

COMPUTER SCIENCE PROGRAMMING C++

Lesson 15: Pointer Reference

Nguyễn Văn Hiếu



Nội dung chính

- Tìm hiểu về tham chiếu là gì, sử dụng tính năng tham chiếu như thế nào ?
- Bộ nhớ lưu trữ các giá trị như thế nào ?
- Cách để chương trình có thể tìm tới nơi giá trị của biến được lưu trữ - khái niệm địa chỉ trong lập trình
- Làm quen với con trỏ, khái niệm, tác dụng, cách thức tạo ra con trỏ và nguyên lý ứng dụng con trỏ trong các bài toán cơ bản
- Một số lưu ý khi sử dụng con trỏ

TABLE OF CONTENTS

01

MAIN CONTENTS

Tìm hiểu về con trỏ, cách sử dụng

02

REQUIREMENTS

Nắm được sự khác biệt giữa tham chiếu và con trỏ

03

NOTES

Một số lưu ý cần nắm khi sử dụng con trỏ

04

LEARNING SKILLS

Ứng dụng con trỏ vào các ví dụ cơ bản



01

Tham chiếu

Tham chiếu là gì ?
Cách sử dụng tham chiếu



Tham chiếu

1. Tham chiếu là tạo ra một biến khác sử dụng chung vùng nhớ với biến được tham chiếu đến.

```
int i1 = 10;  
int &i_ref = i1;  
  
cout << &i1 << endl;  
cout << &i_ref << endl;
```

3. Có thể sử dụng tham chiếu để có thể tác động tới đối tượng ở bên ngoài

Kết quả chúng ta được:

```
0xBFEB475C  
0xBFEB475C
```

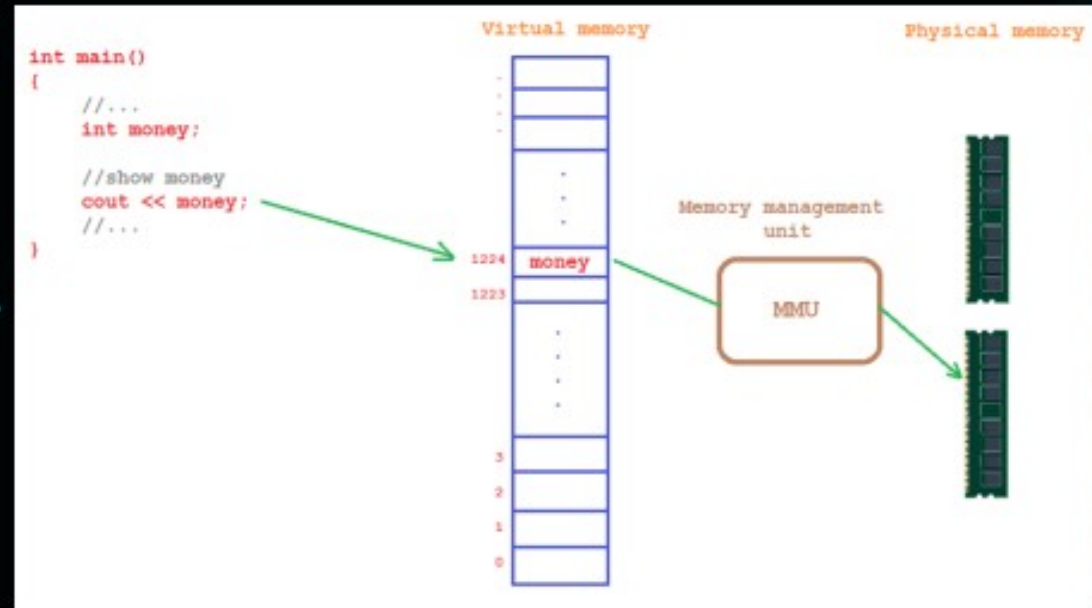
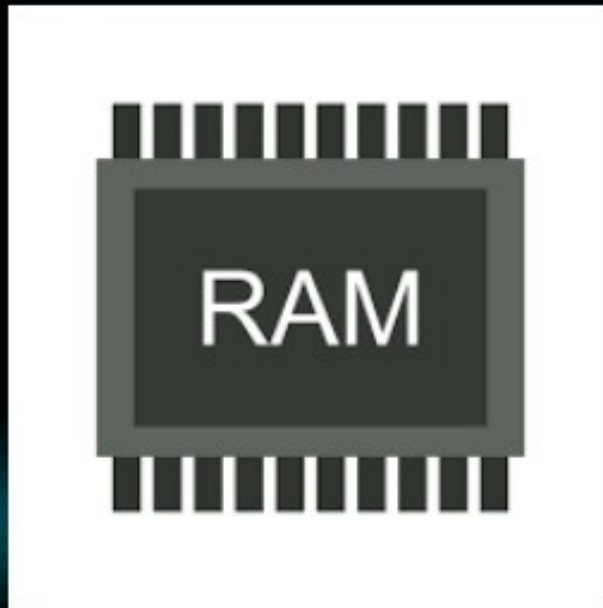
2. Mọi tác động đến tham chiếu sẽ làm thay đổi đến đối tượng được tham chiếu

