

Міністерство освіти і науки України
Харківський національний університет імені В.Н. Каразіна
Кафедра прикладної математики

“ЗАТВЕРДЖУЮ”

Проректор
з науково-педагогічної роботи
Антон ПАНТЕЛЕЙМОНОВ

“ 31 ”



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Алгоритми і структури даних

(назва навчальної дисципліни)

рівень вищої освіти бакалавр

галузь знань 11 – Математика та статистика

спеціальність 113 – Прикладна математика
(шифр і назва)

освітня програма _____
(шифр і назва)

спеціалізація _____
(шифр і назва)

вид дисципліни обов'язкова
(обов'язкова / за вибором)

факультет математики і інформатики

2020 / 2021 навчальний рік

Програму рекомендовано до затвердження вченою радою факультету математики і інформатики

“31” серпня 2020 року, протокол №8

РОЗРОБНИКИ ПРОГРАМИ:

Ігнатович Світлана Юріївна, доктор фізико-математичних наук, доцент, професор кафедри прикладної математики.

Програму схвалено на засіданні кафедри Прикладної математики

Протокол від “31” серпня 2020 року №12

Завідувач кафедри Прикладної математики

 Валерій КОРОБОВ
(підпис) (прізвище та ініціали)

Програму погоджено з гарантом освітньої (освітньо-професійної) програми (керівником проектної групи) Прикладна математика
назва освітньої програми

Гарант освітньо-професійної програми
(керівник проектної групи) Прикладна математика

 Світлана ІГНАТОВИЧ
(підпис) (прізвище та ініціали)

Програму погоджено науково-методичною комісією факультету математики і інформатики Харківського національного університету імені В.Н. Каразіна
назва факультету, для здобувачів вищої освіти якого викладається навчальна дисципліна

Протокол від “31” серпня 2020 року, протокол № 1

Голова науково-методичної комісії факультету математики і інформатики

 Ольга АНОЩЕНКО
(підпис) (прізвище та ініціали)

ВСТУП

Програма навчальної дисципліни «**Алгоритми і структури даних**» складена відповідно до освітньо-професійної (освітньо-наукової) програми підготовки

бакалавр

(назва рівня вищої освіти)

спеціальності (напрямку) 113 – Прикладна математика

спеціалізації _____

1. Опис навчальної дисципліни

1.1. Метою викладання навчальної дисципліни є надання майбутнім спеціалістам теоретичних знань та практичних навичок з основ розробки і аналізу алгоритмів.

1.2. Основні завдання вивчення дисципліни

1. Ознайомлення з основними поняттями і методами аналізу алгоритмів. Ознайомлення з деякими найбільш важливими алгоритмами.
2. Ознайомлення з основними типами структур даних та їх використанням для створення ефективних алгоритмів.
3. Застосування мови Python для розв'язання практичних задач зі створення і аналізу алгоритмів.

1.3. Кількість кредитів – 4

1.4. Загальна кількість годин – 120

1.5. Характеристика навчальної дисципліни	
Обов'язкова	
Денна форма навчання	Заочна (дистанційна) форма навчання
Рік підготовки	
2-й	
Семестр	
3-й	
Лекції	
32 год.	.
Практичні, семінарські заняття	
32 год.	.
Лабораторні заняття	
Самостійна робота	
56 год.	.
у тому числі індивідуальні завдання	
12 год.	

1.6. Заплановані результати навчання

Знати :

1. Методи створення і аналізу алгоритмів, типи структур даних, деякі алгоритми пошуку і сортування.
2. Основи синтаксису і особливості мови Python.

Вміти:

1. Запропонувати алгоритм для розв'язання задачі, записати його за допомогою псевдокоду, провести аналіз складності алгоритму, довести його коректність.
2. Написати програму мовою Python, яка реалізує запропонований алгоритм.

