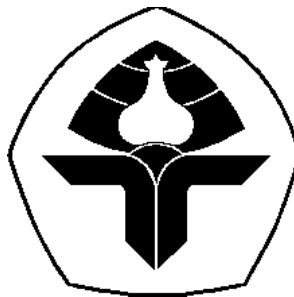


**JOB SHEET**

# **PEMROGRAMAN DASAR**

**MKK-13206**



**Disusun Oleh**  
**Putu Indah Ciptayani, S.Kom. M.Cs**

**PROGRAM STUDI MANAJEMEN INFORMATIKA  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI BALI  
2018**

<b>POLITEKNIK NEGERI BALI</b>	<b>Job Sheet 2</b>  <b>Tipe Data, Variabel, dan Konstanta</b>	<b>Dosen Pengampu : Putu Indah Ciptayani, M.Cs</b>
<b>Bidang Studi : Manajemen Informatika</b>		<b>Waktu : 3 x 50 Menit</b>
<b>Jurusan : Teknik Elektro</b>		<b>Pertemuan 2</b>
<b>Kelas/Sem : IB/ Th Ajaran : 2018/2019</b>		

**A. Standar Kompetensi**

Mahasiswa diharapkan mampu menerjemahkan permasalahan dalam dunia nyata ke dalam algoritma dan mengimplementasikannya ke dalam Bahasa C setelah mempelajari mata kuliah ini.

**B. Kompetensi Dasar**

Setelah mempelajari materi ini, mahasiswa diharapkan mampu memahami dan menggunakan berbagai tipe data dalam Bahasa C.

**C. Indikator**

1. Mahasiswa mampu menerapkan tipe data dalam Bahasa C
2. Mahasiswa mampu menggunakan sintaks keluaran dan masukan dengan berbagai tipe data

**D. Peralatan**

1. Seperangkat computer yang sudah terinstal compiler Bahasa C
2. Buku Ajar
3. Job sheet

**E. Teori**

**1. Tipe Data Dasar**

Data merupakan suatu nilai yang bisa dinyatakan dalam bentuk konstanta atau variabel. Konstanta menyatakan nilai yang tetap, sedangkan variabel menyatakan nilai yang dapat diubah-ubah selama eksekusi berlangsung.

Data berdasarkan jenisnya dapat dibagi menjadi 5 kelompok, yang dapat dinyatakan sebagai tipe data dasar. Kelima tipe data dasar tersebut adalah sebagai berikut :

Tipe	Total Bit	Kapasitas	Keterangan
Char	8	-128 s/d 127	Karakter
Int	16	-32768 s/d 32767	Bil. integer
Float	32	3.4E-38 s/d 3.4E+38	Bil. Real (presisi tunggal)
Double	64	1.7E-308 s/d 1.7E_308	Bil. Real (presisi ganda)
Void	0	-	Tak bertipe

Untuk tipe data short int, long int, signed int dan unsigned int, maka ukuran memori yang diperlukan serta kawasan dari masing-masing tipe data adalah sebagai berikut:

Tipe	Total bit	Kawasan	Keterangan
short int	16	-32768 s/d 32767	short integer
long int	32	-2147483648 s/d 2147483647	long integer
signed int	32	-2147483648 s/d 2147483647	biasa disingkat dengan int
unsigned int	32	0 s/d 4294967295	bilangan int tak bertanda

**Catatan :**

Ukuran dan kawasan dari masing-masing tipe data adalah bergantung pada jenis mesin yang digunakan (misalnya mesin 16 bit bisa jadi memberikan hasil berbeda dengan mesin 32 bit)

## **2. Variabel**

Variabel adalah suatu pengenal (identifier) yang digunakan untuk mewakili suatu nilai tertentu di dalam proses program. Berbeda dengan konstanta yang nilainya selalu tetap, nilai dari suatu variable bisa diubah-ubah sesuai kebutuhan.

