

LAPORAN STRUKTUR DATA

KELOMPOK 6

ANALISIS SORTINGAN PALING CEPAT DAN PALING LAMBAT DIANTARA INSERTION SORT, SELECTION SORT, MERGE SORT, BUBBLE SORT, DAN RADIX SORT

Anggota Kelompok :

- 1. FIFIT SYAFAATY (21091397001)**
- 2. NUR HASLINDA (21091397035)**
- 3. RIZQI CAHYA ANGELITA (21091397047)**
- 4. DWI RAMADHANIASARI (21091397057)**
- 5. PUTRI DWINATRYSKA A.R.F (21091397075)**

Konsep / Cara Kerja :

- **Insertion Sort**

Insertion sort merupakan teknik pengurutan dengan cara membandingkan dan mengurutkan dua data pertama pada array, kemudian membandingkan data array berikutnya apakah sudah berada pada tempat semestinya. Cara kerjanya yaitu dengan mengambil elemen list satu per-satu dan memasukkannya di posisi yang benar. Pengurutan insertion sort merupakan pengurutan data dengan membandingkan dua elemen data pertama, kemudian membandingkan elemen-elemen data yang sudah diurutkan, perbandingan akan terus diulang hingga tidak ada data yang tersisa. Untuk menghemat memori, implementasi insertion sort menggunakan pengurutan di tempat yang membandingkan elemen saat itu dengan elemen sebelumnya yang sudah diurut, lalu menukarnya terus sampai posisinya tepat.

- **Selection Sort**

Algoritma Selection Sort pengurutannya dengan melakukan beberapa kali pass untuk melakukan penyeleksian elemen struktur data. Untuk sorting ascending (menaik), elemen yang paling kecil di antara elemen-elemen yang belum urut, disimpan indeksinya, kemudian dilakukan pertukaran nilai elemen dengan indeks yang disimpan tersebut dengan elemen yang paling depan yang belum urut. Sebaliknya, untuk sorting descending (menurun), elemen yang paling besar yang disimpan indeksinya kemudian ditukar.

Algoritma ini bekerja sebagai berikut:

1. Mencari nilai minimum (jika ascending) atau maksimum (jika descending) dalam sebuah list
2. Menukarkan nilai ini dengan elemen pertama list

3. Mengulangi langkah di atas untuk sisa list dengan dimulai pada posisi kedua

- Merge Sort

Algoritma Merge Sort adalah salah satu algoritma modern yang mirip seperti algoritma Quick Sort. Keduanya sama-sama menggunakan metode Divide and Conquer. dimana sebuah list akan dipecah menggunakan fungsi rekursif pertama kita akan membuat 2 blok dari sebuah list, dengan cara membelahnya menjadi 2 bagian sama rata, jika ternyata list tersebut jumlahnya ganjil, maka akan dibulatkan. setelah menjadi 2 blok, masing masing blok akan di pecah kembali menggunakan fungsi rekursiv hingga setiap blok hanya memiliki 1 index list. jika sudah, kita akan menyatukan 2 blok menjadi satu dan melakukan sorting antara 2 elemen blok tadi, jika ascending maka index dengan angka paling kecil akan berada disebelah kiri dan sebaliknya. kita akan mengulang proses perbandingan tersebut hingga hanya tersisa 1 blok dengan index yang sudah tersorting

- Bubble Sort

Bubble Sort adalah salah satu algoritma untuk sorting data, atau kata lainnya mengurutkan data dari yang terbesar ke yang terkecil atau sebaliknya (Ascending atau Descending). Bubble sort menggunakan metode/algoritma pengurutan dengan dengan cara melakukan penukaran data secara terus menerus sampai sampai tidak ada lagi perubahan. Jika tidak ada perubahan berarti data sudah terurut.

Metode pengurutan gelembung (Bubble Sort) diinspirasi oleh gelembung sabun yang berada dipermukaan air. Karena berat jenis gelembung sabun lebih ringan daripada berat jenis air, maka gelembung sabun selalu terapung ke atas permukaan. Prinsip di atas dipakai pada pengurutan gelembung.

Algoritma bubble sort adalah salah satu algoritma pengurutan yang paling simple, baik dalam hal pengertian maupun penerapannya. Ide dari algoritma ini adalah mengulang proses perbandingan antara tiap-tiap elemen array dan menukarnya apabila urutannya salah. Perbandingan elemen-elemen ini akan terus diulang hingga tidak perlu dilakukan penukaran lagi. Algoritma ini termasuk dalam golongan algoritma comparison sort, karena menggunakan perbandingan dalam operasi antar elemennya.

- Radix Sort

Radix Sort adalah algortima atau bahkan strategi pengurutan (sorting) tanpa perbandingan bersama istilah satu lagi, sorting Non-Comparasion type dimana berat prosesnya tidak mengadakan evaluasi antar information. Kata radix relevan harafiah penempatan berat angka. Di mana sederhananya, berat lukisan desimal, radix adalah digitnya. Dalam implementasinya, Radix Sort merupakan algoritma pengurutan yang segera, nyaman, dan secara substansial membantu. Namun tak terhitung yangmengira bahwa algoritma radix {memiliki} tak terhitung pembatasan di mana untuk kasus-kasus tertentu tidak dapat dilakukan bersama algoritma ini, seperti pengurutan bilangan bagian dan bilangan unfavourable.

Proses sederhana Radix Sort adalah mengkategorikan data-data akhirnya menjadi sub kumpulan information ideal bersama nilai pasar radix-nya (klasifikasi tertentu), dimana berat tiap kategorinya dilakukan pengklasifikasian bahkan lebih dan seterusnya ideal bersama butuh mengkonkatenasinya, dan subkategori-subkategori tersebut digabungkan dapatkan, yang secara dilakukan seluruhnya bersama strategi kecil concatenation.

1. Jelaskan sortiran mana yang paling cepat dan jelaskan alasannya !

Sortiran yang paling cepat adalah merge sort, karena algoritma merge sort selalu membagi 2, maka lebih mudah jika kita memisalkan bahwa n adalah selalu pangkat dari 2. Merge sort memiliki waktu pengerjaan yang konsisten dan algoritma yang efisien. Selanjutnya diikuti oleh sortiran insertion sort, karena algoritmanya stabil pada jumlah elemen yang sedikit dibandingkan dengan bubble sort. Namun, tetap kurang efisien untuk tabel berukuran besar (menyimpan banyak nilai). Lalu selanjutnya, selection sort lebih cepat mengurutkan data dibandingkan algoritma bubble Sort. Hal ini ditunjukkan dengan kecilnya nilai yang didapat oleh algoritma tersebut.

2. Jelaskan sortiran mana yang paling lambat dan jelaskan alasannya!

Sortiran paling lambat adalah bubble sort, karena proses pengurutan data dilakukan secara satu persatu. sehingga ketika data yang dieksekusi oleh program dalam jumlah banyak akan membutuhkan waktu yang relatif lama dan lambat. Selanjutnya diikuti oleh radix sort, karena tidak dapat diterapkan secara universal, biasanya harus disesuaikan dengan penggunaan aktual dan menggunakan banyak memori tambahan.