

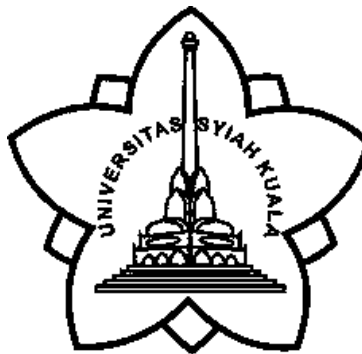
Laporan Tugas Struktur Data dan Algoritma

PENGUJIAN PERFORMA ALGORITMA SORTING

disusun untuk memenuhi
tugas mata kuliah Struktur Data dan Algoritma

oleh:

MUHAMMAD SIDQI ALFAREZA
2308107010040



JURUSAN INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS SYIAH KUALA
DARUSSALAM, BANDA ACEH 2025

1. Deskripsi Algoritma dan Cara Implementasi

Pengujian ini bertujuan untuk membandingkan performa enam algoritma sorting, baik pada data bertipe string maupun integer (int). Adapun algoritma yang diuji adalah:

- Bubble Sort
- Selection Sort
- Insertion Sort
- Merge Sort
- Quick Sort
- Shell Sort

Data yang digunakan memiliki ukuran bertahap, mulai dari 10.000 hingga 2.000.000 entri. Pada data string, diberlakukan pembatasan waktu maksimum 20 menit per algoritmanya dikarenakan waktu yang digunakan telah melebihi 6 jam. Pada data integer, semua algoritma dieksekusi penuh tanpa pembatasan waktu karena setelah pengujian tidak memakan waktu yang berlebihan.

2. Tabel Hasil Eksperimen

- a. Langkah-langkah menjalankan kode program:
 - i. Jalankan kode

```
PS D:\KULIAH\SEMESTER IV\SDA\Tugas Sorting> ./main
```

- ii. Pilih menu data yang akan disorting

```
===== PILIH JENIS DATA =====
1. Data Angka (data_angka.txt)
2. Data Kata (data_kata.txt)
Pilihan Anda: 2
```

- iii. Pilih menu jumlah data yang akan disorting

```
===== PILIH JUMLAH DATA =====
1. 10.000
2. 50.000
3. 100.000
4. 250.000
5. 500.000
6. 1.000.000
7. 1.500.000
8. 2.000.000
Pilihan Anda: 5
```

- iv. Data sedang disorting

```
--- Sorting Data Kata (500000 data) ---
```

b. Data angka

- 10000 Angka

```
--- Sorting Data Angka (10000 data) ---
Bubble Sort: 0.098 detik   Bubble Sort   Memory: 39.06 KB ( 0.04 MB)
Selection Sort: 0.043 detik Selection Sort Memory: 39.06 KB ( 0.04 MB)
Insertion Sort: 0.041 detik Insertion Sort Memory: 39.06 KB ( 0.04 MB)
Merge Sort: 0.000 detik   Merge Sort   Memory: 117.19 KB ( 0.11 MB)
Quick Sort: 0.000 detik   Quick Sort   Memory: 39.06 KB ( 0.04 MB)
Shell Sort: 0.000 detik   Shell Sort   Memory: 39.06 KB ( 0.04 MB)
```

- 50000 Angka

```
--- Sorting Data Angka (50000 data) ---
Bubble Sort: 3.977 detik   Bubble Sort   Memory: 195.31 KB ( 0.19 MB)
Selection Sort: 0.954 detik Selection Sort Memory: 195.31 KB ( 0.19 MB)
Insertion Sort: 1.137 detik Insertion Sort Memory: 195.31 KB ( 0.19 MB)
Merge Sort: 0.007 detik   Merge Sort   Memory: 585.94 KB ( 0.57 MB)
Quick Sort: 0.000 detik   Quick Sort   Memory: 195.31 KB ( 0.19 MB)
Shell Sort: 0.020 detik   Shell Sort   Memory: 195.31 KB ( 0.19 MB)
```

