

Чернівецький національний університет імені Юрія Федьковича

(повне найменування закладу вищої освіти)

Факультет математики та інформатики

(назва факультету/навчально-наукового інституту)

Кафедра математичного моделювання

(назва кафедри)



Декан

Мартинюк О.В.

“ 25 ”

05

2024 року

РОБОЧА ПРОГРАМА

навчальної дисципліни

Теорія алгоритмів

(назва навчальної дисципліни)

обов'язкова

(вказати: обов'язкова / вибіркова)

Освітньо-професійна програма Системний аналіз

(назва програми)

Спеціальність 124 – Системний аналіз

(вказати: код, назва)

Галузь знань 12 – Інформаційні технології

(вказати: шифр, назва)

Рівень вищої освіти перший (бакалаврський)

(вказати: перший бакалаврський/другий магістерський)

факультет математики та інформатики

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

Мова навчання українська

(вказати: на якій мові викладається навчальна дисципліна)

Чернівці 2024 рік

Робоча програма навчальної дисципліни Теорія алгоритмів

(назва навчальної дисципліни)

складена відповідно до освітньо-професійної програми «Системний аналіз», затвердженої Вченою радою Чернівецького національного університету імені Юрія Федьковича (Протокол № 7 від 29.04.2024р.).

(назва освітньо-професійної програми, дата останнього затвердження)

Розробники: Караванова Тетяна Петрівна, асистент кафедри математичного моделювання

(П.І.Б. авторів, посада, науковий ступінь, вчене звання)

Погоджено з гарантом ОП і затверджено на засіданні кафедри математичного моделювання

Протокол № 18 від “ 25 ” червня 2024 року

Завідувач кафедри

(підпис)

Черевко І.М.

(прізвище та ініціали)

Схвалено методичною радою факультету математики та інформатики

Протокол № 11 від “25” червня 2024 року.

Голова методичної ради факультету
математики та інформатики

(підпис)

Сікора В.С.

(прізвище та ініціали)

1. Мета навчальної дисципліни:

- розвиток логічного, аналітичного мислення, а саме уміння використовувати індукцію, дедукцію, аналіз, синтез, робити висновки, узагальнення;
- формування теоретичної бази знань студентів щодо структур даних, побудови алгоритмів з використанням структур даних та їх реалізації мовою програмування;
- розвиток уміння розв'язувати алгоритмічні задачі, застосовуючи класичні базові алгоритми, математичний апарат, відповідну спеціальну літературу та програмне забезпечення;
- бачення студентами можливостей використання набутих знань у їх майбутній професії.

2. Результати навчання:

Загальні компетентності за ОП «Системний аналіз»

ЗК01. Здатність до абстрактного мислення, аналізу та синтезу.

ЗК10. Здатність працювати автономно.

Фахові компетентності за ОП «Системний аналіз»

ФК6. Здатність до комп'ютерної реалізації математичних моделей реальних систем і процесів; проектувати, застосовувати і супроводжувати програмні засоби моделювання, прийняття рішень, оптимізації, обробки інформації, інтелектуального аналізу даних.

Програмні результати навчання за ОП «Системний аналіз»

ПР2. Вміти використовувати стандартні схеми для розв'язання комбінаторних та логічних задач, що сформульовані природною мовою, застосовувати класичні алгоритми для перевірки властивостей та класифікації об'єктів, множин, відношень, графів, груп, кілець, решіток, булевих функцій тощо.

ПР9. Вміти створювати ефективні алгоритми для обчислювальних задач системного аналізу та систем підтримки прийняття рішень.

Студенти повинні знати:

- сутність поняття алгоритму, базових структур даних, пошуку інформації та її впорядкування, використання оптимізаційних методів для розв'язання задач на графах, динамічного програмування, жадібних алгоритмів, обчислювальної геометрії;
- фундаментальні оптимізаційні алгоритми з використанням структур даних: алгоритми на графах, динамічне програмування, жадібні алгоритми, обчислювальна геометрія;

- загальні принципи розв’язування алгоритмічних задач: постановка задачі, побудова алгоритму, реалізація алгоритму мовою програмування, тестування реалізованого алгоритму;
- методи розв’язання алгоритмічних задач.

