**LAPORAN TUGAS STRUKTUR DATA**



**Kelompok 2**

1) Handika Dio Pradana (21091397021)

2) Muhammad Fikri Ramadhana (21091397033)

3) Aditya Putra Pratama (21091397043)

4) A’Ahmes Osama Firmansyah (21091397051)

5) Irfan Rahmat Firmansyah (21091397063)

**A. Pendahuluan Laporan**

Berdasarkan pengembangan dan pengujian terhadap program yang telah dibuat dengan operasi sorting list yang terdiri dari insertion sort, selection sort, shell sort, bubble short, dan merge short dapat disimpulkan bahwa yang memiliki kecepatan yang lebih diantara ke lima tipe sorting list data yaitu merge sort dan yang terlambat bubble sort. Berikut tabel perhitungan berdasarkan detik (*s*).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sorting List** | N | **1** | **5** | **10** |
| Bubble sort | O(n2) | 1 | 25 | 100 |
| Selection sort | O(n2) | 1 | 25 | 100 |
| Intersention sort | O(n2) | 1 | 25 | 100 |
| Shell sort | O(n2) | 1 | 25 | 100 |
| Merge sort | O(n log n) | 2 | 11 | 33 |

Pada tabel nampak sama pada sorting list tipe bubble sort, selection sort, intesection sort, dan shell sort, namun nyatanya dari keempatnya terdapat perbedaan masing-masing yang tidak terlihat. Bubble sort memiliki interval waktu paling lama dikarenakan proses yang dilakukan membutuhkan banyak waktu untuk memproses *pass* terhadap pergeseran elemen. Kemudian selection sort dan intesection sort memiliki kemiripan yang hampir sama namun intersention sort memiliki kecepatan interval waktu 40% lebih banyak daripada selection sort. Sedangkan shell sort memiliki kompleksitas waktu 5 kali lebih cepat daripada bubble sort dan 2 kali lebih cepat daripada intersention sort, namun tipe sorting list ini memiliki kompleksitas algoritma yang cukup sulit.

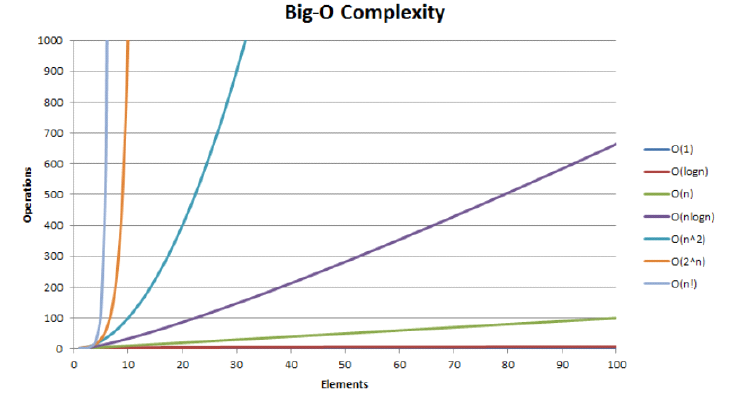
**B. Hasil Diskusi Mengenai Laporan Sorting List yang Tercepat**

Dalam tabel dapat dilihat jika sorting list merge sort memiliki kecepatan proses sorting data tercepat dibandingkan dengan yang lain. Merge sort menggunakan pola divide and conquer dengan melakukan pengurutan dengan memecah array menjadi 2 bagian secara rekursif hingga tersisa 1 elemen yang kemudian akan dihubungkan kembali membentuk data list yang telah diurutkan. Merge sort juga memiliki kelebihan, kelebihan ini jauh lebih cepat daripada algoritma pengurutan sederhana seperti bubble sort ketika berhadapan dengan kumpulan data besar. Tidak seperti beberapa algoritma, kompleksitas waktu terbaik, rata-rata dan terburuk adalah O(n log n). Namun Merge sort memiliki kekurangan seperti tidak sesederhana algoritma pengurutan yang tidak terlalu rumit. Misalkan anda dalam berurusan dengan array 20 item. Sehingga perbedaan antara kompleksitas waktu akan sangat nominal. Juga, tidak seperti algoritma pengurutan yang lebih sederhana, kompleksitas ruangnya adalah O(n), di mana kompleksitas ruang Bubble Sort adalah O(1).

**C. Hasil Diskusi Mengenai Laporan Sorting List yang Terlambat**

Jika dilihat dalam tabel dan berdasarkan diskusi yang telah dilakukan, maka sorting list bubble sort yang menempati tipe terlambat diantara kelima tipe data sorting list. Bubble sort dapat dilihat dari beberapa jenis kasus, yaitu worst-case, average-case, dan best-case. Kondisi *best-case* data yang akan disorting telah terurut sebelumnya, sehingga proses perbandingan hanya dilakukan sebanyak (n-1) kali, dengan satu kali proses tahapan. Persamaan notasi Big-O yang diperoleh dari proses bubble sort *best-case* yaitu O(n). Dengan kata lain, pada kondisi Best-Case algoritma Bubble Sort termasuk pada algoritma lancar. Namun dalam kondisi *worst-case,* data terkecil berada pada ujung *array* yang dimana setiap kali melakukan satu pass, data terkecil akan bergeser ke arah awal sebanyak satu pass elemen saja. Dengan kata lain, untuk menggeser data terkecil dari urutan keempat menuju urutan pertama, dibutuhkan pass sebanyak tiga kali, ditambah satu kali pass untuk memverifikasi dan persamaan notasi big-O nya yang didapat berupa O(n2). Adapun kelebihan yang dimiliki dari sorting list tipe Bubble sort ini yaitu perhitungannya memakai metode yang sederhana, algoritma yang mudah dipahami, dan langkah beserta tahapan yang begitu mudah.

**D. Diagram Kompleksitas Big-O**

****