- 100000 Angka

```
--- Sorting Data Angka (100000 data) ---
Bubble Sort: 16.661 detik   Bubble Sort   Memory: 390.63 KB ( 0.38 MB)
Selection Sort: 3.869 detik Selection Sort Memory: 390.63 KB ( 0.38 MB)
Insertion Sort: 4.257 detik Insertion Sort Memory: 390.63 KB ( 0.38 MB)
Merge Sort: 0.018 detik   Merge Sort   Memory: 1171.88 KB ( 1.14 MB)
Quick Sort: 0.017 detik   Quick Sort   Memory: 390.63 KB ( 0.38 MB)
Shell Sort: 0.017 detik   Shell Sort   Memory: 390.63 KB ( 0.38 MB)
```

- 250000 Angka

```
--- Sorting Data Angka (250000 data) ---
Bubble Sort: 107.130 detik   Bubble Sort   Memory: 976.56 KB ( 0.95 MB)
Selection Sort: 24.141 detik Selection Sort Memory: 976.56 KB ( 0.95 MB)
Insertion Sort: 26.147 detik Insertion Sort Memory: 976.56 KB ( 0.95 MB)
Merge Sort: 0.059 detik   Merge Sort   Memory: 2929.69 KB ( 2.86 MB)
Quick Sort: 0.028 detik   Quick Sort   Memory: 976.56 KB ( 0.95 MB)
Shell Sort: 0.047 detik   Shell Sort   Memory: 976.56 KB ( 0.95 MB)
```

- 500000 Angka

```
--- Sorting Data Angka (500000 data) ---
Bubble Sort: 454.042 detik   Bubble Sort   Memory: 1953.13 KB ( 1.91 MB)
Selection Sort: 99.201 detik Selection Sort Memory: 1953.13 KB ( 1.91 MB)
Insertion Sort: 106.226 detik Insertion Sort Memory: 1953.13 KB ( 1.91 MB)
Merge Sort: 0.106 detik   Merge Sort   Memory: 5859.38 KB ( 5.72 MB)
Quick Sort: 0.042 detik   Quick Sort   Memory: 1953.13 KB ( 1.91 MB)
Shell Sort: 0.106 detik   Shell Sort   Memory: 1953.13 KB ( 1.91 MB)
```

- 1000000 Angka

```
--- Sorting Data Angka (1000000 data) ---
Bubble Sort: 2330.176 detik   Bubble Sort   Memory: 3906.25 KB ( 3.81 MB)
Selection Sort: 2284.756 detik Selection Sort Memory: 3906.25 KB ( 3.81 MB)
Insertion Sort: 946.916 detik Insertion Sort Memory: 3906.25 KB ( 3.81 MB)
Merge Sort: 0.452 detik   Merge Sort   Memory: 11718.75 KB ( 11.44 MB)
Quick Sort: 0.217 detik   Quick Sort   Memory: 3906.25 KB ( 3.81 MB)
Shell Sort: 0.468 detik   Shell Sort   Memory: 3906.25 KB ( 3.81 MB)
```

- 1500000 Angka

```

--- Sorting Data Angka (1500000 data) ---
Bubble Sort: 5182.043 detik   Bubble Sort   Memory: 5859.38 KB ( 5.72 MB)
Selection Sort: 884.756 detik Selection Sort Memory: 5859.38 KB ( 5.72 MB)
Insertion Sort: 941.732 detik Insertion Sort Memory: 5859.38 KB ( 5.72 MB)
Merge Sort: 0.300 detik     Merge Sort   Memory: 17578.13 KB ( 17.17 MB)
Quick Sort: 0.153 detik     Quick Sort   Memory: 5859.38 KB ( 5.72 MB)
Shell Sort: 0.339 detik     Shell Sort   Memory: 5859.38 KB ( 5.72 MB)