Студенти повинні вміти та мати навички:

- застосовувати теоретичні знання щодо структур даних та фундаментальних оптимізаційних алгоритмів для розв’язування практичних завдань;
- використовувати навички роботи з інтегрованим середовищем програмування;
- аналізувати відомі методи побудови алгоритмів та визначати найоптимальніші з них для розв’язування конкретної задачі;
- розробляти власні тести для перевірки коректності розроблених алгоритмів та тестувати розроблені алгоритми;
- використовувати навички техніки програмування.

3. Опис навчальної дисципліни

3.1. Загальна інформація

Форма навчання	Рік підготовки	Семестр	Кількість		Кількість годин						Вид підсумкового контролю
			кредитів	годин	лекції	практичні	семінарські	лабораторні	самостійна робота	індивідуальні завдання	
Денна	2	3	5	150	30	–	–	45	75	–	Екзамен

3.2. Структура змісту навчальної дисципліни

Назви змістових модулів і тем навчальних занять	Кількість годин												
	денна форма							заочна форма					
	усього	у тому числі					усього	у тому числі					с.р.
		лекц.	прак./ семін.	лаб.	інд.	с.р.		лекц.	прак./ сем.	лаб.	інд.	с.р.	
1	2	3	4	5	6	7	8	9	10	11	12	13	
Теми навчальних занять (назва теми й основні питання)	Змістовий модуль 1. Базові структури даних та алгоритми обробки їх елементів												
Тема 1. Теорія алгоритмів. Основні поняття.	7	2				5							
Тема 2. Базові структури даних. Проста змінна. Масив. Стек. Черга.	11	2	2	4		5							

Тема 3. Динамічна структура даних зв'язний список	7	2			5						
Тема 4. Дерева. Бінарні дерева.	11	2	2	4	5						
Тема 5. Хеш- функції та хеш- таблиці.	7	2			5						
Тема 6. Пошукові алгоритми на деревах та у мережах.	11	2	2	4	5						
Разом за ЗМ1	60	12	6	12	30						
Теми навчальних занять (назва теми й основні питання)	Змістовий модуль 2. Оптимізаційні алгоритми з використанням структур даних										
Тема 7. Основні поняття теорії графів. Пошук в глибину та в ширину.	7	2			5						
Тема 8. Ейлерів та Гамільтонів графи.	11	2	2	4	5						
Тема 9. Остовне дерево. Мінімальне остовне дерево.	7	2			5						
Тема 10. Пошук найкоротшого шляху між двома вершинами у зваженому графі.	11	2	2	4	5						
Тема 11. Топологічне сортування.	7	2			5						
Тема 12. Основи динамічного програмування. Базові задачі динамічного програмування.	7	2			5						
Тема 13. Жадібні алгоритми.	11	2	3	6	5						

Тема 14. Основні поняття обчислювальної геометрії. Базові алгоритми обчислювальної геометрії.	7	2			5						
Тема 15. Побудова опуклої оболонки.	11	2	2	4	5						
Разом за ЗМ 2	90	18	9	18	45						
Усього годин	150	30	15	30	75						

3.3. Тематика лабораторних занять

№	Назва теми	Кількість годин
1	Структури даних «Стек», «Черга».	4
2	Структури даних «Зв'язний список», «Бінарні дерева»	4
3	Структура даних «хеш-таблиця». Пошукові алгоритми на деревах.	4
4	Пошукові алгоритми на графах. Ейлереві та Гамільтонові графи.	4
5	Алгоритми на зважених та орієнтованих графах.	4
6	Динамічне програмування. Жадібні алгоритми.	6
7	Реалізація алгоритмів побудови опуклої оболонки.	4

3.4. Самостійна робота студента

№	Назва теми	Кількість балів
1	Створення алгоритму. Оцінка складності та аналіз ефективності алгоритмів	3
2	Поняття про структури даних, обробка інформації. Стек. Черга.	3
3	Динамічний розподіл пам'яті. Зв'язний список.	3
4	Дерево як структура даних.	3
5	Пошукові алгоритми на деревах та у мережах.	3
6	Основні поняття теорії графів. Застосування алгоритмів на графах для розв'язування алгоритмічних задач.	3
7	Основні поняття динамічного програмування та жадібних алгоритмів. Застосування базових алгоритмів динамічного програмування та жадібних алгоритмів для розв'язування алгоритмічних задач.	3
8	Основні поняття обчислювальної геометрії. Застосування базових алгоритмів обчислювальної геометрії для розв'язування алгоритмічних задач.	3

4. Освітні технології, методи навчання і викладання навчальної дисципліни

Даний курс представляє собою розроблений комплекс презентацій лекцій, варіантів лабораторних робіт до курсу, методичних рекомендацій до їх виконання, інтерактивних можливостей здачі лабораторних робіт, консультування, що розміщений на платформі <https://moodle.chnu.edu.ua/>.

