

http://

Tuhoc.cc



LẬP TRÌNH



Dễ hiểu



LESSON

5

Ép kiểu dữ liệu C++

double c = (double) b



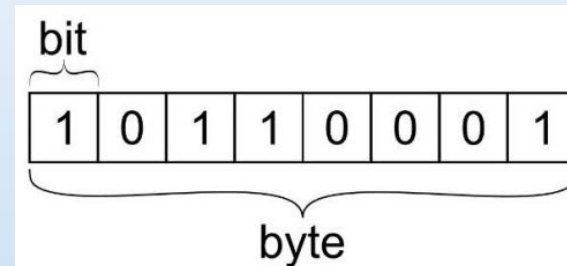
@galailaptrinh



0

Các kiểu dữ liệu cơ sở

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)



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

int main()
{
    //0. ktra giới hạn kiểu dữ liệu
    cout << "max kieu int la: " << INT_MIN << std::endl;
    cout << "min kieu int la: " << INT_MAX << std::endl;

    cout << "max kieu short la: " << SHRT_MIN << std::endl;
    cout << "min kieu short la: " << SHRT_MAX << std::endl;
}
```

```
max kieu int la: -2147483648
min kieu int la: 2147483647
max kieu short la: -32768
min kieu short la: 32767
```

✓ Bool : Kiểu luận lý true, false



1

Tại sao phải ép kiểu

- ✓ *Như bài học trước, chúng ta đã biết các kiểu dữ liệu sinh ra , để phù hợp với mục đích sử dụng, tiết kiệm bộ nhớ hệ thống.*
- ✓ *Trong quá trình tính toán, có thể kiểu dữ liệu trả về không còn giống với kiểu ban đầu chúng ta khai báo => Cần ép kiểu để sử dụng kiểu phù hợp*



2

Các loại ép kiểu C++

□ Có 2 dạng ép kiểu



Ép kiểu rộng :

ép kiểu từ dữ liệu *bé* > *lớn*

Ex: *Int* → *Long* → *Float* → *Double*

Không lo mất dữ liệu



Ép kiểu hẹp :

ép kiểu từ dữ liệu *lớn* > *bé*

Ex: *double* → *float* → *long* → *int*

Có thể mất dữ liệu



3

Ép kiểu rộng



Ép kiểu rộng :

ép kiểu từ dữ liệu bé > lớn

Ex: *Int* → *Long* → *Float* → *Double*

Không lo mất dữ liệu

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

int main()
{
    //check vùng kiểu short, int
    cout << "min kieu int la: " << INT_MIN << std::endl;
    cout << "max kieu int la: " << INT_MAX << std::endl;

    cout << "min kieu short la: " << SHRT_MIN << std::endl;
    cout << "max kieu short la: " << SHRT_MAX << std::endl;

    //1. ép kiểu rộng
    short a = 10; //max short là 32767
    //kiểm tra kiểu dữ liệu của a
    cout << "kieu du lieu cua a: " << typeid(a).name() << endl;
    //xuất giá trị a
    cout << "a= " << a << endl;

    //khai báo biến b, ép kiểu a sang int
    int b = (int)a;
    //int b = a // C++ cho phép ép tắt
    //kiểm tra kiểu dữ liệu của b
    cout << "kieu du lieu cua b: " << typeid(b).name() << endl;
    //xuất giá trị b
    cout << "b= " << b << endl;
}
```



```
min kieu int la: -2147483648
max kieu int la: 2147483647
min kieu short la: -32768
max kieu short la: 32767
kieu du lieu cua a: short
a= 10
kieu du lieu cua b: int
b= 10
```



4

Ép kiểu hẹp



Ép kiểu hẹp :

ép kiểu từ dữ liệu lớn > bé

Ex: double → float → long → int

Có thể mất dữ liệu

```
//2. ép kiểu hẹp
float soA = 1.25;
int soB = 1.25;
cout << "SoA= " << soA << endl;
cout << "SoB= " << soB << endl;
```



```
SoA= 1.25
SoB= 1
```



5

Code trong bài giảng

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

int main()
{
    //1. ép kiểu rộng
    int a = 10;
    //kiểm tra kiểu dữ liệu của a
    cout << "kiểu dữ liệu của a: " << typeid(a).name() << endl;
    //xuất giá trị a
    cout << "a= " << a << endl;

    //khai báo biến b, ép kiểu a sang int
    float b = (float)a; // ép tường minh
    //float b = a; // C++ cho phép ép tắt

    //kiểm tra kiểu dữ liệu của b
    cout << "kiểu dữ liệu của b: " << typeid(b).name() << endl;
    //xuất giá trị b
    cout << "b= " << b << endl;

    //2. ép kiểu hẹp
    float soA = 1.25;
    int soB = 1.25;
    cout << "SoA= " << soA << endl;
    cout << "SoB= " << soB << endl;
}
```

Tài liệu bài giảng, nhóm thảo luận

<http://dc.tuhoc.cc>

