

MODUL PRAKTIKUM
ALGORITMA PEMROGRAMAN 1



Institut Teknologi Sumatera

2018

PETUNJUK PRAKTIKUM

Tahap Pelaksanaan Praktikum

Praktikum dilaksanakan dalam 2 tahap yang dilaksanakan secara berturut-turut, yaitu:

1. Latihan (opsional)
2. Tugas Praktikum

Latihan

Tahap ini bersifat **opsional!** Bisa dilakukan secara formal di lab, bisa juga dilakukan di luar lab. Dilakukan sebelum Tugas Praktikum yang sebenarnya dikerjakan.

Tujuan	Mahasiswa berlatih mengetikkan contoh-contoh program yang sesuai dengan topik yang diberikan dan mengamati hasil eksekusi sekaligus untuk membiasakan dengan lingkungan praktikum
Durasi	±60 menit
Sifat	individual (boleh berdiskusi dengan teman dan asisten)
Pelaksanaan	Mahasiswa mengerjakan latihan-latihan soal yang terdapat pada bab LATIHAN: <ol style="list-style-type: none">1. Baca petunjuk pada soal-soal latihan.2. Salin contoh-contoh program yang ada.3. <i>Compile</i> dan buatlah <i>executable file</i>, lalu eksekusi/jalankan program (jika kompilasi berhasil).4. Amati hasil eksekusi dan bandingkan dengan petunjuknya.5. Bahan latihan dapat dipilih sesuai dengan waktu yang tersedia.

Tugas Praktikum

WAJIB dilakukan dilakukan di lab!

Tujuan	Mahasiswa mengerjakan tugas-tugas praktikum secara mandiri sesuai dengan topik yang diberikan dan mengumpulkan hasilnya untuk penilaian
Durasi	±100 menit
Sifat	individual (tidak diperkenankan bekerja sama dengan pihak mana pun), tetapi masih boleh membuka material kuliah miliknya sendiri
Pelaksanaan	Mahasiswa mengerjakan soal-soal praktikum untuk modul terkait yang terdapat pada bab TUGAS PRAKTIKUM sesuai dengan petunjuk yang ada dalam durasi yang ditetapkan.

Petunjuk Penamaan dan Penulisan File Program (Untuk Peserta)

1. Pada setiap soal baik latihan maupun tugas praktikum, perhatikan petunjuk penamaan file.
2. Pada petunjuk penamaan file, gantilah <NIM> dengan NIM Anda masing-masing dan XX dengan nomor soal dalam 2 digit.

Contoh:

Jika file harus disimpan dengan format: **LP1_Hello1_<NIM>.cpp** dan NIM Anda adalah **19999888**, maka nama file Anda adalah: **LP1_Hello1_19999888.cpp**

Jika file harus disimpan dengan format: **P1_<NIM>_XX.cpp** dan NIM Anda adalah **19999888** serta nomor soal yang sedang dikerjakan adalah **2**, maka nama file Anda adalah: **P1_19999888_02.cpp**

3. Untuk setiap file source code program berikan identitas, minimum:

```
// NIM>Nama :  
// Nama file :  
// Tanggal :  
// Deskripsi :
```

4. Simpan dan upload file source code hasil latihan dan praktikum pada direktori yang ditentukan asisten.

Petunjuk Kompilasi Menggunakan GNU C++

Untuk petunjuk membuka lingkungan *compiler*, silakan berkonsultasi dengan asisten.

Contoh nama file source : **hello.cpp**

Compile program dan membentuk *executable file*, misalnya **hello.exe**:

```
>> g++ -c hello.cpp  
>> g++ -c hello.exe hello.o
```

Compile program sekaligus membentuk *executable file*, misalnya **hello.exe**:

```
>> g++ -o hello.exe hello.cpp
```

Pemanggilan program, misalnya **hello.exe**:

```
>> hello
```

Petunjuk untuk Pengelola Praktikum

1. Untuk penyerahan materi praktikum ke mahasiswa, bagian LATIHAN dan bagian TUGAS PRAKTIKUM harus diletakkan dalam file terpisah.
2. LATIHAN bisa diletakkan dan dibuka kapan saja sepanjang semester sementara TUGAS PRAKTIKUM hanya dibuka pada saat pelaksanaan praktikum saja.
3. Satu sesi praktikum (100 menit) harap diberikan hanya satu versi TUGAS PRAKTIKUM.
4. Form penilaian praktikum untuk satu mahasiswa dapat dilihat di akhir dokumen ini.

Variabel

Variabel adalah sebuah pengenal yang digunakan untuk menyimpan suatu nilai yang nilai tersebut dapat berubah-ubah selama dijalankannya program. Terdapat beberapa aturan yang digunakan dalam penulisan sebuah variabel.

