Return by reference in C++ with Examples

Difficulty Level : Medium • Last Updated : 01 Aug, 2020

Pointers and References in C++ held close relation with one another. The major difference is that the pointers can be operated on like adding values whereas references are just an **alias** for another variable.

- Functions in C++ can return a **reference** as it's returns a **pointer**.
- When function returns a **reference** it means it returns a **implicit** pointer.

Return by reference is very different from Call by reference. Functions behaves a very important **role** when variable or pointers are returned as reference.

See this function signature of Return by Reference Below:

> ***dataType& functionName(parameters);***
> *where,*
> ***dataType** is the **return type** of the **function**,*
> *and **parameters** are the passed **arguments** to it.*

Below is the code to illustrate the Return by reference:

CPP

```cpp
// C++ program to illustrate return by reference
#include <iostream>
using namespace std;

// Function to return as return by reference
int& returnValue(int& x)
{

    // Print the address
    cout << "x = " << x
         << " The address of x is "
         << &x << endl;

    // Return reference
    return x;
}

// Driver Code
int main()
{
    int a = 20;
    int& b = returnValue(a);

    // Print a and its address
    cout << "a = " << a
         << " The address of a is "
         << &a << endl;

    // Print b and its address
    cout << "b = " << b
         << " The address of b is "
         << &b << endl;

    // We can also change the value of
    // 'a' by using the address returned
    // by returnValue function

    // Since the function returns an alias
    // of x, which is itself an alias of a,
    // we can update the value of a
    returnValue(a) = 13;

    // The above expression assigns the
    // value to the returned alias as 3.
    cout << "a = " << a
         << " The address of a is "
         << &a << endl;
    return 0;
}
```

Output:

```
x = 20 The address of x is 0x7fff3025711c
a = 20 The address of a is 0x7fff3025711c
b = 20 The address of b is 0x7fff3025711c
x = 20 The address of x is 0x7fff3025711c
a = 13 The address of a is 0x7fff3025711c
```

Explanation:

Since **reference** is nothing but an **alias**(synonym) of another variable, the address of **a**, **b** and **x** never changes.

> ***Note:** We should never return a **local variable** as a **reference**, reason being, as soon as the functions returns, local variable will be **erased**, however, we still will be left with a **reference** which might be a **security bug** in the code.*

Below is the code to illustrate the Return by reference:

C++

```cpp
// C++ program to illustrate return
// by reference
#include <iostream>
using namespace std;

// Global variable
int x;

// Function returns as a return
// by reference
int& retByRef()
{
    return x;
}

// Driver Code
int main()
{
    // Function Call for return
    // by reference
    retByRef() = 10;

    // Print X
    cout << x;
    return 0;
}
```

Output:

```
10
```

Article Contributed By :

ayush0x00
@ayush0x00

Vote for difficulty

Current difficulty : Medium

Easy | Normal | Medium | Hard | Expert

Article Tags : cpp-references, return, C++ Programs

Improve Article | Report Issue