

NOMBRE Y APELLIDOS:

1. [12 %] Escribir qué se muestra por pantalla cuando se ejecuta el siguiente programa:

| | | |
|---|---|--|
| <pre>def campoo(a): b = True for k in range(len(a)): b = b and decr(a,k) return b def decr(a,k): a[k] = a[k]-1 return a[k] >= 0</pre> | <pre>>>> a = [1,2,3,4] >>> print(campoo(a)) >>> print(a) >>> print(campoo(a)) >>> print(a)</pre> | <p>SOLUCION</p> <pre>-----> -----> -----> -----></pre> |
|---|---|--|

2. [8 %] El texto recuadrado se compone de cinco caracteres, cuyos *code points* UCS, en notación hexadecimal y por orden, son: U+950C, U+7A, U+26A, U+14B y U+6B. Escribe, en notación hexadecimal, su codificación UTF-8.

铉zıjk

3. [6 %] Se considera el siguiente código.

| | | |
|--|------------------------|---|
| <pre>v = (((a%b)<=a) and (b<a or a!=a and a!=b))</pre> | <pre> </pre> | <p>¿cuál es el valor de la variable v, cuando ?</p> <pre>-- a=6 y b=5 -----> v=</pre> <pre>-- a=6 y b=12-----> v=</pre> |
|--|------------------------|---|

4. [12 %] Escribe una función para dibujar triángulos como los de el ejemplo. Esta función ha de tomar dos argumentos: un número entero que actuará como parámetro del dibujo (el número de símbolos que forman los catetos del triángulo) y un valor *booleano* del que dependerá la colocación de la figura:

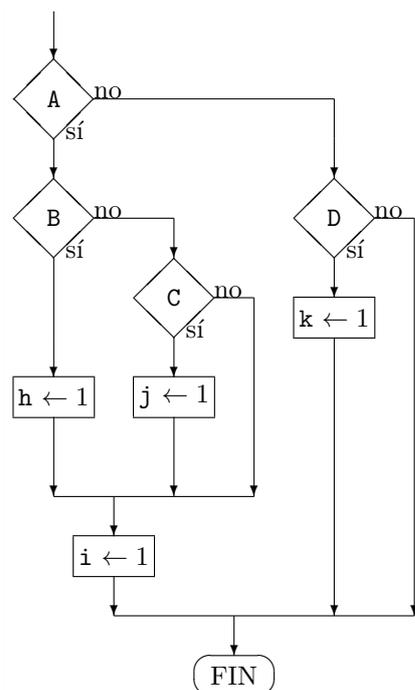
| | |
|---|--|
| <pre>>>> triangulo(1,False) \$ >>> triangulo(2,True) \$ \$ \$ >>> triangulo(3,False) \$ \$ \$ \$ \$ \$</pre> | <pre>>>> triangulo(4,True) \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ >>> triangulo(5,False) \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$ \$</pre> |
|---|--|

5. [4 %] Indicar las principales diferencias entre compilador e intérprete y pon un par de ejemplos de lenguajes de alto nivel para cada uno de ellos.

6. [10 %] Representa en forma de diagrama de flujo el algoritmo 1 y escribe en el lenguaje Python el algoritmo 2.

```
if A:  
    j=1  
    while B:  
        if C:  
            j+=1  
        j+=2  
else:  
    j=2  
print j
```

Algorithmo 1



Algorithmo 2

