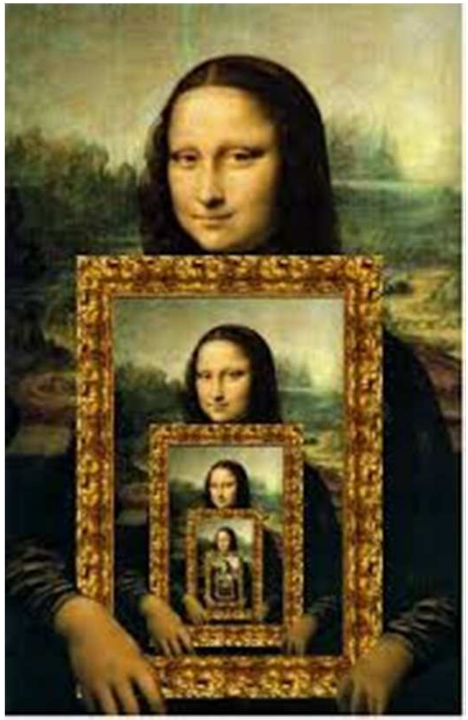
# UF2: Recursividad



**Mª Belén Tortosa**

# Recursividad

Recursividad consiste en que una función se llama a SÍ misma para resolver un problema.

Primero veremos un ejemplo.

Factorial de una función.

$$6!=6*5!$$

$$0!=1$$

Estas dos afirmaciones son las que nos indican que podemos usar recursividad.

Veámoslo….

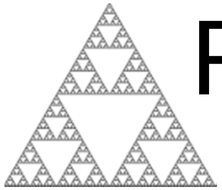
# Recursividad

## 6!=6*5!

Es la manera de pasar de un problema ( calcular factorial de 6 ) a uno más sencillo ( calcular factorial de 5)

## 0!=1

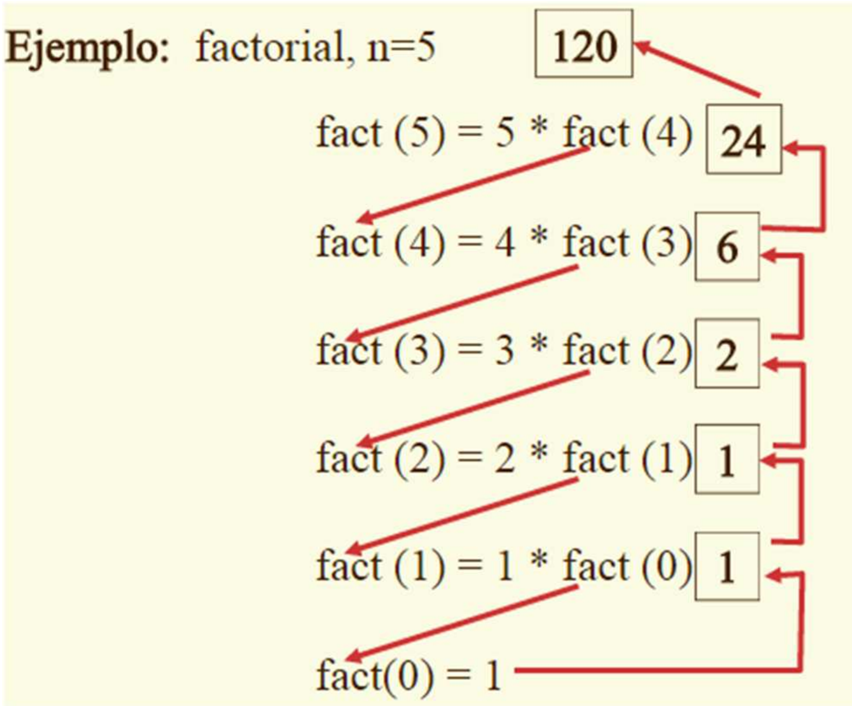
Es el caso OBVIO.



# Recursividad

Veamos como se aplica :

5!=5*4!
4!=4*3!
3!=3*2!
2!=2*1!
1!=1
1!=1*0!
0!=1




A partir del caso obvio, puedo calcular el caso 1!.

A partir del caso 2! puedo calcular el caso 3! etc…

La ventaja es que ANSI C se ocupa de los detalles de las llamadas y de remontar la cadena de llamadas para dar un único resultado.

