

# 1er Parcial de Introducción a la Programación en Python

## Matemática para Economistas III

### IDEI-UNGS.

30 de septiembre de 2022

*Tenés que combatir el algoritmo,  
porque vos buscaste dos o tres  
cosas y te tiran por la cabeza  
70.000 idioteces que no te  
interesan*

---

Adrián Dárgelos

Docente: Mateo Suster — msuster@campus.ungs.edu.ar —  
Fecha límite de entrega: Martes 11 de Octubre 23:59 hs

#### Pautas del examen

El examen deberá ser realizado a partir de los siguientes lineamientos:

1. Se debe resolver individualmente.
2. Se deberá entregar el código de Python utilizado en formato **.ipynb exclusivamente**.
3. **Nombrar** el archivo solo con el número de DNI del estudiante (sin espacios ni puntos ni nada mas). Esto es muy (IMPORTANTE). Incluir dentro del archivo el apellido y nombre del estudiante.
4. Se evaluará positivamente a la inclusión de las buenas prácticas de programación vistas en clase, así como la **prolijidad** y **originalidad** de las soluciones propuestas.
5. Es válido (y preferible) solicitar ayuda entre pares (alumnos y/o docente) por Slack antes que por otro medio. Su consulta no molesta (al contrario, enriquece).
6. Utilizar código ajeno no está prohibido, siempre y cuando se explicita la fuente correspondiente. Se podrá agregar todas las aclaraciones extras que se deseen o se consideren relevantes.
7. La entrega deberá realizarse a través de mail a la casilla msuster@campus.ungs.edu.ar con el asunto "MPE III - Parcial 1 de Python".
8. Último, pero no menos importante: antes de entregar, revise que el código se ejecute de principio a fin sin errores !

#### Ejercicios

1. Elegir una ecuación diferencial (de la práctica o planteada por usted) y presentarla en lenguaje Mark-down.
2. Justificar la elección realizada en el punto anterior (explicación mínima, corta y simple de porqué le interesa resolver la ecuación elegida).

3. Resolver la ecuación diferencial con la librería SymPy. Discutir la estabilidad dinámica de la solución.
4. Resolver nuevamente la misma ecuación diferencial modificando las condiciones iniciales. Visualizar en **un mismo gráfico** la evolución a lo largo del tiempo de la ecuación elegida con las dos condiciones iniciales propuestas. Explicar porqué y cómo se modifica el comportamiento de la función. Tener en cuenta que quizás sea necesario ir variando el rango de tiempo en donde se evalúa la ecuación para lograr una mejor visualización. Bonus: graficar también el equilibrio intertemporal.
5. Definir cuatro (4) listas distintas (inventar todo lo deseado):
  - Nombres de materias (con 5 datos de tipo *string*)
  - Notas personales de un estudiante obtenidas en dichas materia (con 5 datos de tipo entero, *int*)
  - Promedio histórico de notas en dichas materias (con 5 datos de tipo real, *float*)
  - Una lista vacía

A partir de dichos objetos, escribir un programa que evalúe si ‘Matemática’ o ‘Estadística’ se encuentran dentro de la lista de materias. Luego, en caso de ser cierto, el programa debe ejecutar un bucle `while` que estime la diferencia porcentual (u otra métrica de interés) de la nota personal con respecto al promedio histórico para cada una de las materias. Guarde dichos valores en la lista vacía definida anteriormente. En caso contrario, el programa deberá imprimir por pantalla alguna explicación de porqué no ejecutó las instrucciones anteriores. Explicar el procedimiento y evaluar el funcionamiento del programa con una lista que incluya las materias ‘Matemática’ o ‘Estadística’ y con otra que las excluya.

6. Encapsular el programa del cálculo anterior en una función que devuelva la métrica elegida **si y solo si** el promedio de las notas personales del estudiante es menor a 7. En caso contrario, la función deberá retornar cual es el valor máximo de la lista promedios históricos de las materias. Acompañar los valores retornados con algunos *strings* breves que ayuden a entender la función. Evaluar la función con diferentes listas para revisar que corra sin fallas. Denominar la función con un nombre representativo del algoritmo. **Nota:** puede valerse de la función `max()`.