

LAPORAN AKHIR

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-11



Disusun Oleh:

Nama: Andri Firman Saputra

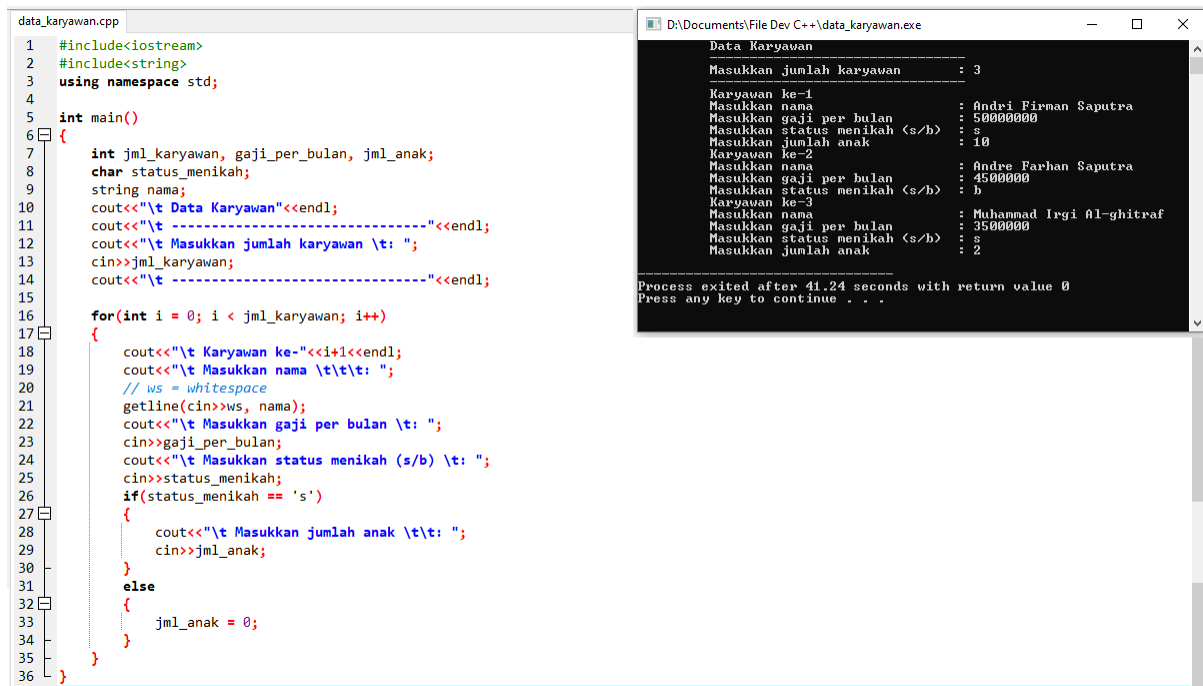
NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Tugas Akhir – Pertemuan 11



The image shows a C++ program named `data_karyawan.cpp` and its execution output in a terminal window.

Source Code:

```
1 #include<iostream>
2 #include<string>
3 using namespace std;
4
5 int main()
6 {
7     int jml_karyawan, gaji_per_bulan, jml_anak;
8     char status_menikah;
9     string nama;
10    cout<<"\t Data Karyawan"<<endl;
11    cout<<"\t -----"<<endl;
12    cout<<"\t Masukkan jumlah karyawan \t: ";
13    cin>>jml_karyawan;
14    cout<<"\t -----"<<endl;
15
16    for(int i = 0; i < jml_karyawan; i++)
17    {
18        cout<<"\t Karyawan ke-"<<i+1<<endl;
19        cout<<"\t Masukkan nama \t\t\t: ";
20        // ws = whitespace
21        getline(cin>>ws, nama);
22        cout<<"\t Masukkan gaji per bulan \t: ";
23        cin>>gaji_per_bulan;
24        cout<<"\t Masukkan status menikah (s/b) \t: ";
25        cin>>status_menikah;
26        if(status_menikah == 's')
27        {
28            cout<<"\t Masukkan jumlah anak \t\t\t: ";
29            cin>>jml_anak;
30        }
31        else
32        {
33            jml_anak = 0;
34        }
35    }
36 }
```

