

[Home](#) / [Tutorial](#) / [C++](#) / [Tipe Data](#)

# Tipe Data

Posted on Januari 22, 2018 | Last Modified Januari 22, 2018

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

int main() {
    dataType variabel1, variabel2;
    cout<<variabel1<<variabel2;
    return 0;
}
```

Fundamental

## Data Type



Tipe data adalah klasifikasi variabel untuk menentukan bentuk keadaan data yang akan disimpan dalam memori dan jenis operasi apa yang dapat dilakukan oleh variabel tersebut. variabel memiliki 3 jenis tipe data yaitu Variabel Numerik, Variabel Text dan Boolean.

Variabel Numerik : Memungkinkan kita untuk menyimpan data dalam bentuk bilangan. Variabel numerik dapat dibagi menjadi beberapa bagian yaitu : Bilangan Bulat (Integer “int”), Bilangan Desimal Berpresisi Tunggal (Floating Point “float”). Bilangan Desimal Berpresisi Ganda (Double Precision “double”).

Variabel Text : memungkinkan kita untuk menyimpan data dalam bentuk Karakter. Bahasa Pemrograman C++ menyediakan beberapa tipe data dasar seperti Char (Character / Karakter Tunggal) dan String (Kumpulan Karakter dan Kata).

Boolean : pada Bahasa Pemrograman C++ ditulis dengan menuliskan “bool” merupakan jenis yang hanya dapat mewakili satu dari dua pilihan yaitu 1 (True) atau 0 (False).

Tipe data yang umum digunakan :

### Tipe Data

### Bentuk Penulisan

Tipe Data	Bentuk Penulisan
Boolean	bool
Character	char
Integer	int
Floating Point	float
Double Floating Point	double
String	string
Valueless	void

Dari tipe data di atas bisa kita modifikasi menggunakan Modifiers Type untuk memberikan fungsi lebih kepada Variabel.

Modifiers Type	Fungsi
signed	Untuk membuat Tipe Data Numerik menjadi Negatif
unsigned	Untuk membuat Tipe Data Numerik menjadi Positif
short	Pemesanan Jarak dan Ukuran memori lebih Kecil
long	Pemesanan Jarak dan Ukuran memori lebih Besar

Setiap tipe data memiliki besar memori yang berbeda dengan yang lain. Ketika kita sedang membuat variabel, kita sebenarnya sedang memesan memori untuk penyimpanan kepada komputer sebesar tipe data yang digunakan. Di bawah ini adalah tabel keterangan detail mengenai masing-masing Tipe Data termasuk Tipe Data yang menggunakan Tipe Modifikasi. keterangan cara penulisan dan mengenai seberapa besar memori yang akan digunakan untuk penyimpanan.

## Daftar Lengkap Tipe data C++

Tipe Data	Bentuk Penulisan	Ukuran Memori
Integer	int	4 byte = 32 bit
	signed int	4 byte = 32 bit
	signed short int	2 byte = 16 bit

Tipe Data	Bentuk Penulisan	Ukuran Memori
	signed long int	4 byte = 32 bit
	signed long long int	4 byte = 32 bit
	unsigned int	4 byte = 32 bit
	unsigned short int	2 byte = 16 bit
	unsigned long int	4 byte = 32 bit
	unsigned long long int	4 byte = 32 bit
Floating Point	float	4 byte = 32 bit
Double Floating Point	double	8 byte = 64 bit
	long double	12 byte = 96 bit
Boolean	bool	1 byte = 8 bit
Character	char	1 byte = 8 bit
(C++ 11)	char16_t	2 byte = 16 bit
(C++ 11)	char32_t	4 byte = 32 bit
	wchar_t	2 byte = 16 bit
	unsigned char	1 byte = 8 bit
	signed char	1 byte = 8 bit
String	string	4 byte = 32 bit
Pointer Null (C++ 11)	decltype(nullptr)	4 byte = 32 bit
Valueless	void	1 byte = 8 bit

Tetapi Untuk ukuran memori mungkin bisa berbeda-beda pada setiap komputer yang anda gunakan. Untuk melihat pasti ukuran memori setiap tipe data pada komputer anda, Anda sbisa menggunakan fungsi “sizeof()” untuk memeriksanya besar memori.

Fungsi “sizeof()” biasanya digunakan untuk mengetahui besar memori dari tipe data, variabel, object dan semua deklarasi yang dapat memakan memori pada komputer.

Contoh Program :

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

int main()
{
    cout << "--> Integer "<< endl;
    cout << "Ukuran Memori dari int : " << sizeof(int) << endl;
    cout << "Ukuran Memori dari signed int : " << sizeof(signed
int) << endl;
    cout << "Ukuran Memori dari signed int : " << sizeof(signed
short int) << endl;
    cout << "Ukuran Memori dari signed int : " << sizeof(signed
long int) << endl;
    cout << "Ukuran Memori dari signed int : " << sizeof(signed
long long int) << endl;
    cout << "Ukuran Memori dari unsigned int : " <<
sizeof(unsigned int) << endl;
    cout << "Ukuran Memori dari unsigned int : " <<
sizeof(unsigned short int) << endl;
    cout << "Ukuran Memori dari unsigned int : " <<
sizeof(unsigned long int) << endl;
    cout << "Ukuran Memori dari unsigned int : " <<
sizeof(unsigned long long int) << endl;

    cout <<endl<< "--> Floating Point "<< endl;
    cout << "Ukuran Memori dari float : " << sizeof(float) <<
endl;

    cout <<endl<< "--> Double Floating Point "<< endl;
    cout << "Ukuran Memori dari Double : " << sizeof(double) <<
endl;
    cout << "Ukuran Memori dari Long Double : " << sizeof(long
double) << endl;

    cout <<endl<< "--> Boolean "<< endl;
    cout << "Ukuran Memori dari Boolean : " << sizeof(bool) <<
endl;

    cout <<endl<< "--> Character "<< endl;
```

```
    cout << "Ukuran Memori dari Char : " << sizeof(char) <<
endl;
    cout << "Ukuran Memori dari Singned Char : " <<
sizeof(signed char) << endl;
    cout << "Ukuran Memori dari Unsigned Char : " <<
sizeof(unsigned char) << endl;
    cout << "Ukuran Memori dari char16_t : " << sizeof(char16_t)
<< endl;
    cout << "Ukuran Memori dari char32_t : " << sizeof(char32_t)
<< endl;
    cout << "Ukuran Memori dari wchar_t : " << sizeof(wchar_t)
<< endl;

    cout <<endl<< "--> String "<< endl;
    cout << "Ukuran Memori dari String: " << sizeof(string) <<
endl;

    cout <<endl<< "--> Pointer Null "<< endl;
    cout << "Ukuran Memori dari decltype(nullptr): " <<
sizeof(decltype(nullptr)) << endl;

    cout <<endl<< "--> Valueless "<< endl;
    cout << "Ukuran Memori dari void: " << sizeof(void) << endl;

    return 0;
}
```

