

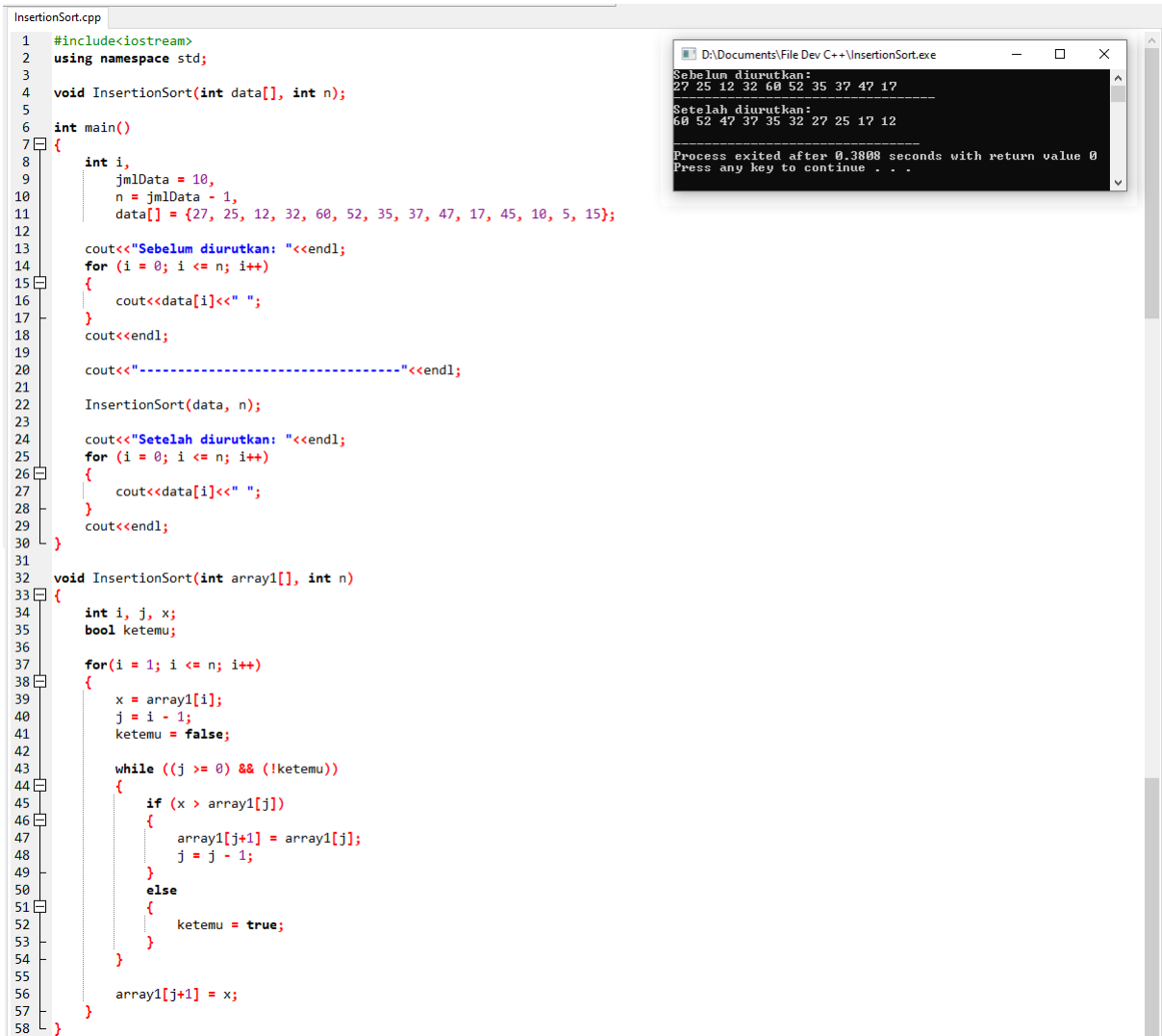
Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023

Tugas : Algoritma II – Pertemuan 13

1.



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

D:\Documents\File Dev C++\InsertionSort.exe

Sebelum diurutkan:
27 25 12 32 60 52 35 37 47 17

Setelah diurutkan:
60 52 47 37 35 32 27 25 17 12

Process exited after 0.3808 seconds with return value 0
Press any key to continue . . .

