**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет   
імені Петра Могили**

**Факультет комп’ютерних наук**

**Кафедра інженерії програмного забезпечення**

**ЗВІТ**

*з лабораторної роботи № 1*

**«Алгоритми сортування»**

**Варіант № 5**

Дисципліна «Алгоритми та структури даних»

Спеціальність «Інженерія програмного забезпечення»

121–ЛР.01–208. 22120802

***Cтудент*** *\_\_\_\_\_\_\_\_\_\_****Д.М.Дзундза***

*(підпис)*

*\_\_27.09.2021 \_\_*

*(дата)*

***Викладач*** *\_****\_\_\_\_\_\_\_\_К. О. Кірей***

*(підпис)*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(дата)*

**Миколаїв – 2021**

Лабораторна робота №1

**Тема роботи:** Алгоритми сортування

**Мета роботи:** ознайомитися з різними методами сортування масивів (bubble

sort, insertion sort, selection sort, merge sort, quick sort); дослідити на практиці щодо кожного методу залежність часу сортування від значень елементів, первинного розташування та первинної впорядкованості, кількості елементів; навчитися застосовувати отримані знання для вирішення практичних завдань.

**Завдання:**

1.1 Проведіть експеримент і дізнайтеся, як впливає на час виконання сортування наступний вміст масиву:

* масив складається з випадкових чисел;
* спочатку масив відсортований за спаданням;
* масив має два однакових значення, що розташовані за
* довільним порядком;
* масив має всі однакові значення.

1.2. Модернізуйте код програми виходячи з умов п. 1, заповніть порівняльну таблицю. Наприклад, для бульбашкового методу одержані наступні дані.

1.3. Ознайомтеся з методами сортування масивів (visualgo.net/en/sorting):

* бульбашковим (Bubble sort);
* вставками (Insertion sort);
* вибором (Selection sort);
* злиттям (Merge sort);
* швидкого сортуванням (Quick sort).

1.4. Заповніть таблицю для інших чотирьох методів.

2. Напишіть програму, що демонструє стійкість або не стійкість методу сортування.

3. Напишіть програму для виміру продуктивності сортування, що викликає функцію сортування для масивів різних розмірів (від 1000 до 100000, крок 1000), заміряє час виконання сортування і виводить дані у файл. За полученими даними побудуйте графіки залежності часу сортування від кількості елементів для всіх методів п. 3 завдання 1. Наприклад, на рис. 1.2 наведено відповідний графік для сортування вставками, побудований у MS Excel, добавлена лінія тренда і виведено

її рівняння, з якого видно, що залежність квадратична.

**Результати роботи програми:**

**Час сортування**

Кількість елементів масиву – **100 000**

ОС - **Windows 10**, процесор - **Intel Core i5-8250U CPU 1.6GHz**,

ОЗУ - **12 ГБайт**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Назва методу сортування | Масив складається з випадкових чисел (елементи масиву ініціалізовані за допомогою функції rand() у діапазоні від -1000 до 1000) | Спочатку масив відсортовано за спаданням | Масив має два однакових значення, розташованих у довільному порядку | Масив має всі однакові значення |
| Бульбашковий (Bubble sort) | 290259 мс | 735372 мс | 275656 мс | 31639 мс |
| Вставками (Insertion sort) | 6642 мс | 33179 мс | 8414 мс | 1мс |
| Вибірковий  (Selection sort) | 12065 мс | 28745 мс | 27994 мс | 28305 мс |
| Злиттям (Merge sort) | 62 мс | 144 мс | 156 мс | 151 мс |
| Швидкий  (Quick sort) | 10мс | 14 мс | 15 мс | 12 мс |

