



DASAR PEMROGRAMAN

- **Identifier**
- **Tipe Data**

IDENTIFIER (pengenal)

- Nama tipe (di bagian deklarasi Type)
- Tempat penyimpanan suatu data :
 - **variable** jika isinya dapat berubah dalam kisaran tertentu
 - **konstanta** jika isinya selalu tetap.
 - **file**, penyimpanan data di *storage*, dan sifatnya menetap.
- Fungsi dan Prosedur
- Modul program
- Algoritma

Penamaan Identifier

- *Identifier* **HARUS dimulai/diawali** dengan huruf, underscore/ garis bawah (_) atau tanda dollar (\$).

Identifier dapat berupa satu atau beberapa karakter

- ✓ Huruf (A s/d Z, a s/d z)
- ✓ Digit (0 1 2 3 4 5 6 7 8 9)
- ✓ Garisbawah (_)

- Untuk selebihnya dapat menggunakan karakter apapun, **kecuali karakter yang digunakan sebagai operator** oleh C++.

- *Identifier* adalah *case sensitive* (membedakan huruf besar/ kecil)

- Variabel yang didefinisikan dengan awalan * (tanda bintang) merupakan **pointer** (bukan berisi data, melainkan alamat dari suatu data)

Contoh variabel pointer : int *pint;

- Identifier haruslah **bukan merupakan keywords** yang dikenal C++

Valid Identifier

- Karakter-karakter di dalam identifier **tidak boleh dipisah dengan spasi**. Hal ini mengingat bahwa nama adalah kesatuan leksikal, maka setiap nama harus ditulis secara utuh.
- Anda dapat mengganti Spasi dengan karakter “_”. Namun, sebagai pengganti bentuk penulisan tradisional untuk identifier, yang menggunakan huruf kecil dengan garis-bawah sebagai pemisah kata
 - nama_karyawan
 - Nama_Karyawan, beberapa identifier dituliskan dengan huruf kapital pada awal kata, seperti :
 - namaKaryawan atau NamaKaryawan

■ Valid

- No
- Kd_barang
- kodeBarang
- \$barang
- bahASA_PEmrogramaN
- jmlBarang

■ Invalid

- 1_harga - tidak boleh diawali dengan angka
- %barang - simbol % tidak boleh digunakan
- Nama pelanggan - tidak boleh menggunakan spasi
- Kd+barang - tidak boleh menggunakan tanda operator

Reserved Words/Keywords

- Kata-kata yang dikenal oleh C++ dan mempunyai arti khusus dalam program.
- *Keywords* tidak boleh digunakan sebagai identifier (nama variabel, kelas, method, dll).
- Contoh keywords dalam C++ :

```

auto      const      double  float  int      short   struct   unsigned
break     continue  else    for    long    signed  switch   void
case      default    enum    goto   register sizeof  typedef  volatile
char      do          extern  if     return  static  union    while

asm                dynamic_cast  namespace  reinterpret_cast  try
bool               explicit      new         static_cast       typeid
catch              false         operator    template          typename
class              friend        private     this               using
const_cast         inline        public       throw              virtual
delete             mutable      protected   true               wchar_t
  
```

Case Sensitive

- Huruf kecil dan huruf kapital pada identifier tidak dianggap sama
- Identifier alamat, Alamat, dan ALAMAT menyatakan tiga identifier yang berbeda.
- Contoh :

```
#include <iostream>
#include <conio.h>
using namespace std;

void main ()
{
    char nama[4] = "Evi";
    char Nama[4] = "Eva";
    char NAMA[4] = "Evo";

    cout<<"String yang dicetak : "<<Nama;
    _getch();
}
```

Variabel

- → lokasi penyimpanan yang ada di memori
- Setiap variabel memiliki kemampuan menyimpan suatu informasi sesuai dengan tipe data yang dideklarasikan untuk variabel tersebut.
 - tipe-data nama-variabel;
 - tipe-data nama-variabel = <pengisian nilai variabel>;

Contoh :

```
■ int counter;
```

```
■ int counter = 3;
```

```
■ int panjang, lebar, tinggi;
```

```
■ int panjang = 10, lebar = 5, tinggi = 1;
```

```
■ int panjang, lebar, tinggi = 1; // hanya var tinggi yg diberikan nilai
```


Type data dasar pada C++

Type data	Ukuran memori	Jangkauan nilai	Jumlah digit presisi
char	1 byte	-128 s/d +127	
short int	2 byte		
int	2 byte	-32768 s/d +32767	
long	4 byte	-2.147.438.648 s/d 2.147.438.647	
float	4 byte	3.4×10^{-38} s/d 3.4×10^{38}	6-7
double	8 byte	1.7×10^{-308} s/d 1.7×10^{308}	15-16
long double	10 byte	3.4×10^{-4932} s/d 1.1×10^{4932}	19

