**Recursion(đệ quy)**

1. Syntax:

Int dequy(int dem)

{

If dem>1

Dequy(dem-1);

}

* Gọi lại hàm đấy trong hàm đấy.
* Nhớ có đk ngừng k là lặp vô hạn.

1. Function:

* Giống như lệnh lặp mà phiên bản hàm
* Về cơ bản thì đệ quy là gọi lại chính hàm đang sử dụng trong hàm ấy.
* Cơ chế hoạt động:

Giống if thôi, gọi lại và giảm đi điều kiện điểm dừng đến khi nào nó ngừng lại. nó chỉ xử lý phần trước khi gọi lại hàm, khi tới điểm dừng thì rồi nó mới xử lý phần sau khi gọi hàm, nó xử lý ngược lại với trước.

VD: Như ngăn xếp ấy: kéo ngăn ra, bỏ cái đầu tiên vô ngăn đầu, xong gọi lại là bỏ cái thứ 2 xuống ngăn dưới,…..xuống tới ngăn n(dừng), khi dừng thì nó mới đóng ngăn lại, lúc đóng nó sẽ thực hiện phần còn lại(phần ở dưới lời gọi hàm), từ phần n đến phần đầu tiên.(maybe ☺)

Vd của vd: void dequy(int count)

{

cout << count << '\n';

if (count > 1)

dequy(count - 1);

cout << count<<endl;

}

* Nó sẽ thực hiện phần trên hàm dequy(count-1) trước, rồi sau khi count=1 rồi thì nó bắt đầu đóng ngăn lại, thực hiện phần count<<count<<endl; ở dưới. vì vậy ta sẽ thấy phần dưới sẽ ngược phần trên.
* Thực hiện kiểu xong ngăn n, đóng hàm n(thực hiện dưới câu lệnh gọi đến đóng), xong ngăn n-1, đóng hàm n-1……
* Ưu tiên dùng vòng lặp hơn đệ quy.