Máster Avanzado de Programación en Python para Hacking, BigData y Machine Learning

ESCUELA INTERNACIONAL POSGRADOS

Asignatura: Buenas Prácticas de Programación en Python
Docente: D. Ramón Rueda
Alumno: Fanor Valencia

ACTIVIDAD PRÁCTICA LECCIÓN 4:

APARTADO 1:

1. *Haciendo uso de comprensión de listas realice un programa que, dado una lista de listas de números enteros, devuelva el máximo de cada lista. Por ejemplo, suponga la siguiente listas de listas: [[2, **4**, 1], [1,2,3,4,5,6,7,**8**], [100,**250**,43]]*
El programa debe devolver el mayor elemento de cada sub-lista (señalado en negrita).

Código:

```
19    # Método de Compresión de lista
20    l_maximo2 = [ maximo(e) for e in l_de_listas ]
21
```

Resultado:

```
De la lista inicial:
 [[3, 54, 2, 5], [7, 8, 1, 3, 2], [99, 287, 54, 87], [1, 2, 3, 4, 5, 6, 7]]
El número mayor de cada lista es:
 [54, 8, 287, 7]
```

Ahora, haciendo uso del pdb, inserte puntos de parada justo en la línea donde ha implementado la compresión de listas. Haga pruebas mostrando el contenido de las variables y continuar con la ejecución línea a línea.

Para esta parte, hemos implementado el inicio del depurador antes de la función implementada.

```
13    print ("Apartado 1. \n ")
14    # Definimos una función que calcule el máximo de una lista y marcamos el inicio del depurador.
15    pdb.set_trace()
16    def maximo(n):
17        return amax(n)
```

Ahora, procedemos a iniciar el código y fijar los puntos de parada. Para este ejercicio, hemos definido 2. El primero en el *return* de la función *maximo(n)* con el fin de ver el valor que irá devolviendo a la lista, el segundo, tal y como se solicita en el enunciado, justo en la línea en la que hemos implementado la lista.

```
PROBLEMS   OUTPUT   DEBUG CONSOLE   TERMINAL                    Python Debug Console

MacBook-Air-de-Materiales:ACT4 materialesingremic$ python3 main.py
Apartado 1.

> /Users/materialesingremic/Desktop/ACT4/main.py(16)<module>()
-> def maximo(n):
(Pdb) break 17
Breakpoint 1 at /Users/materialesingremic/Desktop/ACT4/main.py:17
(Pdb) break 20
Breakpoint 2 at /Users/materialesingremic/Desktop/ACT4/main.py:20
(Pdb)
```

Pasamos a ejecutar el código línea a línea:

¿Qué conclusiones obtiene?
Podemos ver cómo el depurador nos muestra tanto la línea que está en ejecución como su correspondiente resultado. Al entrar en el bucle de la lista de compresión, abre la función *maximo(n)* que regresa un valor, este proceso se repite hasta no quedar más elementos por recorrer. Una vez cumplido el propósito del bucle, los valores que se han ido recolectando y añadiendo en la lista son mostrados como el resultado de la lista de compresión.

Como conclusión de este apartado, podemos comprobar la utilidad del depurador de Python(pdb), que nos permite no sólo depurar en la terminal sino que también nos ayuda a entender cómo se está ejecutando nuestro código.

APARTADO 2:

2. Haga uso de la función filter para construir un programa que, dado una lista de n números devuelva aquellos que son primos. Por ejemplo, dada la lista [3, 4, 8, 5, 5, 22, 13], el programa que implemente debe devolver como resultado [3, 5, 5, 13]

Código:

```
30  #Creamos la lista con los valores a analizar
31  l_numeros = [n for n in range(3,77)]
32
33  # Definimos una función que identifique los números primos
34  def primo(n):
35      for i in range (2,n):
36          if (n%i == 0):
37              return False
38      return True
39
40  # Utilizamos la función filter para crear una lista con los números primos de la lista inicial
41  primos = list(filter(primo,l_numeros))
42
43  print ('De la lista inicial: \n', l_numeros, '\nLos números primos son: \n', primos, '\n')
44
```

Resultado:

```
Apartado 2.

De la lista inicial:
 [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 4
3, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70, 71, 72, 73, 74, 75, 76]
Los números primos son:
 [3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73]

MacBook-Air-de-Materiales:ACT4 materialesingremic$
```