційних, технічних, алгоритмічних та інших методів і засобів захисту інформації.

СК.11. Здатність здійснювати вибір, розгортати, інтегрувати, діагностувати, адмініструвати та експлуатувати комп’ютерні системи та мережі, мережеві ресурси, сервіси та інфраструктуру організації.

СК.13. Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп’ютерних систем, мереж та їх компонентів шляхом використання аналітичних методів і методів моделювання.

СК.14. Здатність оцінювати і враховувати економічні, соціальні, технологічні та екологічні чинники, що впливають на сферу професійної діяльності.

СК.15. Здатність проектувати, розробляти та обслуговувати webзастосунки з динамічним контентом, використовуючи webтехнології, технології комп’ютерної графіки та анімації.

2. Опис навчальної дисципліни

2.1. Загальна інформація

Форма навчання	Рік підготовки	Семестр	Кількість			Кількість годин						Вид підсумкового контролю
			Кредитів	Годин	Модулів	Лекції	Семінарські	Практичні	Лабораторні	Самостійна робота	Підготовка до екзамену	
Денна	2	3	5	150	3	34	-	-	34	52	30	екзамен
	2	4	5	150	2	36	-	-	36	48	30	екзамен

2.2. Дидактична карта навчальної дисципліни

Назви тем	**Кількість годин**						
	денна форма						
	усього	**у тому числі**					
		лекції	семінарські	практичні	лабораторні	самостійна робота	
1	2	3			4	5	
Семестр 3							
МОДУЛЬ 1. Поглиблене програмування Python							
Тема 1. Від програми до програмного продукту. Проєктування власного проєкту	7,1	2	-	-	2	3,1	
Тема 2. Поглиблені можливості Python. Сучасний стиль програмування	7,1	2	-	-	2	3,1	
Тема 3. Функції та декомпозиція складних програм	7,1	2	-	-	2	3,1	
Тема 4. Об’єктно-орієнтоване програмування в Python	7,1	2	-	-	2	3,1	
Тема 5. Наслідування, композиція та поліморфізм. Проєктування класів	7,1	2	-	-	2	3,1	
Тема 6. Модулі, пакети та архітектура Python-проєкту	7,1	2	-	-	2	3,1	
МОДУЛЬ 2. Дані, алгоритми та взаємодія програм							
Тема 7. Обробка помилок та винятків. Надійність програм	7,1	2	-	-	2	3,1	
Тема 8. Робота зі структурованими даними: JSON та CSV	7,1	2	-	-	2	3,1	
Тема 9. Бази даних та SQL у Python	7,1	2	-	-	2	3,1	
Тема 10. Алгоритми та структури даних. Ефективність програм	7,1	2	-	-	2	3,1	
Тема 11. Пошук, сортування та оптимізація роботи з даними	7	2	-	-	2	3	
Тема 12. Тестування, налагодження та якість програмного коду	7	2	-	-	2	3	
Тема 13. API та мережеві запити. Взаємодія Python із зовнішніми сервісами	7	2	-	-	2	3	
МОДУЛЬ 3. Розроблення власного програмного продукту							

Тема 14. Графічний інтерфейс користувача Python-програми	7	2			2	3	
Тема 15. Архітектура та інтеграція компонентів власного проєкту	7	2	-	-	2	3	
Тема 16. Розроблення, тестування та вдосконалення власного програмного продукту	7	2	-	-	2	3	
Тема 17. Презентація та захист власного програмного продукту	7	2	-	-	2	3	
Екзамен	**30**	**-**	**-**	**-**	-	-	
Усього годин за 3 семестр	***150***	***34***	***-***	***-***	***34***	***52***	

