

MI2101

Praktikum Teknik Komputasi

Modul 4

Fadjar Fathurrahman

2018

1 Tujuan

- Mampu membuat program C++ sederhana dengan memanfaatkan array
- Mampu menggunakan *input redirection* pada Linux

2 Perangkat lunak yang diperlukan

- Linux OS
- CodeBlocks yang telah dikonfigurasi untuk kompiler GNU C/C++
- Terminal emulator dengan bash sebagai shell (baris perintah)
- Editor teks seperti gedit

3 Menghitung rata-rata

Buatlah program untuk menghitung rata-rata dari nilai yang ada pada suatu array. Gunakan fungsi dengan prototip sebagai berikut.

```
float hitung_rata2( int Ndata, float *data )
```

Untuk menguji fungsi yang Anda buat, lengkapi program berikut.

```
#include <iostream>

using namespace std;

float hitung_rata2( int Ndata, float *data );

int main()
{
    const int N = 10;

    float nilai[N] = {40.9, 45.3, 55.3, 55.1, 33.1,
                      11.7, 34.1, 13.2, 33.3, 34.1};

    cout << "Nilai rata = " << hitung_rata2(N, nilai) << endl;

    return 0;
}
```

```

}

float hitung_rata2( int Ndata, float *data )
{
    // tuliskan fungsi Anda di sini
}

```

4 Mengolah nilai akhir suatu mata kuliah

Anda diberikan tugas untuk membuat program untuk mengolah nilai akhir mahasiswa dari suatu mata kuliah. Ada tiga komponen nilai yang akan dihitung, yaitu nilai UTS, nilai UAS, dan nilai praktikum. Nilai akhir dihitung dengan persamaan sebagai berikut:

$$\text{NilaiAkhir} = 0.3\text{NilaiUTS} + 0.3\text{NilaiUAS} + 0.4\text{NilaiPraktikum} \quad (1)$$

Selain itu, akan ditentukan juga nilai akhir dalam bentuk huruf, dengan aturan sebagai berikut.

Nilai akhir	Nilai huruf
$90 < \text{NilaiAkhir} \leq 100$	A
$80 < \text{NilaiAkhir} \leq 90$	AB
$70 < \text{NilaiAkhir} \leq 80$	B
$60 < \text{NilaiAkhir} \leq 70$	BC
$50 < \text{NilaiAkhir} \leq 60$	C
$\text{NilaiAkhir} \leq 50$	E

Kode program:

```

#include <iostream>
#include <string>

using namespace std;

void input_data(
    int NDATA,
    string *nama, string *NIM, float *uts, float *uas, float *praktikum
);

void hitung_nilai_akhir(
    int Ndata,
    float *uts, float *uas, float *praktikum,
    float *nilai_akhir
);

void tampilkan_data(
    int idata,
    string *nama, string *NIM, float *uts, float *uas, float *praktikum,
    float *nilai_akhir
);

int main()
{
    const int NDATA = 3;

    string nama[NDATA];
    string NIM[NDATA];
    float uts[NDATA], uas[NDATA], praktikum[NDATA];
    float nilai_akhir[NDATA];

    input_data( NDATA, nama, NIM, uts, uas, praktikum );

    hitung_nilai_akhir( NDATA, uts, uas, praktikum, nilai_akhir );
}

```

```

    for(int i = 0; i < NDATA; i++) {
        tampilkan_data( i, nama, NIM, uts, uas, praktikum, nilai_akhir );
    }

    return 0;
}

void input_data(
    int NDATA,
    string *nama, string *NIM, float *uts, float *uas, float *praktikum
)
{
    for(int i = 0; i < NDATA; i++) {
        cout << endl;
        cout << "-----" << endl;
        cout << "Entri data:" << i+1 << endl;
        cout << "-----" << endl;
        cout << "Masukkan data" << endl;

        // handle inputs here ... (nama, nim, uts, uas, praktikum)
        // gunakan cin
        // .....
    }
}

void tampilkan_data(
    int i,
    string *nama, string *NIM, float *uts, float *uas, float *praktikum,
    float *nilai_akhir
)
{
    cout << endl;
    cout << "-----" << endl;
    cout << "Tampilan data: " << i+1 << endl;
    cout << "-----" << endl;
    cout << "Nama           : " << nama[i] << endl;
    cout << "NIM            : " << NIM[i] << endl;
    cout << "Nilai UTS      : " << uts[i] << endl;
    cout << "Nilai UAS      : " << uas[i] << endl;
    cout << "Nilai praktikum : " << praktikum[i] << endl;
    cout << "-----" << endl;
    cout << "Nilai akhir     : " << nilai_akhir[i];
    cout << " (" << hitung_nilai_huruf(nilai_akhir[i]) << ")" << endl;
}

void hitung_nilai_akhir(
    int Ndata,
    float *uts, float *uas, float *praktikum,
    float *nilai_akhir
)
{
    // lengkapi .....
}

string hitung_nilai_huruf( float nilai_akhir )
{
    // lengkapi .....
}

```

Untuk pengujian awal, gunakan NDATA kecil, misalnya NDATA=2 atau NDATA=3, dan pastikan bahwa data yang dimasukkan dan dihitung adalah benar.

Berikut ini adalah contoh keluaran (tanpa tampilan input):

```

=====
Tampilan data: 1
=====
Nama           : Joko
NIM            : A18001
Nilai UTS      : 80
Nilai UAS      : 90
Nilai praktikum : 91
-----
Nilai akhir    : 87.4 (AB)

=====
Tampilan data: 2
=====
Nama           : Jono
NIM            : A18002
Nilai UTS      : 88
Nilai UAS      : 99
Nilai praktikum : 90
-----
Nilai akhir    : 92.1 (A)

=====
Tampilan data: 3
=====
Nama           : Jojo
NIM            : A18003
Nilai UTS      : 90
Nilai UAS      : 99
Nilai praktikum : 10
-----
Nilai akhir    : 60.7 (BC)

```

Setelah Anda yakin bahwa input Anda benar, uji juga untuk jumlah data yang relatif banyak seperti `NDATA=10`. Pada kasus ini, memasukkan data manual sangat melelahkan. Solusi yang dapat kita gunakan adalah dengan menuliskan input ke suatu file. File ini adalah file teks biasa yang formatnya disesuaikan dengan kebutuhan program kita. Untuk kasus ini pada contoh di atas kita dapat membuat file dengan isi sebagai berikut:

```

Joko A18001 80 90 91
Jono A18002 88 99 90
Jojo A18003 90 99 10

```

Misalkan file yang Anda buat di atas disimpan dengan nama `input.txt`. Dengan menggunakan terminal Anda dapat mengetikkan perintah berikut.

```
./nama_program < input.txt
```

Pada Unix/Linux, hal ini dikenal dengan *input redirection*.

5 Pertanyaan dan tugas

Pada program sebelumnya, `NDATA` ditulis pada program (*hardwired*). Kembangkan program sebelumnya di mana `NDATA` merupakan input dari pengguna dan array-array terkait dialokasikan secara dinamik.

Petunjuk: gunakan operator `new` untuk mengalokasikan memori secara dinamik serta `delete[]` untuk dealokasi memori. Contoh:

```
// Input jumlah data
int NDATA;
cout << "Masukkan NDATA: ";
cin >> NDATA;

// alokasi data
string *nama;
nama = new string[NDATA];

// Lakukan sesuatu dengan array nama
// ...

// dealokasi memori setelah tidak digunakan lagi
delete[] nama;
```