



Introducción a la Programación

Guía de ejercicios 06: Recursividad

OBSERVACIÓN:

Los siguientes ejercicios deben resolverse empleando recursividad.

1. Escriba un programa que calcule la suma de los primeros “n” números a través del empleo de una función recursiva. El programa debe solicitar primero el número “n” y después de los cálculos imprimir la suma.
2. Escriba un programa que calcule la potencia de un número a través del empleo de una función recursiva. El programa debe solicitar primero el número que representa la base y luego el número que representa al exponente. Al finalizar el programa, se debe imprimir la potencia solicitada.
3. Escriba un programa que solicite un número y que calcule su factorial mediante el uso de una función recursiva.
Considere que $n! = n * (n-1)!$, y que $0! = 1$
4. Escriba un programa que solicite dos números, y que convierta el primer número a la base representada por el segundo número. (La base podría ser hasta 16, es decir se podría pasar hasta hexadecimal).
5. Escriba un programa que calcule los términos de la Serie de Fibonacci mediante el uso de una función recursiva.
Considere que el término “n” de la serie está dado por: $f(n) = f(n-1) + f(n-2)$.
Además, debe tener en cuenta que $f(0) = 0$ y que $f(1) = 1$.



6. Escriba un programa que solicite un número decimal y que calcule la suma de los dígitos que lo componen. Al finalizar el programa debe imprimir la suma.

```
def sumaDigitos(n):  
    if n==0:  
        return 0  
    else:  
        return n%10+sumaDigitos(n//10)  
  
def main():  
    n=int(input("Ingrese numero: "))  
    print("La suma de digitos es:", sumaDigitos(n))  
  
main()
```

7. Se le pide que escriba un programa que calcule de forma recursiva los primeros “m” términos de la siguiente serie:

$$\sum_{n=1}^m \frac{1}{n} = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots$$

Al finalizar el programa debe imprimir la sumatoria obtenida.

```
def sumaSerie(n):  
    if n==1:  
        return 1  
    else:  
        return (1/n)+sumaSerie(n-1)  
  
def main():  
    n=int(input("Ingrese cantidad de terminos: "))  
    print("La suma de la serie es:", sumaSerie(n))  
  
main()
```



8. Se le pide que escriba un programa que calcule de forma recursiva los primeros “m” términos de la siguiente serie:

$$\frac{1}{2} + \frac{1}{4} + \frac{1}{8} + \frac{1}{16} + \dots$$

Al finalizar el programa debe imprimir la sumatoria obtenida.

```
import math

def sumaSerieDos(n):
    if n==1:
        return 1/2
    else:
        return (1/math.pow(2,n))+sumaSerieDos(n-1)

def main():
    n=int(input("Ingrese cantidad de terminos: "))
    print("La suma de la serie es:",sumaSerieDos(n))

main()
```

9. Implemente un programa que calcule de forma recursiva la suma de elementos de una lista dada como parámetro.

```
def sumaElemLista(lista):
    if len(lista)==1:
        return lista[0]
    else:
        return lista[len(lista)-1]+sumaElemLista(lista[:len(lista)-1])

def main():
    lista=[10,20,30,40]
    print("La suma de la elementos de la lista es:",sumaElemLista(lista))

main()
```



10. Implemente un programa que calcule de forma recursiva el elemento máximo de una lista dada como parámetro.

```
def mayorElem(lista):  
    if len(lista)==1:  
        return lista[0]  
    else:  
        m=mayorElem(lista[1:])  
        if m>lista[0]:  
            return m  
        else:  
            return lista[0]  
  
def main():  
    lista=[10,50,30,40]  
    print("el mayor elemento de la lista es:",mayorElem(lista))  
  
main()
```

11. Implemente un programa que reciba una lista y un valor X, y calcule de forma recursiva la cantidad de elementos que son mayores que X.

12. Implemente una función recursiva que permita calcular el valor de la función matemática F evaluada en "x", dadas las siguiente condiciones: F(0) es 4 y para todo valor de "x" mayor que 0 se cumple que $F(x) = 16 + F(x - 1)$. Su programa principal debe solicitar al usuario el ingreso de x del teclado, realizar el llamado correspondiente a la función recursiva y mostrar un mensaje con el resultado en pantalla.



13. Escribir un programa con funciones recursivas que permita sumar si el número es par, los 10 anteriores números pares, si es impar los 5 anteriores números impares.

```
def sumaParesImpares(n,cant):  
    if cant==0:  
        return 0  
    else:  
        return n + sumaParesImpares(n-2,cant-1)  
  
def main():  
    n=int(input("ingrese numero: "))  
    cantPares=10  
    cantImpares=5  
    if n%2==0:  
        print("suma pares:",sumaParesImpares(n,cantPares))  
    else:  
        print("suma pares:",sumaParesImpares(n,cantImpares))
```