



# INTRODUCCIÓN

10145 - FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA






# Autograder

Este semestre se utilizará un sistema de autocorrección basado en test de entrada y salida.

Para esto, se le hace entrega de una carpeta que contiene 3 archivos:

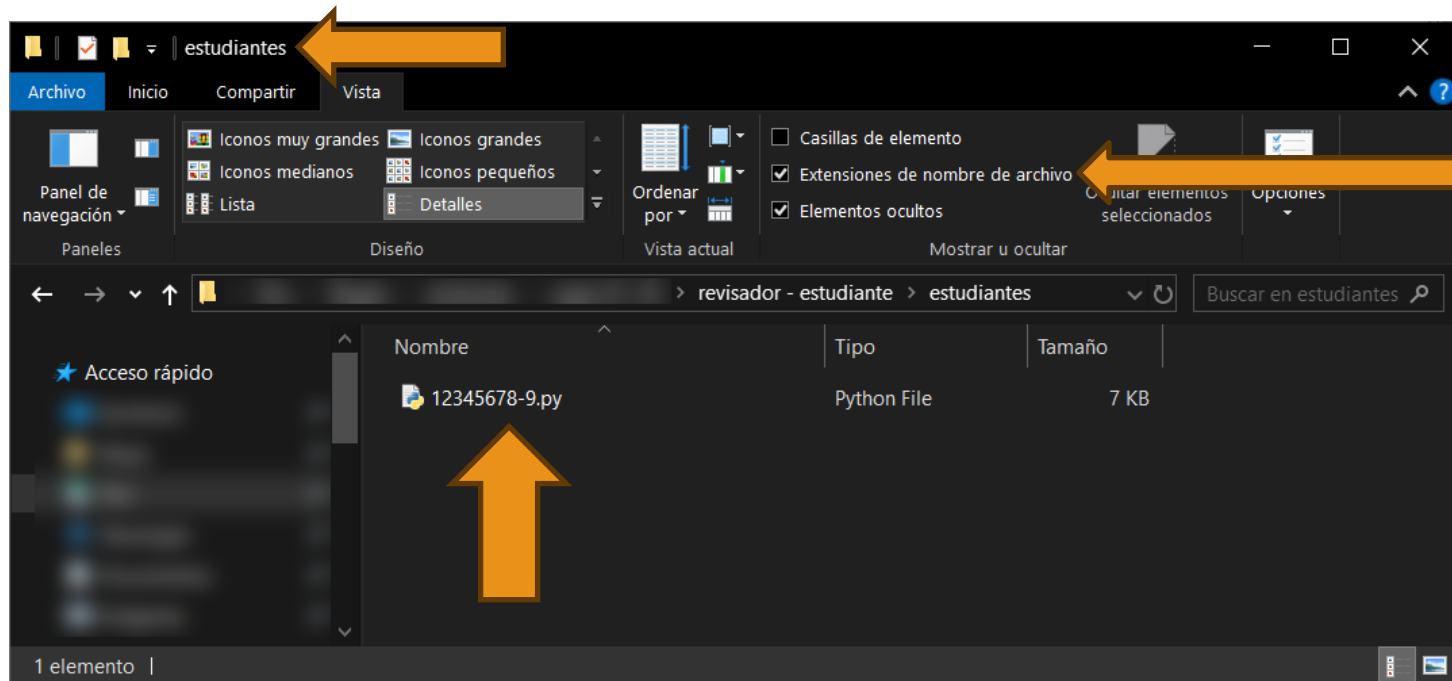
- La carpeta **estudiantes**.
- Un archivo de texto llamado **IO\_Esperado.txt**.
- Un programa de Python **revisor** (Autograder).

Nombre	Tipo	Tamaño
 estudiantes	Carpeta de archivos	
 IO_Esperado.txt	Documento de tex...	1 KB
 revisador 2000 - estudiante.py	Python File	8 KB



# Carpeta estudiante

- Cuando cree un programa y quiera validarlo, debe guardar su programa dentro de la carpeta estudiantes.
- Como nombre de su programa debe utilizar su RUN.



Nota: No tiene que ver con esto, pero ya que estamos aquí...  
Active la visualización de las extensiones si es que no las tiene a la vista.



# Archivo IO\_Esperado.txt

- Este es un archivo de texto que contiene un conjunto de datos de entrada y las salidas esperadas para cada una de estas.
- Este archivo no debe modificarlo, ya que esto variará la revisión de los test mientras usted prueba.
- Cada ejercicio traerá su propia versión de este archivo.
- Puede abrirlo y mirarlo si gusta.

```
IO_Esperado.txt: Bloc de notas
Archivo Edición Formato Ver Ayuda
###ENTRADA###
1
2
3
###SALIDA###
Elija un número entero del 1 al 21.

Grupo 1: ['05', '11', '17', '12', '15', '07', '20']
Grupo 2: ['02', '21', '04', '03', '08', '16', '14']
Grupo 3: ['19', '01', '18', '09', '13', '10', '06']
Indique en qué grupo está su número (1, 2 o 3):
Grupo 1: ['02', '03', '14', '17', '07', '01', '13']
Grupo 2: ['21', '08', '05', '12', '20', '18', '10']
Grupo 3: ['04', '16', '11', '15', '19', '09', '06']
Indique en qué grupo está su número (1, 2 o 3):
Grupo 1: ['02', '17', '13', '05', '18', '16', '19']
Grupo 2: ['03', '07', '21', '12', '10', '11', '09']
Grupo 3: ['14', '01', '08', '20', '04', '15', '06']
Indique en qué grupo está su número (1, 2 o 3): Su número es: 20
###ENTRADA###
3
2
1
```

