

Nama : Zabrina Rulif Aurellia

Nim : 21091397056

Kelas : 2021B

Laporan Individu

➤ Selection Sort

- Selection sort merupakan metode pengurutan yang menyeleksi satu per satu data yang ada dalam array 1 dimensi. Cara yang dilakukan dengan metode ini ialah membandingkan data satu persatu dalam array satu dimensi.
- Algoritma Selection Sort :
 1. Menemukan elemen terkecil dalam array kita dan menukarnya (*swap*) dengan elemen yang ada di posisi pertama
 2. Kemudian algoritma ini akan mengulangi hal yang sama lagi yaitu mencari elemen terkecil yang ada di dalam array dan kemudian menukarnya (*swap*) dengan elemen yang ada di posisi kedua (mengingat elemen di posisi pertama sudah berhasil kita *sorting*)
 3. Proses ini akan terus berlanjut sampai semua elemen yang ada di dalam array telah berhasil kita *sorting*.
- Contoh proses sorting Selection Sort

9	4	7	3	8	2
---	---	---	---	---	---

Anggap-lah kita mempunyai 6 elemen array yang masih tidak berurutan seperti ini

9	4	7	3	8	2
---	---	---	---	---	---

Selection sort akan mencari elemen terkecil dari array yang kita punya, dimana disini elemen terkecil adalah 2

2	4	7	3	8	9
---	---	---	---	---	---

Setelah berhasil mendapatkan elemen terkecil, maka selection sort ini akan menukar elemen tersebut ke posisi pertama (9 dan 2 ditukar)

2	4	7	3	8	9
---	---	---	---	---	---

Proses pengurutan pertama berhasil dilakukan, maka selection sort ini akan mengulangi lagi hal yang sama, yaitu mencari elemen terkecil yang ada di dalam array kita

2	4	7	3	8	9
---	---	---	---	---	---

Ternyata 3 merupakan elemen terkecil yang ada di dalam array, maka selection sort akan menukar posisi elemen 3 ke posisi kedua (mengingat kita sudah berhasil meletakkan elemen terkecil di posisi pertama)

2	3	7	4	8	9
---	---	---	---	---	---

Elemen 3 sudah berhasil kita tukar ke posisi kedua, maka dengan itu proses pengurutan kedua berhasil kita lakukan dan selection sort akan melanjutkannya ke proses pengurutan selanjutnya