## Fungsi dan Cara penggunaan macam-macam dasar Tipe Data

### Boolean (bool)

Boolean adalah salah satu tipe data yang hanya memiliki dua pilihan yaitu True (1) atau False (0). Tipe data ini biasanya digunakan untuk memberikan kepastian dan memastikan kebenaran dari sebuah operasi atau kondisi program.

### Bentuk Penulisan :

```
bool daftar_variabel = inisialisasi;
```

#### Contoh Deklarasi :

```
bool pilihan=0; // Contoh mendirikan variable Boolean dengan
inisialisasi 0 = false.
```

#### Contoh Program :

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

int main()
{
    int angka; //Tipe data Integer untuk menyimpan data bersifat
    bilangan bulat
    bool hasil = true; //Tipe data Boolean yang mempunyai nilai
    awal true ( 1 )

    cout<<"Masukan angka = ";cin>>angka; //Input Bilangan bulat ke
    variabel "angka"

    hasil = angka > 10;
    //variabel "hasil" akan menyimpan nilai kondisi dari variabel
    "angka" > 10.
    cout<<hasil; //jika kondisi True maka akan tertulis angka 1
    atau 0 jika kondisi False.
    return 0;
}
```

#### **Character (char)**

Char Adalah salah satu tipe data sembarang huruf, angka, dan simbol. Yang memungkinkan kita untuk memesan memori berformat text dengan karakter tunggal seperti 'a', '\$' dan lain-lain.

#### Bentuk Penulisan :