```

- 2000000 Angka

```

--- Sorting Data Angka (2000000 data) ---
Bubble Sort: 7038.566 detik   Bubble Sort   Memory: 7812.50 KB ( 7.63 MB)
Selection Sort: 1556.793 detik Selection Sort Memory: 7812.50 KB ( 7.63 MB)
Insertion Sort: 1611.376 detik Insertion Sort Memory: 7812.50 KB ( 7.63 MB)
Merge Sort: 0.414 detik     Merge Sort   Memory: 23437.50 KB ( 22.89 MB)
Quick Sort: 0.206 detik     Quick Sort   Memory: 7812.50 KB ( 7.63 MB)
Shell Sort: 0.459 detik     Shell Sort   Memory: 7812.50 KB ( 7.63 MB)

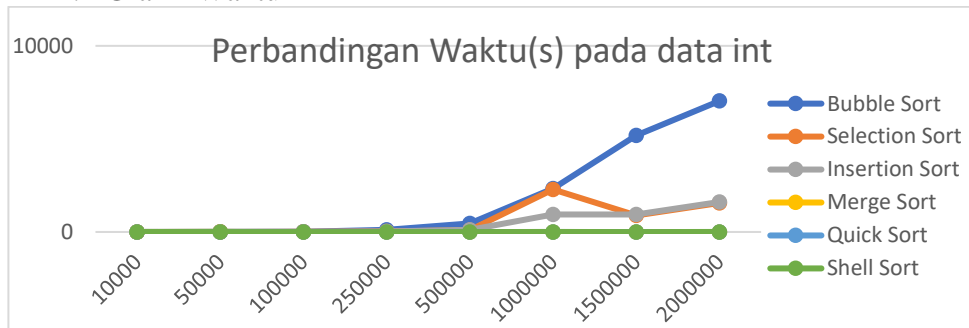
```

c. Data Kata (String)

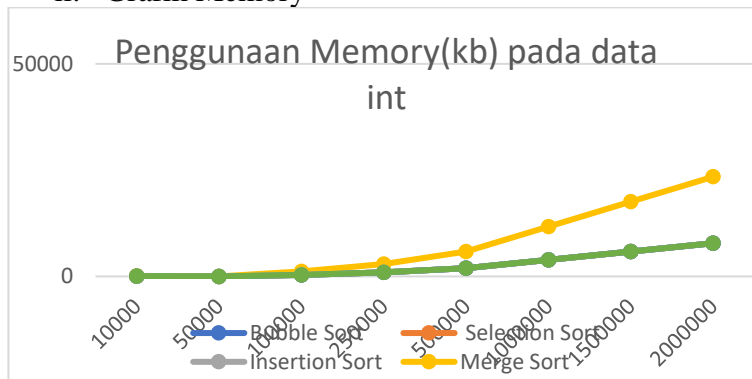
Grafik Perbandingan Waktu dan Memory

b. Data Angka

i. Grafik Waktu



ii. Grafik Memory



c. Data Kata (String)

- o 10000 String

```

--- Sorting Data Kata (10000 data) ---
Bubble Sort: 0.687 detik   Bubble Sort   Memory: 117.19 KB ( 0.11 MB)
Selection Sort: 0.206 detik   Selection Sort   Memory: 117.19 KB ( 0.11 MB)
Insertion Sort: 0.096 detik   Insertion Sort   Memory: 117.19 KB ( 0.11 MB)
Merge Sort: 0.000 detik   Merge Sort   Memory: 234.38 KB ( 0.23 MB)
Quick Sort: 0.000 detik   Quick Sort   Memory: 117.19 KB ( 0.11 MB)
Shell Sort: 0.008 detik   Shell Sort   Memory: 117.19 KB ( 0.11 MB)

```

○ 50000

```

--- Sorting Data Kata (50000 data) ---
Bubble Sort: 20.279 detik   Bubble Sort   Memory: 585.94 KB ( 0.57 MB)
Selection Sort: 5.047 detik   Selection Sort   Memory: 585.94 KB ( 0.57 MB)
Insertion Sort: 2.054 detik   Insertion Sort   Memory: 585.94 KB ( 0.57 MB)
Merge Sort: 0.023 detik   Merge Sort   Memory: 1171.88 KB ( 1.14 MB)
Quick Sort: 0.032 detik   Quick Sort   Memory: 585.94 KB ( 0.57 MB)
Shell Sort: 0.033 detik   Shell Sort   Memory: 585.94 KB ( 0.57 MB)

