

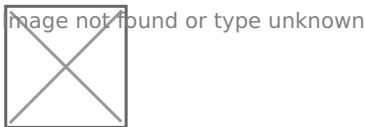
Hàm - Tham số giá trị và tham số biến trong C++

Cho đến nay, trong tất cả các hàm chúng ta đã biết, tất cả các tham số truyền cho hàm đều được truyền theo *giá trị*. Điều này có nghĩa là khi chúng ta gọi hàm với các tham số, những gì chúng ta truyền cho hàm là các *giá trị* chứ không phải bản thân các biến.

Ví dụ, giả sử chúng ta gọi hàm **addition** như sau:

```
int x=5, y=3, z;  
z = addition ( x , y );
```

Trong trường hợp này khi chúng ta gọi hàm **addition** thì các giá trị **5** and **3** được truyền cho hàm, không phải là bản thân các biến.



Đến đây các bạn có thể hỏi tôi: Như vậy thì sao, có ảnh hưởng gì đâu ? Điều đáng nói ở đây là khi các bạn thay đổi giá trị của các biến **a** hay **b** bên trong hàm thì các biến **x** và **y** vẫn không thay đổi vì chúng đâu có được truyền cho hàm chỉ có giá trị của chúng được truyền mà thôi.

Hãy xét trường hợp bạn cần thao tác với một biến ngoài ở bên trong một hàm. Vì vậy bạn sẽ phải truyền tham số dưới dạng tham số biến như ở trong hàm **duplicate** trong ví dụ dưới đây:

```
// passing parameters by reference  
#include <iostream.h>  
  
void duplicate (int& a, int& b, int& c)  
{  
    a*=2;  
    b*=2;  
    c*=2;  
}  
  
int main ()  
{
```

```

int x=1, y=3, z=7;
duplicate (x, y, z);
cout << "x=" << x << ", y=" << y << ", z=" << z;
return 0;
}

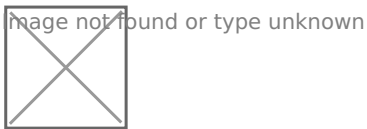
```

Kết quả:

```
x=2, y=6, z=14
```

Điều đầu tiên làm bạn chú ý là trong khai báo của `duplicate` theo sau tên kiểu của mỗi tham số đều là dấu và (&), để báo hiệu rằng các tham số này được truyền theo tham số biến chứ không phải tham số giá trị.

Khi truyền tham số dưới dạng tham số biến chúng ta đang truyền bản thân biến đó và bất kì sự thay đổi nào mà chúng ta thực hiện với tham số đó bên trong hàm sẽ ảnh hưởng trực tiếp đến biến đó.



Trong ví dụ trên, chúng ta đã liên kết **a**, **b** và **c** với các tham số khi gọi hàm (**x**, **y** và **z**) và mọi sự thay đổi với **a** bên trong hàm sẽ ảnh hưởng đến giá trị của **x** và hoàn toàn tương tự với **b** và **y**, **c** và **z**.

Kiểu khai báo tham số theo dạng tham số biến sử dụng *dấu và (&)* chỉ có trong C++. Trong ngôn ngữ C chúng ta phải sử dụng con trỏ để làm việc tương tự như thế.

Truyền tham số dưới dạng tham số biến cho phép một hàm trả về nhiều hơn một giá trị. Ví dụ, đây là một hàm trả về số liền trước và liền sau của tham số đầu tiên.

```

// more than one returning value
#include <iostream.h>

void prevnext (int x, int& prev, int& next)
{
    prev = x-1;
    next = x+1;
}

int main ()
{

```

```
int x=100, y, z;  
prevnext (x, y, z);  
cout << "Previous=" << y << ", Next=" << z;  
return 0;  
}
```

Kết quả:

```
Previous=99, Next=101
```

Revision #1

Created 5 October 2019 05:38:12 by Laptrinh.vn

Updated 5 October 2019 05:43:38 by Laptrinh.vn