

## SELECTION SORT

```
1 //program sortiran menggunakan selection sort
2 #include<iostream>
3 using namespace std;
4
5 int main(){
6
7 //inialisasi variable
8     int jumlah_data, tukar, nilai[100];
9 //inialisasi jumlah angka yang ingin di sorting
10    cout << "Masukkan jumlah data: ";
11    cin >> jumlah_data;
12
13 //--proses looping awal penginputan angka yang akan di masukkan--
14    for (int loop = 0; loop < jumlah_data; loop++){
15        cout << "Data ke-" << loop+1 << " = " ;
16        cin >> nilai[loop];
17        cout << endl;
18    }
19
20 //--proses looping selection sort--
21    for (int loop = 0; loop < jumlah_data-1; loop++) {
22        tukar = loop;
23        int temp;
24
25        for(int j = loop+1; j < jumlah_data; j++){
26            if(nilai[j] < nilai[tukar]){
27                tukar = j;
```

```

        for(int j = loop+1; j < jumlah_data; j++){
            if(nilai[j] < nilai[tukar]){
                tukar = j;
            }
        }
        temp = nilai[tukar];
        nilai[tukar] = nilai[loop];
        nilai[loop] = temp;
    }

    //--looping untuk menampilkan output yang sudah di sorting--
    for(int i = 0; i < jumlah_data; i++){
        cout << nilai[i] << " ";
    }
    cin.get();
    return 0;
}

```

#### PENJELASAN

1. perintah cout masukkan jumlah data, untuk memasukkan jumlah data yang akan disort.
2. for loop pertama, menginputkan data sebanyak jumlah data yang kalian inginkan. contoh : Data ke-1 = 4, dan seterusnya.
3. for loop kedua, memproses data dengan selection sort. Saat for loop luar, perulangan indeks data. Saat for loop ke dua proses menukarkan data, apabila data setelahnya lebih kecil maka akan diswap data sebelumnya sampai persyaratan terpenuhi.
4. for loop ketiga menampilkan hasil akhir data secaraurut.

#### Selection Sort: contoh

	1	2	3	4	5	6	
1	5	2	4	6	1	3	Carilah elemen terkecil & tukar dengan "5"
2	1	2	4	6	5	3	1 fixed. Carilah elemen terkecil & tukar dengan "2"
3	1	2	4	6	5	3	1,2 fixed. Carilah elemen terkecil & tukar dengan "4"
4	1	2	3	6	5	4	1,2,3 fixed. Carilah elemen terkecil & tukar dengan "6"
5	1	2	3	4	5	6	1,2,3,4 fixed. Carilah elemen terkecil & tukar dengan "5"
6	1	2	3	4	5	6	1,2,3,4,5 fixed, otomatis elemen terakhir sudah pada posisi yang benar

#### Kelebihan

1. Algoritma ini sangat rapat dan mudah untuk diimplementasikan.
2. Operasi pertukarannya hanya dilakukan sekali saja.
3. Waktu pengurutan dapat lebih ditekan.
4. Mudah menggabungkannya kembali.
5. Kompleksitas selection sort relatif lebih kecil.

#### Kekurangan

1. Membutuhkan method tambahan.
2. Sulit untuk membagi masalah