1. SORTING

PENGERTIAN SORTING

 sorting  adalah sebuah metode untuk pengurutan data, misalnya dari data yang terbesar ke data yang terkecil.

Artinya jika ada deretan data, maka data yang pertama akan membandingkan dengan data yang kedua. Jika data yang pertama lebih besar dari pada data yang kedua maka data yang pertama akan bertukar posisi dengan data yang kedua, begitu seterusnya sampai benar-benar data terurut dari yang terbesar hingga yang terkecil.

Metode sorting sangat banyak dan berkembang ada Bubble sort, Selection Sort, Insertion sort, Merge sort, Quick sort. Namun tidak semua metode sorting ini efektif. Karena metode sorting yang paling efektif adalah ketika metode tersebut dapat melakukan pengurutan data dengan cepat dan tidak memerlukan banyak memori

1. INSERTION SORT

Algoritma insertion sortpada dasarnya memilah data yang akan diurutkan menjadi dua bagian, yang belum diurutkan dan yang sudah diurutkan. Elemen pertama diambil dari bagian array yang belum diurutkan dan kemudian diletakkan sesuai posisinya pada bagian lain dari array yang telah diurutkan. Langkah ini dilakukan secara berulang hingga tidak ada lagi elemen yang tersisa pada bagian array yang belum diurutkan.

1. SELECTION SORT

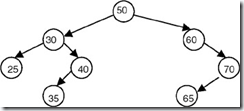
Ide utama dari algoritma selection sortadalah memilih elemen dengan nilai paling rendah dan menukar elemen yang terpilih dengan elemen ke-i. Nilai dari idimulai dari 1 ke n, dimana nadalah jumlah total elemen dikurangi 1.

1. BUBLE SORT

Merupakan algoritma pengurutan paling tua dengan metode pengurutan paling sederhana. Pengurutan yang dilakukan dengan membandingkan masing-masing itemdalam suatu list secara berpasangan, menukar itemjika diperlukan, dan mengulaginya sampai akhir list secara berurutan, sehingga tidak ada lagi item yang dapat ditukar.

1. **BINARY SEARCH TREES**

sebuah binary search tree (bst) adalah sebuah pohon biner yang boleh kosong, dan setiap nodenya harus memiliki identifier/value. value pada semua node subpohon sebelah kiri adalah selalu lebih kecil dari value dari root, sedangkan value subpohon di sebelah kanan adalah sama atau lebih besar dari value pada root, masing – masing subpohon tersebut (kiri&kanan) itu sendiri adalah juga bst. lihat gambar:

[](https://jampasir.files.wordpress.com/2009/01/image.png)

1. OPERASI BINARY SEARCH TREE
2. INSERT

Penyisipan sebuah  node baru, didahului dengan operasi pencarian posisi yang sesuai. Dalam hal ini node baru tersebut akan menjadi daun/leaf.

|  |
| --- |
|  |

1. DELETE

Operasi delete memiliki 3 kemungkinan :

* Delete terhadap node tanpa anak/child (leaf/daun) : node dapat langsung dihapus
* Delete terhadap node dengan satu anak/child : maka node anak akan menggantikan posisinya.
* Delete terhadap node dengan dua anak/child : maka node akan digantikan oleh node paling kiri dari Sub Tree Kanan atau dapat juga digantikan oleh anak paling kanan dari Sub Tree Kiri.

1. SEARCHING

Operasi searching memiliki 3 kemungkinan :

* Jika YANG DICARI = PARENT, pencarian selesai.
* Jika YANG DICARI lebih BESAR dari PARENT, search akan berjalan ke bagian kanan parent, berlaku sebaliknya
* Jika tidak ditemukan dibagian left sub tree, melakukan pencarian di right sub tree secara rekursif, berlaku sebaliknya