



Variabel

(TIB26 – ALGORITMA PEMROGRAMAN)

Pertemuan 3, 4

Sub-CPMK

- Mahasiswa mampu mendeklarasikan variabel dan penugasan sesuai dengan peruntukannya (C3, A3)

Materi

- Tipe Data
- Deklarasi
- Inisialisasi
- Enumerasi
- Penugasan

Perhatian

- Tidak disarankan copy-paste kode program dari presentasi ini, karena ada beberapa symbol yang dianggap sebagai Unicode oleh editor yang anda gunakan, sehingga akan dianggap sebagai symbol yang salah oleh compiler, sebaiknya diketik ulang saja



1.

Tipe Data

1.1 Variabel

- Variabel merupakan penyimpanan data
- Pada pemrograman, variabel akan disimpan pada alamat memory pada lokasi yang sesuai dengan panjang datanya
- Karena itu variabel pada pemrograman harus dideklarasikan terlebih dahulu untuk mengalokasikan alamat memory sebagai tempat penyimpanan data dari variable tersebut

1.2. Ketentuan penamaan variabel

- Case sensitive
- Tidak boleh diawali dengan angka
- Terdiri dari satu kata, tidak boleh dipisahkan dengan spasi
- Tidak menggunakan karakter symbol kecuali underscore
- Tidak Menggunakan keyword

1.3. Alokasi Memori

- Alokasi Memory memerlukan panjang Data dan bagaimana data tersebut diperlakukan, sehingga tidak terjadi kerancuan pada pemanggilan maupun penyimpanan
- Panjang data dan perlakuan data tergantung pada jenis data
- Karena itu variabel harus dideklarasikan sesuai dengan jenis datanya

1.4. Tipe Data Integer

- Tipe integer/bilangan bulat dideklarasikan dengan kata kunci berikut ini:

Keyword	Besarnya Memori	Jangkauan Nilai
Int	16 bit	-32768 s/d 32768
Short/short	16 bit	-32768 s/d 32768
Long int / long	32 bit	-2147483648 s/d 2147483648

1.5. Tipe Data Unsigned Integer

Keyword	Besarnya Memori	Jangkauan Nilai
Unsigned int	16 bit	0 s/d 65535
Unsigned short	16 bit	0 s/d 65535
Unsigned long	16 bit	0 s/d 4294967295

1.6. Tipe Data Floating Point

Keyword	Besarnya Memori	Jangkauan Nilai
Float	32 bit	3.4E-038 s/d 3.4E+038
Double	64 bit	1.7E-038 s/d 1.7E+038
Long double	64 bit	1.7E-038 s/d 1.7E+038



2.

Deklarasi

2.1. Deklarasi

- Pada pemrograman, variabel akan disimpan pada alamat memory pada lokasi yang sesuai dengan panjang datanya
- Karena itu variabel pada pemrograman harus dideklarasikan terlebih dahulu untuk mengalokasikan alamat memory sebagai tempat penyimpanan data dari variabel tersebut

2.2. Syntax Deklarasi

- Pada Bahasa Pemrograman C, deklarasi variabel dilakukan dengan menuliskan perintah deklarasi sesuai dengan tipe datanya diikuti dengan nama variable
- Contoh:
- `int a; //Deklarasi variabel a bertipe data integer`
- `float b; //Deklarasi variabel b bertipe data floating point`
- `char c; //Deklarasi variabel c bertipe data char`

2.3. Keyword Deklarasi

- char → Character
- unsigned char → char tak bertanda
- signed char → char bertanda
- int → integer
- unsigned int → integer tak bertanda
- signed int → integer bertanda
- float → Floating Point 32 bit
- double → floating point 64 bit
- bool → Boolean (0/1)

2.4. Deklarasi Char

- Deklarasi char dapat berupa sebuah character tunggal, dapat juga berupa rangkaian character.
- Deklarasi variable character tunggal dapat dilakukan langsung dengan perintah char

```
char c; //deklarasi variable character tunggal
```

- Deklarasi rangkaian character dapat kita lakukan dengan bantuan array char (array akan di bahas pada pembahasan Array) dengan susunan:

```
char nama_variable[panjangrangkaian];
```

Contoh:

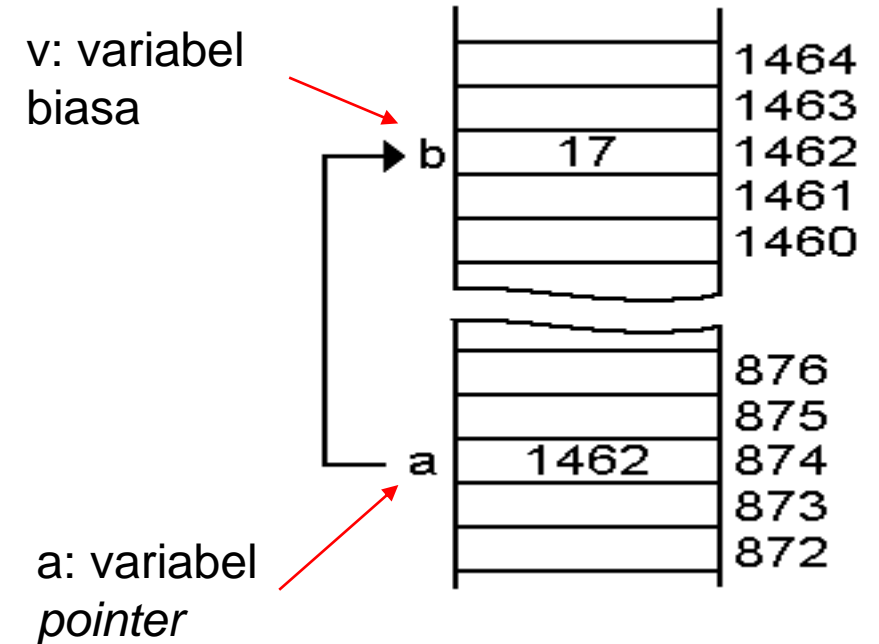
```
char nama[20]; //deklarasi variable nama bertipe character sepanjang 20 char
```


2.5. *Pointer*

- *Pointer* adalah variabel yang berisi alamat dari suatu *memory* yang menyimpan data
- Variabel *pointer* dimaksudkan menyimpan alamat awal dari suatu *memory* yang berisi sel-sel *memory* yang dialokasikan untuk menyimpan data sesuai dengan tipe data yang dialokasikan

2.6. Contoh *Pointer*

- Variabel *Pointer a* menunjuk ke alamat variabel *b*.
- variabel *b* menyimpan sebuah bilangan (17)
- Variabel *a* dapat menyimpan alamat dari variabel *b* pada *memory* (1462)



2.7. Deklarasi Variabel bertipe *Pointer*

- Variabel *Pointer* dapat dideklarasikan dengan mencantumkan karakter * di depan variabel yang dideklarasikan
- Contoh:
`int *a;`
- Deklarasi ini akan mendaftarkan variabel a untuk menyimpan data bertipe *pointer* yang ditujukan untuk menunjuk ke alamat yang berisi data int



3.

Inisialisasi

3.1. Inisialisasi

- Inisialisasi adalah pengisian nilai awal variabel
- Biasanya diisikan pada saat variabel di deklarasikan

Contoh:

```
int a = 100; //deklarasi variable a bertipe integer dan insialisasi dengan nilai 100
```

- Dapat juga melakukan inisialisasi melalui statement terhadap suatu variable

Contoh:

```
int a;  
a = 100;
```

3.2. Aturan penulisan inisialisasi

- Inisialisasi pada variable tunggal

Tipe_data Nama_variable = nilai inisialisasi;

- Inisialisasi variable array dapat dilakukan dengan menggunakan kurung kurawal dengan data set di dalam kurung kurawal tersebut, tiap item data dipisahkan tanda koma

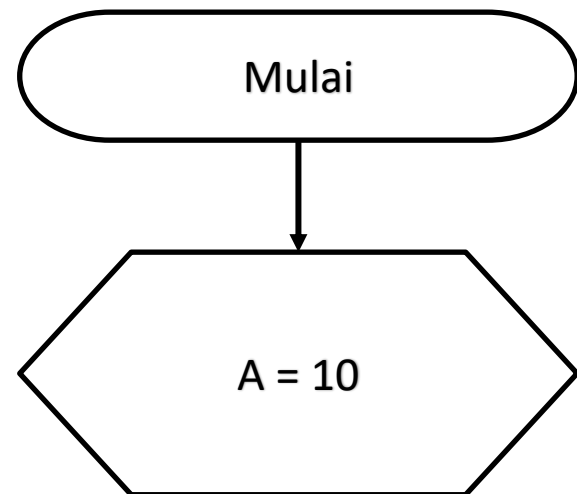
Tipe_data nama_array[panjangArray] = {data_0, data_1, data_2, ...data_n};

3.3. Preparasi

- Inisialisasi dan enumerasi pada flowchart dapat diterapkan menggunakan notasi preparasi

Contoh:

`int a=10;`





4.