Execution Output:

```
Data Karyawan
Masukkan jumlah karyawan : 3
Karyawan ke-1
Masukkan nama : Andri Firman Saputra
Masukkan gaji per bulan : 5000000
Masukkan status menikah (s/b) : s
Masukkan jumlah anak : 10
Karyawan ke-2
Masukkan nama : Andre Farhan Saputra
Masukkan gaji per bulan : 4500000
Masukkan status menikah (s/b) : b
Karyawan ke-3
Masukkan nama : Muhammad Irgi Al-ghitraf
Masukkan gaji per bulan : 3500000
Masukkan status menikah (s/b) : s
Masukkan jumlah anak : 2
Process exited after 41.24 seconds with return value 0
Press any key to continue . . .
```

Source Code:

```
#include<iostream>

#include<string>

using namespace std;

int main()
{

    int jml_karyawan, gaji_per_bulan, jml_anak;
    char status_menikah;
    string nama;

    cout<<"\t Data Karyawan"<<endl;

    cout<<"\t -----"<<endl;

    cout<<"\t Masukkan jumlah karyawan \t: ";

    cin>>jml_karyawan;

    cout<<"\t -----"<<endl;

    for(int i = 0; i < jml_karyawan; i++)
    {

        cout<<"\t Karyawan ke-"<<i+1<<endl;
        cout<<"\t Masukkan nama \t\t\t: ";
```

```
// ws = whitespace
getline(cin>>ws, nama);
cout<<"\t Masukkan gaji per bulan \t: ";
cin>>gaji_per_bulan;
cout<<"\t Masukkan status menikah (s/b) \t: ";
cin>>status_menikah;
if(status_menikah == 's')
{
    cout<<"\t Masukkan jumlah anak \t\t: ";
    cin>>jml_anak;
}
else
{
    jml_anak = 0;
}
}
}
```

Kesimpulan:

Pada pertemuan ke 11 ini saya dapat menarik kesimpulan, Saya memahami tentang struktur (structure). Struktur(structure) adalah sekumpulan variabel yang masing-masing tipe datanya berbeda, dan di kelompokkan ke dalam satu nama. (Menurut pascal struktur juga dikenal sebagai record). Struktur membantu data-data yang rumit, khususnya dalam program yang besar, Karena struktur membiarkan sekelompok variabel diperlakukan sebagai satu unit. Struktur ini sering digunakan untuk mendefinisikan suatu record data yang disimpan di dalam file. Struktur termasuk kedalam tipe data yang di bangkitkan (derived data type), yang disusun menggunakan objek dengan tipe yang lain.

Contoh bentuk umum struktur:

```
struct mhs
{
    char *nama;
    char *nim;
    int uts, uas;
    float akhir;
    char mutu;
}
```

LAPORAN AWAL

ALGORITMA DAN PEMROGRAMAN

LAPORAN KE-12



Disusun Oleh:

Nama: Andri Firman Saputra

NIM : 201011402125

Kelas : 02TPLP023 – Pagi

TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PAMULANG

Jl. Surya Kencana No. 1 Pamulang Telp (021)7412566, Fax. (021)7412566
Tangerang Selatan - Banten

Nama : Andri Firman Saputra
NIM : 201011402125

Praktikum Algoritma
Laporan Awal - Pertemuan 12

Teori Dasar

a) Pendahuluan

Sorting adalah suatu proses pengurutan data yg 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)

descending kebalikannya.

1) Bubble sort

Bubble sort adalah suatu metode pengurutan yg membandingkan elemen yg satu dengan elemen berikutnya. Perbandingan aramatnya dapat dimulai dari data yg paling awal atau yg paling akhir.

