



EXERCICI PUNTUABLE TECPRO

27/05/2014

Grau en Enginyeria de Sistemes TIC

COGNOMS:

NOM:

GRUP de LAB:

Exercici 1. Donada la següent funció recursiva, justifiqueu el resultat de cadascun dels doctests. Pel primer doctest, justifiqueu quantes crides recursives s'efectuaran.

```
def sumatori(llista):  
    """  
    >>> sumatori([1,3,5,7,9])  
    # a contestar  
    >>> sumatori([])  
    # a contestar  
    """  
    if len(llista) == 1:  
        return llista[0]  
    else:  
        return llista[0] + sumatori(llista[1:])
```

Exercici 2. Anàlisi de la complexitat. Donades les funcions que segueixen, **justifiqueu** per cadascuna si són d'ordre logarítmic $O(\log n)$, d'ordre lineal $O(n)$, d'ordre quadràtic $O(n^2)$ o bé d'ordre cúbic $O(n^3)$.

```
def comprovaa(n):  
    x=0  
    for i in range(n):  
        x+=1  
    for j in range(n):  
        x+=2  
    return x  
  
def comprovac(n):  
    l=1  
    m=0  
    p=1  
    for r in range(n):  
        w=1*r+45  
        v=m*m  
    for i in range(n):  
        for j in range(n):  
            a=i*i  
            b=j*j  
            c=i*j  
    z=33
```

Exercici 3. La pila. Suposant la implementació de l'estructura de dades lineal pila implementada en Python d'acord amb les indicacions treballades a classe, **justifiqueu** quin serà el resultat d'executar les instruccions que segueixen.

```

from stack import Stack
s=Stack()
s.push(22)
s.push(34)
s.push(22)
s.push(3)
while not s.isEmpty():
    z=s.top()
    if z==22:
        s.push(1)
    elif z==1:
        print "Trobat 1"
        s.pop()
        s.pop()
    else:
        print z
        s.pop()

```

Exercici 4. Apartat a) Escriu una funció **recursiva** de nom *exponent*, tal que, donada una base i un exponent positiu, retorni el resultat de calcular recursivament base elevat a exponent.

```

def exponent(base,e):
    """
    Retorna el resultat de calcular recursivament base elevat a e, e>=0
    >>> exponent(0,5)
    0
    >>> exponent(2,4)
    16
    >>> exponent(3,0)
    1
    """

```

Apartat b) Escriu una funció **recursiva** tal que, donat un nombre en base decimal, permeti convertir-lo a un nombre binari. No es poden utilitzar piles ni llistes ni funcions predefinides de python (en cas que s'utilitzin, l'exercici no es puntuarà.)

```

def decimal2binari(n):
    """
    retorna un string resultant de la conversio de l'enter n>=0 a binari
    >>> decimal2binari(22)
    '10110'
    """

```

Exercici 5. La llista de k en k. Escriu una funció **recursiva** de nom *novallista*, tal que, donada una llista d'enters, retorni una nova llista on cada element resulta d'agrupar els elements originals agafats de k en k, $k \geq 1$, i representar-los en la nova llista com un nou element que té per valor la mitjana dels originals. Exemples d'execució:

```

def novaLlista(llista,k):
    """
    >>> novaLlista([1,2,3,4,5,6,7,8,9],2)
    [1, 3, 5, 7, 9]
    >>> novaLlista([1,2,3,4,5,6,7,8,9],4)
    [2, 6, 9]
    >>> novaLlista([1,2,3,4,5,6,7,8,9],1)
    [1, 2, 3, 4, 5, 6, 7, 8, 9]
    >>> novaLlista([],3)
    []
    """

```

Exercici 6. L'arbre binari de cerca. Donada la classe *binaryTree*, que representa un arbre binari de cerca, creeu un mètode **recursiu**, de nom *quantsNodes()* que retorni el nombre de nodes de l'arbre.