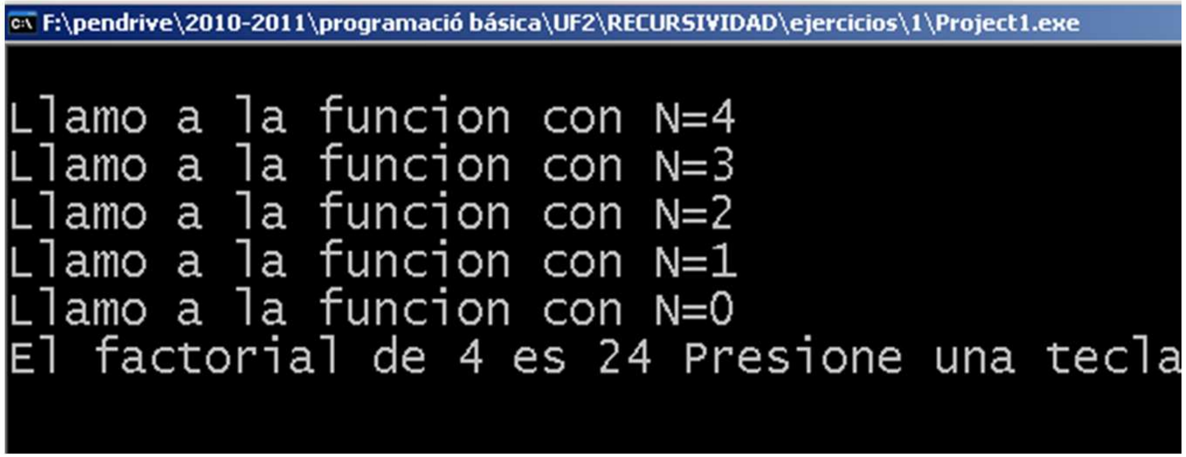
# Recursividad

¿Cómo sería el código de la función?

```
#include <stdio.h>
#include <stdlib.h>
int factorial ( int N);
int main(int argc, char *argv[])
{
  int N=4;
  printf("\nEl factorial de %d es %d ",N,factorial(N) );
  system("PAUSE");
  return 0;
}

int factorial ( int N) {
    int res;
    printf("\nLlamo a la función con N=%d",N);
    if(N<0) res=-1; //Caso que no puedo calcular
    else if (N==0) res=1; // caso obvio
    else res=factorial(N-1)*N; //caso que disminuye la dificult
    return res;
}
```

# Recursividad




Gracias al printf que indica el N con el que llamamos a la función, podemos seguir la pista de con qué valores de N se ha llamado a la función factorial.

# Recursividad

## Ejercicios

1. Escribe la función Potencia(x, y) = x^y de manera recursiva.
   Ayuda: pot(x,n)=x*pot(x,n-1); pot(x,0)=1;

2.Escribe el producto de dos números de manera recursiva.
   Ayuda: 2x3 implica sumar tres veces el número dos

3. Escribir la función que dé el elmento n-esimo de la serie de Números de Fibonacci:

   f(0) = 0; f(1) = 1

   f(x + 1) = f(x) + f(x - 1)

4. Calcula la suma de los elementos de un vector de forma recursiva.

5.Escribe la función de búsqueda binaria de forma recursiva.

6. Calcula los números combinatorios. ( ver diapositiva siguiente)

# Recursividad

## Números combinatorios

Definición y ejemplo de número combinatorio

$$\binom{m}{n} = \frac{m!}{n!(m-n)!} \quad \text{con m} \geq n$$

Para calcular un número combinatorio siempre podem

$$\binom{7}{5} = \frac{7!}{5! \cdot 2!} = \frac{7 \cdot 6 \cdot 5!}{5! \cdot 2!} = \frac{42}{2} = 21$$

Triángulo de Tartaglia:

$$\begin{array}{ccccccccc} & & & 1 & & 1 & & & \\ & & 1 & & 2 & & 1 & & \\ & 1 & & 3 & & 3 & & 1 & \\ 1 & & 4 & & 6 & & 4 & & 1 \end{array}$$

$$\begin{array}{ccccccccc} & & & \binom{1}{0} & & \binom{1}{1} & & & \\ & & \binom{2}{0} & & \binom{2}{1} & & \binom{2}{2} & & \\ & \binom{3}{0} & & \binom{3}{1} & & \binom{3}{2} & & \binom{3}{3} & \\ \binom{4}{0} & & \binom{4}{1} & & \binom{4}{2} & & \binom{4}{3} & & \binom{4}{4} \end{array}$$

$$\binom{m}{n} + \binom{m}{n+1} = \binom{m+1}{n+1}$$

Esta es la manera de pasar de un caso a uno más sencillo. El número combinatorio con m+1 en la parte de arriba se calcula a partir de dos con m.

Casos obvios.

1. $\binom{m}{0} = \frac{m!}{0!(m-0)!} = 1$
2. $\binom{m}{m} = \frac{m!}{m!(m-m)!} = 1$
3. $\binom{m}{1} = m$