Nama : Shabinna Rahmadilla Santoso

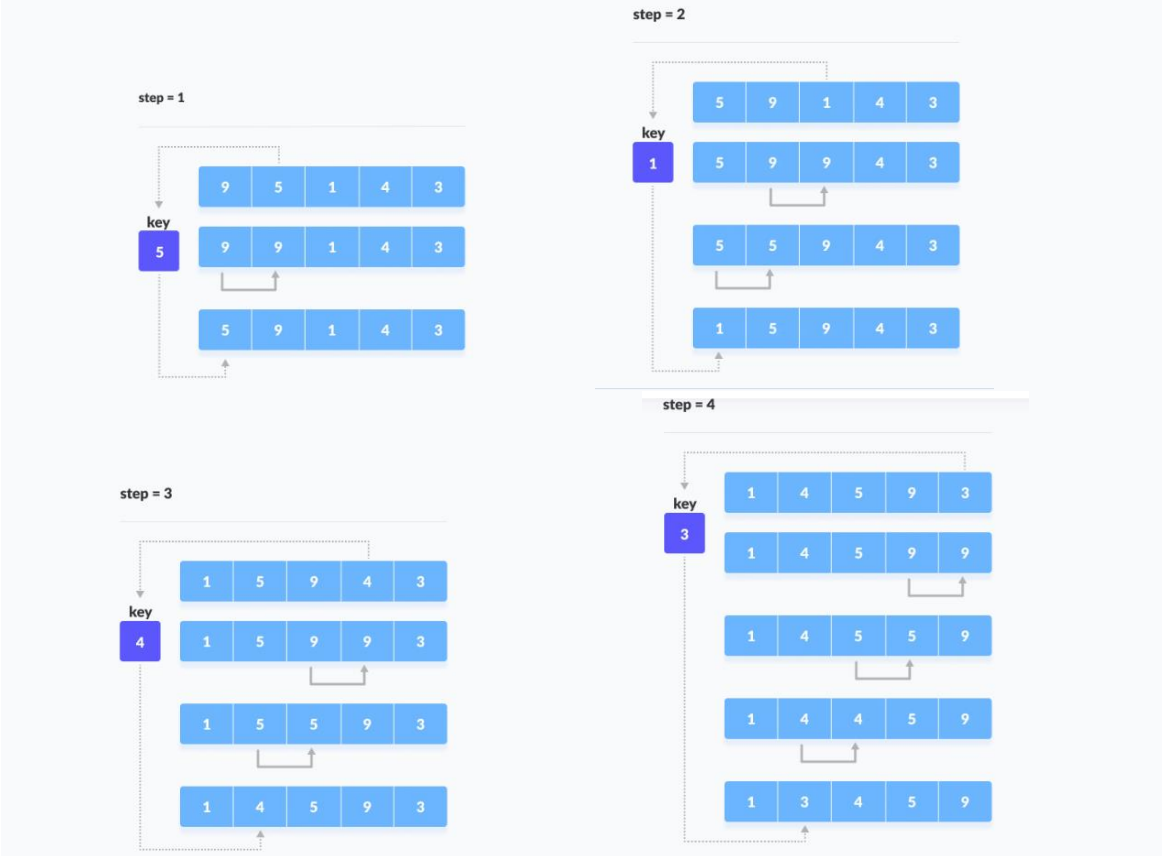
Nim : 21091397004

Kelas : 2021B

Insertion Sort

Insertion sort adalah algoritma pengurutan yang menempatkan elemen yang tidak disortir pada tempatnya yang sesuai dalam setiap iterasi. Pengurutan penyisipan bekerja sama seperti kita mengurutkan kartu di tangan kita dalam permainan kartu. Jika kartu pertama sudah diurutkan kemudian, memilih kartu yang tidak disortir. Jika kartu yang tidak disortir lebih besar dari kartu yang ada di tangan, maka kartu tersebut ditempatkan di sebelah kanan, sebaliknya, di sebelah kiri. Dengan cara yang sama, kartu lain yang tidak disortir diambil dan diletakkan di tempat yang tepat. Pendekatan serupa digunakan oleh insertion sort.

- Cara Kerja Insertion Sort



- Pemograman C++ Insertion Sort

```cpp
// Shabinna Rahmadilla Santoso - 004 - 2021B
//Insertion Sort

#include <iostream>
using namespace std;

// Fungsi Mengurutkan Array Menggunakan Insertion Sort
int main()
{
    //Proses Memasukkan Array
    int data_array;
    cout << " Masukkan Jumlah Array : ";
    cin >>  data_array;
    int array [data_array]; //Deklarasi Array

    // Mencetak Angka
    // Proses Memasukkan Angka
    for ( int i = 0; i < data_array; i++ ){
        cout << "Masukkan Angka Ke Indeks " << i << " : ";
        cin >> array [i];
        cout << endl;
    }

    // Proses Penyortingannya
    // Membandingkan kunci ke setiap elemen sebelah kiri hingga elemen terkecil
    for ( int i = 1; i < data_array; i++ ){
        int key = array [i]; //Menandakan keberadaan angka, Menentukkan proses
        int j = i - 1;
        while ( j >= 0 && array [j] > key){
            array [j+1] = array [j];
            j--; //Untuk menghentikan looping jika angka minus
        }
        array [j+1] = key; //Memasukkan array di tempat yang benar
    }

    // Proses Menampilkan Hasil Penyortingan dengan Insertion Sort
    cout <<   " Hasil Penyortingann" << endl;
    for ( int m = 0; m <  data_array; m++){
        cout << array [m] << " ";
    }
    return 0;
}
```

