

Materia:	Programación I		
Nivel:	1º Cuatrimestre		
Tipo de Examen:	Primer Parcial		
Apellido ⁽¹⁾ :		Fecha:	03/06/2025
Nombre/s ⁽¹⁾ :		Docente a cargo ⁽²⁾ :	Fernández - Meloni
División ⁽¹⁾ :	313-1	Nota ⁽²⁾ :	
DNI ⁽¹⁾ :		Firma ⁽²⁾ :	

(1) Campos a completar solo por el estudiante en caso de imprimir este enunciado en papel.

(2) Campos a completar solo por el docente en caso de imprimir este enunciado en papel.

A TENER EN CUENTA:

Está prohibido el uso de métodos de listas como `append()`, `remove()`, `insert()`, entre otros.

No se permite el uso de funciones predefinidas como `max()`, `min()`, `sum()`. Se espera que implementen sus propias funciones para estas operaciones.

Resolver el siguiente problema realizando un menú de opciones

Una academia de baile organiza una competencia abierta para definir a su nuevo o nueva representante nacional. La competencia cuenta con un jurado compuesto por tres personas, cada una de las cuales puntúa a los participantes del 1 al 10.

Se requiere un sistema que permita:

- **Registrar los nombres de los participantes (5 en total)**
- **Registrar las puntuaciones que cada jurado otorga a cada participante**
- **Procesar y mostrar diferentes informaciones relevantes a partir de los datos cargados**

Cada participante debe almacenar la siguiente información:

- **Nombre del participante**
- **Puntuación del Jurado 1**
- **Puntuación del Jurado 2**
- **Puntuación del Jurado 3**

Se requiere lo siguiente:

1. **Cargar participantes:** Ingresar los nombres de cinco participantes. Cada nombre debe tener al menos 3 caracteres y solo contener letras y espacios.
2. **Cargar puntuaciones:** Ingresar la puntuación de cada jurado para cada participante. Las puntuaciones deben estar entre 1 y 10.
3. **Mostrar puntuaciones:** Mostrar para cada participante: nombre, puntajes individuales y promedio general. Usar formato claro y ordenado.

```
NOMBRE PARTICIPANTE: Angela  
PUNTAJE JURADO 1: 10  
PUNTAJE JURADO 2: 8  
PUNTAJE JURADO 3: 7  
PUNTAJE PROMEDIO: 8.33/10
```

4. **Participantes con promedio mayor a 4:** Mostrar los que hayan tenido un promedio de puntuación mayor a 4. Si no hay ninguno, mostrar mensaje de error.
5. **Participantes con promedio mayor a 7:** Mostrar los que hayan tenido un promedio de puntuación mayor a 7. Si no hay ninguno, mostrar mensaje de error.
6. **Promedio de cada jurado:** Mostrar el promedio de puntuaciones otorgadas por cada jurado.
7. **Jurado más estricto:** Indicar cuál jurado otorgó los puntajes promedio más bajos.
8. **Buscar participante por nombre:** Permitir ingresar un nombre y mostrar sus datos si existe. Si no existe, mostrar mensaje de error.

PUNTOS EXTRA (SUBEN NOTA)

9. **Top 3:** Mostrar los tres participantes con mayor puntaje promedio
10. **Participantes ordenados alfabéticamente:** Mostrar los participantes ordenados de (A a Z) con sus datos.

- Deben usar su propia biblioteca para el ingreso de datos ([Inputs.py](#)).
- Debe haber un menú de opciones funcional.
- Deben implementar un módulo aparte con funciones ([Funciones.py](#)).
- El programa no debe romperse si el usuario ingresa letras u opciones no válidas en el menú o en las cargas.
- Siempre que sea posible, las funciones deben devolver un valor (al menos un booleano), salvo aquellas que solo impriman datos.
- Todas las funciones deben estar documentadas.
- Si no se cargaron los participantes ni los puntajes no se debería poder acceder a ninguna de las otras opciones.
- En el punto 7 podría haber más de un jurado estricto, si logran resolverlo de esa forma suman puntos extra.