

## Call the function by reference in C ++

The method of calling a function by reference of parameters passed to a function copies the address of a parameter into the official parameter. Within that function, the address used to access the parameter is actually used in calling the function. This means that changes are made with parameters that affect the passed parameters.

The method of **calling a function by reference** of parameters passed to a function copies the address of a parameter into the official parameter. Within that function, the address used to access the parameter is actually used in calling the function. This means that changes are made with parameters that affect the passed parameters.

To pass values ??by reference, parameter pointers are passed to functions like any other value. Therefore, you need to declare the function parameters as pointer types as in the **traodoi ()** function as follows, but change the value of the two integer variables pointed to by its parameters.

```
// phan dinh nghia ham de trao doi cac gia tri. void traodoi ( int & x , int &
```

Now, we call the function traodoi () by passing the values ??by reference as follows:

```
#include using namespace std ; // phan khai bao ham void traodoi ( int & x , i
```

You put the above function definition at the end of this code, then compile and run the above C ++ program will produce the following result:

```
Truoc khi trao doi, gia tri cua a la: 100
Truoc khi trao doi, gia tri cua b la: 200
Sau khi trao doi, gia tri cua a la: 200
Sau khi trao doi, gia tri cua b la: 100
-----
```

This indicates that the changes reflect outside the function, unlike the function by the value that the changes do not reflect outside the function.

### According to Tutorialspoint

Last lesson: Call the function by pointer in C ++

Next lesson: Number in C ++

You finished reading the article "**Call the function by reference in C ++**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.