

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

**Факультет інформаційних технологій
Кафедра комп'ютерної інженерії та інформаційних систем**

Декан Факультету Інформаційних Технологій
Говорушко Т.О.
1.09.2024 р.

СИЛАБУС



Вибіркова дисципліна _____ Структури даних і алгоритми _____

Загальна інформація

Позиція	Зміст інформації
Викладач(і)	Мельниченко Олександр Вікторович
Профайл викладача	http://kiis.khmnu.edu.ua/personnel/melnychenko-oleksandr/
Е-маїл викладача(ів)	melnychenko@khmnu.edu.ua
Контактний телефон	+380972863460
Сторінка дисципліни в ІСУ	https://msn.khmnu.edu.ua/course/view.php?id=553
Навчальний рік	2024-2025
Консультації	Очні: понеділок, 6-а пара, 1-114; онлайн: за необхідністю та попередньою домовленістю

Характеристика дисципліни

Статус дисципліни	Форма здобуття освіти	Курс	Семестр	Загальний обсяг		Кількість годин						Форма семестрового контролю			
				Кредити ЄКТС	Години	Аудиторні заняття				Індивідуальна робота студента	Самостійна робота, в т.ч. ІРС	Курсовий проєкт	Курсова робота	Залік	Іспит
						Разом	Лекції	Лабораторні роботи	Практичні заняття						
В	О Д		не парний	8	240	102	34	34	34		138			+	

Анотація дисципліни

Дисципліна “Структури даних і алгоритми” відноситься до циклу вибіркової дисципліни, забезпечує підготовку студентів з розроблення алгоритмів обробки даних та принципів побудови структур для збереження даних.

При викладанні дисципліни використовуються активні і творчі форми проведення занять, зокрема, методи проблемного навчання.

Мета і завдання дисципліни

Метою курсу є оволодіння студентами основними принципами побудови структур для збереження даних і алгоритмами та методами обробки даних, які в подальшому можуть сприяти їх успішному застосуванню в професійній діяльності.

Предмет дисципліни. Принципи побудови структур для збереження даних; концепція абстрактних типів даних, основні структури, що реалізують абстрактні типи даних і їх представлення; методи та алгоритми обробки даних; оцінка складності алгоритмів.

Завдання дисципліни:

- навчити застосовувати сумісне збереження даних різних типів на прикладі компіляторів C++;
- розглянути основні можливості мов програмування C/C++ по організації складних структур даних;
- навчити використовувати базові алгоритми та структури даних для розв'язку прикладних задач;
- навчити застосовувати алгоритми обробки даних на всіх етапах життєвого циклу прикладної програмної системи, починаючи з аналізу вимог до програмної системи і її попереднього проектування, і закінчуючи її реалізацією, тестуванням і наступним супроводом.

Очікувані результати навчання.

В результаті вивчення курсу «Структури даних і алгоритми» студент повинен досягти таких результатів навчання (сукупність знань, умінь, навичок, компетентностей):

знати:

- теоретичні положення, що лежать в основі функціонування алгоритмічного та програмного забезпечення; концепцію абстрактних типів даних та основні структури, що реалізують абстрактні типи даних; принципи побудови структур для збереження даних; базові алгоритми обробки даних; основи алгоритмізації та сучасної методології розробки алгоритмічного програмного забезпечення; методи оцінки складності алгоритмів;

вміти:

- здійснювати пошук, оброблення та аналіз інформації з різних джерел для розв'язання прикладних задач з використанням теорії алгоритмів та принципів побудови структур для збереження даних;
- застосовувати знання з теорії алгоритмів та основних принципів побудови структур для збереження даних при розв'язуванні задач аналізу та синтезу програмних систем;
- застосовувати знання з теорії алгоритмів для ідентифікації, формулювання і розв'язування прикладних задач, використовуючи методи побудови алгоритмів, алгоритми обробки даних, принципи побудови структур для збереження даних та абстрактні типи даних, що є найбільш придатними для досягнення поставлених цілей;
- поєднувати теорію і практику побудови структур для збереження даних та алгоритмів обробки даних, а також приймати оптимальні рішення при виробленні стратегії розв'язку прикладних задач з урахуванням виробничих інтересів;
- виконувати експериментальні дослідження з метою пошуку оптимальних алгоритмів розв'язку прикладних задач;
- системно мислити та застосовувати творчі здібності до формування нових ідей при побудові алгоритмів для розв'язування прикладних задач;
- оцінювати отримані результати та аргументовано захищати прийняті рішення при реалізації алгоритмів та виборі структур даних для розв'язку прикладних задач;