```

○ 100000

```

--- Sorting Data Kata (100000 data) ---
Bubble Sort: 99.218 detik   Bubble Sort   Memory: 1171.88 KB ( 1.14 MB)
Selection Sort: 47.581 detik   Selection Sort   Memory: 1171.88 KB ( 1.14 MB)
Insertion Sort: 25.858 detik   Insertion Sort   Memory: 1171.88 KB ( 1.14 MB)
Merge Sort: 0.078 detik   Merge Sort   Memory: 2343.75 KB ( 2.29 MB)
Quick Sort: 0.063 detik   Quick Sort   Memory: 1171.88 KB ( 1.14 MB)
Shell Sort: 0.125 detik   Shell Sort   Memory: 1171.88 KB ( 1.14 MB)

```

○ 250000

```

--- Sorting Data Kata (250000 data) ---
Bubble Sort: 787.472 detik   Bubble Sort   Memory: 2929.69 KB ( 2.86 MB)
Selection Sort: 152.370 detik   Selection Sort   Memory: 2929.69 KB ( 2.86 MB)
Insertion Sort: 78.585 detik   Insertion Sort   Memory: 2929.69 KB ( 2.86 MB)
Merge Sort: 0.140 detik   Merge Sort   Memory: 5859.38 KB ( 5.72 MB)
Quick Sort: 0.104 detik   Quick Sort   Memory: 2929.69 KB ( 2.86 MB)
Shell Sort: 0.241 detik   Shell Sort   Memory: 2929.69 KB ( 2.86 MB)

```

○ 500000

```

--- Sorting Data Kata (500000 data) ---
Bubble Sort: 900.007 detik   Bubble Sort   Memory: 5859.38 KB ( 5.72 MB)
Selection Sort: 900.005 detik   Selection Sort   Memory: 5859.38 KB ( 5.72 MB)
Insertion Sort: 900.010 detik   Insertion Sort   Memory: 5859.38 KB ( 5.72 MB)
Merge Sort: 0.000 detik   Merge Sort   Memory: 11718.75 KB ( 11.44 MB)
Quick Sort: 0.000 detik   Quick Sort   Memory: 5859.38 KB ( 5.72 MB)
Shell Sort: 0.898 detik   Shell Sort   Memory: 5859.38 KB ( 5.72 MB)

```

○ 1000000

```

--- Sorting Data Kata (1000000 data) ---
Bubble Sort: 900.008 detik   Bubble Sort   Memory: 11718.75 KB ( 11.44 MB)
Selection Sort: 900.002 detik   Selection Sort   Memory: 11718.75 KB ( 11.44 MB)
Insertion Sort: 900.002 detik   Insertion Sort   Memory: 11718.75 KB ( 11.44 MB)
Merge Sort: 0.000 detik   Merge Sort   Memory: 23437.50 KB ( 22.89 MB)
Quick Sort: 0.000 detik   Quick Sort   Memory: 11718.75 KB ( 11.44 MB)
Shell Sort: 3.156 detik   Shell Sort   Memory: 11718.75 KB ( 11.44 MB)

```


○ 1500000

```

--- Sorting Data Kata (1500000 data) ---
Bubble Sort: 900.000 detik    Bubble Sort    Memory: 17578.13 KB ( 17.17 MB)
Selection Sort: 967.220 detik  Selection Sort  Memory: 17578.13 KB ( 17.17 MB)
Insertion Sort: MAX           Insertion Sort  Memory: 17578.13 KB ( 17.17 MB)
Merge Sort: 0.000 detik       Merge Sort      Memory: 35156.25 KB ( 34.33 MB)
Quick Sort: 0.000 detik       Quick Sort      Memory: 17578.13 KB ( 17.17 MB)
Shell Sort: 4.344 detik       Shell Sort      Memory: 17578.13 KB ( 17.17 MB)