2	3	7	4	8	9
---	---	---	---	---	---

2	3	4	7	8	9
---	---	---	---	---	---

2	3	4	7	8	9
---	---	---	---	---	---

2	3	4	7	8	9
---	---	---	---	---	---

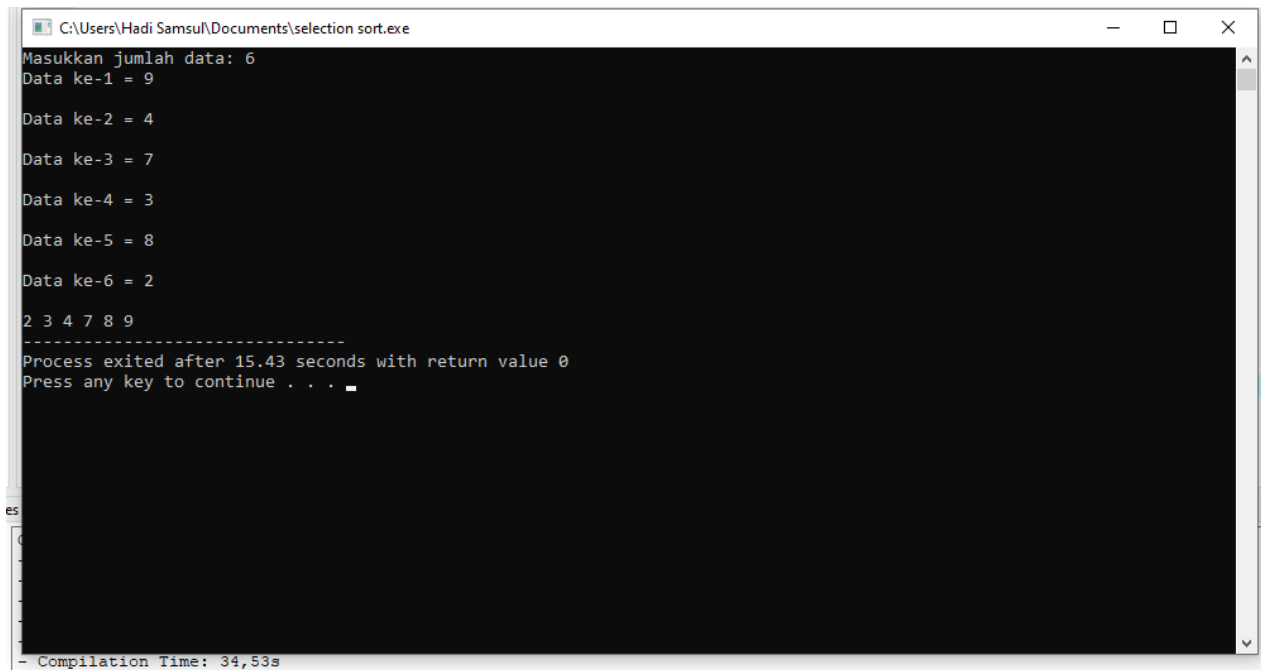
2	3	4	7	8	9
---	---	---	---	---	---

➤ Coding

```
selection sort.cpp
1  #include<iostream>
2
3  //variabel standard
4  using namespace std;
5
6  // fungsi utama atau program utama
7  int main() {
8
9      // deklarasi variabel
10     int n, tukar, A[100];
11
12     // menampilkan pesan input
13     cout << "Masukkan jumlah data: ";
14
15     // pesan input akan di simpan di variabel n
16     cin >> n;
17
18     // perulangan yang digunakan oleh input n
19     for (int i = 0; i < n; i++) { // n merupakan jumlah data yang ingin dimasukkan
20         cout << "Data ke-" << i + 1 << " = ";
21         cin >> A[i];
22         cout << endl;
```

```
selection sort.cpp
22     }
23     }
24     // perulangan perbandingan input (sort)
25     for (int i = 0; i < n-1; i++) { // n merupakan jumlah data yang ingin dimasukkan
26         tukar = i;
27         int temp;
28         for(int j = i+1; j < n; j++){
29             if(A[j] < A[tukar]){
30                 tukar = j;
31             }
32         }
33         temp = A[tukar];
34         A[tukar] = A[i];
35         A[i] = temp;
36     }
37
38     // menampilkan hasil perbandingan (sorting)
39     for(int i = 0; i < n; i++){ // n merupakan jumlah data yang ingin dimasukkan
40         cout << A[i] << " ";
41     }
42     // mengambil inputan dari nilai n
43     cin.get();
```

```
selection sort.cpp
28     }
29     }
30     }
31     }
32     }
33     temp = A[tukar];
34     A[tukar] = A[i];
35     A[i] = temp;
36 }
37
38 // menampilkan hasil perbandingan (sorting)
39 for(int i = 0; i < n; i++){ // n merupakan jumlah data yang ingin dimasukkan
40     cout << A[i] << " ";
41 }
42 // mengambil inputan dari nilai n
43 cin.get();
44
45 // end code
46 return 0;
47 }
48
```



```
C:\Users\Hadi Samsu\Documents\selection sort.exe
Masukkan jumlah data: 6
Data ke-1 = 9
Data ke-2 = 4
Data ke-3 = 7
Data ke-4 = 3
Data ke-5 = 8
Data ke-6 = 2
2 3 4 7 8 9
-----
Process exited after 15.43 seconds with return value 0
Press any key to continue . . .
```

- Compilation Time: 34,53s

➤ Kelebihan dan Kekurangan Selection Sort

- Kelebihan Selection Sort :
 1. Algoritma ini sangat rapat dan mudah untuk diimplementasikan.
 2. Mudah menentukan data maksimum /minimum.
 3. Mudah menggabungkannya kembali.
 4. Kompleksitas selection sort relatif lebih kecil.
- Kekurangan Selection Sort :
 1. Membutuhkan method tambahan
 2. Sulit untuk digabungkan kembali
 3. Perlu dihindari untuk penggunaan data lebih dari 1000 tabel, karena akan menyebabkan kompleksitas yang lebih tinggi dan kurang praktis.