Nama dari suatu variable dapat ditentukan sendiri oleh pemrogram dengan aturan sebagai berikut :

- Pengenal harus diawali dengan huruf (A..Z, a..z) atau karakter garis bawah (\_).
- Selanjutnya dapat berupa huruf, digit (0..9) atau karakter garis bawah atau tanda dollar (\$).
- Panjang pengenal boleh lebih dari 31 karakter, tapi hanya 31 karakter pertama yang akan dianggap berarti.
- Pengenal tidak boleh menggunakan nama yang tergolong sebagai kata-kata cadangan (reserved words) seperti int, if, while, dan sebagainya.

### **2.1 Mendeklarasikan Variabel**

Variabel digunakan dalam program untuk menyimpan suatu nilai, dan nilai yang ada padanya dapat diubah-ubah selama eksekusi program berlangsung. Variabel yang akan digunakan dalam program harus dideklarasikan terlebih dahulu. Pengertian deklarasi disini berarti memesan memori dan menentukan jenis data yang bisa disimpan di dalamnya. Bentuk umum deklarasi variabel adalah sebagai berikut:

```
tipe daftar-variabel;
```

Pada pendeklarasian variabel, daftar variabel dapat berupa variabel atau beberapa variabel yang dipisahkan dengan koma.

Contoh:

```
Int var_bulat1;
```

```
Float var_pecahan1, var_pecahan2;
```

## 2.2 Memberikan nilai ke variabel

Untuk memberikan nilai ke variabel yang telah dideklarasikan, maka bentuk umum pernyataan yang digunakan adalah sebagai berikut:

```
nama_variabel = nilai;
```

Contoh:

```
int var_bulat = 10;
```

```
double var_pecahan = 10.5;
```

## 2.3 Inisialisasi variabel

Adakalanya dalam penulisan program, setelah dideklarasikan variabel langsung diberi nilai awal. Sebagai contoh yaitu variabel nilai:

```
int nilai;
```

```
nilai = 10;
```

Dua pernyataan diatas sebenarnya dapat disingkat melalui pendeklarasian

yang disertai penugasan nilai, sebagai berikut:

```
int nilai = 10;
```

## 3. Konstanta

Konstanta menyatakan nilai yang tetap. Berbeda dengan variabel, suatu konstanta tidak dideklarasikan. Namun seperti halnya variabel, konstanta juga memiliki tipe. Penulisan konstanta mempunyai aturan tersendiri, sesuai dengan tipe masing-masing.

- Konstanta karakter misalnya ditulis dengan diawali dan diakhiri dengan tanda petik tunggal, contohnya: 'A' dan '@'.
- Konstanta integer ditulis dengan tanda mengandung pemisah ribuan dan tak mengandung bagian pecahan, contohnya : -1 dan 32767.
- Konstanta real (float dan double) bisa mengandung pecahan (dengan tanda berupa titik) dan nilainya bisa ditulis dalam bentuk eksponensial (menggunakan tanda e) contohnya: 27.5 f ( untuk

tipe float) atau 27.5 (untuk tipe double) dan 2.1e+5 (maksudnya 2,1x10<sup>5</sup>)

- Konstanta string merupakan deretan karakter yang diawali dan diakhiri dengan tanda petik ganda ("), contohnya : "Pemrograman Dasar C"

#### 4. Fungsi printf()

Bentuk umum pernyataan printf():

```
printf("string kontrol", argumen1, argumen 2,...)
```

String kontrol dapat berupa keterangan yang akan ditampilkan pada layar beserta penentu format (seperti %d, %f, %c). Penentu format dipakai untuk memberi tahu kompiler jenis data yang akan ditampilkan. Argumen sesudah string kontrol (Argumen1, argumen2,...) adalah data yang akan ditampilkan ke layar. argumen ini dapat berupa variabel, konstanta, bahkan ungkapan. Misal:

```
printf("%d",20); //argumen berupa konstanta
printf("%d,a"); //argumen berupa variabel
printf("%d",a+20); //argumen berupa ungkapan
```

Berikut ini adalah tabel yang berisi penentu format pada printf() :