- Terdiri dari huruf, angka dan karakter garis bawah (_) atau kombinasi antara ketiganya.
- Tidak boleh mengandung spasi dan terdiri dari minimal satu karakter.
- Dimulai dengan huruf atau garis bawah, tidak boleh diawali dengan angka.
- Bersifat case sensitive, yaitu huruf besar dan kecil dibedakan.

Berikut ini adalah contoh penulisan variabel yang benar: **nama_mhs**, **alamat**, **NIM**, **total_harga**.

Sedangkan penulisan variabel yang salah: **%nilai**, **pent!ng**, **ga lau**.

Tipe Data

Untuk memudahkan dalam menulis program, C++ mengelompokkan setiap variabel menjadi beberapa jenis. Misalnya jenis variabel yang mengandung nilai bilangan bulat dibedakan dengan variabel yang mengandung nilai bilangan pecahan. Berikut ini adalah tipe data yang dimiliki oleh C++.

Tipe Data	Ukuran Memori	Rentang Nilai
char	1 bytes	-127 to 127 or 0 to 255
unsigned char	1 bytes	0 to 255
signed char	1 bytes	-127 to 127
int	4 bytes	-2147483648 to 2147483647
signed int	4 bytes	0 to 4294967295
short int	2 bytes	-32768 to 32767
unsigned short int	Range	0 to 65,535
signed long int	4 bytes	same as long int
unsigned long int	4 bytes	0 to 4,294,967,295
float	4 bytes	+/- 3.4e +/- 38 (~7 digits)
double	8 bytes	+/- 1.7e +/- 308 (~15 digits)
long double	8 bytes	+/- 1.7e +/- 308 (~15 digits)
wchar_t	2 or 4 bytes	1 wide character

Deklarasi Variabel

Sebelum digunakan didalam program, sebuah variabel perlu dideklarasikan terlebih dahulu. Deklarasi variabel juga berguna untuk menentukan tipe data variabel itu sendiri. Bentuk umum deklarasi dari sebuah variabel adalah sbb:

```
tipe_data nama_variabel;
```

Contoh deklarasi tipe data:

```
int x;           // deklarasi variable bertipe integer
char nama;      // deklarasi variable bertipe character
float ipk;      // deklarasi variable bertipe bilangan real
double beta;    // deklarasi variable bertipe bilangan real
```

Konstanta

Konstanta adalah sebuah tetapan nilai dalam sebuah program. Nilai konstanta tidak dapat diubah selama program dijalankan.

Deklarasi konstanta dapat dilakukan dengan cara sebagai berikut:

```
const tipe_data nama_konstanta = nilai;
```

```
#define nama_konstanta nilai_konstanta
```

Operator

Operator digunakan untuk memberikan nilai bagi sebuah variabel. Dalam pemrograman terdapat beberapa jenis operator, yaitu operator aritmatika, operator penugasan, operator perbandingan, operator logika, dll.

Dari sekian banyak operator yang ada, terlebih dahulu akan dipelajari operator aritmatika. Operator ini digunakan dalam melakukan perhitungan matematika.

Operator	Nama Operasi	Contoh
+ (Unary)	Positif	+5
- (Unary)	Negatif	-5
+	Penjumlahan	5 + 5
-	Pengurangan	5 - 5
*	Perkalian	5 * 5
/	Pembagian	5 / 5
%	Hasil Bagi	5 % 5

Percabangan

Percabangan adalah suatu proses pemilihan aksi di antara beberapa alternatif yang diberikan.

Terdiri atas :

Kondisi : ekspresi yang menghasilkan true dan false

Aksi : statement yang dilaksanakan jika kondisi yang berpasangan terpenuhi

1. Struktur if (1 kondisi) :

```
if (kondisi){  
    aksi;                // aksi dilakukan jika kondisi terpenuhi  
}
```

2. Struktur if (2 kondisi):

```
if (kondisi){  
    aksi_1;              // aksi_1 dilakukan jika kondisi terpenuhi  
} else {  
    aksi_2;              // aksi_2 dilakukan jika kondisi tidak terpenuhi  
}
```

3. Struktur if (3 atau lebih kondisi):

```
if (kondisi_1){  
    aksi_1;              // aksi_1 dilakukan jika kondisi_1 terpenuhi  
} else if(kondisi_2){  
    aksi_2;              // aksi_2 dilakukan jika kondisi_2 terpenuhi  
} else if(kondisi_3){  
    aksi_3;              // aksi_3 dilakukan jika kondisi_3 terpenuhi  
}
```

4. Struktur switch:

```
switch(variabel){        // variabel bertipe ordinal  
    case value_1:         // value_1, value_2, dst bersifat konstan  
        statement_1;  
        break;           // break berguna untuk keluar dari switch  
    case value_2:  
        statement_2;  
        break;  
    default:  
        statement_3;  
}
```

LATIHAN

Latihan 1. Input dan Output

Nama file: LP2_<NIM>_01.cpp

Deskripsi:

Buatlah program yang digunakan untuk menerima masukan sebuah **nama** orang (bertipe string) dan **umur** dari orang tersebut (bertipe integer) dan menampilkan ke layar:

Halo, **nama**!
Umurmu adalah **umur** tahun.

