

## Práctica 2

Seminario de Python - Facultad de Informática - Universidad Nacional de La Plata

2020

1. Dada una lista de strings con el siguiente formato:

```
tam = ['im1 4,14', 'im2 33,15', 'im3 6,34', 'im4 410,134']
```

Donde **im1**, **im2**, **etc** son los nombres de las imágenes y la parte de números representa el valor de una coordenada  $(x, y)$ . Se solicita que arme dos listas que contengan, nombre y luego una tupla de las coordenadas en formato de números. Una de las listas debe contener los datos en las cuales el valor de  $x$  es mayor o igual a un número ingresado por teclado y la otra, contendrá los datos de las imágenes cuyo valor de coordenada sea menor al número ingresado. Tener en cuenta que los datos de la lista original son strings y que los números de la lista generada deben ser enteros. Se espera que se muestre el siguiente resultado, si el dato ingresado por el teclado fuera 30:

```
lista1 = ['im2', (33, 15), 'im4', (410, 134)]
lista2 = ['im1', (4, 14), 'im3', (6, 34)]
```

Nota: puede utilizar la función *string.partition* que permite separar un string y asignar a variables. Investigue su utilización.

2. Dada una lista como la utilizada en el ejercicio 1 genere una lista nueva que contenga solamente las coordenadas y ordénelas. Investigue la función **sort**. Ejemplo: si la lista original es:

```
tam = ['im1 4,14', 'im2 33,15', 'im3 6,34', 'im4 410,134']
```

La lista generada sería:

```
[(4, 14), (6, 34), (33, 15), (410, 134)]
```

3. Dada una lista con varios strings, generar una nueva lista que contenga SOLO aquellos string que representen valores enteros. La misma debe quedar ordenada. Investigue la función **string.isdecimal()**. Ejemplo: si la lista original es:

```
['Auto', '123', 'Viaje', '50', '120']
```

La lista generada sería:

```
[50, 120, 123]
```

4. Dada una lista con preguntas que se responden por 'sí' o 'no' y sus respuestas correctas, armar un juego que muestre cada una de las preguntas al jugador, verifique si es correcta o no e incremente el puntaje en caso de acertar. Se debe seleccionar en forma aleatoria la pregunta a mostrar y eliminarla una vez que ya la mostraron. El juego finaliza cuando no hay más preguntas en la lista. Un ejemplo de la lista con las preguntas podría ser:

```
preguntas = [['Buenos Aires limita con Santiago del Estero', 'no'], ['Jujuy limita con Bolivia', 'si'], ['San Juan limita con Misiones', 'no']]
```

Tener en cuenta que las respuestas solicitadas al jugador por teclado, 'si' y 'no', pueden darse en mayúsculas o minúsculas.

5. Generar un menú que te permita realizar las siguientes opciones:

```
Menú de opciones para la lista de números a ingresar:  
1: ingresar números  
2: ordenar números  
3: calcular el máximo  
4: calcular el mínimo  
5: calcular el promedio  
0: para terminar
```

Se debe repetir hasta que se ingrese la opción **0**. Se debe permitir agregar números aún luego de haber utilizado las demás operaciones utilizando la opción **1**. En caso que no se haya ingresado ningún número indicar que la lista está vacía. Investigue las funciones **max**, **min** y **sum**.

6. Generar un menú que te permita ingresar dos números por teclado y luego realizar una de las siguientes operaciones:

```
suma: +  
resta: -  
multiplicación: *  
división: /
```

Se debe mostrar el resultado de la operación.

7. Dado un string ingresado por teclado, determinar si es un palíndromo. Investigue la operación sobre strings que permite invertirlos. Tener en cuenta que puede haber mayúsculas y minúsculas mezcladas en el string ingresado.
8. Dado un string ingresado por teclado determinar si la cantidad total de veces que aparece cada letra es un número primo. Veamos un ejemplo:

```
Adivina  
La letra a aparece: 2 veces  
La letra d aparece: 1 vez  
La letra i aparece: 2 veces  
La letra v aparece: 1 vez  
La letra n aparece: 1 vez  
Por lo tanto las letras 'a' e 'i' son letras que aparecen un número  
primo de veces.
```

9. Observar el programa GUESS THE NUMBER en la página web, modifíquelo para que el rango del número a adivinar sea de 1 hasta 50 y que corte su ejecución cuando lo adivine. Agregar además que luego de 3 intentos mostrar la pista, una única vez, si el número es par o impar.
10. Dada una lista con nombres de imágenes:

```
imagenes=['im1','im2','im3']
```

Generar una estructura que asocie 3 coordenadas ingresadas por teclado  $(x_1, y_1)$ ,  $(x_2, y_2)$  y  $(x_3, y_3)$ , con cada elemento de la lista (en el mismo orden en que son ingresadas). Además verifique, mientras se van ingresando las coordenadas, que no hayan repetidas para una misma imagen; en dicho caso deberá volver a ingresarla.