```
char nama_variabel[array] = inisialisasi ;
```

untuk array di dalam sepasang tanda “[“ dan “]”, anda dapat tidak untuk menggunakannya. Tipe data char hanya dapat menyimpan data berbentuk karakter dan hanya satu karakter. Bagian [array] digunakan untuk menentukan deret jumlah memori yang akan kita pesan, yang akan memungkinkan kita dapat menyimpan data dalam bentuk rangkaian huruf atau rangkaian kata (lebih dari satu karakter), satu karakter biasanya akan menempati satu memori yang kita pesan dan dalam peraturan pemesanan variabel char berderet harus mempunyai satu memori tambahan dari jumlah karakter. Hal ini akan penulis jelaskan lebih dalam di artikel selanjutnya.

### Contoh Penulisan

```
char grade = 'a'; //Variabel char tanpa array  
char nama[15] = "belajarcpp.com" //Variabel char dengan array  
memesan 15 memori bertipe char secara berderet (satu karakter  
akan menempati satu memori).
```

### Contoh Program

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

int main(){
    char nama[15] = "Belajarcpp.com";//Variabel Char + Array =
    memesan memori sebanyak 15 sebesar tipe data char.
    char kelas='A'; //Variabel Char tanpa array dan hanya dapat
    memuat satu karakter.

    cout<<"Masukan nama = ";cin>>nama;//jika anda memasukan
    karakter lebih dari 14 kemungkinan program akan berhenti
    (error).
    cout<<"Masukan kelas = ";cin>>kelas; //Jika anda memasukan
    karakter lebih dari satu maka variabel "kelas" hanya akan
    mengambil satu karakter pertama.
    cout<<endl<<"Yang anda masukan adalah"
    <<endl<<nama<<endl<<kelas;

    return 0;
}
```

### Integer (int)

Integer adalah salah satu tipe data numerik yang memungkinkan kita untuk menyimpan data dalam bentuk bilangan bulat.

#### Bentuk penulisan :

```
int daftar_variabel = inisialisasi;
```

#### Contoh Deklarasi :

```
int bilangan = 10;
```

#### Contoh Program :



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

int main(){
    int bilangan = 0, hasil=0;

    cout<<"Masukan angka = ";cin>>bilangan;
    hasil = bilangan * 2;
    cout<<"Hasil dari "<<bilangan<<" X 2 adalah "<<hasil;

    return 0;
}
```

### Float Point (float)

Float adalah salah satu tipe data Numerik yang memungkinkan untuk menyimpan nilai dalam memori bersifat bilangan pecahan atau real, maupun eksponensial.

#### Bentuk Penulisan

```
float daftar_variabel = inisialisasi;
```

#### Contoh Deklarasi

```
float p=3,14;
```

#### Contoh Program

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

int main(){
    float jari, hasil ;
    const float p=3.14; //variabel konstanta

    cout<<"Masukan Jumlah jari-jari = ";cin>>jari;
    hasil = (jari * p)*2; //disini kita menggunakan variabel float
    //karena hasil dari penghitungan lingkaran biasanya berbentuk
    bilangan pecahan
    cout<<"Keliling dari Lingkaran adalah "<<hasil;

    return 0;
}
```

### Double Floating Point (double)

Double sama seperti float adalah salah satu tipe data yang bersifat menyatakan bilangan pecahan atau real, maupun eksponensial. Bedanya adalah penyimpanan angka maksimal lebih besar daripada float otomatis double juga akan membutuhkan memori yang lebih besar.