2. Тематичний план навчальної дисципліни.**Розділ 1. Алгоритми***Тема 1. Вступ до мови Python.*

Особливості мови Python, вбудовані типи даних, особливості синтаксису, підключення модулів. Приклади програм.

Тема 2. Аналіз алгоритмів

Запис алгоритму у вигляді псевдокоду. Обчислювальна складність алгоритмів, знаходження обчислювальної складності. Коректність алгоритмів. Методи доведення коректності.

Тема 3. Рекурсивні та ітеративні алгоритми.

Рекурсивні алгоритми та їх аналіз. Особливості розподілу пам'яті при реалізації рекурсивних алгоритмів. Приклади рекурсивних алгоритмів. Хвостова рекурсія. Жадібні алгоритми, динамічне програмування.

Тема 4. Пошук і сортування

Алгоритми пошуку і сортування. Аналіз складності, приклади.

Розділ 2. Структури даних*Тема 5. Список, дерево, стек, черга.*

Основні структури даних: список, дерево, стек, черга. Особливості і приклади використання.

Тема 6. Множина, словник, хеш-таблиця, граф.

Структури даних множина, словник, хеш-таблиця. Особливості і приклади використання.

3. Структура навчальної дисципліни

Назви модулів і тем	Кількість годин					
	Денна форма					
	Усього	у тому числі				
л		п	лаб	інд	ср	
1	2	3	4	5	6	7
Розділ 1. Алгоритми						
Тема 1. Вступ до мови Python	16	4	4			8
Тема 2. Аналіз алгоритмів	24	8	8			8
<i>Індивідуальне завдання №1</i>	4					4
Тема 3. Рекурсивні та ітеративні алгоритми	22	8	6			8
Тема 4. Пошук і сортування	18	4	6			8
<i>Індивідуальне завдання №2</i>	4					4
Разом за розділом 1	88	24	24			40

Розділ 2. Структури даних						
Тема 5. Список, дерево, стек, черга	14	4	4			6
Тема 6. Множина, словник, хеш-таблиця, граф	14	4	4			6
<i>Індивідуальне завдання №3</i>	4					4
Разом за розділом 3	32	8	8			16
Усього годин	120	32	32			56

4. Тематики практичних занять

№ з/п	Назва теми	Кількість годин
1	Знайомство з мовою Python, написання найпростіших програм.	4
2	Приклади алгоритмів, їх запис, складання програм. Цикли. Реалізація алгоритмів на мові Python з використанням списків (list).	4
3	Аналіз обчислювальної складності алгоритмів і доведення їх коректності	4
4	Рекурсивні алгоритми. Реалізація рекурсивних алгоритмів на мові Python.	4
5	Деякі оптимізаційні алгоритми: жадібний алгоритм, метод динамічного програмування.	2
6	Алгоритми пошуку і сортування і їх реалізація за допомогою мови Python.	2
7	Бібліотека Turtle і візуалізація алгоритмів сортування.	4
8	Реалізація типів даних за допомогою вбудованих типів даних у мові Python.	2
9	Класи у мові Python. Створення і використання класів для моделювання структур даних.	2
10	Графи і їх застосування для розв'язання задач пошуку і оптимізації.	4
Разом		32

5. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
1	Виконання домашніх завдань з ознайомлення з особливостями синтаксису і написання найпростіших програм на мові Python.	8
2	Виконання домашніх завдань з аналізу обчислювальної складності і доведення коректності алгоритмів.	8
3	Виконання домашніх завдань з написання та аналізу програм мовою Python, які використовують рекурсивні функції.	4
4	Виконання домашніх завдань з написання та аналізу програм для задач оптимізації.	4
5	Виконання домашніх завдань з алгоритмів пошуку і сортування.	4
6	Виконання домашнього завдання з використання модуля Turtle для візуалізації.	4
7	Виконання домашніх завдань зі створення і використання класів на мові Python.	6
8	Виконання домашніх завдань з графів і їх використання.	6
9	Виконання індивідуальних завдань.	12
Разом		56

6. Індивідуальні завдання

Завдання №1: обрати одну із запропонованих задач, скласти алгоритм, записати його у вигляді псевдокоду, довести коректність, проаналізувати обчислювальну складність, скласти програму на мові Python, протестувати її і знайти час виконання.