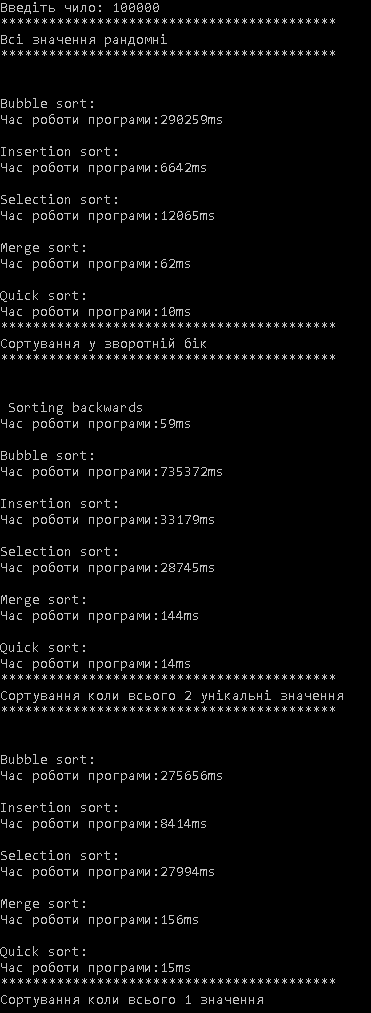


Рисунок 1 – Заміри часу виконання алгоритмів частина 1

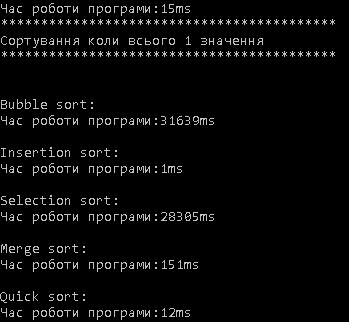


Рисунок 2 – Заміри часу виконання алгоритмів частина 2

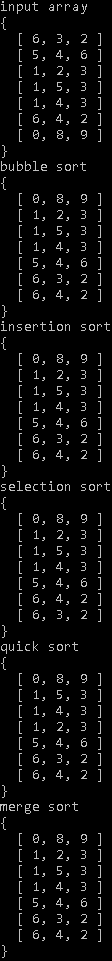


Рисунок 3 - Демонстрація стабільності алгоритму

|  |  |
| --- | --- |
| Назва | Стабільність алгоритму |
| Bubble sort | Стабільний |
| Insertion sort | Стабільний |
| Selection sort | Не стабільний |
| Quick sort | Не стабільний |
| Merge sort | Стабільний |

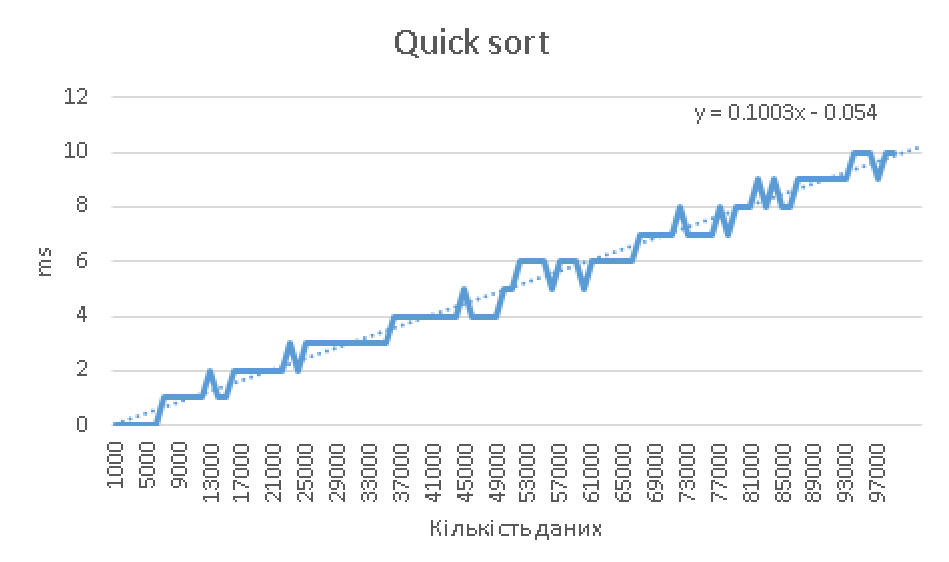


Рисунок 4 – Залежність часу виконання алгоритму від кількості елементів для Quick sort