```

○ 2000000

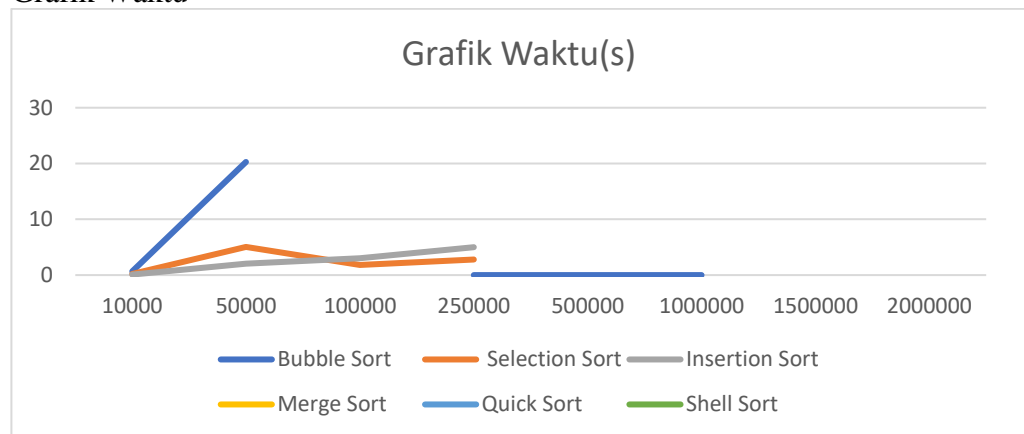
```

--- Sorting Data Kata (2000000 data) ---
Bubble Sort: 900.005 detik    Bubble Sort    Memory: 23437.50 KB ( 22.89 MB)
Selection Sort: MAX           Selection Sort  Memory: 23437.50 KB ( 22.89 MB)
Insertion Sort: 900.012 detik  Insertion Sort  Memory: 23437.50 KB ( 22.89 MB)
Merge Sort: 0.000 detik       Merge Sort      Memory: 46875.00 KB ( 45.78 MB)
Quick Sort: 0.000 detik       Quick Sort      Memory: 23437.50 KB ( 22.89 MB)
Shell Sort: 5.290 detik       Shell Sort      Memory: 23437.50 KB ( 22.89 MB)

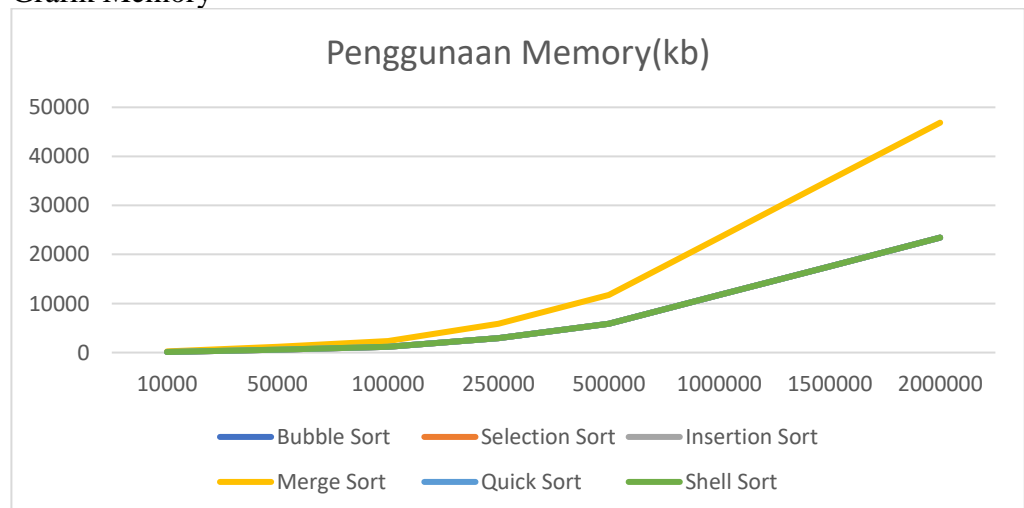
```

