

# Pemrograman Dasar

Pertemuan III

# Pembahasan Tugas

- .....

# Prosedur

- Prosedur memanaskan makanan dalam microwave :
  - Taruhlah makanan dalam wadah yang aman untuk microwave
  - Tutuplah pintu microwave dengan rapat
  - Tancapkan steker ke posisi stop kontak
  - Putarlah knop ke posisi 4 menit
  - Tunggu sampai lampu mati dan ada bunyi 'ting'
  - Lepaskan steker dari stop kontak
  - Bukalah pintu microwave dan keluarkan wadah yang berisi makanan tersebut
- PROSEDUR : Urutan untuk memandu orang dalam melakukan sebuah proses
- Apa bedanya dengan algoritma ??

# Prosedur vs Algoritma

- Perhatikan prosedur berikut
  1. Ambil lima buah kartu
  2. Urutkan kartu tersebut (→ PROSEDUR)
- Pada langkah kedua, manusia bisa mengurutkannya dengan mudah. Namun, tidak dengan komputer. Komputer tidak mampu memaknai prosedur
- Komputer membutuhkan petunjuk yang sangat detail tentang tata cara pengurutan kartu, yaitu **algoritma**
- Contoh : Urutkan semua kartu yang anda miliki dengan urutan prioritas (dari rendah ke tinggi) seperti berikut : angka ascending (2, 3, ... 10) , kemudian dilanjutkan dengan 'J', 'Q', 'K', 'A' → **ALGORITMA**

# Struktur Program C

- Kode “Selamat Belajar”

```
#include <stdio.h>      Menyertakan berkas header  
                          stdio.h  
  
int main( )  Tipe kode output program  
{           adalah integer  
    print(“Selamat Belajar C\n”); Perintah untuk  
    return 0; menampilkan string  
}           Kode keluar program
```

- Berkas header file `.h` adalah berkas yang berisi prototipe fungsi, konstanta, dan definisi variabel
- Perintah `#include` adalah perintah yang digunakan untuk mengatur kompiler agar membaca header yang ditunjuk
- `main()` adalah fungsi pertama yang akan dijalankan oleh compiler ketika mengeksekusi program
- Nilai balik program dinyatakan dengan perintah *return*. Karena tipe `main` nya adalah `int` (integer / bil.bulat), maka `return`-nya diberikan 0.

# C dan C++

- C++ adalah versi bahasa C yang mendukung Pemrograman Berorientasi Objek
- C++ menggunakan library yang berbeda dengan bahasa C
- Berbeda dengan bahasa C yang merupakan pemrograman terstruktur (*structured programming*), C++ mengedepankan modular dan object di dalam implementasinya
- Karena berbasis objek dan mengedepankan modular, C++ biasa digunakan dalam sebuah implementasi / project berskala besar yang melibatkan tim.

# C vs C ++

- Kode “Selamat Belajar”

```
#include <stdio.h>
```

```
int main( )  
{  
    printf(“Selamat Belajar C\n”);  
    return 0;  
}
```

```
//pertama.c
```

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

```
int main()  
{  
    cout << “Selamat Belajar C++” << endl;  
    return 0;  
}
```

```
//pertama.cpp
```

# Kompilasi (compile) & IDE

- Kompilasi adalah proses penyusunan kumpulan source code yang sudah teratur menjadi bahasa mesin yang siap dieksekusi.
- Dalam proses kompilasi bahasa C, kita memerlukan IDE (Integrated Development Environment) dan Compiler
- IDE digunakan sebagai tool editor atau alat bantu dalam proses pembuatan program
- Compiler sebagai alat untuk melakukan kompilasi
- Bloodshed Dev C++ adalah contoh IDE yang sudah ditanamkan compiler di dalamnya.
- Mempunyai 2 langkah utama
  - Build – melakukan kompilasi dari source code menjadi bahasa mesin siap dieksekusi
  - Run - mengeksekusi hasil kompilasi



# Type data

- C dan C++ menyediakan berbagai tipe data dasar/primitif :

Tipe	Keterangan
char	Menyatakan sebuah karakter. Contoh : A, f, 9, atau *
double	Menyatakan bilangan real dengan ketelitian tinggi (15 digit). Dapat menampung dari $10^{-308}$ hingga $10^{308}$
float	Menyatakan bilangan real dengan ketelitian rendah (6-7 digit). Dapat menampung $10^{-38}$ hingga $10^{38}$
short	Menyatakan bilangan bulat antara -32768 hingga +32767
int	Menyatakan bilangan bulat antara -2147483648 hingga +2147483647
long	Menyatakan bilangan bulat antara -9232372036854775808 hingga +9232372036854775807
boolean	Menyatakan nilai logika, <b>true (1)</b> atau <b>false (0)</b>

# Penting ...

- Harus berhati-hati, karena anda tidak bisa memasukkan nilai sebuah tipe data ke variabel dengan tipe data lain di C atau C++
- Misal :

```
double PHI = 3.14;  
int temp = 0;
```

`temp = PHI;` ← tidak mungkin, akan error. Karena temp bertipe bilangan bulat (integer) dan tidak akan bisa menampung double.

# Variabel

- Variabel secara bahasa, artinya kiasan.
- Variabel : nama yang menyatakan tempat dalam memori komputer yang digunakan untuk menyimpan suatu nilai, dan nilai tersebut dapat diubah sewaktu-waktu saat program sedang dieksekusi
- Sebelum menggunakan variabel dalam program, kita harus mendeklarasikannya di awal program.

