

Laporan individu

Selection sort

```
1  #include<iostream>
2  using namespace std;
3
4  int main() {
5
6  // Variable terdiri dari jumlah_array yang berisi jumlah array yang akan diurutkan, tukar digunakan untuk mengurutkan data, dan isi adalah data yang akan diurutkan
7  int jumlah_array, tukar, isi[100];
8
9  // input jumlah array
10 cout << "Masukkan jumlah data: ";
11 cin >> jumlah_array;
12
13 // menginputkan data sebanyak jumlah array
14 for (int i = 0; i < jumlah_array; i++) {
15     cout << "Data ke-" << i + 1 << " = ";
16     cin >> isi[i];
17     cout << endl;
18 }
19
20 // proses perulangan data
21 for (int i = 0; i < jumlah_array-1; i++) {
22     tukar = i;
23
24 // proses menukarkan data, jika data setelahnya lebih kecil maka akan ditukar dengan data sebelumnya sampai persyaratan terpenuhi
25 int temp;
26
27 for(int j = i+1; j < jumlah_array; j++){
28     if(isi[j] < isi[tukar]){
29         tukar = j;
30     }
31 }
32
33 temp = isi[tukar];
34 isi[tukar] = isi[i];
35
36     isi[i] = temp;
37 }
38
39 // menampilkan output data secara urut
40 cout << "Deret array yang sudah disortir: ";
41 for(int i = 0; i < jumlah_array; i++){
42     cout << "[" << isi[i] << " ";
43 }
44
45 cin.get();
46
47 return 0;
48 }
49
50 }
```

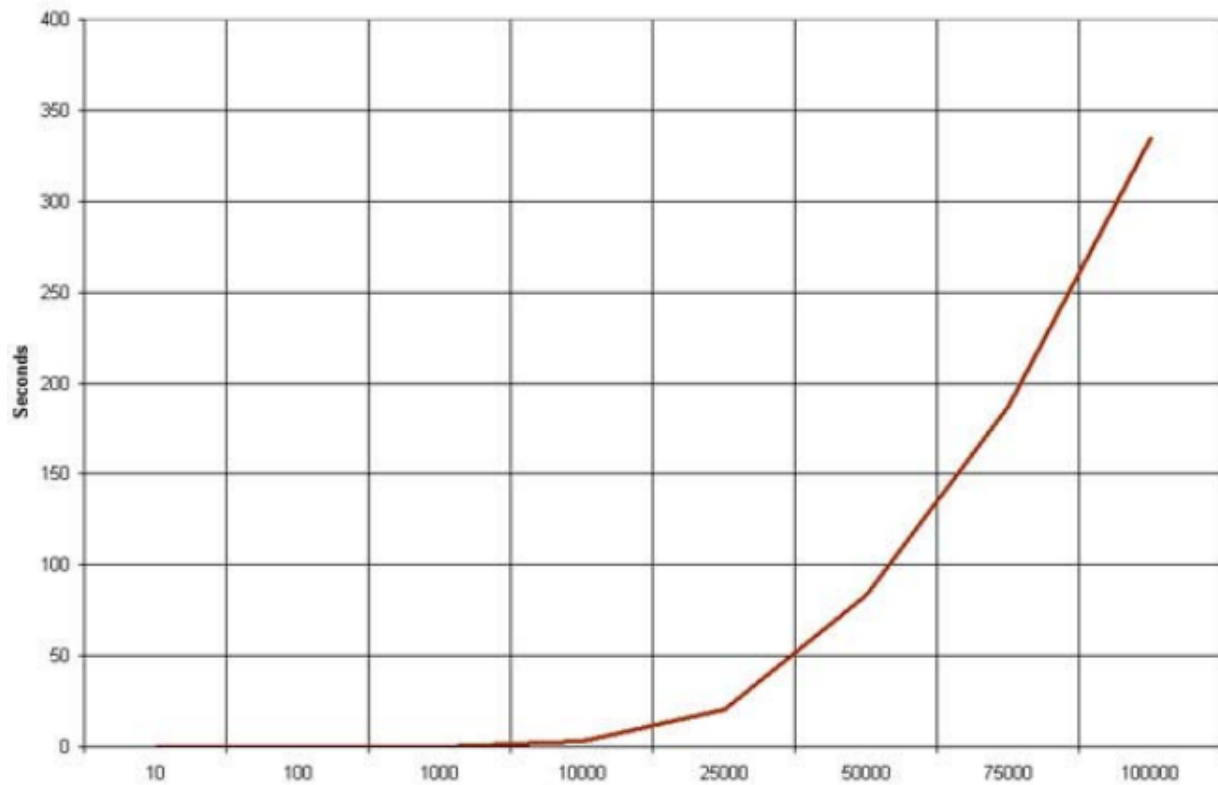
Kelebihan Selection sort:

- Algoritmanya mudah diimplementasikan.
- Mudah menentukan data maksimum dan minimum.
- Kompleksitas selection sort relatif kecil.

Kelemahan Selection sort:

- Kompleksitas selection sort akan meningkat dan tidak praktis jika jumlah data mencapai 1000.
- Membutuhkan metode tambahan.
- Dibandingkan dengan insertion sort, algoritma dari selection memiliki kesamaan dalam mudahnya diimplementasikan namun memiliki performa yang lebih buruk dari insertion sort.

Kompleksitas selection sort berdasarkan jumlah data



Hasil run:

```
Masukkan jumlah data: 5
Data ke-1 = 4
Data ke-2 = 2
Data ke-3 = 3
Data ke-4 = 5
Data ke-5 = 1
Deret array yang sudah disortir: [1][2][3][4][5]
-----
Process exited after 15.09 seconds with return value 0
Press any key to continue . . .
```