

КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інформаційних систем та технологій

«ЗАТВЕРДЖУЮ»
Заступник декана з навчально-виховної роботи
Наталія ТМЄНОВА
(підпис)
«__» __ 2024 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Алгоритми та структури даних

для студентів

галузь знань 12 «Інформаційні технології»
спеціальність 126 «Інформаційні системи та технології»
освітній рівень бакалавр
Освітня програма Програмні технології інтернет речей
вид дисципліни обов'язкова

Форма навчання	денна
Навчальний рік	2024/2025
Семестр	2
Кількість кредитів ECTS	5
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладач: Володимир Дружинін, д.т.н, професор

Пролонговано: на 20__/20__ н.р. (_____) «__» 20__ р.
(підпис, ПІБ, дата)
на 20__/20__ н.р. (_____) «__» 20__ р.
(підпис, ПІБ, дата)
на 20__/20__ н.р. (_____) «__» 20__ р.
(підпис, ПІБ, дата)

КИЇВ – 2024

Розробник: Володимир Дружинін, д.т.н, професор., завідувач кафедри інформаційних систем та технологій

ЗАТВЕРДЖЕНО

«___» _____ 20__ р.

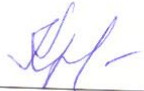
Завідувач кафедри інформаційних систем та технологій

 д.т.н., проф. В. Дружинін

Протокол № 10-24/24 від «27» червня 2024 року

Схвалено науково - методичною комісією факультету інформаційних технологій

Протокол № 9 від «27» червня 2024 року

Голова науково-методичної комісії ФІТ  (Ганна КРАСОВСЬКА)

«___» _____ 20__ року

Пропраць перевірено



ВСТУП

1. Мета дисципліни:

Вивчення методів аналізу складності алгоритмів. Ознайомлення з поняттям динамічної структури даних, а також абстрактного типу даних. Ознайомлення із поняттям спискової структури, оволодіння та вивчення реалізацій основних динамічних структур даних на основі спискових структур; ознайомлення з поняттями теорії графів, їх основних властивостей, а також вивчення, реалізацію і дослідження основних алгоритмів на графах; ознайомлення з поняттям асоціативних списків; вивчення та дослідження бінарних дерев пошуку та алгоритмів балансування бінарних дерев, вміння програмно реалізувати бінарні дерева на алгоритмічних мовах програмування (C++); вивчення та оволодіння основними методами сортування та пошуку для лінійних структур даних, зокрема, масивів; оволодіння методами хешування, принципами побудови хеш-таблиць.

2. Попередні вимоги до опанування або вибору навчальної дисципліни:

Вміти будувати блок-схеми алгоритмів, мати знання та навички розробки програм на алгоритмічних мовах програмування, таких як C++, вміти розв'язувати задачі із застосуванням таких структур даних, як одновимірні та двовимірні масиви, а також рядків. Володіти структурним підходом до розробки програм.

3. Анотація навчальної дисципліни:

Навчальна дисципліна «Алгоритми та структури даних» охоплює певне коло питань, які пов'язані з вивченням, дослідженням та реалізацією різних структур даних та алгоритмів для кожної зі структур, що розглядаються. Розглядаються такі теми: основи аналізу складності алгоритмів, абстрактні типи даних, динамічні типи даних на основі спискових структур, збалансовані бінарні дерева пошуку, методи побудови хеш-таблиць, основи теорії графів, методи реалізації структури даних для графів, алгоритми на графах, алгоритми сортування та пошуку для лінійних структур даних. Також в дисципліні «Алгоритми та структури даних» розглядаються способи реалізації алгоритмів та структур даних, що вивчаються на алгоритмічних мовах програмування. Дисципліна «Алгоритми та структури даних» є платформою для вивчення основних фахових дисциплін освітньої програми «Програмні технології інтернет речей» таких як: «Організація баз даних», «Технології програмування», «Об'єктно-орієнтоване програмування», «Програмування розподілених систем» тощо.

