

Autoevaluación

Programación 1

Primer Semestre 2017

1. Dada la siguiente lista definida por compresión, indicar cuál es su resultado: `[x+8 for x in range(3,7)]`
 - a. [8, 9, 10, 11]
 - b. [3, 4, 5, 6]
 - c.** [11, 12, 13, 14]
 - d. [8, 8, 8, 8]
2. Dada la siguiente lista definida por compresión, indicar cuál es su resultado: `[c for c in "programa"]`
 - a. ["programa"]
 - b.** ['p', 'r', 'o', 'g', 'r', 'a', 'm', 'a']
 - c. ['a', 'm', 'a', 'r', 'g', 'o', 'r', 'p']
 - d. ["programa", "programa", "programa", "programa", "programa", "programa", "programa", "programa"]
 - e. Ninguna de las anteriores
3. Dada la siguiente lista definida por compresión, indicar cuál es su resultado: `[[z,k] for z in range(3) for k in range(3,5)]`
 - a.** [[0, 3], [0, 4], [1, 3], [1, 4], [2, 3], [2, 4]]
 - b. [[3, 3], [3, 5], [3, 3], [5, 3]]
 - c. [[1, 3], [1, 4], [2, 3], [2, 4], [3, 3], [3, 4]]
 - d. [[1, 3], [1, 4], [1, 5], [2, 3], [2, 4], [2, 5], [3, 3], [3, 4], [3, 5]]
4. Dada la siguiente lista definida por compresión, indicar cuál es su resultado: `[s.upper() for s in "hoy es viernes"]`
 - a. ['HOY', 'ES', 'VIERNES']
 - b. ['H', 'O', 'Y', 'E', 'S', 'V', 'I', 'E', 'R', 'N', 'E', 'S']
 - c.** ['H', 'O', 'Y', ' ', 'E', 'S', ' ', 'V', 'I', 'E', 'R', 'N', 'E', 'S']
 - d. ['HOY ES VIERNES']
5. Dada la siguiente lista definida por compresión, indicar cuál es su resultado: `[len(z) for z in "hoy es viernes 24".split()]`
 - a. [1, 1, 1, 1]
 - b. [3, 1, 2, 1, 7, 1, 2]
 - c. [4, 4, 4, 4]
 - d.** Ninguna de las anteriores

Laboratorio 10

Programación 1

Segundo Semestre 2017

Metas del módulo

- Capacidad para trabajar con listas por comprensión
 - Capacidad para definir y trabajar con funciones anónimas.
1. Crear una función que retorne una lista creada por comprensión con los números pares hasta un número dado por parámetro.
 2. Utilizar comprensión de listas para obtener los números que se obtienen como $(n+1)*(n-1)$ para $n > 1$, y que son menores de 1000. La lista de estos números comienza con [0, 3, 8, 15, 24, ...].
 3. Utilizando listas por comprensión implementar una función que toma una cadena y devuelve la cadena en mayúsculas sin espacios en blanco ni símbolos de puntuación (los dígitos no se cambian).
 4. Dado una cadena de caracteres con fechas en formato "MM/DD/YYYY" separadas por coma, obtener una lista de cadenas de caracteres con las fechas en formato "DD/MM/YYYY"

Ej.: "10/11/2016,01/02/2016" → ['11/10/2016', '02/01/2016']

5. Dada la siguiente clase:

```
class Mail(object):  
  
    def __init__(self, asunto, cuerpo, destinatarios):  
  
        self.asunto = asunto  
  
        self.cuerpo = cuerpo  
  
        self.destinatarios = [ ... ]  
  
        self.copia = [ ... ]  
  
        self.copia_oculta = [ ... ]
```

- a. Utilizando listas por comprensión, completar el constructor de la clase a partir de la lista de destinatarios (cadenas de caracteres con las direcciones de email) que se recibe como parámetro siguiendo las siguientes condiciones:
- b. Las direcciones de email que pertenezcan al dominio "ucu.edu.uy" van en el campo (lista) copia.
- c. Las direcciones de email que pertenezcan al dominio "gmail.com" van en el campo (lista) copia_oculta.
- d. Todas las demás dirección van en la lista de destinatarios del mail