Program:

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

int main (){
    // Kamus
    string nama;
    int umur;

    // Algoritma
    cout << "Masukkan Nama :";
    cin >> nama;
    cout << "Masukkan Umur :";
    cin >> umur;
    cout << "Halo, " << nama << "!" << endl;
    cout << "Umurmu adalah " << umur << " tahun." << endl;

    return 0;
}
```

Contoh Input dan Output:

Input		Output
Nama	Umur	
Eko	17	Halo, Eko! Umurmu adalah 17 tahun
Citra	20	Halo, Citra! Umurmu adalah 20 tahun

Latihan 2. Integer dan Float

Nama file: LP2_<NIM>_02.cpp

Deskripsi:

Integer adalah tipe bilangan bulat, sedangkan float adalah tipe bilangan real. Kita harus tahu kapan menggunakan float, kapan menggunakan integer. Program yang akan dibuat akan membandingkan tipe bilangan integer dan tipe bilangan float.

Variabel X bertipe float dan variabel Y bertipe integer. Kemudian diberikan input yang sama, kita akan lihat, apakah output yang dihasilkan juga sama.

Program:

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

int main(){
    // Kamus
    float x;
    int y;

    // Algoritma
    cout << "Masukkan nilai x : ";
    cin >> x;
    cout << "Masukkan nilai y : ";
    cin >> y;
    cout << "Nilai x yang dimasukkan adalah " << x << endl;
    cout << "Nilai y yang dimasukkan adalah " << y << endl;

    return 0;
}
```

Contoh Input dan Output:

Input		Output
X	Y	
12	12	Nilai x yang dimasukkan adalah 12 Nilai y yang dimasukkan adalah 12
12.6	12.6	Nilai x yang dimasukkan adalah ??? Nilai y yang dimasukkan adalah ???

Apakah hasilnya sama? Jika tidak sama, konsultasi dengan asisten, mengapa tidak sama.

Latihan 3. String dan Char

Nama file: LP2_<NIM>_03.cpp

Deskripsi:

String dan char. Tidak sedikit yang bingung dengan perbedaan keduanya. Program yang dibuat akan membandingkan tipe data string dan tipe data char.

Variable X bertipe string dan variable Y bertipe char. Kemudian diberikan input yang sama, kita akan lihat, apakah output yang diberikan juga sama.

h

Program:

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

int main(){
    // Kamus
    string x;
    char y;

    // Algoritma
    cout << "Masukkan nilai x : ";
    cin >> x;
    cout << "Masukkan nilai y : ";
    cin >> y;
    cout << "Nilai x yang dimasukkan adalah " << x << endl;
    cout << "Nilai y yang dimasukkan adalah " << y << endl;

    return 0;
}
```

Contoh Input dan Output:

Input		Output
X	Y	
D	D	Nilai x yang dimasukkan adalah D Nilai y yang dimasukkan adalah D
Data	Data	Nilai x yang dimasukkan adalah ??? Nilai y yang dimasukkan adalah ???

Apakah hasilnya sama? Jika tidak sama, konsultasi dengan asisten, mengapa tidak sama.

Latihan 4. Menentukan Wujud Air (Percabangan IF-ELSE-IF)

Nama file: LP2_<NIM>_04.cpp

Deskripsi:

Buatlah program untuk menentukan wujud air (uap, cair, es) jika temperaturnya berbeda (dalam Celcius). Jika temperatur ≥ 100 maka wujud air adalah gas, jika temperatur ≤ 0 wujud air adalah es/beku, jika temperaturnya diantara kedua tersebut maka wujud airnya adalah cair.

Program:

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

int main(){
    // Kamus
    int suhu;

    // Algoritma
    cout << "Masukkan temperatur dalam satuan Celcius : ";
    cin >> suhu;

    if (suhu >= 100){
        cout << "Wujud air adalah gas";
    } else if (suhu <= 0){
        cout << "Wujud air adalah es";
    } else {
        cout << "Wujud air adalah cair";
    }

    return 0;
}
```

Contoh Input dan Output:

Input	Output
124	Wujud air adalah gas
-18	Wujud air adalah es
88	Wujud air adalah cair