Завдання №2: обрати метод сортування, написати програму на мові Python з використанням модуля Turtle для візуалізації.

Завдання №3: Реалізувати тип даних на мові Python.

7. Методи навчання

Лекційні та практичні заняття проводяться аудиторно. У разі оголошення карантину заняття проводяться відповідно до Наказу ректора Харківського національного університету імені В.Н. Каразіна (аудиторно або дистанційно за допомогою платформ Google Meet або Zoom).

8. Методи контролю

1. Перевірка виконання домашніх завдань, поточне опитування за лекційним матеріалом.
2. Перевірка індивідуальних завдань (розрахунково-графічні роботи).
3. Перевірка залікової роботи.

9. Схема нарахування балів

Поточний контроль, самостійна робота, індивідуальні завдання						Залікова робота	Сума
Розділ 1		Розділ 2	Індивідуальні завдання				
T1-T2	T3-T4	T5-T6	№1	№2	№3		
10	10	10	10	10	10	40	100

T1 – T6 – теми розділів.

Шкала оцінювання

Сума балів за всі види навчальної діяльності протягом семестру	Оцінка	
	для чотирирівневої шкали оцінювання	для дворівневої шкали оцінювання
90 – 100	відмінно	зараховано
70-89	добре	
50-69	задовільно	
1-49	незадовільно	

Критерії оцінювання навчальних досягнень

Оцінка в балах	Оцінка за національною шкалою	
Оцінка	Пояснення	
90 – 100	Зараховано	Теоретичний зміст курсу освоєний цілком, необхідні практичні навички роботи з освоєним матеріалом сформовані, всі навчальні завдання, які передбачені програмою навчання, виконані в повному обсязі, відмінна робота без помилок або з однією незначною помилкою.
70 – 89	зараховано	Теоретичний зміст курсу освоєний цілком, практичні навички роботи з освоєним матеріалом в основному сформовані, всі навчальні завдання, які передбачені програмою навчання, виконані, якість виконання жодного з них не оцінено мінімальним числом балів, деякі види завдань виконані з помилками, робота з декількома незначними помилками, або з однією – двома значними помилками.

50 –69	зараховано	Теоретичний зміст курсу освоєний не повністю, але прогалини не носять істотного характеру, необхідні практичні навички роботи з освоєним матеріалом в основному сформовані, більшість передбачених програмою навчання навчальних завдань виконано, деякі з виконаних завдань містять помилки, робота з трьома значними помилками.
1–49	Не зараховано	Теоретичний зміст курсу не освоєно, необхідні практичні навички роботи не сформовані, виконані навчальні завдання містять грубі помилки, робота, що потребує повної переробки

10. Рекомендована література

Основна література

1. Д. Кнут, «Искусство программирования», Т. 1-3.
2. Н. Вирт, «Алгоритмы и структуры данных».
3. А. Ахо, Д. Ульман, Д. Хопкрофт, «Структуры данных и алгоритмы».
4. К. Штайн, Р. Ривест, Т. Кормен, Ч. Лейзерсон, «Алгоритмы: построение и анализ».
5. А. Шень, «Программирование: теоремы и задачи».

Допоміжна література

1. С. Дасгупта, Х. Пападимитриу, У. Вазирани, «Алгоритмы».
2. Р. Седжвик, К. Уэйн, Р. Дондеро, Программирование на языке Python: учебный курс. 2017.
3. М. Т. Goodrich, R. Tamassia, М. Н. Goldwasser, Data Structures and Algorithms in Python. 2013.

11. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

1. www.python.org
2. <http://www.pythontutor.com>