Назви модулів і тем	**Кількість годин**					
	усього	у тому числі				
		лекції	семінарські	практичні	лабораторні	самостійна робота
1	2	3	4	5	6	7
Семестр 4						
Модуль 1. Процедурне програмування мовою С/С++.						
Тема 1. Структуровані типи даних	8	2			2	4
Тема 2. Динамічна програмна пам'ять. Вказівники.	8	2			2	4
Тема 3. Динамічний розподіл пам'яті для одновимірних масивів	8	2			2	4
Тема 4. Динамічний розподіл пам'яті для двовимірних масивів	8	2			2	4
Тема 5. Розподіл пам'яті для структур та масивів структур	8	2			2	4
Тема 6. Процедурна парадигма. Створення та використання функцій.	8	2			2	4
Тема 7. Функції зі скалярними параметрами	8	2			2	4
Тема 8. Функції з адресними параметрами	8	2			2	4
Тема 9. Область дії змінних. Правила використання параметрів.	8	2			2	4
Разом за Модулем 1	**72**	**18**			**18**	**36**
Модуль 2. Керування пам'яттю та обробка файлової інформації засобами С/С++.						
Тема 10. Поняття рекурсії. Організація рекурсивних обчислень	8	2			2	4
Тема 11. Динамічні структури даних. Списки.	8	2			2	4
Тема 12. Маніпуляції даними в списках. Вилучення та включення елементів	8	2			2	4
Тема 13. Створення систематизованих списків	8	2			2	4
Тема 14. Ієрархічні структури даних. Створення та обробка дерев	8	2			2	4
Тема 15. Рекурсивні методи обробки ієрархічних структур даних	8	2			2	4
Тема 16. Обробка файлової інформації. Загальні засоби управління роботи з файлами	10	2			2	6
Тема 17. Засоби та методи обробки текстових файлів	10	2			2	6
Тема 18. Засоби та методи обробки файлів прямого доступу	10	2			2	6
Разом за Модулем 2	**72**	**18**			**18**	**36**
Усього за дисципліну (семестр 4)	**150**	**36**			**36**	**78**

3. Теми лекційних занять

№ з/п	Назви тем	Кількість годин
Семестр 3		
1	Тема 1. Від програми до програмного продукту. Проєктування власного проєкту	2
2	Тема 2. Поглиблені можливості Python. Сучасний стиль програмування	2
3	Тема 3. Функції та декомпозиція складних програм	2
4	Тема 4. Об’єктно-орієнтоване програмування в Python	2
5	Тема 5. Наслідування, композиція та поліморфізм. Проєктування класів	2
6	Тема 6. Модулі, пакети та архітектура Python-проєкту	2
7	Тема 7. Обробка помилок та винятків. Надійність програм	2
8	Тема 8. Робота зі структурованими даними: JSON та CSV	2
9	Тема 9. Бази даних та SQL у Python	2
10	Тема 10. Алгоритми та структури даних. Ефективність програм	2
11	Тема 11. Пошук, сортування та оптимізація роботи з даними	2
12	Тема 12. Тестування, налагодження та якість програмного коду	2
13	Тема 13. API та мережеві запити. Взаємодія Python із зовнішніми сервісами	2
14	Тема 14. Графічний інтерфейс користувача Python-програми	2
15	Тема 15. Архітектура та інтеграція компонентів власного проєкту	2
16	Тема 16. Розроблення, тестування та вдосконалення власного програмного продукту	2
17	Тема 17. Презентація та захист власного програмного продукту	2
Разом за 3 семестр		**34**
Семестр 4		
1	Тема 1. Структуровані типи даних 1. Поняття структури даних. Прості методи оголошення 2. Структура як тип даних. Створення нових структурних типів 3. Стереотипні операції обробки структурованих даних	2
2	Тема 2. Динамічна програмна пам'ять. Вказівники. 1. Поняття адрес даних та вказівників. 2. Вказівники на скалярні дані. 3. Операції над вказівниками.	2
3	Тема 3. Динамічний розподіл пам'яті для одновимірних масивів 1. Вказівники на одновимірні масиви. Переваги використання вказівників. 2. Функції диспетчеризації даних при роботі з одновимірними масивами. 3. Алгоритми для роботи з динамічною пам’яттю	2
4	Тема 4. Динамічний розподіл пам'яті для двовимірних масивів 1. Поняття вказівника на масив вказівників. Дворівнева структура 2. Правила виділення пам’яті для двовимірних масивів 3. Адресна та індексна форма локалізації елементів	2
5	Тема 5. Розподіл пам'яті для структур та масивів структур 1. Лексика використання вказівників на структури 2. Правила керування пам’яттю при роботі з масивами структур	2
6	Тема 6. Процедурна парадигма. Створення та використання функцій.	2