Source Code:

```
#include<iostream>
using namespace std;

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

int main()
{
    int i,
        jmlData = 10,
        n = jmlData - 1,
        data[] = {27, 25, 12, 32, 60, 52, 35, 37, 47, 17, 45, 10, 5, 15};

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

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

    InsertionSort(data, n);

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

void InsertionSort(int array1[], int n)
{
    int i, j, x;
    bool ketemu;

    for(i = 1; i <= n; i++)
    {
        x = array1[i];
        j = i - 1;
        ketemu = false;

        while ((j >= 0) && (!ketemu))
        {
            if (x > array1[j])
            {
                array1[j+1] = array1[j];
                j = j - 1;
            }
            else
            {
                ketemu = true;
            }
        }

        array1[j+1] = x;
    }
}
```

2.

```
ShellSort.cpp
1 #include<iostream>
2 using namespace std;
3
4 void InsertionSort(int data[], int n, int start, int step);
5 void ShellSort(int data[], int n);
6
7 int main()
8 {
9     int i,
10     jmlData = 10,
11     n = jmlData - 1,
12     data[] = {27, 25, 12, 32, 60, 52, 35, 37, 47, 17, 45, 10, 5, 15};
13
14     cout<<"Sebelum diurutkan: "<<endl;
15     for (i = 0; i <= n; i++)
16     {
17         cout<<data[i]<<" ";
18     }
19     cout<<endl;
20
21     cout<<"-----"<<endl;
22
23     ShellSort(data, n);
24
25     cout<<"Setelah diurutkan: "<<endl;
26     for (i = 0; i <= n; i++)
27     {
28         cout<<data[i]<<" ";
29     }
30     cout<<endl;
31 }
32
33 void InsertionSort(int data[], int n, int start, int step)
34 {
35     int i, j, x;
36     bool ketemu;
37     i = start + step;
38
39     while (i <= n)
40     {
41         x = data[i];
42         j = i - step;
43
44         ketemu = false;
45
46         while ((j >= 0) && (!ketemu))
47         {
48             if (x > data[j])
49             {
50                 data[j+step] = data[j];
51                 j = j - step;
52             }
53             else
54             {
55                 ketemu = true;
56             }
57         }
58
59         data[j+step] = x;
60         i += step;
61     }
62 }
63
64 void ShellSort(int data[], int n)
65 {
66     int start, step;
67     for (step = 5; step >= 1; step -= 2)
68     {
69         for (start = 0; start <= step; start++)
70         {
71             InsertionSort(data, n, start, step);
72         }
73     }
74 }
```

DA\Documents\File Dev C++\ShellSort.exe

Sebelum diurutkan:
27 25 12 32 60 52 35 37 47 17

Setelah diurutkan:
60 52 47 37 35 32 27 25 17 12

Process exited after 0.06735 seconds with return value 0
Press any key to continue . . .

Source Code:

```
#include<iostream>

using namespace std;

void InsertionSort(int data[], int n, int start, int step);
void ShellSort(int data[], int n);

int main()
{
    int i,
        jmlData = 10,
        n = jmlData - 1,
        data[] = {27, 25, 12, 32, 60, 52, 35, 37, 47, 17, 45, 10, 5, 15};

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

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

    ShellSort(data, n);

    cout<<"Setelah diurutkan: "<<endl;
    for (i = 0; i <= n; i++)
    {
        cout<<data[i]<<" ";
    }
    cout<<endl;
}
```

```

void InsertionSort(int data[], int n, int start, int step)
{
    int i, j, x;
    bool ketemu;
    i = start + step;

    while (i <= n)
    {
        x = data[i];
        j = i - step;

        ketemu = false;

        while ((j >= 0) && (!ketemu))
        {
            if(x > data[j])
            {
                data[j+step] = data[j];
                j = j - step;
            }
            else
            {
                ketemu = true;
            }
        }

        data[j+step] = x;
        i += step;
    }
}

```

```
void ShellSort(int data[], int n)
{
    int start, step;
    for (step = 5; step >= 1; step -= 2)
    {
        for(start = 0; start <= step; start++)
        {
            InsertionSort(data, n, start, step);
        }
    }
}
```