

## Лабораторна робота 2: Емпіричний аналіз алгоритмів

### Мета роботи:

- 1) проаналізувати умови застосування емпіричного методу для аналізу ефективності алгоритмів;
- 2) набути практичних навичок з організації експерименту: проектування експерименту, проведення експерименту, аналіз експериментальних даних і складання звіту;
- 3) навчитися вимірювати час виконання алгоритму програмними засобами мови Java;
- 4) набути практичних навичок з аналізу та візуалізації експериментальних даних.

### Необхідні теоретичні знання мови Java

1. Генерація випадкових чисел:  
<https://docs.oracle.com/javase/7/docs/api/java/util/Random.html>
2. Отримання поточного значення часу JVM:  
<http://www.javawithus.com/tutorial/array-of-objects>

### Інструментальні і програмні засоби:

Java JDK 8  
IDE Eclipse  
MS Office Excel

### Рекомендації щодо організації експерименту

#### Проектування експерименту

- сформулювати мету експерименту;
- спроектувати порядок вимірювань відповідно до мети експерименту;
- описати апаратну і програмну платформи експериментального середовища;
- реалізувати алгоритм (алгоритми) на заданій структурі даних певною мовою програмування відповідно до мети експерименту;
- поставити вимоги до набору тестових даних відповідно до мети експерименту;
- реалізувати генератор даних, який постачає початкові дані для алгоритму, різні за обсягом і характером. Для генерації випадкових чисел можна використовувати клас Random та його методи;
- реалізувати компонент, який зберігає результати експерименту;
- описати та обґрунтувати припущення щодо результатів експерименту.

#### Проведення експерименту

## Алгоритми та структури даних. Частина 2

- налаштувати експериментальне середовище для проведення експерименту;
- генерувати початкові набори даних відповідно до поставлених при проектуванні вимог;
- отримати час виконання алгоритму для наборів даних різного розміром: час виконання алгоритму обчислюється шляхом заміру поточного часу перед початком роботи алгоритму та після його закінчення;
- провести вимірювання часу декілька раз (приблизно п'ять раз) і отримати середнє значення часу.

*Аналіз експериментальних даних*

- побудувати графік залежностей часу виконання алгоритму від кількості елементів набору даних. Графіки залежностей будуються з використанням програмного застосування Microsoft Excel або Google spreadsheet ([sheets.google.com](https://sheets.google.com)). Для того, щоб порівняти час виконання двох алгоритмів або операцій, графіки залежностей слід розмістити в одній координатній площині;
- прогнозувати Big-O для коду, що аналізується;
- порівняти результати експерименту і припущення, зроблені при проектуванні;
- зробити висновок щодо результату експерименту відповідно до його мети.

*Звіт про експеримент*

- 1) Опис середовища: апаратна частина, операційна система, інтегроване середовище розробки, мова програмування
- 2) Псевдокод алгоритму і його асимптотичний аналіз
- 3) Код на мові Java
- 4) Специфікація вхідного набору даних
- 5) Таблиці з результатами вимірювань
- 6) Графіки
- 7) Висновок

*Завдання*

Провести два експерименти, керуючись рекомендаціями до організації експерименту.

**Мета експерименту 1:** дослідити час виконання алгоритму залежно від обсягу і характеру вхідних даних на двох різних апаратних середовищах.

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

Алгоритм обрати з таблиці 1 згідно з варіантом індивідуального завдання

**Мета експерименту 2:** порівняти час виконання двох алгоритмів на одній апаратній платформі

Алгоритм обрати з таблиці 2 згідно з варіантом індивідуального завдання

*Питання для самоперевірки:*

1. Емпіричний аналіз: припущення з використання порівняння, умови застосування. Методика проведення емпіричних досліджень.
2. Підрахунок кількості простих операцій у послідовних, повторювальних і вкладених алгоритмах.
3. Математичний аналіз алгоритмів. Побудова аналітичного виразу, що оцінює ефективність алгоритму. Аналіз найгіршого і найкращого випадку роботи алгоритму
4. Асимптотичний аналіз верхньої та середньої оцінок складності алгоритмів. Асимптотична нотація:  $\theta$ -нотація,  $O$ -нотація,  $\Omega$ -нотація. Порівняння найкращих, середніх і найгірших оцінок.
5. Зростання функції
6. Алгоритм «швидкого» сортування. Особливості операції розділення за опорним елементом. Аналіз алгоритму швидкого сортування
7. Алгоритм сортування злиттям. Сортування низхідного і висхідного злиття. Аналіз алгоритму сортування злиттям
8. Математичний аналіз рекурсивних алгоритмів. Знаходження розв'язку рекурентного рівняння. Метод індукції для рекурентного рівняння. Побудова дерева рекурсії. Основна теорема для рекурентного рівняння