	1. Поняття підпрограми. Основні положення процедурної парадигми 2. Прототипна та сучасна форма оголошення функції 3. Результат функції. Правила звернення до функцій	
7	Тема 7. Функції зі скалярними параметрами 1. Параметри функції. Значення та адреси параметрів 2. Правила використання функції з параметрами значеннями 3. Приклади створення та використання функцій з вхідними параметрами	2
8	Тема 8. Функції з адресними параметрами Особливості функцій з вихідними параметрами. Правила передачі адресних скалярних параметрів Операції адресації та розадресації формальних та фактичних параметрів	2
9	Тема 9. Область дії змінних. Правила використання параметрів. 1. Глобальні та локальні розділи програмних даних. 2. Правила використання значень параметрів, глобальних та локальних даних 3. Правила оголошення глобальних та локальних даних	2
10	Тема 10. Поняття рекурсії. Організація рекурсивних обчислень 1. Поняття рекурсивного алгоритму та рекурсивної функції 2. Класифікація рекурсій 3. Приклади організації рекурсивних обчислень.	2
11	Тема 11. Динамічні структури даних. Списки. 1. Поняття динамічної структури даних. 2. Класифікація динамічних структур. 3. Створення черг та стеків.	2
12	Тема 12. Маніпуляції даними в списках. Вилучення та включення елементів 1. Алгоритми додавання нових елементів до списку 2. Алгоритм вилучення елементів зі списку 3. Прості операції обробки елементів списку	2
13	Тема 13. Створення систематизованих списків 1. Алгоритм циклічного створення впорядкованого списку 2. Алгоритм циклічного пошуку особливих елементів списку 3. Алгоритм вилучення відмічених елементів списку	2
14	Тема 14. Ієрархічні структури даних. Створення та обробка дерев 1. Поняття ієрархічної структури даних 2. Бінарні дерева. Алгоритм створення.	2
15	Тема 15. Рекурсивні методи обробки ієрархічних структур даних 1. Бінарні дерева. Алгоритм відображення. 2. Алгоритм пошуку у бінарному дереві 3. Алгоритм вилучення дерева	2
16	Тема 16. Обробка файлової інформації. Загальні засоби управління роботи з файлами 1. Поняття «зовнішнього файлу». Класифікація. 2. Функції керування зв’язком між програмою та файлом 3. Керування режимом доступу	2
17	Тема 17. Засоби та методи обробки текстових файлів 1. Поняття файлів послідовного доступу. Правила зміни режимів 2. Операції читання з текстового файлу та запису у текстовий файл. 3. Особливості циклічної роботи з регулярними текстовими даними	2
18	Тема 18. Засоби та методи обробки файлів прямого доступу 1. Поняття файлів прямого доступу 2. Операції читання та запису при роботі з бінарними файлами прямого доступу	2

	3. Керування розміщенням у файлі прямого доступу для обміну інформацією,	
Разом за семестр 4		**36**

Контроль виконання завдань, винесених на підготовку до лекційних занять та опрацювання лекційного матеріалу, здійснюється в межах модульного контролю. Однією з опор для опрацювання лекційного матеріалу є конспект лекцій, що ведеться студентами.

Бали за цю роботу входять у загальну кількість балів за конкретний модуль.

4. Теми лабораторних завдань

№ з/п	Назви тем	Кількість годин
Семестр 3		
1	Тема 1. Від програми до програмного продукту. Проєктування власного проєкту	2
2	Тема 2. Поглиблені можливості Python. Сучасний стиль програмування	2
3	Тема 3. Функції та декомпозиція складних програм	2
4	Тема 4. Об’єктно-орієнтоване програмування в Python	2
5	Тема 5. Наслідування, композиція та поліморфізм. Проєктування класів	2
6	Тема 6. Модулі, пакети та архітектура Python-проєкту	2
7	Тема 7. Обробка помилок та винятків. Надійність програм	2
8	Тема 8. Робота зі структурованими даними: JSON та CSV	2
9	Тема 9. Бази даних та SQL у Python	2
10	Тема 10. Алгоритми та структури даних. Ефективність програм	2
11	Тема 11. Пошук, сортування та оптимізація роботи з даними	2
12	Тема 12. Тестування, налагодження та якість програмного коду	2
13	Тема 13. API та мережеві запити. Взаємодія Python із зовнішніми сервісами	2
14	Тема 14. Графічний інтерфейс користувача Python-програми	2
15	Тема 15. Архітектура та інтеграція компонентів власного проєкту	2
16	Тема 16. Розроблення, тестування та вдосконалення власного програмного продукту	2
17	Тема 17. Презентація та захист власного програмного продукту	2
Разом за 3 семестр		**34**
Семестр 4		
1	Тема 1. Структуровані типи даних	2
2	Тема 2. Динамічна програмна пам'ять. Вказівники.	2
3	Тема 3. Динамічний розподіл пам'яті для одновимірних масивів	2
4	Тема 4. Динамічний розподіл пам'яті для двовимірних масивів	2
5	Тема 5. Розподіл пам'яті для структур та масивів структур	2
6	Тема 6. Процедурна парадигма. Створення та використання функцій.	2
7	Тема 7. Функції зі скалярними параметрами	2
8	Тема 8. Функції з адресними параметрами	2
9	Тема 9. Область дії змінних. Правила використання параметрів.	2
10	Тема 10. Поняття рекурсії. Організація рекурсивних обчислень.	2
11	Тема 11. Динамічні структури даних. Списки.	2
12	Тема 12. Маніпуляції даними в списках. Вилучення та включення елементів	2
13	Тема 13. Створення систематизованих списків	2
14	Тема 14. Ієрархічні структури даних. Створення та обробка дерев	2
15	Тема 15. Рекурсивні методи обробки ієрархічних структур даних	2