*tipe data nama variabel*

- Contoh deklarasi variabel :

Deklarasi	Keterangan
int jumlah	Variabel jumlah bertipe int untuk menyimpan bilangan bulat
long int jum_penduduk	Variabel jum_penduduk bertipe long int untuk menyimpan bilangan bulat hingga angka 2 miliar
char huruf	Variabel huruf bertipe char untuk menampung satu karakter
bool selesai	Variabel selesai bertipe bool, nilainya benar atau salah

# Pemberian Nilai ke Variabel

- `int jumlah = 64;`  
//pendeklarasian variabel jumlah bertipe integer dan pemberian nilai awal yaitu 64
- `jumlah = 85;`  
//mengisikan nilai 85 ke variabel jumlah
- `jumlah = jumlah + 1;`  
//menambahkan nilai 1 ke variabel jumlah, sehingga menjadi 86
- `jumlah--;`  
//mengurangkan nilai jumlah saat ini dengan 1, sehingga menjadi 85

# Literal / Konstanta

- Adalah sebuah variabel yang menyatakan nilai tetap di dalam sebuah program
- C dan C++ mempunyai 6 jenis konstanta
  - short
  - long int atau int
  - float
  - double
  - bool
  - char
- Cara penggunaan :  
*const tipe data nama variabel*
- Misal :  
const double PHI 3.14

# Contoh konstanta

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    const double PHI = 3.14;
```

```
    double luas, jari_jari;
```

```
    jari_jari = 20;
```

```
    luas = PHI * jari_jari * jari_jari;
```

```
    printf("Luas lingkaran = %lf", luas);
```

```
    return 0;
```

```
}
```

```
//initstr.c
```

# String

- String artinya deretan karakter.
- String bisa mengandung sebuah char, banyak char, dll.
- String ditulis dengan awalan dan akhiran tanda petik ganda (“).
- Contoh :

String	Keterangan
“Yogya”	String dengan panjang 5 karakter
“Y”	String dengan panjang 1 karakter
“”	String kosong (tidak ada satu karakter pun)
“Jl. Beo 45/B”	String mengandung karakter huruf, angka, simbol
“123”	String berisi 3 karakter angka, “1”, “2”, dan “3”. Berbeda dengan bilangan bulat int angka 123.

# String

- String selalu diakhiri dengan karakter NULL (`\0`)

Y	O	G	Y	A	\0
---	---	---	---	---	----

- Contoh deklarasi string :

```
char kota[6];
```

mendeklarasikan variabel kota yang dapat menampung 6 karakter huruf.

- Secara umum, jika kita menghendaki membuat variabel string yang dapat menampung  $n$  karakter, maka kita harus mendeklarasikannya dengan setidaknya  $n+1$  karakter untuk menampung karakter NULL.
- Berbeda dengan variabel selain string, untuk pemberian nilai ke string tidak bisa menggunakan "=", namun dengan menggunakan perintah **strcpy()**.



# Contoh String

```
#include <stdio.h>
#include <string.h>

int main ()
{
    char info[27];
    strcpy(info, "Kelas Dasar Pemrograman");
    printf(info);
    printf("\n");

    strcpy(info, "Teknik Informatika UPN");
    printf(info);

    return 0;
}
```

# Menampilkan info ke layar

- Menggunakan perintah :
  - puts() – menampilkan string ke layar, sudah mengandung karakter *newline*
  - putchar() – menampilkan sebuah karakter saja.
  - printf
- Penentu format pada printf :
  - %u – menampilkan bilangan bulat tak bertanda
  - %d atau %i – menampilkan bilangan bulat bertanda
  - %o – menampilkan bilangan bulat tak bertanda dalam bentuk sistem oktal (berbasis 8)
  - %x – menampilkan bilangan bulat tak bertanda dalam bentuk sistem oktal (berbasis 16). Huruf dinyatakan dalam huruf kecil
  - %X – menampilkan bilangan bulat tak bertanda dalam bentuk sistem oktal (berbasis 16). Huruf dinyatakan dalam huruf besar
  - %f – menampilkan bilangan real dalam bentuk ddd.ddddd
  - %e atau %E – menampilkan bilangan real dalam bentuk sains
  - %g atau %G – mirip dengan %f atau %F tergantung pada kepresisian data
  - l – digunakan untuk menyatakan long int atau long double. Contoh : %ld atau %lf
  - %% - untuk mencetak simbol %

# Lebar tampilan data

- Printf(“Cacah %2d”, 23);

C	a	c	a	h		2	3
---	---	---	---	---	--	---	---

- Printf(“Cacah %3d”, 23);

C	a	c	a	h			2	3
---	---	---	---	---	--	--	---	---

- Printf(“Cacah %4d”, 23);

C	a	c	a	h				2	3
---	---	---	---	---	--	--	--	---	---

- Printf(“Cacah %8.2d”, 52.00);

			5	2	.	0	0
--	--	--	---	---	---	---	---

- Printf(“%8s”, “Hai”)

					H	a	i
--	--	--	--	--	---	---	---

- Printf(“%-8s”, “Hai”)

H	a	i					
---	---	---	--	--	--	--	--

# Latihan Soal

- Tipe data apa yang anda perlukan ?
  - Usia orang
  - Sebuah huruf kapital
  - Jumlah penduduk di Indonesia
  - Tahun kelahiran seseorang
  - Alamat rumah
- Ubahlah notasi matematika berikut ke dalam operasi C :
  - $(2x + 3y)(2z - 4y)$
  - $\left(3 + \frac{a}{b}\right)^{m-1}$
  - $\frac{\frac{a}{b}+6}{x-\frac{y}{z}}$
  - $\frac{\frac{1}{x}+\frac{1}{y}}{\frac{1}{x}-\frac{1}{y}} + \frac{x}{y}$