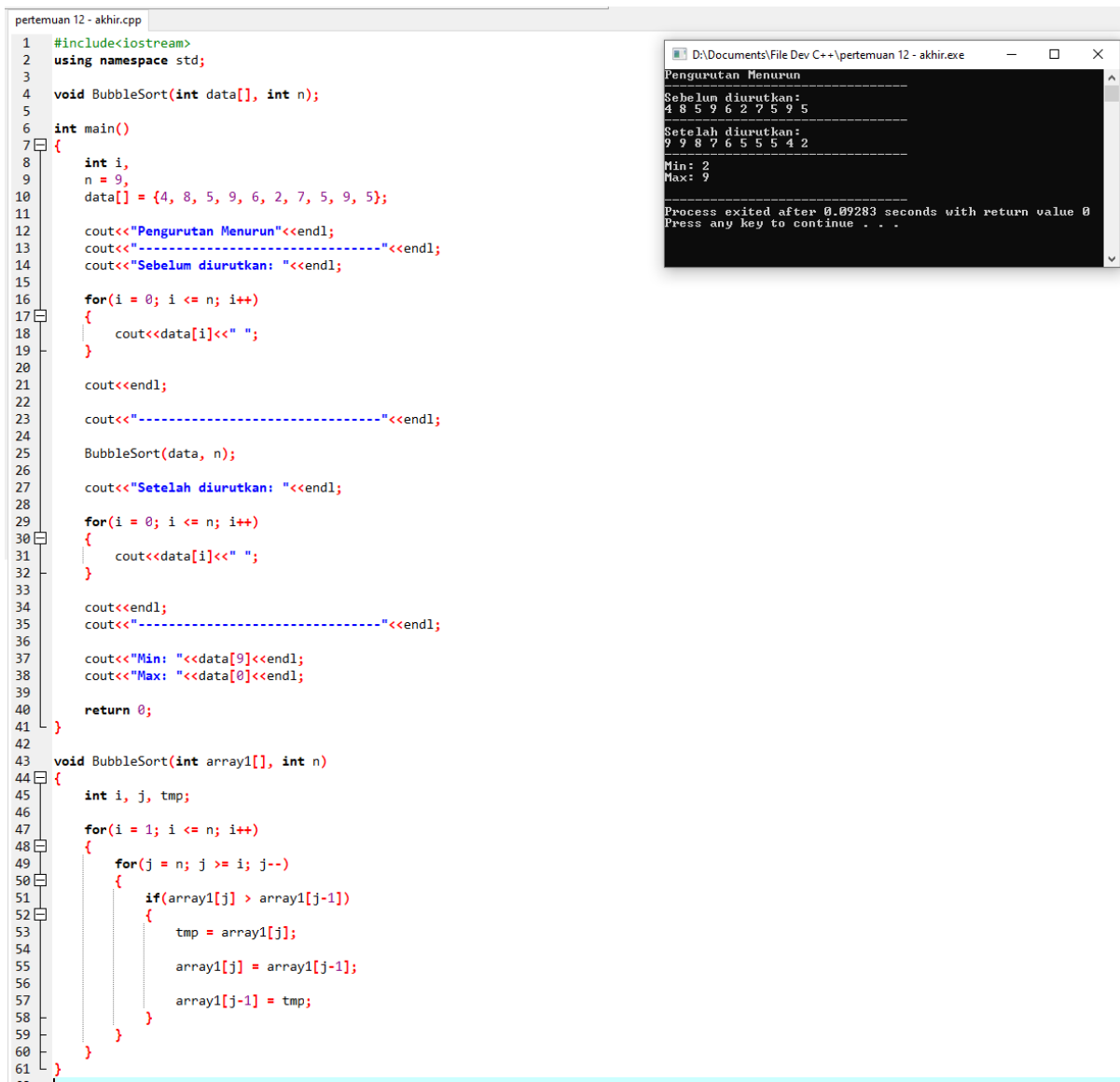


Tugas Akhir – Pertemuan 12



The image shows a C++ IDE with a file named `pertemuan 12 - akhir.cpp`. The code implements a bubble sort algorithm. The `main` function initializes an array `data` with the values `{4, 8, 5, 9, 6, 2, 7, 5, 9, 5}` and calls the `BubbleSort` function. The `BubbleSort` function sorts the array in descending order. The output of the program is displayed in a separate window titled `D:\Documents\File Dev C++\pertemuan 12 - akhir.exe`.

```
1 #include<iostream>
2 using namespace std;
3
4 void BubbleSort(int data[], int n);
5
6 int main()
7 {
8     int i,
9     n = 9,
10    data[] = {4, 8, 5, 9, 6, 2, 7, 5, 9, 5};
11
12    cout<<"Pengurutan Menurun"<<endl;
13    cout<<"-----"<<endl;
14    cout<<"Sebelum diurutkan: "<<endl;
15
16    for(i = 0; i <= n; i++)
17    {
18        cout<<data[i]<<" ";
19    }
20
21    cout<<endl;
22
23    cout<<"-----"<<endl;
24
25    BubbleSort(data, n);
26
27    cout<<"Setelah diurutkan: "<<endl;
28
29    for(i = 0; i <= n; i++)
30    {
31        cout<<data[i]<<" ";
32    }
33
34    cout<<endl;
35    cout<<"-----"<<endl;
36
37    cout<<"Min: "<<data[9]<<endl;
38    cout<<"Max: "<<data[0]<<endl;
39
40    return 0;
41 }
42
43 void BubbleSort(int array1[], int n)
44 {
45     int i, j, tmp;
46
47     for(i = 1; i <= n; i++)
48     {
49         for(j = n; j >= i; j--)
50         {
51             if(array1[j] > array1[j-1])
52             {
53                 tmp = array1[j];
54                 array1[j] = array1[j-1];
55                 array1[j-1] = tmp;
56             }
57         }
58     }
59 }
60
61 }
```

Pengurutan Menurun

Sebelum diurutkan:
4 8 5 9 6 2 7 5 9 5

Setelah diurutkan:
9 9 8 7 6 5 5 4 2

Min: 2
Max: 9

Process exited after 0.09283 seconds with return value 0
Press any key to continue . . .

Source Code:

```
#include<iostream>

using namespace std;

void BubbleSort(int data[], int n);

int main()
{
    int i,
    n = 9,
    data[] = {4, 8, 5, 9, 6, 2, 7, 5, 9, 5};

    cout<<"Pengurutan Menurun"<<endl;
    cout<<"-----"<<endl;
    cout<<"Sebelum diurutkan: "<<endl;

    for(i = 0; i <= n; i++)
    {
        cout<<data[i]<<" ";
    }

    cout<<endl;

    cout<<"-----"<<endl;

    BubbleSort(data, n);

    cout<<"Setelah diurutkan: "<<endl;