16	Тема 16. Обробка файлової інформації. Загальні засоби управління роботи з файлами	2
17	Тема 17. Засоби та методи обробки текстових файлів	2
18	Тема 18. Засоби та методи обробки файлів прямого доступу	2
Разом за семестр 4		**36**

5. Самостійна робота

№ з/п	Назви тем	Кількість годин
Семестр 3		
1	Тема 1. Від програми до програмного продукту. Проєктування власного проєкту	3,1
2	Тема 2. Поглиблені можливості Python. Сучасний стиль програмування	3,1
3	Тема 3. Функції та декомпозиція складних програм	3,1
4	Тема 4. Об’єктно-орієнтоване програмування в Python	3,1
5	Тема 5. Наслідування, композиція та поліморфізм. Проєктування класів	3,1
6	Тема 6. Модулі, пакети та архітектура Python-проєкту	3,1
7	Тема 7. Обробка помилок та винятків. Надійність програм	3,1
8	Тема 8. Робота зі структурованими даними: JSON та CSV	3,1
9	Тема 9. Бази даних та SQL у Python	3,1
10	Тема 10. Алгоритми та структури даних. Ефективність програм	3,1
11	Тема 11. Пошук, сортування та оптимізація роботи з даними	3
12	Тема 12. Тестування, налагодження та якість програмного коду	3
13	Тема 13. API та мережеві запити. Взаємодія Python із зовнішніми сервісами	3
14	Тема 14. Графічний інтерфейс користувача Python-програми	3
15	Тема 15. Архітектура та інтеграція компонентів власного проєкту	3
16	Тема 16. Розроблення, тестування та вдосконалення власного програмного продукту	3
17	Тема 17. Презентація та захист власного програмного продукту	3
Разом за 3 семестр		**52**
Семестр 4		
1	Тема 1. Структуровані типи даних	4
2	Тема 2. Динамічна програмна пам'ять. Вказівники.	4
3	Тема 3. Динамічний розподіл пам'яті для одновимірних масивів	4
4	Тема 4. Динамічний розподіл пам'яті для двовимірних масивів	4
5	Тема 5. Розподіл пам'яті для структур та масивів структур	4
6	Тема 6. Процедурна парадигма. Створення та використання функцій.	4
7	Тема 7. Функції зі скалярними параметрами	4
8	Тема 8. Функції з адресними параметрами	4
9	Тема 9. Область дії змінних. Правила використання параметрів.	4
10	Тема 10. Поняття рекурсії. Організація рекурсивних обчислень.	4
11	Тема 11. Динамічні структури даних. Списки.	4
12	Тема 12. Маніпуляції даними в списках. Вилучення та включення елементів	4
13	Тема 13. Створення систематизованих списків	4
14	Тема 14. Ієрархічні структури даних. Створення та обробка дерев	4
15	Тема 15. Рекурсивні методи обробки ієрархічних структур даних	4

16	Тема 16. Обробка файлової інформації. Загальні засоби управління роботи з файлами	6
17	Тема 17. Засоби та методи обробки текстових файлів	6
18	Тема 18. Засоби та методи обробки файлів прямого доступу	6
Разом за семестр 4		**78**

Принципи організації самостійної роботи

−використання міждисциплінарних зв’язків з уже вивченими курсами;

−робота з фрагментами першоджерел, зокрема, й англомовними;

−втілення результатів виконання роботи у різних формах творчих завдань;

−діалогічність і ідейний взаємообмін (зокрема, позаурочний) студентів щодо змісту самосійної роботи; взаємне збагачення студентами знань одне одного через представлення власних тематичних траєкторій.