#### Bentuk penulisan

```
double daftar_variabel = inisialisasi;
```

#### Contoh deklarasi

```
double angka = 3.14;
```

#### Contoh Program

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

int main(){
    double jari, hasil ;
    const double p=3.14;

    cout<<"Masukan Jumlah jari-jari = ";cin>>jari;
    hasil = jari*(jari * p);
    cout<<"Luas dari lingkaran 3.14 X "<<jari<<" X "<<jari<<"
    adalah "<<hasil;

    return 0;
}
```

### String (string)

Merupakan tipe data Text yang memungkinkan kita menyimpan nilai dengan bentuk text, kumpulan dari character. Berbeda dengan char array, Tipe data string kita tidak perlu untuk menentukan jumlah digit pada saat deklarasi variabel.

#### Bentuk penulisan

```
string daftar_variabel = inisialisasi;
```

#### Bentuk penulisan

```
string kalimat = "Selamat Belajar Di belajarcpp.com" ;
```

#### Contoh Program

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

int main(){
    string pesan = "Selamat Belajar Di belajarcpp.com", kalimat =
    "";

    cout<<pesan<<endl;
    getline(cin,kalimat); //Fungsi input agar dapat membaca
    kalimat panjang dan yang menggunakan spasi.
    cout<<kalimat<<endl;

    return 0;
}
```

### Valueless (void)

Void adalah salah satu tipe data yang berarti “tidak ada” atau “tidak mempunyai tipe data”. Void masih di dalam katagori tipe data tetapi kita tidak bisa menggunakannya pada variabel biasa, void biasanya digunakan pada function yang tidak mempunyai return value “void myFunction()”, pada function parameter yang tidak mempunyai parameter “void myfunction(void)” dan pada pointer “void \*myPointer;”.

f Share

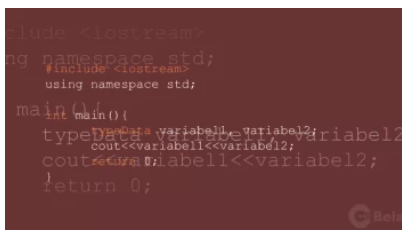
🐦 Tweet

G+ Share

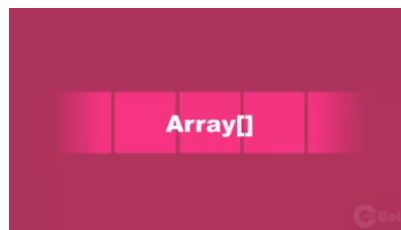
📌 Pin

in Share

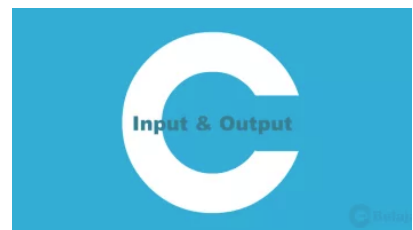
### Terkait



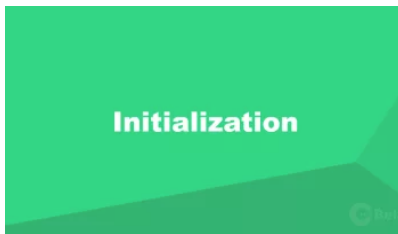
Variabel  
Januari 22, 2018  
dalam "C++"



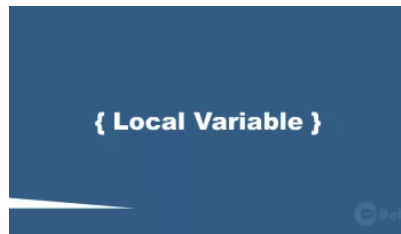
Array  
Januari 23, 2018  
dalam "C++"



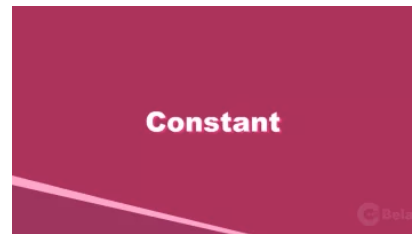
Dasar Input Output pada C  
Januari 22, 2018  
dalam "C"



Inisialisasi  
Januari 22, 2018  
dalam "C++"



Variabel Lokal  
Januari 22, 2018  
dalam "C++"



Konstanta  
Januari 22, 2018  
dalam "C++"