%u	untuk menampilkan data bilangan tak bertanda ( <i>unsigned</i> ) dalam bentuk desimal.
%d	untuk menampilkan bilangan integer bertanda ( <i>signed</i> ) dalam bentuk desimal
%i	
%o	untuk menampilkan bilangan bulat tak bertanda dalam bentuk oktal.
%x	untuk menampilkan bilangan bulat tak bertanda dalam bentuk heksadesimal
%X	(%x → notasi yang dipakai : a, b, c, d, e dan f sedangkan %X → notasi yang dipakai : A, B, C, D, E dan F )
%f	untuk menampilkan bilangan real dalam notasi : dddd.ddddd
%e	untuk menampilkan bilangan real dalam notasi eksponensial
%E	
%g	untuk menampilkan bilangan real dalam bentuk notasi seperti %f, %E atau %F
%G	bergantung pada kepresisian data (digit 0 yang tak berarti tak akan ditampilkan)
l	merupakan awalan yang digunakan untuk %d, %u, %x, %X, %o untuk menyatakan long int (misal %ld). Jika diterapkan bersama %e, %E, %f, %F, %g atau %G akan menyatakan <i>double</i>
L	Merupakan awalan yang digunakan untuk %f, %e, %E, %g dan %G untuk menyatakan <i>long double</i>
h	Merupakan awalan yang digunakan untuk %d, %i, %o, %u, %x, atau %X, untuk menyatakan <i>short int</i> .

#### 5. Fungsi scanf()

Fungsi ini digunakan untuk memasukkan berbagai jenis data. Misalnya untuk memasukkan data jari-jari lingkaran pada program menghitung luas lingkaran.

```
scanf("%f", &radius);
```

Bentuk scanf() sesungguhnya menyerupai printf(). Fungsi ini melibatkan penentu format yang pada dasarnya sama digunakan pada printf(). Secara umum bentuk scanf() adalah sebagai berikut:

- scanf("string kontrol", daftar argumen);

## **F. Keselamatan Kerja**

1. Ikuti langkah kerja yang ada pada job sheet
2. Perhatikan keamanan instalasi listrik saat menghidupkan dan mematikan computer
3. Pastikan lingkungan kerja (meja, kursi dan computer) bersih, baik sebelum maupun sesudah penggunaan
4. Jika ada hal yang kurang jelas, tanyakan kepada dosen pengampu

## **G. Langkah Kerja**

### **1. Program 1**

- a) Bukalah aplikasi Bahasa C pada computer
- b) Ketikkan program berikut :

```
//PROGRAM SEDERHANA UNTUK MENGENAL TIPE DATA
#include <stdio.h>
int main()
{
    int arus, hambatan, tegangan;
    arus = 10;
    hambatan = 500;
    tegangan = arus*hambatan;
    printf("Voltase = %d", tegangan);
    return 0;
}
```

- c) Simpan pekerjaan Anda dengan nama prak2\_1.c
- d) Lakukan kompilasi dan jalankan program
- e) Perhatikan hasil program Anda

### **2. Program 2**

- a) Ketikkan program berikut :

```
//PROGRAM SEDERHANA UNTUK MENGENAL TIPE DATA
#include <stdio.h>
int main()
{
    char nama_komponen[15];
    int jumlah, harga, bayar;
    printf("Nama      komponen      =      ");    scanf("%s",
        &nama_komponen);
    printf("Harga Komponen = "); scanf("%d", &harga);
    printf("jumlah beli = "); scanf("%d",&jumlah);
    bayar=jumlah*harga;
    printf("Yang harus dibayar sejumlah = %d\n\n",
```

```
        bayar);  
    return 0;  
}
```

- b) Simpan pekerjaan Anda dengan nama prak2\_2.c
- c) Lakukan kompilasi dan jalankan program
- d) Perhatikan hasil program Anda

#### **H. Evaluasi**

Kerjakan soal berikut dan kumpulkan hasilnya dalam bentuk hard copy pada pertemuan berikutnya!

1. Buatlah program untuk mencari sisi miring segitiga, keliling dan luas lingkaran dan volume kubus.