

Phạm vi biến trong C++

Một scope (phạm vi) là một khu vực của chương trình nơi biến hoạt động, và nói chung có thể có 3 khu vực mà biến có thể được khai báo:

- Bên trong một hàm hoặc một khối, được gọi là biến cục bộ (local).
- Trong định nghĩa của các tham số hàm, được gọi là các tham số chính thức (formal).
- Bên ngoài của tất cả hàm, được gọi là biến toàn cục (global).

Chúng ta sẽ học hàm và các tham số của hàm là gì trong chương tới. Dưới đây chúng tôi sẽ giải thích khái niệm về biến cục bộ và biến toàn cục.

Biến cục bộ trong C++

Các biến được khai báo bên trong một hàm hoặc khối là các biến cục bộ (local). Chúng chỉ có thể được sử dụng bởi các lệnh bên trong hàm hoặc khối code đó. Các biến cục bộ không được biết ở bên ngoài hàm đó (tức là chỉ được sử dụng bên trong hàm hoặc khối code đó). Dưới đây là ví dụ sử dụng các biến cục bộ:

```
#include <iostream>

using namespace std;

int main ()

{

    // phan khai bao bien cuc bo:

    int a, b;
```

```
int c;  
  
// phan khoi tao bien  
a = 10;  
b = 20;  
c = a + b;  
  
cout << c;  
  
return 0;  
}
```

Biến toàn cục trong C++

Biến toàn cục (global) trong C++ được định nghĩa bên ngoài các hàm, thường ở phần đầu chương trình. Các biến toàn cục giữ giá trị của nó trong suốt vòng đời chương trình của bạn.

Một biến toàn cục có thể được truy cập bởi bất kỳ hàm nào. Tức là, một biến toàn cục là có sẵn cho bạn sử dụng trong toàn bộ chương trình sau khi đã khai báo nó. Dưới đây là ví dụ sử dụng biến toàn cục và biến nội bộ trong C++:

```
#include <iostream>  
  
using namespace std;  
  
// phan khai bao bien toan cuc:  
int g;  
  
int main ()
```

```

{

    // phan khai bao bien cuc bo:

    int a, b;


    // phan khoi tao bien

    a = 10;

    b = 20;

    g = a + b;


    cout << g;


    return 0;

}

```

Một chương trình có thể có các biến toàn cục và biến cục bộ cùng tên với nhau, nhưng trong một hàm thì giá trị của biến cục bộ sẽ được ưu tiên. Ví dụ:

```

#include <iostream>

using namespace std;


// phan khai bao bien toan cuc:

int g = 20;


int main ()

{

    // phan khai bao bien cuc bo:

    int g = 10;

```

```
cout << g;
```

```
return 0;
```

```
}
```

Khi code trên được biên dịch và thực thi, nó cho kết quả sau:

```
10
```

Khởi tạo biến cục bộ và biến toàn cục bởi hệ thống trong C++

Khi một biến cục bộ được định nghĩa, nó không được khởi tạo bởi hệ thống, chính bạn phải khởi tạo nó. Các biến toàn cục được khởi tạo tự động bởi hệ thống khi bạn định nghĩa chúng, như sau:

Kiểu dữ liệu	Giá trị khởi tạo
int	0
char	'\0'
float	0
double	0
pointer	NULL

Khởi tạo biến một cách chính xác là một sự thực hành tốt, nếu không, đôi khi chương trình sẽ cho kết quả không mong đợi.