Apabila elemen yg satu (sebelumnya) lebih besar dari elemen berikutnya, maka posisi ditukar, kalau tidak posisinya tetap atau tidak perlu ditukar.

2) Selection Sort

Selection sort adalah suatu metode pengurutan yg membandingkan elemen yg satu dengan elemen berikutnya sampai elemen terakhir.

Jika ditemukan elemen lain yg lebih kecil dari elemen satu maka dicatat posisinya dan langsung ditukar.

3) Quick Sort

Quick sort adalah suatu metode pengurutan yg membandingkan suatu elemen (pivot) dengan elemen yg lain dan menyusunnya sedemikian rupa sehingga elemen

yg lain lebih kecil dari pada pivot terletak disebelah kiri pivot sedangkan elemen yg lebih besar dari pivot diletakkan disebelah kanan pivot.

4) Merge sort

Merge sort adalah suatu metode pengurutan yg membandingkan elemen. Satu dgn elemen yg lain, apabila nilai nya lebih kecil maka datanya ditampung di elemen yg lain lagi.

Tugas Pendahuluan

1. Apa yg dimaksud dgn Sorting?

Sorting adalah suatu proses pengurutan data yg sebelumnya disusun secara acak atau tidak teratur menjadi urut dan teratur menurut suatu aturan tertentu.

2. Sebutkan dan jelaskan jenis-jenis Sorting pada bahasa C/C++!

- 1) Bubble Sort
- 2) Selection Sort
- 3) Quick Sort
- 4) Merge Sort

3. Se bagai Programmer, mengapa anda menggunakan sorting? Karena jika tidak menggunakan sorting data akan menjadi berantakan dan sulit di baca, sebaliknya jika kita menggunakan sorting data lebih mudah dibaca dan diolah.

4. Buatlah contoh algoritma dan program sederhana menggunakan sorting dgn memauai flowchart!

Contoh Program Sorting:

```
order asc.cpp
1 #include<iostream>
2 using namespace std;
3
4 void BubbleSort(int data[], int n);
5
6 int main()
7 {
8     int i, n = 9, data[] = {10, 40, 20, 60, 15, 4, 8, 100, 200, 25};
9
10    cout<<"Pengurutan Menaik"<<endl;
11    cout<<"-----"<<endl;
12    cout<<"Sebelum diurutkan: "<<endl;
13
14    for(i = 0; i <= n; i++)
15    {
16        cout<<data[i]<<" ";
17    }
18
19    cout<<endl;
20
21    cout<<"-----"<<endl;
22
23    BubbleSort(data, n);
24
25    cout<<"Setelah diurutkan: "<<endl;
26
27    for(i = 0; i <= n; i++)
28    {
29        cout<<data[i]<<" ";
30    }
31
32    cout<<endl;
33
34    return 0;
35 }
36
37 void BubbleSort(int array1[], int n)
38 {
39     int i, j, tmp;
40
41     for(i = 1; i <= n; i++)
42     {
43         for(j = n; j >= i; j--)
44         {
45             // controller asc and desc
46             if(array1[j] < array1[j-1])
47             {
48                 tmp = array1[j];
49                 array1[j] = array1[j-1];
50                 array1[j-1] = tmp;
51             }
52         }
53     }
54 }
55
56 }
```

```
D:\Documents\File Dev C++\order asc.exe
Pengurutan Menaik
-----
Sebelum diurutkan:
10 40 20 60 15 4 8 100 200 25
-----
Setelah diurutkan:
4 8 10 15 20 25 40 60 100 200
-----
Process exited after 0.05929 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[] = {10, 40, 20, 60, 15, 4, 8, 100, 200, 25};

    cout<<"Pengurutan Menaik"<<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;

    return 0;
}
```

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

    for(i = 1; i <= n; i++)
    {
        for(j = n; j >= i; j--)
        {
            // controller asc and desc
            if(array1[j] < array1[j-1])
            {
                tmp = array1[j];

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

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