Nama : Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023

Tugas : Algoritma II – Pertemuan 13

1.

InsertionSort.cpp

```cpp
#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;
    }
}
```

```
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

```cpp
#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);
        }
    }
}
```

```
D:\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);
		}
	}
}
```