6. Методи навчання

При вивченні дисципліни передбачене проведення лекцій та лабораторних занять. Самостійна робота, пов’язана з опрацюванням відповідного теоретичного та практичного матеріалу, може супроводжуватись підготовкою конспектів, розв’язуванням завдань домашніх , виконанням лабораторних робіт, підготовкою програмних проєктів.

Викладання дисципліни передбачає поєднання основних традиційних та інноваційних методів навчання з елементами електронного консультування, що передбачає використання сучасних інформаційних технологій, платформ та забезпечує засвоєння програмних результатів навчання.

Для досягнення освітньої мети й прогнозованих результатів навчання у дисципліні «Комп’ютерні науки» використовуються такі технології та методи навчання:

1. Пояснювально-ілюстративний метод (виклад теоретичного матеріалу на лекціях, ілюстрація методів розв’язання завдань та прикладів застосування в професійній сфері, показ презентацій).

2. Метод проблемного навчання, що передбачає засвоєння студентами фундаментальних знань з дисципліни та пошуку способів розв'язання поставлених на лекції завдань.

3. Репродуктивний метод, що передбачає безпосереднє застосування набутих базових знань до демонстрації практичних умінь та навичок при розв'язуванні поставлених завдань.

4. Частково-пошуковий метод – організація пошуку об’єктів дослідження, розв'язування запропонованих викладачем завдань.

5. Дослідницький метод, що передбачає пошук розв'язку практичних завдань дисципліни з можливістю консультацій з викладачем.

6. Використання мультимедійних та хмарних технологій при організації комунікації з викладачем чи при онлайн-навчанні, а також використання платформи для електронного навчання Moodle (https://moodle.chnu.edu.ua).

7. Групові та індивідуальні методи роботи на практичних заняттях та при виконанні завдань самостійної роботи, формуванням напрацювань для виконання і захисту лабораторних робіт.

9. Застосування електронних курсів та ресурсів, а також платформи для електронного навчання Moodle (https://moodle.chnu.edu.ua).

Отримані навички та знання в подальшому будуть використані під час виконання завдань практичного характеру.

Політика академічної доброчесності

Дотримання політики щодо академічної доброчесності учасниками освітнього процесу при вивченні освітнього компонента регламентовано чинними документами ВСП «Фаховий коледж ЧНУ»:

Етичний кодекс Відокремленого структурного підрозділу «Фаховий коледж Чернівецького національного університету імені Юрія Федьковича»
Положенням про виявлення та запобігання академічному плагіату у Відокремленому структурному підрозділі «Фаховий коледж Чернівецького національного університеті імені Юрія Федьковича»
Політика використання штучного інтелекту в Чернівецькому національному університету імені Юрія Федьковича

Використання здобувачами фахової передвишої освіти інструментів штучного інтелекту (ШІ) під час виконання навчальних завдань:

- *допускається за умови* дотримання принципів академічної доброчесності (здобувач освіти несе відповідальність за зміст, достовірність і результати поданої роботи, а у разі використання ШІ зобов'язаний зазначити факт і спосіб його застосування);
- *не допускається* використання ШІ як заміни самостійного виконання завдань або з порушенням інших вимог академічної доброчесності.

7. Система контролю та оцінювання

Методи контролю

Методи та засоби оцінювання та демонстрування результатів навчання даної дисципліни наступні:
демонстрація роботи розроблених програмних засобів згідно завдань лабораторних робіт;
розрахункові, графічні та схематичні викладки та аналітичні звіти щодо виконаних лабораторних завдань;
інші види індивідуальних та групових завдань.
У разі проведення навчального процесу та оцінювання в дистанційній формі використовуються засоби Moodle (у тому числі тестування).
Основними формами контролю самостійної роботи є:
проведення екзаменів;
тестування;
перевірка домашнього виконання лабораторних завдань підвищеної складності;
письмові чи усні опитування студентів;
захист лабораторних завдань в режимі тестування програм.
Форми контролю
Даною дисципліною передбачені наступні форми поточного контролю:
усне опитування (фронтальне, індивідуальне та комбіноване), захист завдань практичного характеру, а також усні відповіді студентів;
письмове опитування (модульні лабораторні завдання (за останньою темою модуля), тестування;
Форма проміжного контролю після третього семестру – екзамен.
Форма підсумкового контролю – екзамен.

