

## Recursion

Recursion is the process of calling one's self, that is, inside a function the function calls itself. For example:

```
void Foo(int x)
{
    if( x > 0)
        Foo(x / 2);
    cout << x % 2;
}
```

All recursive routines must have the following:

- Base condition
- Pattern

The base condition is when the recursion terminates, when the function stops calling itself. The recursive calling must stop at some point, otherwise you will have an infinite recursion (like an infinite loop).

There must be some pattern that is repeated that can be reproduced in the code.

For example to write a recursive function to compute the factorial of a number ( $4! = 4 \cdot 3 \cdot 2 \cdot 1$ ), we define the base case:

$$F(0) = 1$$

and the pattern to be:

$$F(n) = n \cdot F(n-1)$$

We can translate this to the C++ function:

```
int Fact(int n)
{
    if (0 == n)
        return 1;
    else
        return n * Fact(n-1);
}
```