Таблиця 1. Варіанти індивідуальних завдань лабораторної роботи 2 (частина 1)

Варіант	Алгоритм	Структура даних
1	2	3
1	Сортування – швидке базове	Одновимірний масив
2	Сортування – порозрядне за молодшими розрядами	Одновимірний масив рядків
3	Сортування – низхідне злиття	Одновимірний масив
4	Сортування – Шелла (класичний)	Одновимірний масив
5	Сортування – висхідне злиття	Одновимірний масив
6	Сортування – Шелла (за Кнутом)	Одновимірний масив
7	Сортування – розподіленого підрахунку	Одновимірний масив
8	Сортування – бульбашкове	Одновимірний масив
9	Сортування – вставкою	Одновимірний масив
10	Пошук – лінійний	Односпрямований список
11	Вставка елемента у сортовану послідовність	Двоспрямований список
12	Пошук – бінарний	Одновимірний масив
13	Сортування – швидке з обчисленням медіани з трьох елементів	Одновимірний масив
14	Пошук – інтерполяційний	Одновимірний масив
15	Сортування – швидке з розділенням на три частини	Одновимірний масив
16	Сортування – вибіркою	Односпрямований список
17	Сортування – бульбашкове двоспрямоване	Одновимірний масив
18	Сортування – комірками	Одновимірний масив цілих чисел
19	Злиття – двоколіїне	Одновимірний масив
20	Сортування – пірамідальний базовий	Одновимірний масив
21	Сортування – порозрядне за старшими розрядами	Одновимірний масив рядків
22	Сортування – швидке з випадковим вибором опорного елемента	Одновимірний масив
23	Сортування – вибіркою	Односпрямований масив
24	Пошук – експоненціальний	Одновимірний масив

Таблиця 2. Варіанти індивідуальних завдань лабораторної роботи 2 (частина 2)

Варіант	Алгоритм	Структура даних
1	2	3
1	Сортування порозрядне за старшими розрядами	Одновимірний масив рядків
	Сортування порозрядне за молодшими розрядами	
2	Сортування – низхідне	Одновимірний масив
	Сортування – висхідне	
3	Сортування – вставкою	Одновимірний масив

## Алгоритми та структури даних. Частина 2

Варіант	Алгоритм	Структура даних
1	2	3
	Сортування – розподіленого підрахунку	
4	Сортування – пірамідальне з використанням черги за пріоритетом	Одновимірний масив
	Сортування – пірамідальне базове	
5	Сортування – порозрядне	Одновимірний масив цілих чисел
	Сортування – вставками	
6	Сортування – порозрядне	Одновимірний масив рядків
	Сортування – вставками	
7	Сортування вставкою	Одновимірний масив
		Двоспрямований список
8	Сортування – швидке базове з розділенням за Гоаром	Одновимірний масив
	Сортування – швидке базове з розділенням за Ломуто	
9	Пошук – бінарний	Одновимірний масив
	Пошук – інтерполяційний	
10	Сортування – Шелла (за Кнутом)	Одновимірний масив
	Сортування – вставкою	
11	Сортування – швидке базове	Одновимірний масив
	Сортування – швидке з випадковим вибором опорного елемента	
12	Сортування – розподіленого підрахунку	Одновимірний масив
	Сортування – комірками	
13	Сортування – вибіркою	Одновимірний масив
	Сортування – бульбашкове оптимізоване	
14	Сортування – комірками	Одновимірний масив дійсних чисел
	Сортування – вставками	
15	Сортування – швидке з розділенням на три частини	Одновимірний масив
	Сортування – швидке базове	
16	Сортування – швидке з обчисленням медіани з трьох елементів	Одновимірний масив
	Сортування – швидке базове	
17	Сортування – пірамідальне базове	Одновимірний масив
	Сортування – швидке базове	
18	Сортування – Шелла (класичний)	Одновимірний масив
	Сортування – Шелла (за Кнутом)	
19	Сортування – бульбашкове двоспрямоване	Одновимірний масив
	Сортування – бульбашкове	
20	Злиття – двоколіїне	Одновимірний масив
	Злиття – обмінне	
21	Сортування вставками	Одновимірний масив рядків однієї довжини
	Сортування порозрядне за молодшими розрядами	
22	Сортування вставками	Одновимірний масив цілих чисел
	Сортування порозрядне	
23	Пошук – лінійний	Одновимірний масив
	Пошук – бінарний	
24	Сортування вставками	Одновимірний масив рядків різної довжини
	Сортування порозрядне за молодшими розрядами	