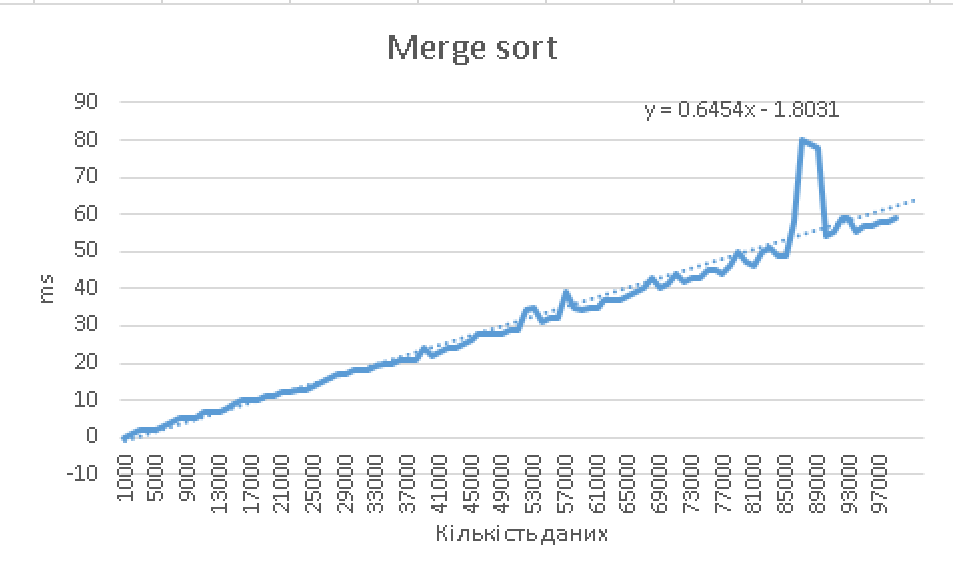


Рисунок 5 – Залежність часу виконання алгоритму від кількості елементів для Merge sort

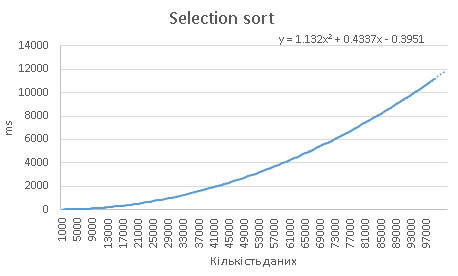


Рисунок 6 – Залежність часу виконання алгоритму від кількості елементів для Selection sort

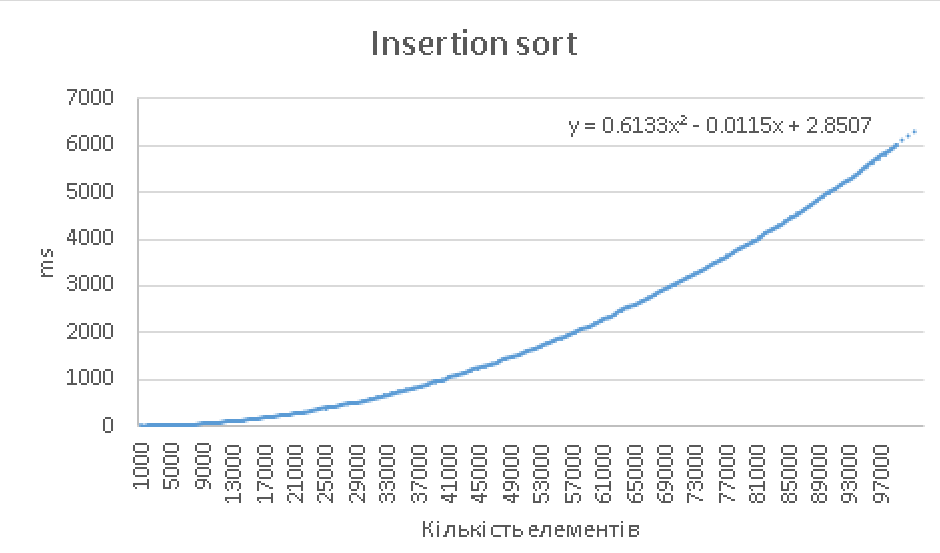


Рисунок 7 – Залежність часу виконання алгоритму від кількості елементів для Insertion sort

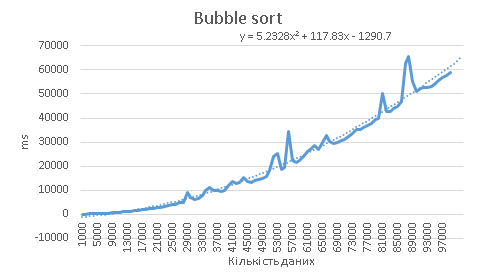


Рисунок 8 – Залежність часу виконання алгоритму від кількості елементів для Bubble sort

**Висновки:** Протягом виконання лабораторної роботи було розглянути алгоритми сортування. Було створено алгоритм для відображення швидкості виконання алгоритму. Було продемонстровано стійкість алгоритмів.