бути здатним:

- до абстрактного мислення, аналізу і синтезу при побудові алгоритмів для розв'язування прикладних задач;
- до навчання та оволодіння сучасними знаннями з метою вироблення стратегії побудови алгоритмів для розв'язку прикладних задач та застосовувати одержані знання у практичних ситуаціях;
- до розуміння теорії алгоритмів та принципів побудови структур для збереження даних;
- застосовувати знання з теорії алгоритмів, оцінки складності алгоритмів та принципи побудови структур для збереження даних у практичних ситуаціях;
- використовувати поняття теорії алгоритмів та концепції абстрактних типів даних, а також принципи побудови структур для збереження даних та оцінки складності алгоритмів для розроблення алгоритмічного та програмного забезпечення на сучасних мовах програмування;
- аргументувати вибір методів розв'язування алгоритмічних прикладних задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення;
- розв'язувати складні задачі під час розробки прикладного програмного забезпечення, що передбачає застосування понять теорії алгоритмів та принципи побудови структур для збереження даних і характеризується комплексністю та невизначеністю умов.

Тематичний і календарний план вивчення дисципліни

№ тижня	Тема лекції*	Тема практичного заняття*	Тема лабораторної роботи*	Самостійна робота студентів		
				Зміст	Год.	Література
1.	Основні поняття структур даних. Основні поняття структур даних. Абстрактні типи даних. Статичні структури даних. Динамічні структури даних. Лінійні структури даних. Нелінійні структури даних.	Сортування обміном (бульбашкою).	Алгоритми нестійкого сортування: швидке сортування, сортування Шелла.	Опрацювання лекційного матеріалу. Підготовка до ЛР1.	8	[1, 2, 4, 7, 8]
2.		Сортування вибором.	Алгоритми нестійкого сортування: швидке сортування, сортування Шелла.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР1.	8	[1, 2, 4, 7, 8]
3.	Складність алгоритмів. Асимптотичні оцінки складності. Класи складності. Клас P (задачі з поліноміальною складністю). Клас NP (поліноміальної перевірки). Клас NPC (NP - повні задачі). Інші класи складності.	Пірамідалне сортування.	Алгоритми стійкого сортування: сортування вставками, сортування злиттям, кишенькове сортування, порозрядне сортування.	Опрацювання лекційного матеріалу. Підготовка до ЛР2.	8	[1, 4, 6, 7, 8, 10]
4.		Сортування прямим злиттям.	Алгоритми стійкого сортування: сортування вставками, сортування злиттям, кишенькове сортування, порозрядне сортування.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР2.	8	[1, 4, 6, 7, 8, 10]
5.	Рекурсивні алгоритми. Рекурсія. Поняття та основні визначення. Рекурентності. Принципи побудови рекурсивних алгоритмів. Етапи розроблення рекурсивних алгоритмів. Параметризація. База рекурсії. Декомпозиція. Умови завершення рекурсії. Перехід від рекурсивної до ітеративної форми. Особливості роботи рекурсивних функцій. Доцільність застосування рекурсії в програмуванні. Аналіз трудомісткості рекурсивних алгоритмів методом підрахунку вершин дерева рекурсії.	Зовнішнє сортування природним злиттям.	Рекурсія.	Опрацювання лекційного матеріалу. Підготовка до ЛР3.	8	[1, 2, 5]
6.		Рекурсивні алгоритми.	Рекурсія.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР3.	8	[1, 2, 5]
7.	Методи сортування. Параметри оцінки алгоритмів сортування. Класифікація алгоритмів сортування за сферою застосування. Стійке	Використання Standard Template Library та алгоритмів стандартної бібліотеки C++	Методи пошуку.	Опрацювання лекційного матеріалу. Підготовка до ЛР4.	8	[1, 3, 8, 10]

