

PrograUDD Ayudantía 03

Instrucciones: Resuelva con un compañero/a la siguiente lista de ejercicios. Recuerde discutir la estrategia para resolver cada problema, y luego escribir el código solución. Suba su solución como un archivo `ay03_apellido1apellido2.py` a Canvas, Sección Tareas > Ayudantías.

Guarde una copia de su solución en Google Drive.

1. Hervidor. La temperatura óptima para preparar el café va de los 92°C a los 96°C. Si el agua supera esta temperatura esta altera las propiedades de los granos de café, haciendo que este presente un sabor más amargo. Escriba un programa que imprima la temperatura del hervidor cada 10 grados hasta llegar a la temperatura óptima para preparar café (94°C). Cuando el hervidor llegue a los 94°C, el programa debe indicar que el agua se encuentra en la temperatura óptima para preparar café. Considere que la temperatura inicial del agua es aleatoria y oscila entre los 18 y 25°C.

Para generar enteros aleatorios utilice el método `randint` del módulo `random`. Ver documentación en <https://docs.python.org/3/library/random.html#random.randint>

```
from random import randint

randint(18,25) #genera numeros aleatorios entre 18 y 25
```

2. Sumatoria. Escriba un programa que calcule el resultado de la siguiente sumatoria considerando un número n ingresado por el usuario.

$$\sum_{k=1}^n \frac{k^2 + 7}{n}$$

3. Lanzador. Loreto ha creado un lanzador de pelotas para su perro, el cual funciona lanzando una bola de tenis con una trayectoria de tipo proyectil con una velocidad inicial v_0 de 16.5 [m/s]. A Loreto le gustaría saber cual es la distancia horizontal máxima d_x que puede alcanzar la bola dependiendo del grado de inclinación θ con que lance la pelota. Ayude a Loreto creando un programa que calcule la distancia horizontal que la pelota alcanza lanzandola desde diferentes ángulos y le diga a Loreto cual es la mayor distancia que puede alcanzar la pelota y con qué ángulo puede obtenerla. Considere ángulos de 0 a 90 grados, con incrementos de 5 grados (0, 5, 10, ...90).

Recuerde que la fórmula de la distancia horizontal d_x de un lanzador en función de la velocidad inicial v_0 y el ángulo θ es:

$$d_x = \frac{v_0 \sin(2\theta)}{g}, \text{ donde } g \text{ es la aceleración de gravedad.}$$

Para calcular $\sin(2\theta)$ utilice la función `sin` del módulo `math`. Recuerde que la función `sin` recibe el ángulo en radianes (haga la conversión). Ver documentación en <https://docs.python.org/3/library/math.html#math.sin>

```
from math import sin
from math import pi

s = sin(2*pi)
```

4. Naipes. Programe un juego que retire cinco cartas de un naipes ingles al azar y se detenga después de extraer 100 fulls (un trio y un par).

Puede simular la extracción de una carta al azar utilizando como base el siguiente código:

```
from random import randrange

TRAJES = ['Picas', 'Diamantes', 'Treboles', 'Corazones']
VALORES = ['2', '3', '4', '5', '6', '7', '8', '9', '10',
            'Jota', 'Reina', 'Rey', 'As']

index_val = randrange(0, len(VALORES))
index_traje = randrange(0, len(TRAJES))

print(VALORES[index_val], 'de', TRAJES[index_traje])
```

5. Amigos. Cree un programa que reciba el nombre de algun amigo o enemigo y retorne el tipo de amigo que es. Para esto cree cuatro listas diferentes con un mínimo de cuatro elementos para cada una. La primera lista debe contener los nombres de sus `mejores_amigos`, la segunda los de sus `amigos`, la tercera la de sus `conocidos` y la última los de las personas con las que no se lleva bien o `enemigos`. Uno de sus amigos (o enemigos) deberá ingresar su nombre al programa y este debe indicar a qué categoría de amigos pertenece y la cantidad de personas que se encuentran en la misma categoría.

6. Usuarios. Programe un registro de usuario en donde se ingrese el nombre del usuario, su e-mail y su contraseña y se cumplan las siguientes reglas:

- El nombre de `usuario` no puede ser igual al de su `e-mail` antes del caracter @. Por

ejemplo pedrojuandiego@gmail.com no puede registrar el nombre de usuario pedrojuandiego.

- La `contraseña` debe tener un largo de 8 o más caracteres, debe contener al menos un número y no puede contener los caracteres \$, #, %, &, /.

7. Fecha. Escriba un programa que reciba un string que contenga una fecha y una hora en el formato `AAAAMMDDHHMMSS` y lo convierta al formato `AAAA-MM-DD HH:MM:DD`. Por ejemplo, si el programa recibe el string `20170122145521`, que representa el día 22 de Enero del 2017 a las 14:55:21 hrs, este debe retornar `2017-01-22 14:55:21`.