    for(i = 0; i <= n; i++)
    {
        cout<<data[i]<<" ";
    }

    cout<<endl;
```

```

        cout<<"-----"<<endl;

        cout<<"Min: "<<data[9]<<endl;
        cout<<"Max: "<<data[0]<<endl;

        return 0;
    }

void BubbleSort(int array1[], int n)
{
    int i, j, tmp;

    for(i = 1; i <= n; i++)
    {
        for(j = n; j >= i; j--)
        {
            if(array1[j] > array1[j-1])
            {
                tmp = array1[j];

                array1[j] = array1[j-1];

                array1[j-1] = tmp;
            }
        }
    }
}

```

Kesimpulan:

Pada pertemuan ke 12 ini saya dapat menarik kesimpulan, Sorting adalah suatu proses pengurutan data yang sebelumnya disusun secara acak atau tidak teratur menjadi urut dan teratur menjadi urut dan teratur menurut suatu aturan tertentu. Biasanya pengurutan terbagi menjadi 2 yaitu : ascending (pengurutan dari karakter/ angka kecil ke karakter / angka besar ke karakter/ angka kecil) Ada banyak cara yang dapat dilakukan untuk melakukan proses pengurutan dari paling atas ke paling bawah atau sebaliknya. Untuk melakukan proses pengurutan dapat menggunakan beberapa metode antara lain:

1) Bubble sort

Bubble sort adalah suatu metode pengurutan yang membandingkan elemen yang sekarang dengan elemen berikutnya. Perbandingan alamatnya dapat dimulai dari data yang paling awal atau yang paling akhir. Apabila elemen yang sekarang (sebelumnya) lebih besar dari elemen berikutnya, maka posisi di tukar, kalau tidak posisinya tetap atau tidak perlu ditukar.

2) Selection sort

Selection sort adalah suatu metode pengurutan yang membandingkan elemen yang sekarang dengan elemen berikutnya sampai elemen terakhir. Jika ditemukan elemen lain yang lebih kecil dari elemen sekarang maka di catat posisinya dan langsung ditukar.

3) Quick sort

Quick sort adalah suatu metode pengurutan yang membandingkan suatu elemen (pivot) dengan elemen yang lain dan menyusun sedemikian rupa sehingga elemen yang lain lebih kecil dari pada pivot terletak disebelah kiri pivot sedangkan elemen yang lebih besar dari pivot diletakan disebelah kanan pivot.

4) Merge sort

Merge sort adalah suatu metode pengurutan yang membandingkan elemen satu dengan elemen yang lain, apabila nilainya lebih kecil maka datanya ditampung di elemen yang lain lagi.