

ЛАБОРАТОРНА РОБОТА 1.6 ДОСЛІДЖЕННЯ МЕТОДІВ АНАЛІЗУ АЛГОРИТМІВ

Мета – дослідження методів аналізу ефективності алгоритмів та набуття практичних навичок з емпіричного дослідження швидкодії алгоритмів залежно від обсягу та структурованості вхідних даних.

Завдання

Виконати такі дії:

- реалізувати два алгоритми для одного набору (табл. 6.1, варіанти 1–5, 7–11, 13, 15–17, 19, 20) або один алгоритм для двох наборів даних (табл. 6.1, варіанти 6, 12, 14, 18);
- отримати час виконання алгоритмів для наборів даних розміром N , N^2 , N^3 , де $N = 100$;
- побудувати графіки залежностей часу виконання від кількості елементів набору даних для двох реалізацій алгоритму (табл. 6.1, варіанти 1–5, 7–11, 13, 15–17, 19, 20) або двох наборів даних (табл. 6.1, варіанти 6, 12, 14, 18).

Методичні рекомендації

Набори даних різного обсягу, необхідні для виконання завдання, слід формувати за допомогою програмного генератора випадкових чисел (метод *public static double random()* класу *java.lang.Math*, методи класу *java.util.Random*).

Зверніть увагу, що під час дослідження двох алгоритмів на одній структурі даних набір початкових даних має бути однаковим.

Час виконання алгоритму обчислюється шляхом заміру поточного часу перед початком роботи алгоритму та після її закінчення (метод *public static long nanoTime()* класу *java.lang.System*).

Значення часу виконання алгоритмів, що застосовуватиметься для побудови графіків залежностей, обчислюється як усереднене значення виконання алгоритму в наносекундах. Графіки залежностей будуються з використанням програмного застосування Microsoft Excel. Для того щоб порівняти час виконання алгоритмів графіки залежностей слід розмістити в одній координатній площині.

Контрольні запитання

1. Для чого необхідно аналізувати алгоритми? Які методи застосовуються для аналізу алгоритмів?
2. У чому полягає сутність емпіричного аналізу алгоритмів? Які умови його застосування?
3. У чому полягає сутність математичного аналізу? Коли він застосовується?
4. Що розуміється під «рост-функцією»? Які є типи «рост-функцій»?
5. Що означає « θ -нотація», « O -нотація» та « Ω -нотація»? Як можна їх зобразити графічно?
6. Як застосовується математичний аналіз для оцінки ефективності алгоритмів?

Таблиця 6.1 – Завдання

Варіант	Алгоритм	Структура даних
1	2	3
1.	Сортування – швидке базове	Одновимірний масив
	Сортування – бульбашкове	
2.	Сортування – порозрядне	Одновимірний масив
	Сортування – висхідне злиття	
3.	Сортування – низхідне злиття	Одновимірний масив
	Сортування – бульбашкове	
4.	Сортування – Шелла (класичний)	Одновимірний масив
	Сортування – Шелла (за Кнутом)	
5.	Сортування – висхідне злиття	Одновимірний масив
	Сортування – низхідне злиття	
6.	Пошук – індексуванням за ключем	Хеш-таблиця з відкритою адресацією (квадратичне зондування)
		Хеш-таблиця з відкритою адресацією (лінійне зондування)
7.	Сортування – розподіленого підрахунку	Одновимірний масив
	Сортування – порозрядне	
8.	Сортування – пірамідальне з використанням черги за пріоритетом	Одновимірний масив
	Сортування – пірамідальне базове	
9.	Пошук – бінарний	Одновимірний масив
	Пошук – інтерполяційний	

1	2	3
10.	Сортування – Шелла (за Кнудом)	Одновимірний масив
	Сортування – вставкою	
11.	Пошук – лінійний	Одновимірний масив
	Пошук – бінарний	
12.	Пошук – лінійний	Односпрямований список
		Одновимірний масив
13.	Сортування – швидке з обчисленням медіани з трьох елементів	Одновимірний масив
	Сортування – швидке базове	
14.	Пошук – індексуванням за ключем	Хеш-таблиця з відкритою адресацією (лінійне зондування)
		Хеш-таблиця з роздільним зв'язуванням
15.	Сортування – швидке з розділенням на три частини	Одновимірний масив
	Сортування – швидке базове	
16.	Сортування – вибіркою	Односпрямований список
		Одновимірний масив
17.	Сортування – бульбашкове двоспрямоване	Одновимірний масив
	Сортування – бульбашкове	
18.	Сортування – карманне	Одновимірний масив впорядкований
		Одновимірний масив неупорядкований
19.	Злиття – двокодійне	Одновимірний масив
	Злиття – обмінне	
20.	Сортування – пірамідальне базове	Одновимірний масив
	Сортування – бульбашкове оптимізоване	

СПИСОК ЛИТЕРАТУРЫ

1. Алгоритмы: построение и анализ, 2-е изд. / Т. Кормен, Ч. Лейзерон, Р. Ривест, К. Штайн. – М. : Издательский дом «Вильямс», 2005. – 1296 с.
2. Гудрич М. Т. Структуры данных и алгоритмы в Java / М. Т. Гудрич, Р. Тамассия; Пер. с англ. А. М. Чернухо. – Мн. : Новое знание, 2003. – 671 с.
3. Кнут Д. Искусство программирования. – Т.1. Основные алгоритмы, 3-е изд. / Д. Кнут. – М. : Издательский дом «Вильямс», 2010. – 720 с.
4. Роберт Седжвик. Фундаментальные алгоритмы на Java. Анализ/Структуры данных/Сортировка/Поиск. / Р. Седжвик. – К. : ООО «ТИД ДС», 2003. – 688 с.

Учебные материалы ИПО НАУ специальности ПОС