Type data dasar pada C++

Type	Length	Range	
unsigned char	8 bits(1 byte)	0	to 255
char	8 bits	-128	to 127
enum	16 bits	-32,768	to 32,767
unsigned int	16 bits	0	to 65,535
short int	16 bits	-32,768	to 32,767
int	16 bits	-32,768	to 32,767
unsigned long	32 bits	0	to 4,294,967,295
long	32 bits	-2,147,483,648	to 2,147,483,647
float	32 bits	3.4×10^{-38}	to $3.4 \times 10^{+38}$
double	64 bits	1.7×10^{-308}	to $1.7 \times 10^{+308}$
long double	80 bits	3.4×10^{-4932}	to $1.1 \times 10^{+4932}$

short, int, long int

- bilangan bulat
- tidak mengandung titik desimal
- deklarasi variabel bertipe int pada C++:

```
int var;
```

```
int monsterKilled;
```

```
int totalScore;
```

```
long hargaBarang;
```

float, double, long double

- deklarasi variabel pada C++:
`float var;`
`double var;`
`long double var;`
- Nilai bilangan pecahan dapat dinyatakan dalam berbagai bentuk :
`3.14159 // 3.14159`
`6.02e23 // 6.02 x 1023`
`1.6e-19 // 1.6 x 10-19`
`3.0 // 3.0`

boolean

- tidak ada di C++
- untuk tipe boolean, dibuat dengan tipe bentukan :

```
typedef enum {false=0,true=1} boolean;  
  
boolean var;
```

string

- String merupakan dari karakter-karakter. Salah satu cara adalah menggunakan tipe terstruktur array dengan tipe char :

```
char var[banyaknyakarakter];
```

(lih. Slide 17)

Signed and Unsigned

Signed → bilangan positif dan negatif
Unsigned → bilangan positif

- Contoh pendeklarasian :
 - Signed int jmlPoint;
 - Unsigned int umurKaryawan;
- Bila tidak didefinisikan, maka secara default akan bernilai signed
 - Int jml; // serupa dengan signed int jml;

Lebih lanjut tentang Char

- Char mendeklarasikan tipe data karakter.
- Karakter adalah sembarang huruf, angka, tanda baca tunggal.

- `char letter;` ←
- `char letter = 'A';`

bentuk pendeklarasian tersebut hanya bisa memuat satu karakter (+ karakter NULL)

Bila hanya satu karakter maka gunakan tanda petik satu ‘

- `char nama[10];`
- `char nama[10] = "Palembang";`

Lebih lanjut tentang Char

- Variabel string tidak lain adalah array bertipe char yang mengandung element berupa karakter NULL yang menyatakan akhir dari string.
- `char nama[10];` merupakan string dengan panjang maksimal 9 karakter, hal ini disebabkan karakter NULL akan ditambahkan pada akhir string.
- `char nama[10] = "Palembang";`

P	A	L	E	M	B	A	N	G	\0
---	---	---	---	---	---	---	---	---	----

↑
karakter ber-ASCII 0
sebagai pengakhir string

Karakter NULL merupakan ciri khas pengakhir suatu string pada C++.

String

- Cara 1 : menggunakan tipe terstruktur array dengan tipe char yaitu `char var[banyaknyakarakter];`
- Cara 2 :
 - Definisikan `#include<string>`
 - `string namaMahasiswa;`
 - `cout<<"Input string : ";`
 - `getline(cin, namaMahasiswa);`
 - `cout<<namaMahasiswa;`

Menentukan Tipe Variabel

- Jika variabel digunakan untuk menyimpan data **bilangan bulat** saja, maka pilihannya adalah tipe bilangan bulat (seperti **int**, **long**).
- Jika variabel hendak dipakai untuk menampung data bilangan pecahan, maka variabel harus didefinisikan bertipe bilangan pecahan (seperti float, double)

Menentukan Tipe Variabel

Long atau int?

- Penentuannya didasarkan oleh kemungkinan besarnya nilai yang akan ditampung variabel.
- Misalnya bila dipastikan nilai yang akan ditampung tidak akan melebihi 32,767, tetapi juga tidak kurang dari -32,768 maka pilihannya adalah tipe int

Menentukan Tipe Variabel

- Pemrosesan aritmatika yang menggunakan tipe seperti int akan **lebih cepat** dibandingkan jika menggunakan tipe long.
- Jika suatu variabel diisi dengan nilai diluar jangkauannya, maka nilai yang tersimpan akan diubah sesuai jangkauannya. Misalnya variabel bertipe int diberikan nilai 75000, maka yang akan tersimpan adalah 9464. sebab nilai positif terbesar pada tipe int adalah 32767. hal ini bekerja sebagaimana speedometer, apabila nilai maksimumnya terlampaui maka akan dimulai dari nilai terendahnya lagi.

Berikan keterangan apakah identifier berikut ini valid atau tidak valid :

- avanZA
- from-to
- \$camry
- v10s
- _avaLoN
- Roger7
- Good/bad
- 4ever
- NiCKnAMe%
- A valid one
- Valid?
- (yes)
- pemrogramanBerorientasiObjek
- netto|bruto
- NAMA
- PT-1
- #brg
- Nama barang
- 2semester