	сортування. Нестійке сортування. Оцінка складності алгоритмів сортування.	<algorithm> для розв'язування прикладних задач.				
8.	Найпростіші методи сортування. Пришвидшені методи сортування. Спеціальні методи сортування. Сортування обміном (бульбашкою). Сортування вибором. Сортування вставками. Сортування Шелла. Швидке сортування. Пірамідальне сортування. Сортування злиттям. Кишенькове сортування. Кишенькове сортування для повторюваних ключів. Порозрядне сортування. Прийоми при роботі з масивами.	Хешування. Хеш-таблиці.	Методи пошуку.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР4.	8	[1, 3, 8, 10]
9.	Методи пошуку. Класифікація методів інформаційного пошуку. Лінійний пошук. Лінійний пошук з бар'єром. Пошук з перестановкою в початок. Пошук із транспозицією. Бінарний пошук. Рекурсивний та ітеративний підходи реалізації бінарного пошуку. Індексно - послідовний метод пошуку.	Хеш-таблиці. Невпорядковані асоціативні контейнери.	Хешування. Пошук з використанням хеш-функцій.	Опрацювання лекційного матеріалу. Підготовка до ЛР5.	8	[1, 3, 8, 10]
10.	пошук з бар'єром. Пошук з перестановкою в початок. Пошук із транспозицією. Бінарний пошук. Рекурсивний та ітеративний підходи реалізації бінарного пошуку. Індексно - послідовний метод пошуку.	Графи. Пошук в глибину.	Хешування. Пошук з використанням хеш-функцій.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР5.	8	[1, 3, 8, 10]
11.	Хешування. Поняття хешування. Хеш-таблиці. Хеш-функції. Мультиплікативна хеш-функція для ключів з плаваючою точкою. Модульна хеш-функція для цілих ключів. Об'єднання мультиплікативного і модульного методів. Модульна хеш-функція для довгих ключів. Вирішення проблеми колізій за допомогою ланцюжків (відкрите хешування). Відкрита адресація (закрите хешування). Поширені способи переборів: лінійне зондування, квадратичне зондування, подвійне хешування. Якою повинна бути хороша хеш-функція. Ключі як натуральні числа. Ділення з остачею. Мультиплікативна схема хешування. Хешування рядків змінної довжини. Сфери застосування хешування.	Графи. Обхід в ширину.	Неорієнтовані граfi.	Опрацювання лекційного матеріалу. Підготовка до ЛР6.	8	[1, 3, 8, 10]
12.	функція для ключів з плаваючою точкою. Модульна хеш-функція для цілих ключів. Об'єднання мультиплікативного і модульного методів. Модульна хеш-функція для довгих ключів. Вирішення проблеми колізій за допомогою ланцюжків (відкрите хешування). Відкрита адресація (закрите хешування). Поширені способи переборів: лінійне зондування, квадратичне зондування, подвійне хешування. Якою повинна бути хороша хеш-функція. Ключі як натуральні числа. Ділення з остачею. Мультиплікативна схема хешування. Хешування рядків змінної довжини. Сфери застосування хешування.	Топологічне сортування	Неорієнтовані граfi.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР6.	8	[1, 3, 8, 10]