4. Завданням вивченням дисципліни «Алгоритми та структури даних» є теоретична та практична підготовка майбутніх фахівців з таких питань: вивчення методів аналізу складності алгоритмів, опанування різних структур даних, такі як спискові структури, масиви, асоціативні списки, графи, бінарні дерева, черга з пріоритетом, вивчення алгоритмів балансування бінарних дерев, методів пошуку та сортування, алгоритмів на графах, методів побудови хеш-таблиць тощо. Дисципліна підтримує та розширює набуті, а також надає нові навички сучасного рівня технологій інформаційних систем, методів та підходів розв'язування задач, архітектури обчислювальних систем, аналізу та моделювання програмного забезпечення, використовувати засоби реалізації інформаційних систем, які є невід'ємною частиною складних спеціалізованих задач та практичних проблем в області ICT та інтернету речей. Дисципліна надає можливість фахівцям: теоретичні знання із вибору, аналізу та дослідження алгоритмів та структур даних при вирішенні практичних завдань програмної обробки даних різного виду.

5. Результати навчання за дисципліною:

<i>Результат навчання</i> (1. Знати; 2. Вміти; 3. Комунікація; 4. Автономність та відповідальність)		<i>Форми (та / або методи і технології) викладання і навчання</i>	<i>Методи оцінювання та пороговий критерій оцінювання</i>	<i>Відсоток у підсумковій оцінці з дисципліни</i>
Код	Результат навчання			
1.1	Знати та розуміти методи аналізу та оцінки складності алгоритмів.	<i>Лекція, лабораторна робота</i>	<i>Індивідуальне завдання на виконання лабораторних робіт, іспит</i>	7%
1.2	Знати та вміти використовувати такі структури даних, як: спискові структури, графи, бінарні дерева пошуку, хеш-таблиці, черга з пріоритетом, масиви.			8%

1.3	Знати основи теорії графів.			8%
1.4	Знати поняття асоціативного списку, лінійного списку, черги з пріоритетом, масиву даних.			7%
2.1	Вміти реалізувати на алгоритмічних мовах програмування динамічні структури: стек, черга, дек, зв'язний список.			10%
2.2	Знати та вміти реалізувати та дослідити основні алгоритми для графів (зокрема, обходи, побудова остовного дерева тощо).			10%
2.3	Знати, вміти реалізувати та дослідити різні методи сортування та пошуку в масивах даних.			10%
2.4	Опанувати методи побудови хеш-таблиць.			10%
2.5	Знати та вміти реалізувати алгоритми балансування бінарних дерев пошуку.			10%
2.6	Вміти порівнювати методи та алгоритми для однотипних задач між собою.			5%
2.7	Вміти проводити топологічне сортування графів різними методами.			5%
3.1	Вміти розв'язувати прикладні математичні та комп'ютерні задачі із застосуванням різних обчислювальних методів та способів організації даних.			10%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання

Результати навчання дисципліни (код)	1.1	1.2	1.3	1.4	2.1	2.2	2.3	2.4	2.5	2.6	2.7	3.1
Програмні результати навчання (назва)												
ПР 2. Застосовувати знання фундаментальних і природничих наук, системного аналізу та технологій моделювання, стандартних алгоритмів та дискретного аналізу при розв'язанні задач проектування і використання інформаційних систем та технологій.								+		+		+
ПР 3. Використовувати базові знання інформатики й сучасних інформаційних систем та технологій, навички програмування, технології безпечної роботи в комп'ютерних мережах, методи створення баз даних та інтернет-ресурсів, технології розроблення алгоритмів і комп'ютерних програм мовами високого рівня із застосуванням об'єктно-орієнтованого програмування для розв'язання задач проектування і використання інформаційних систем та технологій.					+	+	+	+	+		+	
ПР 15. Демонструвати знання основних методів та технологій об'єктно-орієнтованого програмування, а також основ інших парадигм програмування, основ конструювання програмного забезпечення, раціональні алгоритми вирішення задач оптимізації та оптимального керування, уміння ставити конкретну прикладну задачу, знаходити оптимальні рішення за допомогою методів прийняття рішень.	+	+	+	+						+		
ПР 16. Будувати ефективні обчислювальні алгоритми для розрахункових задач, визначати ефективність програм за допомогою програмного					+	+	+	+	+		+	+

таблиці.