- Penjelasan Pemograman

//Proses Memasukkan Array
int data_array;
cout << " Masukkan Jumlah Array : ";
cin >> data_array;
int array [data_array]; //Deklarasi Array
	- Membuat variabel data array dengan tipe integer
	- Cin >> dataarray menandakan jumlah berapa angka yang akan di sorting
	- Mendeklarasikan array dengan tipe integer, sehingga jumlah array sesuai dengan yang kita masukkan (array dinamis)

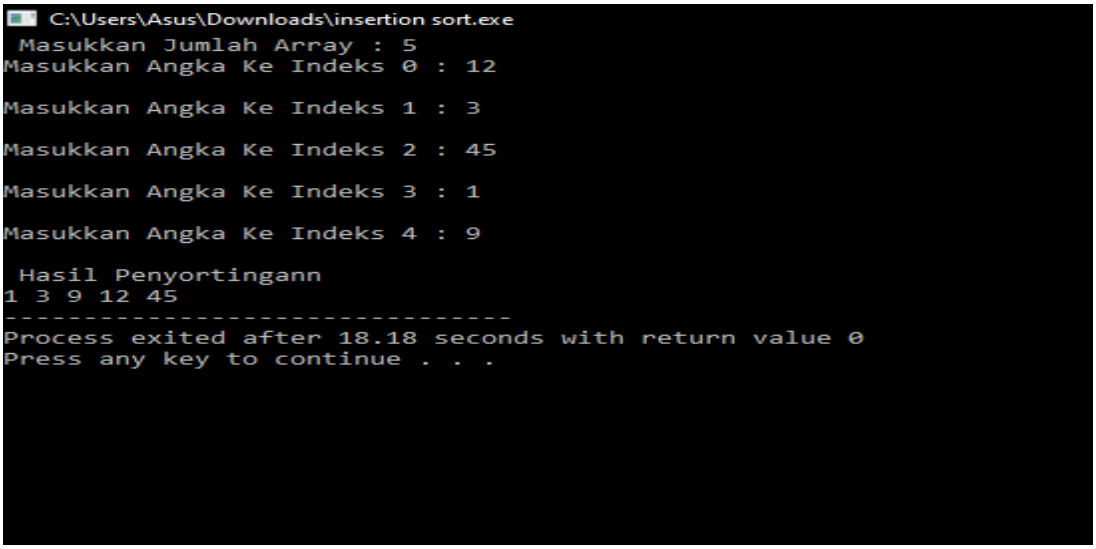
// Mencetak Angka
// Proses Memasukkan Angka
for (int i = 0; i < data_array; i++){
cout << "Masukkan Angka Ke Indeks " << i << " : ";
cin >> array [i];
cout << endl;
}
	- i dimisalkan adalah angka looping 1
	- disini ada proses memasukkan angka ke dalam array, jadi kita bisa memasukkan angka berapa aja

// Proses Penyortingannya
// Membandingkan kunci ke setiap elemen sebelah kiri hingga elemen terkecil
for (int i = 1; i < data_array; i++){
int key = array [i]; //Menandakan keberadaan angka, Menentukkan proses
int j = i - 1;
while (j >= 0 && array [j] > key){
	- bagian ini masuk proses per-loopingan
	- I<data_array, agar tidak melebihi jumlah array yang kita punya
	- int key = array [i] untuk menandakkan keberadaan angka, menentukkan sedang dimana prosesnya. = array agar tahu sedang di array indeks ke berapa
	- j dimisalkan angka looping 2
	- j = i – 1 atau indeks ke-0, untuk membandingkan 2

array [j+1] = array [j];
j--; //Untuk menghentikan looping jika angka minus
}
array [j+1] = key; //Memasukkan array di tempat yang benar
}
	variabel
	- proses ini nilai akan bertukar jika nilai kiri lebih besar dari kanan, j >= 0 && array [j] > key, jika j lebih besar dari 0 dan array j lebih besar dari key maka akan bertukar
	- array [j+1] = array [j], indeks 1 sama dengan indeks 0 (proses bertukar)
	- j--, ketika hasil minus dia akan keluar dari looping
	- array [j+1] = key adalah perubahan posisi key disetiap loopingan, menetapkan key baru, menjadikan key ke variabel kanan

// Proses Menampilkan Hasil Penyortingan dengan Insertion Sort
cout << " Hasil Penyortingann" << endl;
for (int m = 0; m < data_array; m++){
cout << array [m] << " ";
}
return 0;
}
	- m adalah angka looping 3 ata angka hasil
	- proses terakhir untuk menampilkan hasil looingan

- Hasil Output Insertion Sort



- Kelebihan dan Kekurangan

Kelebihan dari Insertion Sort

1. Sederhana dalam penerapannya.
2. Mangkus dalam data yang kecil.
3. Jika list sudah terurut atau sebagian terurut maka Insertion Sort akan lebih cepat dibandingkan dengan Quick sort.
4. Mangkus dalam data yang sebagian sudah terurut.
5. Lebih mangkus dibanding Bubble Sort dan Selection Sort.
6. Loop dalam pada Inserion Sort sangat cepat, sehingga membuatnya salah satu algoritma pengurutan tercepat pada jumlah elemen yang sedikit.

Kekurangan dari Insertion Sort

1. Banyaknya operasi yang diperlukan dalam mencari posisi yang tepat untuk elemen larik.
2. Untuk larik yang jumlahnya besar ini tidak praktis.
3. Jika list terurut terbalik sehingga setiap eksekusi dari perintah harus memindai dan mengganti seluruh bagian sebelum menyisipkan elemen berikutnya.
4. Membutuhkan waktu O(n2) pada data yang tidak terurut, sehingga tidak cocok dalam pengurutan elemen dalam jumlah besar.