Розподіл балів, які отримують студенти

<table>
<tr><td colspan="19">Семестр 3</td></tr>
<tr><td colspan="15">Поточне тестування та самостійна робота Всього 50</td><td colspan="2">Екзамен</td><td>Сума</td></tr>
<tr><td colspan="5">Модуль 1
12</td><td colspan="5">Модуль 2
14</td><td colspan="5">Модуль 3
14</td><td colspan="4">Індивідуальне науково дослідне завдання 10</td><td colspan="2">50</td><td>100</td></tr>
<tr><td colspan="19">Семестр 4</td></tr>
<tr><td colspan="18">Поточне оцінювання
(аудиторна та самостійна робота)</td><td rowspan="2">Кількість балів</td><td rowspan="2">Кількість балів (екзамен)</td><td rowspan="2">Сумарна</td></tr>
<tr><td colspan="9">Модуль №1</td><td colspan="9">Модуль №2</td></tr>
<tr><td>Т1</td><td>Т2</td><td>Т3</td><td>Т4</td><td>Т5</td><td>Т6</td><td>Т7</td><td>Т8</td><td>Т9</td><td>Т10</td><td>Т11</td><td>Т12</td><td>Т13</td><td>Т14</td><td>Т15</td><td>Т16</td><td>Т17</td><td>Т18</td><td rowspan="2">50</td><td rowspan="2">50</td><td rowspan="2">100</td></tr>
<tr><td>2</td><td>2</td><td>2</td><td>2</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td><td>3</td></tr>
</table>

Критерії оцінювання відповідей студентів

Рівень знань	Бали	Критерії оцінювання відповіді
Високий (A) відмінно	90 -100 % від максимальної кількості встановлених балів	Відповідь студента: - містить повний, розгорнутий, правильний виклад матеріалу з поставленого питання; - демонструє знання основних понять і категорій та взаємозв'язку між ними, правильне розуміння змісту основних теоретичних положень; - вказує на вміння давати змістовний та логічний аналіз матеріалу з поставленого питання; - містить послідовний та аргументований розв'язок задачі (завдання), правильно зроблені розрахунки до задачі; - здатність робити власні висновки в разі неоднозначності, спірного чи проблемного характеру поставленого питання чи проблеми.
Хороший (B) добре	80-89 % від максимальної кількості встановлених балів	Студент дав досить змістовну відповідь на поставлене питання, але відповідь містить наступні недоліки: - недостатня повнота, незначні неточності чи прогалини при поясненні того чи іншого аспекту питання;
Достатній (C) добре	70-79 % від максимальної кількості встановлених балів	Студент дав досить змістовну відповідь на поставлене питання, але відповідь містить наступні недоліки: - окремі формулювання є нечіткими, міститься інформація, котра не відноситься до змісту екзаменаційного питання; - алгоритм розв'язку задачі є правильним, однак допущені помилки при розрахунках.
Середній (D) задовільно	60-69 % від максимальної кількості встановлених балів	Студент дав відповідь на поставлене питання, однак допустив суттєві помилки як змістовного характеру, так і при відповіді на питання: - зміст відповіді свідчить про прогалини у знаннях з відповідного питання або ж про неправильне розуміння окремих аспектів поставленого питання; - порушено алгоритм розв'язку задачі і/або присутні помилки при розрахунках.

Нижче середнього (E) задовільно	50-59 % від максимальної кількості встановлених балів	Студент дав відповідь на поставлене питання, однак допустив суттєві помилки як змістовного характеру, так і при відповіді на питання: - зміст відповіді свідчить про прогалини у знаннях з відповідного питання або ж про неправильне розуміння окремих аспектів поставленого питання; - відповідь викладена недостатньо аргументовано та/або з порушенням правил логіки при поданні матеріалу; - порушено алгоритм розв'язку задачі і/або присутні помилки при розрахунках.
Початковий (E) незадовільно	0-49 % від максимальної кількості встановлених балів	Студент взагалі не відповів на питання, або його відповідь є неправильною, тобто містить грубі змістовні помилки щодо принципових аспектів поставленого питання. Аргументація відсутня взагалі (є безсистемною чи алогічною). Задачі розв'язані неправильно.