Під час викладання курсу застосовуються різноманітні методи навчання. Лекційні заняття проводяться аудиторно з використанням традиційних методик, а саме подання матеріалу студентам у вигляді лекції. Однак, такий формат передбачає застосування сучасних інформаційних технологій, а саме розроблених до всього курсу презентацій лекцій на платформі MS PowerPoint з використанням відповідних програмних та технічних засобів.

Під час знайомства з лекційним матеріалом використовуються інтерактивні форми навчальної діяльності, а саме на початку кожної лекції розглядаються питання щодо зв'язку попереднього матеріалу з поточним, випереджальні логічні запитання, щодо подальшого викладення матеріалу (наступний слайд), запитання щодо практичного використання алгоритмів, що розглядаються тощо.

Ключовим елементом курсу є індивідуальний захист лабораторних робіт. Усі завдання лабораторних є авторськими і передбачають реалізацію коду програми відповідно до розробленого оптимізаційного алгоритму, підготовку комплексу власних тестів, що демонструють коректність і оптимальність розв'язаної задачі. Під час захисту студенти мають продемонструвати достатній і високий рівень програмування, розуміння матеріалу, якого стосується дане лабораторне завдання, довести достатність і повноту комплексу підібраних тестів.

Програмою курсу передбачено надання онлайн консультацій студентам. Також у разі непередбачуваних ситуацій (повітряна тривога тощо) заняття можуть проводитися дистанційно за вказаним за сайті курсу посиланням.

За бажанням з усіма презентаціями студенти мають можливість ознайомитися самостійно, що дозволяє їм достроково виконувати лабораторні завдання.

Вся активна початкова діяльність студентів заохочується бонусними балами.

5. Критерії та засоби оцінювання результатів навчання з навчальної дисципліни

5.1. Критерії підсумкового оцінювання.

На підсумковому модулі (іспиті) студентам пропонуються білети, які містять три завдання. Два з них – теоретичні (кожне оцінюється в 10 балів), третє завдання передбачає розв'язання задачі на одну з тем «Структури даних», «Алгоритми на графах», «Алгоритми динамічного програмування», «Жадібні алгоритми», «Алгоритми обчислювальної геометрії» яке також оцінюється у 10 балів.

Диференціація оцінювання відповідей студентів здійснюється за таким алгоритмом: 100% балів - студент обирає усі правильні відповіді; 80% балів - студент обирає більшість правильних відповідей; 60% балів - студент обирає біля половини правильних відповідей; 40% балів - студент обирає менше половини правильних відповідей; 20% балів - студент не обирає практично жодної правильної відповіді.

5.2. Шкала оцінювання: національна та ECTS

Оцінка за національною шкалою	Оцінка за шкалою ECTS	
	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 повторним курсом

5.3. Засоби оцінювання

Засобами оцінювання та демонстрування результатів навчання є:

- усне та письмове опитування;
- виконання практичних та лабораторних робіт.

6. Форми поточного та підсумкового контролю

Поточний контроль знань відбувається протягом лабораторних занять шляхом усного та письмового опитування теоретичних основ теми, виконання лабораторних завдань, а також у вигляді контрольних робіт та індивідуальних завдань.

Студенти можуть отримати додаткові бали (до 10) після виконання індивідуальних навчально-дослідницьких завдань. Ці завдання видає та перевіряє лектор. Додаткові бали враховуватимуться під час здачі іспиту.

Теоретичне опитування може здійснюватися на лекціях. Студент, який повно й вичерпно відповів на питання лектора, отримує додатково 1 бал.

Форма підсумкового контролю – екзамен.

