

## Práctica de recursión

1. Escribir una función recursiva que devuelva el número de dígitos de un entero no negativo
2. Escribir `invertir()` que muestre los dígitos de un número en orden inverso en forma recursiva y no recursiva y comparar.
3. Escribir un algoritmo recursivo que invierta una cadena de caracteres.
4. Escriba una función recursiva que diga si una palabra es o no un palíndromo (igual leída de derecha a izquierda o de izquierda a derecha).
5. Escribir una función recursiva, `vocales`, que devuelva el número de vocales dentro de un string.
6. Escribir una función recursiva `sumSquares` que devuelva la suma de los cuadrados de los números de 0 a N, donde N es no negativo.
7. Escribir una función recursiva que encuentre y devuelva la suma de los elementos en un array entero.
8. Problema combinatorio: ¿De cuántas formas diferentes pueden elegirse 4 personas de un grupo de 10?

El problema general puede ser: encontrar el número de formas que  $r$  cosas diferentes pueden ser elegidas de un conjunto de  $n$  ítems.

$C(n, r)$  denota el número de formas diferentes que pueden elegirse:

$$C(n, r) = \frac{n!}{r!(n-r)!}$$

$$C(n, 0) = C(n, n) = 1.$$

$$\text{Se sabe que } C(n, r) = C(n-1, r-1) + C(n-1, r).$$

- Escribir un algoritmo recursivo  $a$  para determinar  $C(n, r)$ .
- Identifique el caso base y el caso general.
- Usando su algoritmo, determine:  $C(5, 3)$  y  $C(9, 4)$ .