

PROGRAMACION II - GUIA DE EJERCICIOS DE RECURSIVIDAD

Ejercicio 1:

Realizar un procedimiento que imprima de manera recursiva los números del 1 al 10.

Ejercicio 2:

Realizar un procedimiento que imprima de manera recursiva la tabla del 9.

Ejercicio 3:

Realizar un procedimiento que encuentre el número mayor de un vector de n números enteros de manera recursiva.

Ejercicio 4:

La secuencia fibonacci es una parte famosa de la matemática. La misma posee una definición recursiva en donde los dos primeros valores de la secuencia son 0 y 1 (esencialmente 2 casos base) y cada valor posterior es la suma de los dos valores anteriores, por lo que toda la secuencia es: 0, 1, 1, 2, 3, 5, 8, 13, 21 y así sucesivamente.

Se solicita definir un método de fibonacci recursivo (n) que devuelve el número fibonacci n-ésimo, con n = 0 representando el inicio de la secuencia. Ej:

fibonacci(0) → 0

fibonacci(1) → 1

fibonacci(2) → 1

Ejercicio 5:

Tenemos un número de conejitos y cada conejito tiene dos grandes orejas. Se quiere calcular el número total de orejas en todos los conejos recursivamente (sin bucles o multiplicación). Escriba una función recursiva para obtener dicho total. Ej:

bunnyEars(0) → 0

bunnyEars(1) → 2

bunnyEars(2) → 4

Ejercicio 6:

Tenemos conejitos de pie en una línea, numerados 1, 2, ... Los conejitos en posiciones impares (1, 3, ...) tienen las 2 orejas normales. Los conejos en las posiciones pares (2, 4, ...) diremos que tienen 3 orejas, porque cada uno tiene un pie levantado. Escriba una función que devuelva recursivamente el número de "orejas" en la línea de conejos 1, 2, ... n (sin bucles o multiplicación).

bunnyEars2(0) → 0

bunnyEars2(1) → 2

bunnyEars2(2) → 5

Ejercicio 7:

Dado un número entero N, obtener el factorial del mismo siguiendo el algoritmo $n * (n-1) * (n-2) \dots 1$.
Escribir una función recursiva (sin bucles o multiplicación).

factorial(1) → 1

factorial(2) → 2

factorial(3) → 6

Ejercicio 8:

Escribir una función que realice la división de 2 número enteros mediante sucesivas restas utilizando recursividad.

Ejercicio 9:

Escribir un algoritmo recursivo que permita invertir un número entero: Ej: 123 => 321.

Sabiendo que:

- El módulo de 10 de cualquier número entero permite obtener el dígito que se encuentra más a la derecha. Ej $126 \% 10 \Rightarrow 6$.
- Dividiendo cualquier número entero por 10 permite eliminar el dígito que se encuentra más a la derecha. Ej $126 / 10 \Rightarrow 12$.

Ejercicio 10:

Escribir un algoritmo recursivo que permita obtener la suma de los dígitos de un número entero N.

Sabiendo que:

- El módulo de 10 de cualquier número entero permite obtener el dígito que se encuentra más a la derecha. Ej $126 \% 10 \Rightarrow 6$.
- Dividiendo cualquier número entero por 10 permite eliminar el dígito que se encuentra más a la derecha. Ej $126 / 10 \Rightarrow 12$.

Ejemplo:

sumDigits(126) → 9

sumDigits(49) → 13

sumDigits(12) → 3

Ejercicio 11:

Escribir un algoritmo recursivo que permita contar la ocurrencia del dígito 7 en cualquier número entero ingresado por teclado.

Sabiendo que:

- El módulo de 10 de cualquier número entero permite obtener el dígito que se encuentra más a la derecha. Ej $126 \% 10 \Rightarrow 6$.
- Dividiendo cualquier número entero por 10 permite eliminar el dígito que se encuentra más a la derecha. Ej $126 / 10 \Rightarrow 12$.

Ejemplo:

`contar7(717) → 2`

`contar7(7) → 1`

`contar(123) → 0`

Ejercicio 12:

Escriba un algoritmo recursivo que permita sumar todos los valores contenidos en un arreglo de números enteros. Solicitar al usuario que defina el tamaño del arreglo y luego cargar los valores de dicho arreglo.

Ejercicio 13:

Escriba un algoritmo recursivo que permita saber si un número es positivo o negativo utilizando Recursividad Indirecta (implementando llamada entre 2 métodos entre sí).

Ejercicio 14:

Escriba un algoritmo recursivo que permita saber si un número es impar utilizando Recursividad Indirecta (implementando llamada entre 2 métodos entre sí).

Ejercicio 15:

Escribir un algoritmo recursivo que permita realizar la búsqueda del mayor número entero dentro de un arreglo de números enteros. Solicitar al usuario que defina el tamaño del arreglo y luego cargar los valores de dicho arreglo.