7. Рекомендована література

7.1. Фахова (основна)

1. Jeffrey D. Aho, Alfred V. Hopcroft, John E. Ullman. Data Structures and Algorithms, Addison-Wesley, 1983 – 427 pages

2. Niklaus Wirth Algorithms and Data Structures — Publisher: Pearson Education, 2004. — 179 pages.
3. Donald E. Knuth The Art of Computer Programming, Volumes 1-4b — Publisher: Addison-Wesley Professional, 2022.
4. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction To Algorithms — MIT Press, 2001 - 1180 pages.
5. Tim Roughgarden Algorithms Illuminated: Part 1: The Basics — Publisher: Soundlikeyourself Publishing, LLC; Illustrated edition, 2017. — 235 pages.
6. Tim Roughgarden Algorithms Illuminated: (Part 2): Graph Algorithms and Data Structures — Publisher: Soundlikeyourself Publishing; Illustrated edition, 2018. — 226 pages.
7. Tim Roughgarden Algorithms Illuminated: (Part 3): Greedy Algorithms and Dynamic Programming — Publisher: Soundlikeyourself Publishing, LLC; Illustrated edition, 2019. — 232 pages.
8. Tim Roughgarden Algorithms Illuminated: (Part 4): Algorithms for NP-Hard Problems — Publisher: Soundlikeyourself Publishing, LLC, 2020. — 273 pages.
9. Караванова Т.П. Методи побудови алгоритмів та їх аналіз: необчислювальні алгоритми: Навч. посіб. — К.: Генеза, 2007. — 224 с.: іл.
10. Караванова Т.П. Методи побудови алгоритмів та їх аналіз: обчислювальні алгоритми: Навч. посіб. — К.: Генеза, 2008. — 336 с.: іл.
11. Караванова Т.П. Теорія алгоритмів. Частина 1. Необчислювальні алгоритми : навч. посіб. / Т.П. Караванова. Чернівці : Чернівець. нац. ун-т ім. Ю. Федьковича. 2022. 268 с.
12. Караванова Т.П. Теорія алгоритмів. Частина 2. Обчислювальні алгоритми : навч. посіб. / Т.П. Караванова. Чернівці : Чернівець. нац. ун-т ім. Ю. Федьковича. 2022. 288 с.
13. Структура даних [http://uk.wikipedia.org/wiki/Структура даних](http://uk.wikipedia.org/wiki/Структура_даних)

7.2. Допоміжна

1. Караванова Т.П. Інформатика: основи алгоритмізації та програмув.: 777 задач з рек. та прикл.: Навч. посіб. для 8-9 кл. із поглибленим вивч. інф-ки / За заг. ред. М.З.Згуровського — К.: Генеза, 2012. — 286 с.: іл. — Бібліограф.: 286 с.
2. Караванова Т.П. Методика розв'язування алгоритмічних задач. Основи алгоритмізації та програмування: Навчально-методичний посібник для вчителів/Т.П.Караванова. — Кам'янець-Подільський: Аксіома, 2013. — 460с.
3. Караванова Т.П. Методика розв'язування алгоритмічних задач. Побудова алгоритмів: Навчально-методичний посібник для вчителів / Т.П.Караванова. — Кам'янець-Подільський: Аксіома, 2014. — 344 с.

8. Інформаційні ресурси

1. https://n-knigi.com.ua/p1635080348-iskusstvo-programmirovaniya-tom.html?source=merchant_center&gclid=Cj0KCQjw2MWVBhCQARIsAIjbw

[oO7iJrOZhaEOSNEgl7y10_4iiN2qqeC1F1o_XcvCOqtjCoEnslQ58MaAlZREA
Lw_wcB](https://n-knigi.com.ua/ua/p1358924896-algoritmy-postroenie-analiz.html)

2. [http://uk.wikipedia.org/wiki/Структура даних](http://uk.wikipedia.org/wiki/Структура_даних)
3. <https://n-knigi.com.ua/ua/p1358924896-algoritmy-postroenie-analiz.html>
4. <https://n-knigi.com.ua/p1358926396-algoritmy-postroenie-analiz.html>

Додатково

(для контролю та самоконтролю роботи студента)

Розподіл балів, які отримують студенти

Поточне оцінювання (<i>аудиторна та самостійна робота</i>)							Кількість балів (екзамен)	Сумарна к-ть балів
Змістовий модуль №1			Змістовий модуль №2				30	100
T1	T2	T3	T4	T5	T6	T7		
10	10	10	10	10	10	10		

T1, T2 ... T7 – теми змістових модулів