13.	Графи. Основні означення. Типи графів. Спеціальні види графів. Основні властивості графів. Способи представлення	Пошук розділяючих вершин та мостів в графі.	Орієнтовані графи.	Опрацювання лекційного матеріалу. Підготовка до ЛР7.	8	[1, 2, 4]
14.	графів. Матриця інцидентності. Матриця суміжності. Списки суміжності. Перетворення матриці суміжності на список суміжності. Перерахування ребер. Перетворення перерахування ребер на список суміжності. Методи обходу графів. Пошук в глибину. Пошук в ширину. Пошук компонент зв'язності. Пошук циклів в неорієнтованому графі. Пошук циклів в орієнтованому графі. Перевірка графа на дводольність. Алгоритм Дейкстри для визначення найкоротших відстаней в зваженому графі. Алгоритм Флойда-Уоршелла для визначення найкоротших відстаней в зваженому графі.	Задачі на графах.	Орієнтовані графи.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР7.	8	[1, 2, 4]
15.	Дерева. Впорядковані та невлпорядковані дерева. Правила обходу дерев. Прямий, зворотний та симетричний	Двійкове дерево пошуку. Способи обходу дерев.	Дерева.	Опрацювання лекційного матеріалу. Підготовка до ЛР8.	8	[1, 2, 4]
16.	обходи дерева. Абстрактний тип даних TREE. Подання дерев за допомогою масивів. Подання дерев з використанням списків синів. Двійкові дерева. Представлення двійкових дерев. Реалізація двійкових дерев за допомогою вказівників. Збалансоване дерево. Дерево бінарного пошуку. Префіксні дерева. Б-дерево. AVL-дерево. Червоно-чорне дерево. Розширюване дерево.	Префіксні дерева.	Дерева.	Опрацювання лекційного матеріалу. Підготовка до захисту ЛР8.	8	[1, 2, 4]
17.	Методи розроблення алгоритмів. Метод грубої сили (повний перебір). Декомпозиція ("розділяй і володарюй"). Зменшення розміру задачі ("зменшуй і володарюй"). Перетворення ("перетворюй і володарюй"). Жадібні алгоритми. Динамічне програмування. Пошук з поверненням. Локальний пошук.	Підсумкове заняття.	Підсумкове заняття.	Опрацювання лекційного матеріалу. Підготовка до підсумкового лабораторного заняття. Підготовка до тестування.	10	[1, 3—5, 7, 10]

Примітка: * Лекції, практичні заняття проводяться по дві години; послідовність проведення занять визначається розкладом (може не відповідати нумерованим тижням)

Політика дисципліни.

Організація освітнього процесу з дисципліни відповідає вимогам положень про організаційне і навчально-методичне забезпечення освітнього процесу, освітній програмі та навчальному плану. Студент зобов'язаний відвідувати лекції, лабораторні заняття згідно з розкладом, не запізнюватися на заняття, завдання виконувати відповідно до графіка. Пропущене лабораторне заняття студент зобов'язаний опрацювати самостійно у повному обсязі і відвітати перед викладачем не пізніше, ніж за тиждень до чергової атестації. До лабораторних занять студент має підготуватися за відповідною темою і проявляти активність. Набуті особою знання з дисципліни або її окремих розділів у неформальній освіті зараховуються відповідно до Положення про порядок перезарахування результатів навчання у ХНУ.

Критерії оцінювання результатів навчання.

Кожний вид роботи з дисципліни оцінюється за **чотирибальною** шкалою. Семестрова підсумкова оцінка визначається як середньозважена з усіх видів навчальної роботи, виконаних і зданих позитивно з врахуванням коефіцієнта вагомості. Вагові коефіцієнти змінюються залежно від структури дисципліни і важливості окремих її видів робіт. При оцінюванні знань студентів використовуються різні засоби контролю, зокрема: усне опитування перед допуском до виконання лабораторної роботи – здійснюється на їх початку; засвоєння теоретичного матеріалу з тем перевіряється тестовим контролем; якість виконання, набуття теоретичних знань і практичних навичок перевіряється шляхом захисту кожної лабораторної роботи та індивідуального завдання згідно з робочою програмою дисципліни і робочим навчальним планом.

Структурування дисципліни за видами робіт і оцінювання результатів навчання студентів денної форми навчання у семестрі за ваговими коефіцієнтами

Аудиторна робота		Самостійна, індивідуальна робота		Семестровий контроль (залік)
Лабораторні роботи №:	Практичні роботи №:	Контрольна робота	Тестовий контроль	за рейтингом
1-8	1-16		Т 1-4	
ВК: 0,7	0,1	0,1	0,1	0

Примітка: ВК – ваговий коефіцієнт.

