

Nama : Salsabilla Octavianingrum

Kelas : 2021 A

Nim : 21091397005

1. Program C++ Insertion sort

```
Start here X insert sort.cpp X
1  #include <iostream>
2  using namespace std;
3
4  int main() {
5
6      //inisialisasi variabel
7      int input_batas;
8
9      cout<<"Masukkan batas array : ";
10     cin>>input_batas;
11     cout<<endl;
12
13     cout<<"Masukkan elemen array : \n";
14
15     //Looping untuk memasukkan angka
16     int input_angka[input_batas];
17
18     for(int loop_1=0; loop_1<input_batas; loop_1++){
19         cout<<"masukan angka ke "<<loop_1<<" :";
20         cin>>input_angka[loop_1];
21     }
22     cout<<endl;
23
24     //Proses Sorting
25     for(int loop_1=1; loop_1<input_batas; loop_1++){
26
27         int key = input_angka[loop_1];
```

```

26
27     int key = input_angka[loop_1];
28     int loop_2 = loop_1-1;
29
30     while(loop_2>=0 && input_angka[loop_2] > key){
31         input_angka[loop_2+1] = input_angka[loop_2];
32         loop_2--;
33     }
34
35     input_angka[loop_2+1] = key;
36 }
37
38 //Perintah menampilkan hasil sorting
39 cout<<"hasil akhir"<<endl;
40 for(int loop_3=0; loop_3<input_batas; loop_3++){
41     cout << "[" << input_angka[loop_3] << "]" ";
42 }
43 }
44

```

Hasil run :

```

C:\Users\user\Downloads\insert sort.exe
Masukkan batas array : 5
Masukkan elemen array :
masukan angka ke 0 :6
masukan angka ke 1 :4
masukan angka ke 2 :8
masukan angka ke 3 :2
masukan angka ke 4 :5

hasil akhir
[2] [4] [5] [6] [8]
Process returned 0 (0x0)   execution time : 13.052 s
Press any key to continue.

```

2. Kelebihan Insertion sort

- Sederhana dalam penerapannya.
- Mangkus dalam data yang kecil.
- Jika list sudah terurut atau sebagian terurut maka Insertion Sort akan lebih cepat dibandingkan dengan Quicksort.
- Mangkus dalam data yang sebagian sudah terurut.
- Lebih mangkus dibanding Bubble Sort dan Selection Sort.

- Loop dalam pada Insertion Sort sangat cepat, sehingga membuatnya salah satu algoritma pengurutan tercepat pada jumlah elemen yang sedikit.
- Lebih stabil.

3. Kekurangan insertion sort

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