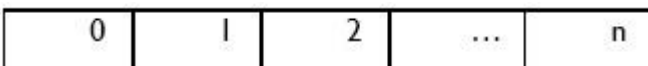


Listados de Ejercicios de Recursividad

- 1) Escriba una función recursiva llamada *calcularFactorial* que calcule el factorial de un número X.
- 2) Escriba una función recursiva llamada *multiplicarConSuma* que calcule $x*y$ utilizando sumas.
- 3) Escriba una función recursiva llamada *sumarArreglo* que sume los elementos de un arreglo A.
- 4) Escriba una función recursiva llamada *imprimirArreglo* que imprima los elementos del arreglo A.
- 5) Escriba una función recursiva llamada *buscarX* que busque un valor X en un arreglo A.
- 6) Escriba una función recursiva llamada *calcularOcurrencia* que cuente la cantidad de ocurrencias de un elemento x en una lista L.
- 7) Escriba una función recursiva llamada *sumaLista* que retorne la suma de los elementos de una lista de enteros L.
- 8) Escriba una función recursiva llamada *listarLista* que imprima los valores de una lista simplemente enlazada.
- 9) Escriba una función booleana recursiva llamada *sonIguales* que reciba dos listas como parámetros y devuelva TRUE si son iguales (mismos elementos en el mismo orden), o FALSE en caso contrario.
- 10) Escriba una función recursiva llamada *existeElemento* que verifique si un elemento x se encuentra en una lista L.
- 11) Escriba una función recursiva llamada *eliminarX* que elimine el elemento x de la lista L.
- 12) Escriba una función recursiva llamada *calcularMaximo* que encuentre el máximo de una lista de naturales L.
- 13) Escriba una función recursiva llamada *insertarOrdenado* que inserte en forma ordenada un elemento x en una lista ordenada L.
- 14) Escriba una función recursiva llamada *ordenarLista* que ordene una lista L.
- 15) Escriba una función recursiva llamada *Merge* que, a partir de dos listas ordenadas L1 y L2, genere una lista ordenada L3, a través de un proceso de intercalación de elementos (merge).
- 16) Escriba una función recursiva llamada *invertirLista* que, dada una lista L, la invierta.
- 17) Escriba una función recursiva llamada *invertirPila* que, dada una pila P, la invierta.

- 18) En una hilera de una calle con adoquines unos niños juegan a la rayuela. Para esto numeran los adoquines de la siguiente forma:



Los movimientos permitidos del juego son:

- Al principio del juego los niños se ubican en el adoquín 0.
- De un adoquín numerado i se puede saltar al adoquín numerado i+1.
- De un adoquín numerado i se puede saltar al adoquín numerado i+2 (sin pasar por el adoquín i+1)
- Por ejemplo, el número de caminos posibles para n=3 es 3 y son los siguientes: (0,1,2,3), (0,2,3) y (0,1,3).

Escriban una función recursiva llamada *CaminosPosibles* que calcule el número de caminos posibles para alcanzar un adoquín objetivo numerado con n (mayor que cero).

- 19) Escriba una función recursiva *SumaC* que, dada una lista de Enteros positivos, indique la mínima cantidad de elementos consecutivos al final de la lista cuya suma sea mayor o igual que una determinada cantidad C, que se recibe como parámetro. Algunos ejemplos son:

1, 2, 3, 4, 5, 6, 7 con $C = 8$ da 2 porque $(6+7=13 > 8)$

7, 6, 5, 4, 3, 2, 1 con $C = 8$ da 4 porque $(4+3+2+1=10 > 8)$

3, 6, 1, 7, 5, 2, 9 con $C = 13$ da 3 porque $(5+2+9=16 > 13)$

Nota: Si C supera la suma de todos los elementos de la lista deberá retornarse la cantidad de elementos que tiene la lista.

Para este ejercicio no se permite invertir la lista y luego resolverlo para los primeros elementos.