

RECURSIVIDAD

PRÁCTICA 2

Darío G. Cardacci
2020

1. Escribir una función recursiva que devuelva la suma de los primeros N enteros.
2. Escribir una función recursiva que devuelva los primeros N números pares.
3. Escribir una función recursiva que resuelva N^p donde N y p son números positivos.
4. Escribir una función recursiva que retorne los primeros N números de la serie de Fibonacci.
5. Escribir una función recursiva SumaHasta(n, desde) -> numero. Debe retornar la suma de los números desde el valor "desde" hasta los N consecutivos a él. Por ejemplo. SumaHasta(5,10) = 10 + 11 + 12 + 13 + 14 => 60.
6. Escribir un programa que encuentre la suma de los enteros positivos pares desde N hasta 2. Chequear que si N es impar se imprima un mensaje de error.
7. Escribir un función recursiva Aparear(unaLista, otra) -> lista de pares (x, y): tal que "x" pertenece a "unaLista", e y pertenece a "otra". Ejemplo: Aparear([1,2,3], ['a','b','c']) -> [(1,'a'), (2,'b'), (3,'c')].
8. Escribir una función recursiva Mayúsculas(unString) -> otro string igual pero en mayusculas. Tratar el strings como un vector de caracteres. Ejemplo Mayúscula("hola") -> Hola.
9. Programe un método recursivo que transforme un número entero positivo a notación binaria.
10. Programe un método recursivo que transforme un número expresado en notación binaria a un número entero.
11. Implemente una función recursiva que nos diga si una cadena es palíndromo.
12. ¿Cuál es el resultado de esta función para distintos valores de X?

```
Static int f(int x)
{
    if (x > 100) {return (x-10);}
    else {return(f(f(x+11)));}
}
```