4. Chỉ tham chiếu 1 lần, không thể thay đổi



Địa chỉ trong bộ nhớ

Tất cả mọi thứ trong chương trình đều được lưu trữ trên bộ nhớ



Mọi thứ được lưu trữ đều có địa chỉ, địa chỉ được đánh dấu bắt đầu từ 0 tương tự mảng



Địa chỉ trong bộ nhớ

Để biết được địa chỉ của một biến tại ô thứ bao nhiêu trong bộ nhớ, chúng ta sử dụng toán tử &

```
int n = 5;  
  
cout << n << endl;  
cout << &n << endl;  
cout << *(&n) << endl;
```



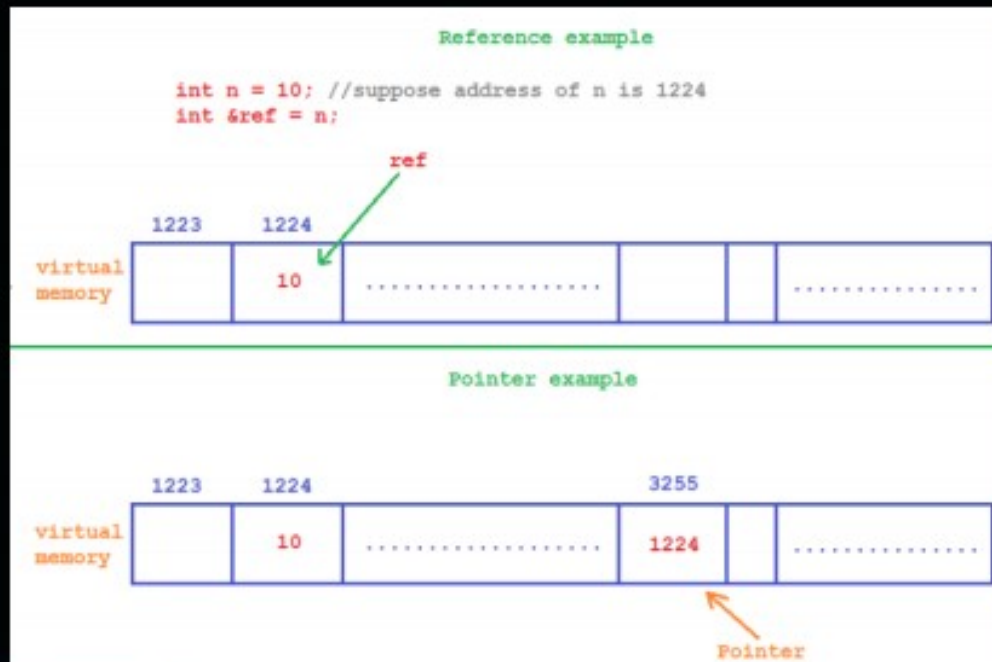
```
5  
0xBFDD181AC  
5
```

Thông qua địa chỉ, chúng ta có thể truy cập giá trị dựa vào toán tử *



Con trỏ

Con trỏ là biến dùng để lưu trữ địa chỉ của một vùng nhớ trong bộ nhớ ảo mà OS cấp phát cho chương trình của chúng ta sử dụng



Khác với tham chiếu, con trỏ là một biến có địa chỉ độc lập so với vùng nhớ mà nó trỏ đến, nhưng giá trị bên trong vùng nhớ của con trỏ chính là địa chỉ của biến (hoặc địa chỉ ảo) mà nó trỏ tới.



Khai báo, sử dụng con trỏ

Khai báo con trỏ

<datatype> name_variable;

VD : `int* p; double** pD;`

Dấu * trong việc khai báo là cú pháp định nghĩa một con trỏ

```
#include <iostream>

using namespace std;

int main()
{
    int n = 5;
    // Variable n is integer, value = 5
    int *a = &n;
    // Variable a is pointer to integer (int*), value = address of n in storage
    cout << *a; // Print value on address of a point
    return 0;
}
```

Cho con trỏ trỏ vào biến cần trỏ và sử dụng con trỏ để truy cập



Sử dụng con trỏ

Sử dụng hàm để hoán vị 2 giá trị

Thực hiện quản lý dữ liệu sử dụng con trỏ không cần thông qua biến

<code>ptr_a < ptr_b</code>	Trả về giá trị true nếu a được lưu trữ ở vị trí trước b
<code>ptr_a > ptr_b</code>	Trả về giá trị true nếu a được lưu trữ ở vị trí sau b
<code>ptr_a <= ptr_b</code>	Trả về giá trị true nếu a được lưu trữ ở vị trí trước b hoặc ptr_a và ptr_b trỏ đến cùng một vị trí
<code>ptr_a >= ptr_b</code>	Trả về giá trị true nếu a được lưu trữ ở vị trí sau b hoặc ptr_a và ptr_b trỏ đến cùng một vị trí
<code>ptr_a == ptr_b</code>	Trả về giá trị true nếu cả hai con trỏ ptr_a và ptr_b trỏ đến cùng một phần tử dữ liệu.
<code>ptr_a != ptr_b</code>	Trả về giá trị true nếu cả hai con trỏ ptr_a và ptr_b trỏ đến các phần tử dữ liệu khác nhau nhưng có cùng kiểu dữ liệu.
<code>ptr_a == NULL</code>	Trả về giá trị true nếu ptr_a được gán giá trị NULL (0)

Các phép toán con trỏ thường dùng



THANK
FOR WATCHING



FullHouse