

## Kiểu dữ liệu trong C/C++

Kiểu dữ liệu trong C/C++ Trong khi làm việc với bất kỳ ngôn ngữ lập trình nào, bạn cần sử dụng các kiểu biến đa dạng để lưu giữ thông tin. Các biến, không gì khác ngoài các vị trí bộ nhớ được dành riêng để lưu giá trị. Nghĩa là, khi bạn tạo một biến, bạn dành riêng một số không gian trong bộ nhớ cho biến đó.

Bạn có thể thích lưu thông tin của các kiểu dữ liệu (Data Type) đa dạng như Character, Wide Character, integer, floating-point, double floating point, Boolean, .... Dựa trên kiểu dữ liệu của một biến, hệ thống sẽ cấp phát bộ nhớ và quyết định những gì có thể được lưu giữ trong bộ nhớ dành riêng đó.

### Kiểu dữ liệu nguyên thủy trong C/C++

Tên tiếng Anh là Primitive Type, còn có thể gọi là kiểu dữ liệu gốc, kiểu dữ liệu có sẵn trong C/C++. Bên cạnh các kiểu dữ liệu gốc này, C/C++ cũng cung cấp các kiểu dữ liệu user-defined. Bảng dưới đây liệt kê 7 kiểu dữ liệu cơ bản trong C/C++:

Kiểu dữ liệu	Từ khóa
Boolean	bool
Ký tự	char
Số nguyên	int
Số thực	float
Số thực dạng Double	double
Kiểu không có giá trị	void

Kiểu Wide character	wchar_t
---------------------	---------

Một số kiểu cơ bản có thể được sửa đổi bởi sử dụng một hoặc nhiều modifier này:

- signed (kiểu có dấu)
- ADVERTISING
- unsigned (kiểu không có dấu)
- short
- long

Bảng sau hiển thị kiểu biến, lượng bộ nhớ nó dùng để lưu giá trị trong bộ nhớ, và giá trị lớn nhất và nhỏ nhất có thể được lưu giữ với các kiểu biến đó:

Kiểu	Độ rộng bit	Dãy giá trị
char	1 byte	-127 tới 127 hoặc 0 tới 255
unsigned char	1 byte	0 tới 255
signed char	1 byte	-127 tới 127
int	4 byte	-2147483648 tới 2147483647
unsigned int	4 byte	0 tới 4294967295
signed int	4 byte	-2147483648 tới 2147483647
short int	2 byte	-32768 tới 32767
unsigned short int	Range	0 tới 65,535

signed short int	Range	-32768 tới 32767
long int	4 byte	-2,147,483,647 tới 2,147,483,647
signed long int	4 byte	Tương tự như long int
unsigned long int	4 byte	0 tới 4,294,967,295
float	4 byte	+/- 3.4e +/- 38 (~7 chữ số)
double	8 byte	+/- 1.7e +/- 308 (~15 chữ số)
long double	8 byte	+/- 1.7e +/- 308 (~15 chữ số)
wchar_t	2 hoặc 4 byte	1 wide character

Kích cỡ của các biến có thể khác với những gì hiển thị trên bảng, phụ thuộc vào compiler và máy tính bạn đang sử dụng.

Dưới đây là ví dụ sẽ đưa ra kích cỡ chính xác của các kiểu dữ liệu đa dạng trên máy tính của bạn.

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Kích co cua char la: " << sizeof(char) << endl;

    cout << "Kích co cua int la: " << sizeof(int) << endl;

    cout << "Kích co cua short int la: " << sizeof(short int) << endl;

    cout << "Kích co cua long int la: " << sizeof(long int) << endl;

    cout << "Kích co cua float la: " << sizeof(float) << endl;
```

```
cout << "Kích co cua double la: " << sizeof(double) << endl;

cout << "Kích co cua wchar_t la: " << sizeof(wchar_t) << endl;

return 0;

}
```

Ví dụ này sử dụng endl, mà chèn một ký tự newline (dòng mới) sau mỗi dòng, và toán tử << được sử dụng để truyền nhiều giá trị tới màn hình. Chúng tôi cũng sử dụng toán tử sizeof() để lấy kích cỡ của các kiểu dữ liệu đã dạng.

Khi code trên được biên dịch và thực thi, nó cho kết quả sau (kết quả có thể đa dạng tùy thuộc vào compiler và máy tính bạn đang sử dụng).

```
Kích co cua char la: 1
Kích co cua int la: 4
Kích co cua short int la: 2
Kích co cua long int la: 4
Kích co cua float la: 4
Kích co cua double la: 8
Kích co cua wchar_t la: 4
```

## Khai báo typedef trong C/C++

Bạn có thể tạo một tên mới cho một kiểu dữ liệu đang tồn tại bởi sử dụng typedef trong C/C++. Cú pháp đơn giản sau để định nghĩa một kiểu dữ liệu mới bởi sử dụng typedef:

```
typedef kieu_du_lieu ten_moi;
```

Ví dụ sau nói cho compiler rằng sothuc là tên khác của float:

```
typedef float sothuc;
```

Bây giờ, khai báo sau là hoàn toàn hợp lệ và sẽ tạo một biến số thực gọi là *vantoc*:

```
sothuc vantoc;
```

## Kiểu liệt kê enum trong C/C++

Kiểu liệt kê enum khai báo một tên kiểu tùy ý và một tập hợp của 0 hoặc nhiều Identifier (Định danh) mà có thể được sử dụng như là các giá trị của kiểu đó. Mỗi Enumerator là một constant có kiểu là kiểu liệt kê (enumeration).

Để tạo một Enumeration, bạn sử dụng từ khóa enum trong C/C++. Form chung của kiểu liệt kê enum là:

```
enum ten_cua_enum { danh_sach_cac_ten } danh_sach_bien;
```

Tại đây, *ten\_cua\_enum* là tên kiểu liệt kê. Danh sách tên được phân biệt bởi dấu phẩy.

Ví dụ, code sau định nghĩa một tên kiểu liệt kê hàng hóa gọi là *hanghoa* và biến *c* là kiểu của *hanghoa*. Cuối cùng, *c* được gán giá trị *nuocngot*.

```
enum hanghoa { sua, nuocngot, biachai } c;
```

```
c = nuocngot;
```

Theo mặc định, trong danh sách các tên thì giá trị của tên đầu tiên là 0, tên thứ hai là 1 và tên thứ 3 là 2, .... Nhưng bạn có thể cung cấp cho một tên một giá trị cụ thể bằng việc thêm một Initializer (giá trị khởi tạo). Ví dụ, trong enumeration sau, *nuocngot* sẽ có giá trị là 40:

```
enum hanghoa { sua, nuocngot=40, biachai };
```

Ở đây, *biachai* sẽ có giá trị là 41 bởi vì mỗi tên sẽ có giá trị lớn hơn của tên trước đó là 1.