Студент не допускається до іспиту, якщо під час семестру в сумі набрав менше 36 балів. Оцінка за іспит не може бути меншою **24 балів** для отримання позитивної оцінки за іспит.

Підсумкова оцінка визначається шляхом підсумовування балів семестрової роботи та іспиту. Якщо у підсумку студент набрав менше 60 балів, йому ставиться оцінка «незадовільно».

При простому розрахунку отримаємо:

	ЗМ1	ЗМ2	Підсумкова семестрова оцінка	іспит	Підсумкова оцінка
Мінімум	15	21	36	24	60
Максимум	26	34	60	40	100

При цьому, кількість балів:

- **0-59** відповідає оцінці «незадовільно»;
- **60-74** відповідає оцінці «задовільно»;
- **75-89** відповідає оцінці «добре»;
- **90-100** відповідає оцінці «відмінно».

8. Структура навчальної дисципліни. Тематичний план лекцій і практичних занять

№ п/п	Назва теми	Кількість годин			
		лекції	ПЗ	ЛЗ	С/Р
Змістовий модуль 1. Динамічні структури даних.					
1	Тема 1. Методи аналізу алгоритмів.	2		2	4
2	Тема 2. Абстрактні типи даних. Спискові структури. Стек. Двонаправлена черга.	2		4	8
3	Тема 3. Однонаправлена черга.	2		2	8
4	Тема 4. Зв'язний список.	2		4	8
5	Тема 5. Асоціативні списки. Бінарні дерева пошуку.	2		4	8
6	Тема 6. Асоціативні списки. Хеш-таблиці.	2		4	8
Змістовий модуль 2. Графи. Масиви. Черга з пріоритетом.					
7	Тема 7. Основи теорії графів. Обходи графів. Побудова остовного та мінімального остовного дерев.	2		4	8
8	Тема 8. Алгоритми знаходження найкоротших шляхів на графах.	2		4	8
9	Тема 9. Топологічне сортування.	2		4	8
10	Тема 10. Алгоритми сортування та пошуку в масивах.	2		4	8
11	Тема 11. Черга з пріоритетом.	2		4	8
	Всього за семестр	22		40	84

Загальний обсяг – 150 год., в тому числі:

Лекцій – **22 год.**

Лабораторні заняття – **40 год.**

Консультацій – **4 год.**

Самостійна робота – **84 год.**

РЕКОМЕНДОВАНА ЛІТЕРАТУРА:

Основна:

1. Bruno R. Preiss. Data Structures and Algorithms with Object-Oriented Design Patterns in Python / Bruno R. Preiss., 2003. – 566 с.
2. Kent D. Lee. Data Structures and Algorithms with Python / Kent D. Lee, Steve Hubbard., – Springer, 2015. – 363 с.
3. Коротєєва Т. О. Алгоритми та структури даних: навч. посібник / Т. О. Коротєєва. – Львів: Львівської політехніки, 2014. – 280 с.
4. International Standard ISO/IEC 14882:2014(E) – Programming Language C++. – Режим доступу: <https://isocpp.org/std/the-standard>.

Додаткова:

1. Problem Solving with Algorithms and Data Structures using Python [Електронний ресурс] – Режим доступу до ресурсу: <https://runestone.academy/runestone/books/published/pythonds/index.html>.
2. Локазюк В.М. Надійність, помилки і тестування програмного забезпечення комп'ютерних пристроїв та систем: Навчальний посібник. – Хмельницький: ТУП, 2003. – 74 с.
3. Співаковський О. В., Осипова Н. В., Львов М. С., Бакуменко К. В. Основи алгоритмізації та програмування. Обчислювальний експеримент. Розв'язання проблем ефективності в алгоритмах пошуку та сортування: Навчальний посібник. – Херсон: Айлант. – 2010. – 100 с.

Інформаційні ресурси:

1. The Java Tutorials. – <http://docs.oracle.com/javase/tutorial/>;
2. C++ Tutorials. – <http://www.cplusplus.com/doc/tutorial/>.