Співвідношення інституційної шкали оцінювання і шкали оцінювання ЄКТС

Оцінка ЄКТС	Інституційна шкала балів	Інституційна оцінка	Критерії оцінювання	
A	4,75-5,00	5	Зараховано	Відмінно – глибоке і повне опанування навчального матеріалу і виявлення відповідних умінь та навичок.
B	4,25-4,74	4		Добре – повне знання навчального матеріалу з кількома незначними помилками.
C	3,75-4,24	4		Добре – в загальному правильна відповідь з двома-трьома суттєвими помилками.
D	3,25-3,74	3		Задовільно – неповне опанування програмного матеріалу, але достатне для практичної діяльності за професією.
E	3,00-3,24	3		Задовільно – неповне опанування програмного матеріалу, що задовольняє мінімальні критерії оцінювання
FX	2,00-2,99	2	Незараховано	Незадовільно – безсистемність одержаних знань і неможливість продовжити навчання без додаткових знань з дисципліни
F	0,00-1,99	2		Незадовільно – необхідна серйозна подальша робота і повторне вивчення дисципліни.

Питання для підсумкового контролю з дисципліни

1. Основні поняття структур даних.
2. Абстрактні типи даних.
3. Статичні структури даних.
4. Динамічні структури даних.
5. Лінійні структури даних.
6. Нелінійні структури даних.

7. Асимптотичні оцінки складності.
8. Класи складності.
9. Клас P (задачі з поліноміальною складністю).
10. Клас NP (поліноміальної перевірки).
11. Клас NPC (NP - повні задачі).
12. Параметри оцінки алгоритмів сортування.
13. Класифікація алгоритмів сортування за сферою застосування.
14. Стійке сортування. Нестійке сортування.
15. Оцінка складності алгоритмів сортування.
16. Найпростіші методи сортування.
17. Пришвиджені методи сортування.
18. Спеціальні методи сортування.
19. Сортування обміном (бульбашкою).
20. Сортування вибором.
21. Сортування вставками.
22. Сортування Шелла.
23. Швидке сортування.
24. Пірамідальне сортування.
25. Сортування злиттям.
26. Кишенькове сортування.
27. Кишенькове сортування для повторюваних ключів.
28. Порозрядне сортування.
29. Рекурсія. Поняття та основні визначення. Рекурентності.
30. Принципи побудови рекурсивних алгоритмів.
31. Етапи розроблення рекурсивних алгоритмів.
32. Умови завершення рекурсії.
33. Перехід від рекурсивної до ітеративної форми.
34. Особливості роботи рекурсивних функцій.
35. Доцільність застосування рекурсії в програмуванні.
36. Аналіз трудомісткості рекурсивних алгоритмів методом підрахунку вершин дерева рекурсії.
37. Класифікація методів інформаційного пошуку.
38. Лінійний пошук.
39. Лінійний пошук з бар'єром.
40. Пошук з перестановкою в початок.
41. Пошук із транспозицією.
42. Бінарний пошук.
43. Рекурсивний та ітеративний підходи реалізації бінарного пошуку.
44. Індексно - послідовний метод пошуку.
45. Поняття хешування.
46. Хеш-таблиці.
47. Хеш-функції.
48. Мультиплікативна хеш-функція для ключів з плаваючою точкою.
49. Модульна хеш-функція для цілих ключів.
50. Об'єднання мультиплікативного і модульного методів.
51. Модульна хеш-функція для довгих ключів.
52. Вирішення проблеми колізій за допомогою ланцюжків (відкрите хешування).
53. Відкрита адресація (закрите хешування).
54. Поширені способи переборів: лінійне зондування, квадратичне зондування, подвійне хешування.
55. Якою повинна бути хороша хеш-функція.
56. Ключі як натуральні числа.
57. Ділення з остачею.
58. Мультиплікативна схема хешування.
59. Хешування рядків змінної довжини.
60. Сфери застосування хешування.
61. Типи графів. Спеціальні види графів.
62. Основні властивості графів.
63. Матриця інцидентності.
64. Матриця суміжності.
65. Списки суміжності.

