

Widya Sari Wibowo
21091397070

LAPORAN INDIVIDU TUGAS 1

SELECTION SORT

Selection Sort

```
#include<iostream>
```

```
using namespace std;
```

```
// SELECTION SORT
```

```
int main() {
```

```
    int angka, tukar_angka, data[100];
```

```
    cout << "Masukkan jumlah data: ";
```

```
    cin >> angka;
```

```
    // Perulangan for dibawah ini berguna untuk melakukan perulangan data  
    sesuai jumlah data yang kita masukkan
```

```
    for (int i = 0; i < angka; i++) {
```

```
        cout << "Data ke-" << i + 1 << " = " ;
```

```
        cin >> data[i];
```

```
        cout << endl;
```

```
    }
```

```
    // Perulangan Selection Sort
```

```
    for (int i = 0; i < angka-1; i++) {
```

```
        tukar_angka = i;
```

```
        int angka_final;
```

```
        for(int j = i+1; j < angka; j++){
```

```
            if(data[j] < data[tukar_angka]){
```

```
                tukar_angka = j;
```

```
            }
```

```
        }
```

```
        angka_final = data[tukar_angka];
```

```
        data[tukar_angka] = data[i];
```

```

data[i] = angka_final;

// Untuk menampilkan proses sorting
cout << "Proses Sorting" << endl;
    for(int m = 0; m < angka; m++){
        cout << data[m] << " ";
    }
    cout<<endl;
}

// Untuk menampilkan hasil sorting
cout << endl << "Hasil Sorting" << endl;
for(int i = 0; i < angka; i++){
    cout << data[i] << " ";
}

cin.get();
return 0;
}

```

Penjelasan

```

7      int angka, tukar_angka, data[100];
8
9      cout << "Masukkan jumlah data: ";
10     cin >> angka;

```

Pada baris ke 7 terdapat variabel yang bertuliskan **int angka, tukar_angka, data[100]** **Variabel angka** berguna untuk menampung jumlah data yang nantinya di inputkan oleh user.

Variabel tukar_angka berfungsi untuk menukar isi data dalam array sesuai instruksi program.

Variabel data berfungsi untuk membatasi jumlah data array yang dimasukkan oleh user. Misalnya kita

Pada baris ke-10 terdapat **Cin** yang berfungsi untuk menerima inputan dari user dan kemudian dimasukkan ke dalam variable angka.

```

13     for (int i = 0; i < angka; i++) {
14         cout << "Data ke-" << i + 1 << " = " ;
15         cin >> data[i];
16
17         cout << endl;
18     }

```

Perulangan for ini berguna untuk melakukan perulangan data sesuai jumlah data yang kita masukkan.

Pada baris ke-15 terdapat **Cin** yang berfungsi untuk menerima inputan data yang dimasukkan oleh user dan kemudian ditampung oleh variabel data dengan index ke I, index i ini, didapat dari perulangan for pada baris ke 13.

```
20 // Perulangan Selection Sort
21 for (int i = 0; i < angka-1; i++) {
22     tukar_angka = i;
23     int angka_final;
24     for(int j = i+1; j < angka; j++){
25         if(data[j] < data[tukar_angka]){
26             tukar_angka = j;
27         }
28     }
29
30     angka_final = data[tukar_angka];
31     data[tukar_angka] = data[i];
32     data[i] = angka_final;
33 }
```

Variabel tukar_angka berfungsi untuk menampung perulangan selection sort yang ditandai dengan variabel integer i.

Variabel angka_final berfungsi untuk menampung index array yang sudah di urutkan dari angka yang terkecil sampai ke yang besar.

Variabel data yang menyimpan index array dari variabel tukar angka berfungsi untuk menampung variabel data.

Data[i] berfungsi untuk menampung angka final yang sudah ditukar sesuai index yang sudah dilakukan perulangan pada baris 20 – 28.

```
34 // Untuk menampilkan proses sorting
35 cout << "Proses Sorting" << endl;
36 for(int m = 0; m < angka; m++){
37     cout << data[m] << " ";
38 }
39 cout<<endl;
40 }
41 // Untuk menampilkan hasil sorting
42 cout << endl << "Hasil Sorting" << endl;
43 for(int i = 0; i < angka; i++){
44     cout << data[i] << " ";
45 }
46
47 cin.get();
48 return 0;
49 }
```

Pada baris ke 34-40 berfungsi untuk menampilkan proses sorting.

Pada baris ke 41-45 berfungsi untuk menampilkan hasil sorting.

Pengurutan pada algoritma selection sort, dilakukan dengan cara mencari suatu nilai ekstrim (minimal atau maksimal) untuk ditukarkan dengan elemen terujung yang ada pada suatu proses loop. Elemen terujung ini tidak akan dimasukkan pada proses loop berikutnya karena nilai terujung ini sudah menjadi nilai maksimum atau minimum. Proses ini dilakukan juga terhadap sisa dari elemen array.

Output Hasil Run Codingan

```
C:\Users\widya\OneDrive\Documents\SMT 2\Struktur Data\Tugas Praktikum\Selection Sortt.exe
Masukkan jumlah data: 10
Data ke-1 = 6
Data ke-2 = 12
Data ke-3 = 33
Data ke-4 = 2
Data ke-5 = 9
Data ke-6 = 18
Data ke-7 = 30
Data ke-8 = 27
Data ke-9 = 1
Data ke-10 = 24

Proses Sorting
1 12 33 2 9 18 30 27 6 24
Proses Sorting
1 2 33 12 9 18 30 27 6 24
Proses Sorting
1 2 6 12 9 18 30 27 33 24
Proses Sorting
1 2 6 9 12 18 30 27 33 24
Proses Sorting
1 2 6 9 12 18 30 27 33 24
Proses Sorting
1 2 6 9 12 18 30 27 33 24
Proses Sorting
1 2 6 9 12 18 24 27 33 30
Proses Sorting
1 2 6 9 12 18 24 27 33 30
Proses Sorting
1 2 6 9 12 18 24 27 30 33

Hasil Sorting
1 2 6 9 12 18 24 27 30 33
-----
```

Big O Selection Sort

Karena Selection Sort terdiri dari perulangan bersarang. Dimana Perulangan tingkat pertama (disebut pass) berlangsung $N-1$ kali. Di dalam perulangan kedua, dicari elemen dengan nilai terkecil. Jika didapat, indeks yang didapat ditimpakan ke variabel. Lalu dilakukan proses penukaran. Begitu seterusnya untuk setiap Pass. Pass sendiri makin berkurang hingga nilainya menjadi semakin kecil. Berdasarkan operasi perbandingan elemennya:

$$T(n) = (n-1) + (n-2) + \dots + 2 + 1 = \sum_{i=1}^{n-1} n - i$$

$$= \frac{n(n-1)}{2} = O(n^2)$$

Berdasarkan kodingan diatas, berapa kali cout array akan dijalankan dengan nilai n sebagai berikut:

- n = 1, maka akan dijalankan 1 kali
- n = 5, maka akan dijalankan 5 kali
- n = 10, maka akan dijalankan 10 kali
- dan seterusnya...

sehingga cout array akan dijalankan sebanyak

$$1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 + \dots + n = \frac{n^2 + n}{2}$$

Kelebihan dan Kekurangan Selection Sort

Kelebihan:

1. Algoritma nya mudah untuk diimplementasikan.
2. Mudah menentukan nilai maksimum / minimum.
3. Mudah melakukan pertukaran nilai.
4. Pertukaran hanya dilakukan sekali.

Kekurangan:

1. Waktu pengecekan lama.