Enumerasi

4.1. Enumerasi

- Enumerasi adalah sekumpulan konstanta bilangan bulat / integer yang masing-masing konstanta akan memiliki nama dan nilai yang berbeda.
- Enumerasi lebih ditujukan untuk penanganan kesalahan proses input, output maupun proses pengolahan data dalam internal CPU.
- Penggunaan konstanta yang didefinisikan atau pun enumerasi akan mempermudah penanganan kesalahan daripada saat hanya menggunakan angka.

4.2. Aturan Penulisan

- Enumerasi menggunakan keyword enum
- Syntax enumerasi:

```
enum nama_variable{enumerasi1, enumerasi2, enumerasi3,  
...enumerasi_n};
```

Enumerasi pertama akan bernilai 0, dan enumerasi berikutnya akan melanjutkan nilai setelah 0: 1, 2, 3, dst

Contoh:

```
enum bilangan{nol, satu, dua, tiga, empat, lima}
```

4.2. Aturan Penulisan (lanj...)

- Syntax enumerasi:

```
enum nama_variable{enumerasi1=angkamulai, enumerasi2, enumerasi  
3, ...enumerasi n};
```

Enumerasi pertama akan bernilai 1, dan enumerasi berikutnya akan melanjutkan nilai setelah 1: 2, 3, dst

Contoh:

```
enum bilangan{satu=1, dua, tiga, empat, lima}
```

4.3. Keluaran Enumerasi

- Keluaran enumerasi berupa angka yang diwakilinya

Contoh

```
enum bilangan{satu=1, dua, tiga, empat, lima};  
enum kelas{I=1, II, III, IV, V, VI};  
cout<<tiga; //akan tercetak 3 di layar  
Cout<<IV; //akan tercetak 4 di layar
```



5.

Penugasan

5.1 Penugasan

- Penugasan / Statement merupakan pemberian nilai pada variabel
- Penugasan dapat dilakukan langsung dengan nilai yang akan diberikan, dapat juga dengan menuliskan rumus ataupun fungsi mengikuti variable yang akan diisi

5.1 Penugasan (lanj...)

- Contoh Penugasan langsung:
a = 10;
cout<<a; //akan tercetak 10
- Contoh penugasan dengan rumus
a = 10;
b = a + 5;
cout<<b; //akan tercetak 15

5.2. Penugasan Langsung

- Contoh

a = 10;

Cout<<a; //pada layar akan tercetak 10

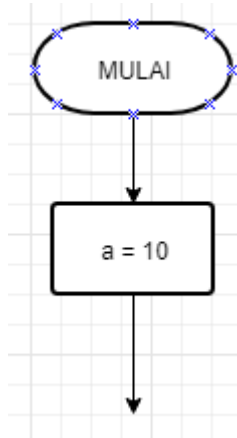
5.3. Penugasan Berupa Rumus

- Contoh:
a = 10; //penugasan langsung
b = a + 5; //penugasan dengan rumus
cout<<b; //akan tercetak 15

5.4. Notasi Penugasan

- Pada flowchart dapat menggunakan bangun persegi panjang berisi rumus.

- Contoh:



- Pada pseudo-code dapat berupa rumus pada baris program
- Contoh:
a = 10;
- Pada kalimat deskriptif dapat dituliskan dengan pembacaan rumusnya:
- Contoh:
[step 11] isi variable a dengan 10

Ringkasan

- Variabel merupakan penyimpanan data
- variabel harus dideklarasikan sesuai dengan jenis datanya
- Inisialisasi adalah pengisian nilai awal variabel
- Enumerasi adalah sekumpulan konstanta bilangan bulat / integer yang masing-masing konstanta akan memiliki nama dan nilai yang berbeda.
- Penugasan / Statement merupakan pemberian nilai pada variabel



Terimakasih

TUHAN Memberkati Anda

Teady Matius Surya Mulyana (tmulyana@bundamulia.ac.id)