Критерії оцінювання результатів навчання з навчальної дисципліни

Шкала оцінювання: національна та ЄКТС

<table>
<tr><td rowspan="2">Сума балів за всі види навчальної діяльності</td><td rowspan="2">Оцінка ЄКТС</td><td colspan="2">Оцінка за національною шкалою</td></tr>
<tr><td>для екзамену, курсового проекту (роботи), практики</td><td>для заліку</td></tr>
<tr><td>90-100</td><td>A</td><td>відмінно</td><td rowspan="5">Зараховано</td></tr>
<tr><td>80-89</td><td>B</td><td rowspan="2">добре</td></tr>
<tr><td>70-79</td><td>C</td></tr>
<tr><td>60-69</td><td>D</td><td rowspan="2">задовільно</td></tr>
<tr><td>50-59</td><td>E</td></tr>
<tr><td>35-49</td><td>FX</td><td>незадовільно з можливістю повторного складання</td><td>не зараховано з можливістю повторного складання</td></tr>
<tr><td>0-34</td><td>F</td><td>незадовільно з обов’язковим самостійним повторним вивченням дисципліни</td><td>не зараховано з обов’язковим самостійним повторним вивченням дисципліни</td></tr>
</table>

9. Перелік питань екзаменів

Семестр 3

1. Від програми до програмного продукту.
2. Проєктування програмного продукту та визначення вимог.
3. Постановка задачі та межі програмного проєкту.
4. Декомпозиція та проєктування складної задачі.
5. Основні можливості сучасного Python.
6. Функціональні та компактні конструкції Python.
7. Читабельність, стиль та сучасна організація Python-коду.
8. Функції як засіб декомпозиції програм.
9. Області видимості, LEGB та побічні ефекти.
10. Чисті функції, єдина відповідальність та декомпозиція.
11. Рекурсія та ітеративні алгоритми.
12. Основні поняття об’єктно-орієнтованого програмування.
13. Інкапсуляція, властивості та спеціальні методи Python.
14. Моделювання предметної області за допомогою класів.
15. Наслідування та ієрархія класів.
16. Поліморфізм, абстракція та композиція.
17. Проєктування взаємодії класів.
18. Модулі, пакети та організація Python-проєкту.
19. Залежності між модулями та архітектура програми.
20. Розділення інтерфейсу, логіки та даних.
21. Обробка помилок та винятків у Python.
22. Надійність програми та проєктування обробки помилок.
23. Структуровані дані: CSV та JSON.
24. Вибір способу зберігання даних.
25. Реляційні бази даних та SQL.
26. Взаємодія Python із базою даних.
27. Ефективність алгоритмів та складність.
28. Структури даних та вибір оптимального способу представлення інформації.
29. Пошук, сортування та оптимізація алгоритмів.
30. Повний цикл розроблення, тестування та захисту власного програмного продукту.

Семестр 4

1. Особливості роботи з текстовими рядками.
2. Методи оголошення структурованих типів та даних.
3. Особливості роботи із структурованими даними. Операції зі структурами.
4. Вказівники на змінні та масиви. Операції зі вказівниками.
5. Динамічний розподіл пам’яті для одновимірних масивів.
6. Динамічний розподіл пам’яті для двовимірних масивів.
7. Динамічний розподіл пам’яті для масивів структур.
8. Підпрограми. Класичний метод оголошення.
9. Підпрограми. Сучасний метод оголошення. Область дії змінних.
10. Підпрограми. Параметри скалярного типу.
11. Підпрограми. Параметри типу одновимірний масив.
12. Підпрограми. Параметри типу двовимірний масив.
13. Підпрограми. Параметри адресного типу для скалярних даних.
14. Підпрограми. Адресні параметри для одновимірних масивів.
15. Підпрограми. Адресні параметри ** для двовимірних масивів.
16. Підпрограми. Адресні параметри для масивів структур.
17. Підпрограми. Рекурсивні функції.
18. Динамічні структури даних. Списки.
19. Динамічні структури даних. Черги та стеки.
20. Динамічні структури даних. Систематизовані списки.

21. Динамічні структури даних. Дерева. Бінарні дерева.
22. Бінарні дерева. Операції створення, пошуку інформації, перегляду.
23. Засоби роботи з файлами. Текстові файли.
24. Засоби роботи з файлами. Бінарні файли.
25. Засоби роботи з файлами. Файли прямого доступу.

