**Reference(Tham Chiếu)**

Defind: Là 1 loại biến luôn.

Cách hoạt động: Khi truyền đối số(bắt buộc là biến) vào 1 hàm thì tham chiếu sẽ tự động lấy vùng dữ liệu của biến đó xài.

VD: void Test(int &y);

{}

Main()

{

Test(x);

}

Lúc này y sẽ tham chiếu vùng dữ liệu của x, nó sẽ sử dụng vùng dữ liệu của biến x, vì thế khi dùng tham chiếu ta có thể:

* Reference:
* Notes:

+ Thay đổi được giá trị của biến được nhập vào đối số(Argument) cụ thể là biến x.

+ Nó sẽ tiết kiệm dữ liệu và tăng hiệu xuất vì không cần phải truy cập thêm 1 vùng dữ liệu trong bộ nhớ.

+ Có thể trả lại được nhiều giá trị, cụ thể là trả về biến trong đối số . ☺

+ Cứ thay đổi biến tham chiếu thì biến được tham chiếu cũng thay đổi.

+ Chỉ có thể truyền đối số là biến (do nó lấy vùng dữ liệu xài mà ☺).

* Syntax: <tên hàm>(<Kiểu dữ liệu> &<Biến>,…);

Vd: Reference\_ex(int &x)

* Const Reference(Tham Chiếu Hằng):
* Notes:

+ Mục đích: Không thay đổi giá trị của đối số.

+ Mà tham chiếu cũng không thay đổi được nốt.

+ Có thể lấy đối số là biến, hằng, số, biểu thức(do nó không đổi nên k cần nhất thiết phải lấy vùng dữ liệu).

+ Tiết kiệm dữ liệu giống tham chiếu bình thường.

* Syntax: <tên hàm>(<Const> <Kiểu dữ liệu> &<Biến>,….);

Vd: CReference\_ex(const int &x)

**Reference (part 2)**

1. Chức năng:

* Tham chiếu là 1 biến có thể truy cập vùng nhớ của biến khác và sửa chữa dữ liệu trong vùng nhớ ấy (con trỏ bản vip).
* Tham chiếu như 1 tên gọi khác của biến cần tham chiếu ấy, nó cho phép truy cập trực tiếp đến bộ nhớ của biến cần dùng.
* Lợi ích: Nó sẽ giúp ta đỡ tốn bộ nhớ khi phải tạo ra bản sao 1 đống biến, hay không gặp trục trặc lung tung giống như con trỏ.

1. Syntax:

Int &ref=value;

* Nhớ là & ở ngữ cảnh chỗ khai báo là tham chiếu k phải địa chỉ.
* Tham chiếu hằng thì mới truy cập vào bộ nhớ của hằng được và nó không đổi giống hằng.

1. Notes:

* Tham chiếu chỉ có thể tham chiếu 1 biến thôi, đừng tham lam ☺

Vd:

Int value1=10;

Int value2=15;

Int &ref=value1;

Ref=value2;

Thì giá trị của value1=ref=value2=15 thật, nhưng địa chỉ của tham chiếu vẫn ở &value1.