Grafik Perbandingan Waktu dan Memory

i. Grafik Waktu



ii. Grafik Memory



3. Analisis dan Kesimpulan

a. Analisis

i. Data int

Pada pengujian sorting data integer, Bubble Sort menunjukkan performa yang sangat buruk. Waktu eksekusi algoritma ini meningkat secara drastis ketika jumlah data mencapai 500.000, dan mencapai lebih dari 7000 detik saat memproses 2 juta data. Selection Sort dan Insertion Sort juga mengalami peningkatan waktu eksekusi yang signifikan seiring bertambahnya jumlah data, meskipun tidak seekstrem Bubble Sort. Sebaliknya, algoritma Merge Sort, Quick Sort, dan Shell Sort mampu mempertahankan performa waktu yang sangat cepat dan stabil, bahkan saat menangani dataset integer berukuran besar. Dari sisi penggunaan memori, Merge Sort diketahui membutuhkan alokasi memori tambahan yang cukup besar akibat proses penggabungan data, sedangkan algoritma lainnya tetap menunjukkan efisiensi memori yang baik.

ii. Data String

Pada pengujian sorting data string, Bubble Sort mengalami lonjakan waktu eksekusi yang sangat besar bahkan ketika ukuran data baru mencapai 50.000 kata. Untuk ukuran data yang lebih besar, proses sorting menggunakan Bubble Sort dihentikan karena telah melebihi batas waktu maksimal 20 menit yang telah ditentukan. Selection Sort dan Insertion Sort menunjukkan kinerja yang lebih stabil dibanding Bubble Sort, namun tetap jauh lebih lambat dibanding algoritma berbasis divide and conquer seperti Merge Sort dan Quick Sort. Algoritma Merge Sort, Quick Sort, dan Shell Sort menunjukkan performa yang sangat baik. Waktu eksekusi ketiga algoritma ini relatif konstan meskipun ukuran data bertambah hingga jutaan kata. Dari sisi penggunaan memori, Merge Sort tercatat menggunakan memori paling besar karena kebutuhan alokasi array tambahan dalam proses sorting. Sementara itu, Quick Sort dan Shell Sort lebih hemat dalam penggunaan memori.

b. Kesimpulan

Dari hasil pengujian, dapat disimpulkan bahwa pemilihan algoritma sorting sangat menentukan efisiensi proses pengurutan data, baik dari sisi waktu maupun memori.

Pada data integer, algoritma sederhana seperti Bubble Sort, Selection Sort, dan Insertion Sort menunjukkan keterbatasan performa, terutama pada dataset besar. Bubble Sort menjadi yang paling tidak efisien dengan waktu eksekusi yang sangat tinggi. Sebaliknya, Merge Sort, Quick Sort, dan Shell Sort mampu menjaga kecepatan sorting dengan penggunaan memori yang relatif efisien, meskipun Merge Sort membutuhkan memori tambahan yang lebih besar.

Pada data string, pola serupa ditemukan. Bubble Sort gagal menyelesaikan

sorting dalam batas waktu 20 menit pada dataset besar, sedangkan Selection Sort dan Insertion Sort juga kurang efektif. Merge Sort, Quick Sort, dan Shell Sort kembali menjadi pilihan utama, dengan waktu eksekusi yang cepat dan penggunaan memori yang lebih terkontrol.

Secara umum, Quick Sort dan Shell Sort menjadi algoritma paling direkomendasikan untuk sorting data berskala besar, karena mampu memberikan keseimbangan antara kecepatan dan efisiensi memori. Merge Sort juga layak dipilih jika kapasitas memori mencukupi. Sementara itu, algoritma sederhana lebih cocok untuk dataset kecil.