66. Перетворення матриці суміжності на список суміжності.
67. Перерахування ребер.
68. Перетворення перерахування ребер на список суміжності.
69. Методи обходу графів.
70. Пошук в глибину.
71. Пошук в ширину.
72. Пошук компонент зв'язності.
73. Пошук циклів в неорієнтованому графі.
74. Пошук циклів в орієнтованому графі.
75. Перевірка графа на дводольність.
76. Топологічне сортування.
77. Алгоритм Дейкстри для визначення найкоротших відстаней в зваженому графі.
78. Алгоритм Флойда-Уоршелла для визначення найкоротших відстаней в зваженому графі.
79. Впорядковані та невпорядковані дерева.
80. Правила обходу дерев. Прямий, зворотний та симетричний обходи дерева.
81. Абстрактний тип даних TREE.
82. Подання дерев за допомогою масивів.
83. Подання дерев з використанням списків синів.
84. Двійкові дерева.
85. Представлення двійкових дерев.
86. Реалізація двійкових дерев за допомогою вказівників.
87. Збалансоване дерево.
88. Дерево бінарного пошуку.
89. Префіксні дерева.
90. Б-дерево.
91. AVL-дерево.
92. Червоно-чорне дерево.
93. Розширюване дерево.
94. Метод грубої сили (повний перебір).
95. Декомпозиція ("розділяй і володарюй").
96. Зменшення розміру задачі ("зменшуй і володарюй").
97. Перетворення ("перетворюй і володарюй").
98. Жадібні алгоритми.
99. Динамічне програмування.
100. Пошук з поверненням.

9. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

Навчальний процес з дисципліни забезпечений необхідними навчально-методичними розробками в модульному середовищі.

10. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Шаховська Н.Б., Голощук Р.О. Алгоритми і структури даних: навч. посіб. Львів : Видавництво «Магнолія 2006», 2024. 215 с.
2. Бармак О.В., Манзюк Е.А., Радюк П.М. Теорія алгоритмів. Теоретичний курс та лабораторний практикум : навч. посіб. Хмельницький : ХНУ, 2023. 168 с.
3. Кублій, Л. І. Алгоритми та структури даних. Основи алгоритмізації [Електронний ресурс] : підручник для здобувачів ступеня бакалавра за спеціальністю 121 «Інженерія програмного забезпечення» / Л. І. Кублій ; КПІ ім. Ігоря Сікорського. Київ : КПІ ім. Ігоря Сікорського, 2022. 528 с.
4. Mehlhorn, Kurt. Data structures and algorithms 2: graph algorithms and NP-completeness. Vol. 2. - Springer Science & Business Media, 2021. p. 542.
5. Sedgewick, Robert, and Kevin Wayne. Algorithms. Addison-wesley professional, 2021. p. 504.
6. Крєневич А.П. Алгоритми і структури даних. Підручник. Київ: ВПЦ "Київський Університет", 2021. 200 с.
7. Berztsiss, Alfs T. Data structures: theory and practice. Academic press, 2020. p. 682.
8. Sanders, P., Mehlhorn, K., Dietzfelbinger, M., & Dementiev, R. Sequential and Parallel Algorithms and Data Structures. Springer, 2019. p. 302.
9. Storer, James Andrew. An introduction to data structures and algorithms. Springer Science & Business Media, 2019. p. 348.
10. Adamson, Iain T. Data structures and algorithms: a first course. -Springer Science & Business Media, 2019. p. 460.
11. Електронна бібліотека університету. Доступ до ресурсу: http://lib.khnu.km.ua/asp/php_f/page_lib.php.

ІНФОРМАЦІЙНЕ ЗАБЕЗПЕЧЕННЯ

Електронний університет:

1. Модульне середовище для навчання (розміщені усі необхідні матеріали з дисципліни, в тому числі тестові завдання для поточного та семестрового контролю знань).
2. Електронна бібліотека університету

Розробник:



док. філ., Мельниченко О.В.

Погоджено:



Зав. каф. КІС:

к.т.н., доцент. Засорнова І.О.