10. Рекомендована література

Основна (базова)

1. Фратавчан В.Г., Фратавчан Т.М., Лазорик В.В. Алгоритмізація та програмування : навчальний посібник для закладів вищої освіти. Чернівці : ЧНУ, 2022. 286 с.
2. Програмування мовою С++: навчальний посібник / укл. Танасюк Ю.В. Чернівці : ЧНУ, 2022. 208 с.
3. Васильєв О. Програмування С++ в прикладах і задачах : навч. посібник / Васильєв О. Київ: Ліра К, 2019. 382 с.
4. Трофименко О. Г., Прокоп Ю. В., Задерейко О. В. Алгоритмізація та програмування : навчально-методичний посібник. Одеса : Фенікс, 2020. 310 с.
5. Козак Л. І., Костюк І. В., Стасевич С. П. Основи програмування: навчальний посібник – Львів : «Новий Світ-2000», 2020. 328с.
6. Караванова Т.П. Інформатика. Основи алгоритмізації та програмування: 777 задач з рекомендаціями та прикладами. Київ : "Генеза", 2006. 286 с.
7. Prata S. C++ Premier Plus. Kindle Edition, 2018. 1440 p.
8. Straustrup B. Tour of C++, 2nd edition. Addison-Wesley Professional, 2018. 256 p.
9. Meyers S. Effective modern C++: 42 specific ways to improve your use of C++11and C++14. O'Reilly Media, 2019. 334 p.
10. Deitel P., Deitel H. C++20 for programmers, 3rd edition. Pearson Education Limited, 2021. 1008 p.
11. 2. Josuttis N. M. C++20 – The complete guide. Leanpub, 2021.764 p.
12. Козак Л. І. Основи програмування : навчальний посібник / Л. І. Козак, І. В. Костюк, С. П. Стасевич. – Львів: "Новий Світ-2000", 2019. – 326 с
13. Ришковець Ю. В. Алгоритмізація та програмування. Ч. 1 : навчальний посібник / Ю. В. Ришковець, В. А. Висоцька. – Львів : "Новий Світ-2000", 2020. – 337 с.
14. Ришковець Ю. В. Алгоритмізація та програмування. Ч. 2 : навчальний посібник / Ю. В. Ришковець, В. А. Висоцька. – Львів : "Новий Світ-2000", 2020. – 314 с. – ц. 300,00 (10 екз.)

Допоміжна

1. Фратавчан В.Г., Фратавчан Т.М., Лазорик В.В. Алгоритмізація та програ-мування : методичні вказівки та завдання для лабораторних робіт (для сту-дентів спеціальностей «Комп'ютерні науки», «Програмна інженерія», «Комп'ютерна інженерія»). Чернівці : ЧНУ, 2022. 44 с.
2. Brian W. Kernighan, Dennis M. Ritchie, C Programming Language, 2nd Edition, Pearson, 2015, 288 h.
3. Bjarn Stroustrup, The C++ Programming Language (4th Edition). Addison-Wesley ISBN 978-0321563842, 2013, 1366 p..
4. Татарчук Д. Д., Діденко Ю. В. Програмування мовами С та С++: навчаль-ний посібник посіб.. Київ , 2012. 112 с.
5. Грицюк Ю.І, Рак Т.Є. Програмування мовою С++ : Навчальний посібник, Львів : ЛДУ БЖД, 2011р., 291 с.
6. Ковалюк Т.В.. Основи програмування. Київ : Вид. група BHV, 2005. 384 ст.
7. Bruneau Babet, Lean & Mean Borland C++, Brady, 1994. 471 p.

Інформаційні ресурси:

1. C++. Теорія та практика: навчальний посібник.
https://dut.edu.ua/ua/lib/1/category/1064/view/539
2. C++. Основи програмування. Теорія та практика: підручник.
https://dut.edu.ua/ua/lib/1/category/1064/view/538
3. Програмування. Поглиблений курс.
4. http://www.cyb.univ.kiev.ua/library/books/zubenko-omelchuk-2.pdf
5. The C++ Programming Language Fourth Edition Bjarne Stroustrup.
http://chenweixiang.github.io/docs/The_C++_Programming_Language_4th_Edition_Bjarne_Stroustrup.pdf
6. http://www.netacad.com
7. Програмування. Поглиблений курс.
http://www.cyb.univ.kiev.ua/library/books/zubenko-omelchuk-2.pdf