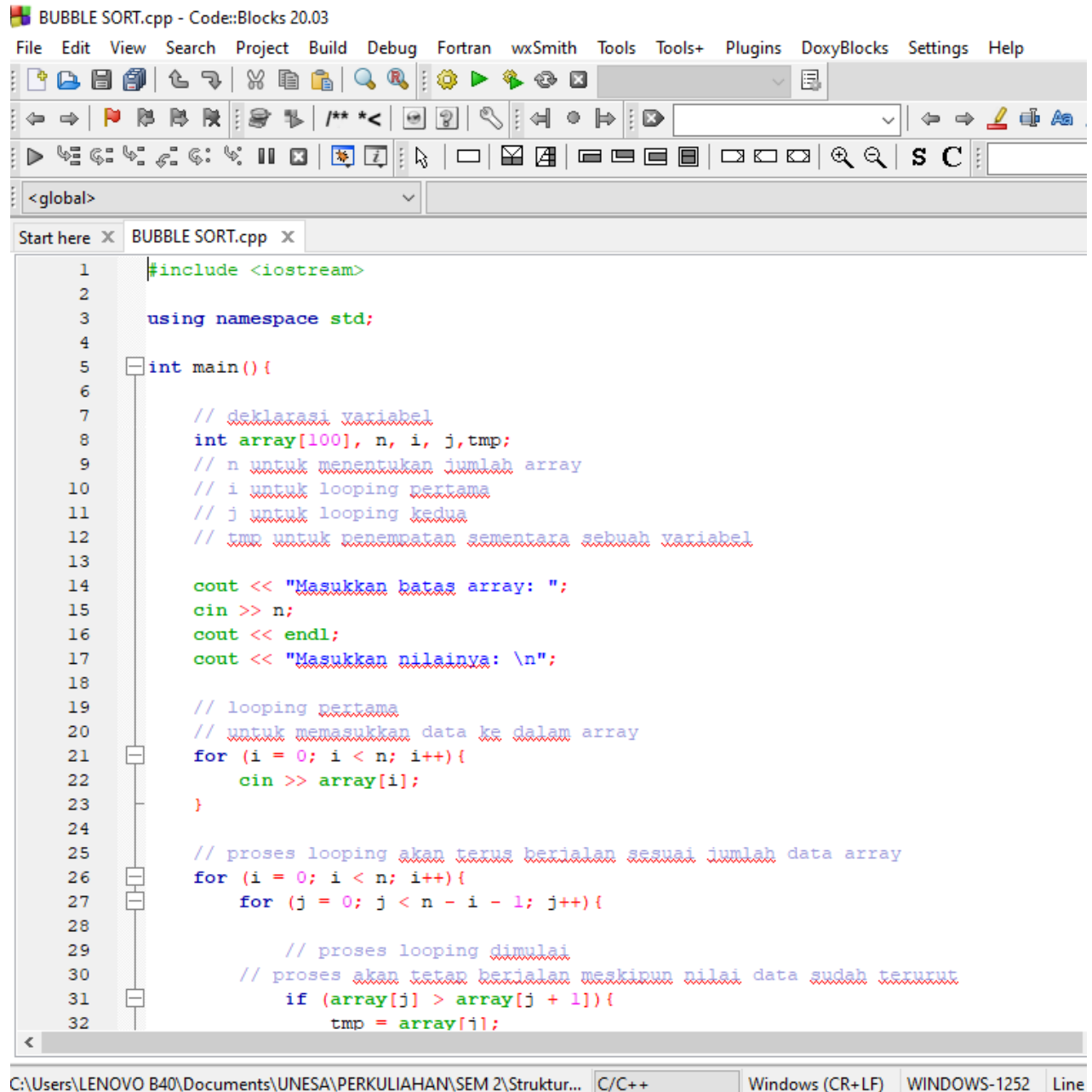


Berikut adalah hasil Codingan 'Bubble Sort' beserta penjelasannya.



```
1  #include <iostream>
2
3  using namespace std;
4
5  int main() {
6
7      // deklarasi variabel
8      int array[100], n, i, j, tmp;
9      // n untuk menentukan jumlah array
10     // i untuk looping pertama
11     // j untuk looping kedua
12     // tmp untuk penempatan sementara sebuah variabel
13
14     cout << "Masukkan batas array: ";
15     cin >> n;
16     cout << endl;
17     cout << "Masukkan nilainya: \n";
18
19     // looping pertama
20     // untuk memasukkan data ke dalam array
21     for (i = 0; i < n; i++){
22         cin >> array[i];
23     }
24
25     // proses looping akan terus berjalan sesuai jumlah data array
26     for (i = 0; i < n; i++){
27         for (j = 0; j < n - i - 1; j++){
28
29             // proses looping dimulai
30             // proses akan tetap berjalan meskipun nilai data sudah terurut
31             if (array[j] > array[j + 1]){
32                 tmp = array[j];
```

C:\Users\LENOVO B40\Documents\UNESA\PERKULIAHAN\SEM 2\Struktur... C/C++ Windows (CR+LF) WINDOWS-1252 Line



&lt;global&gt;

Start here x BUBBLE SORT.cpp x

```
17     cout << "Masukkan nilainya: \n";
18
19     // looping pertama
20     // untuk memasukkan data ke dalam array
21     for (i = 0; i < n; i++){
22         cin >> array[i];
23     }
24
25     // proses looping akan terus berjalan sesuai jumlah data array
26     for (i = 0; i < n; i++){
27         for (j = 0; j < n - i - 1; j++){
28
29             // proses looping dimulai
30             // proses akan tatan berialan meskipun nilai data sudah terurut
31             if (array[j] > array[j + 1]){
32                 tmp = array[j];
33                 array[j] = array[j + 1];
34                 array[j + 1] = tmp;
35             }
36         }
37     }
38
39     // perintah menampilkan hasil akhir
40     cout << endl;
41     cout << "Hasil akhir sorting: \n";
42     for (i = 0; i < n; i++){
43         cout << array[i] << " ";
44     }
45     cout << "\n";
46 }
47
```

&lt;

Berikut adalah hasil Run Codingan 'Bubble Sort'

```
"C:\Users\LENOVO B40\Documents\UNESA\PERKULIAHAN\SEM 2\Struktur Data\Tugas Individu\BUBBLE SORT.exe"
Masukkan batas array: 5

Masukkan nilainya:
4
20
3
9
13

Hasil akhir pengurutan:
3 4 9 13 20

Process returned 0 (0x0)   execution time : 28.704 s
Press any key to continue.
```

Berikut adalah kelebihan & kekurangan 'Bubble Sort'

**Kelebihan:**

- Dilihat dari codingannya, bubble sort yang saya buat merupakan codingan yang paling simple.
- Algoritma bubble sort yang saya buat juga lebih mudah dipahami.

**Kekurangan:**

- Bubble sort tidak efisien. Looping tetap dilakukan meskipun data sudah terurut karena setiap data dibandingkan dengan setiap data yang lain untuk menentukan posisinya.
- Pada saat mengurutkan data yang sangat besar akan mengalami kelambatan atau kinerjanya yang memburuk ketika mengolah banyak data.