Línea 31, columna 30 100% Windows (CRLF) UTF-8



# Autograder

- El archivo .py que se incluye es el programa encargado de comparar las soluciones.
- Este es una versión simplificada de la que utilizará su profesor/a.
- Esto no significa que validará de manera menos estricta su código, sino que el de su profesor/a obtiene algunas estadísticas adicionales de su código y cancela ejecuciones que puedan ser problemáticas.
- Por si se lo pregunta, **dentro de este programa NO viene la solución de los ejercicios**. No hay problema con que mire el código en su interior, aunque utiliza algunos elementos que no se incluyen en el alcance de contenidos de la asignatura.
- Al igual que antes, este programa no debe modificarlo.
- Este es el programa que debe ejecutar cuando usted quiera validar su solución (La que puso en la carpeta “estudiantes”)



# Autograder

- Al ejecutar este programa, su solución se ejecutará tantas veces como test tenga el archivo IO\_Esperado.txt.
- Por cada ejecución se le indicará por la consola si el test fue o no superado.
- En caso de fallar el test, se indica el tipo de error y una breve explicación de qué significa un error de ese tipo.

## Test 8: Error de Semántica

El código presentado está correctamente escrito en Python.

Pero tiene problemas con el planteamiento del ejercicio y el cómo resolverlo.

(Esto incluye que el formato requerido en el ejercicio no se ha respetado completamente.)



# Resultados.xlsx

- Una vez ejecutado el autograder, se genera un archivo .xlsx (Excel).
- En este puede encontrar el resultado de cada ejecución mostrando la entrada utilizada, salida entregada por el estudiante y salida esperada para ese test.
- Si el test se marca con un cero es porque el test falló y si tiene un 1 es porque el test ha sido superado con éxito.



# Resultados.xlsx

	A	B	C	D
1	Test 1	Entrada 1	Salida Estudiante 1	Salida Esperada 1
			Y los participantes seleccionados son: Picollo Ten Shin Han Gokú Krillin Milk Tao Pai Pai Yamcha Shen  Ya tenemos los combates!!!! Milk vs Gokú Ten Shin Han vs Tao Pai Pai Yamcha vs Shen Krillin vs Picollo  Y se lleva a cabo el torneo: Milk vs Gokú -> Gana Gokú Ten Shin Han vs Tao Pai Pai -> Gana Ten Shin Han Yamcha vs Shen -> Gana Shen Krillin vs Picollo -> Gana Krillin Gokú vs Ten Shin Han -> Gana Gokú Shen vs Krillin -> Gana Krillin Gokú vs Krillin -> Gana Gokú Y el campeon y poseedor del título del mas fuerte bajo el cielo es: Gokú !!!!!	Y los participantes seleccionados son: Picollo Ten Shin Han Gokú Krillin Milk Tao Pai Pai Yamcha Shen  Ya tenemos los combates!!!! Milk vs Gokú Ten Shin Han vs Tao Pai Pai Yamcha vs Shen Krillin vs Picollo  Y se lleva a cabo el torneo: Milk vs Gokú -> Gana Gokú Ten Shin Han vs Tao Pai Pai -> Gana Ten Shin Han Yamcha vs Shen -> Gana Shen Krillin vs Picollo -> Gana Picollo Gokú vs Ten Shin Han -> Gana Gokú Shen vs Picollo -> Gana Picollo Gokú vs Picollo -> Gana Gokú Y el campeon y poseedor del título del mas fuerte bajo el cielo es: Gokú !!!!!
2	0	10 8 Tao Pai Pai,122 Chaos,104 Yajirobe,85 Ten Shin Han,190 Gokú,178 Milk,130 Picollo,MAQUINA DESTRUIDA Krillin,166 Yamcha,112 Shen,105 8,3,2,7,1,4,5,6		





# Consideraciones

- Al momento de la corrección de las evaluaciones se evalúan 3 partes.
  1. Sistema que evalúa copias entre estudiantes.
  2. Autograder que asigna puntaje por cantidad de test superados.
  3. Revisión de su profesor/a de su algoritmo.



# Consideraciones

- En el caso del autograder, se generarán unas 100 entradas y salidas a partir de la solución creada por la coordinación.
- De esas entradas y salidas a usted se le hará entrega de un conjunto de 10 aproximadamente para que pruebe su solución.
- Al momento de la corrección de la evaluación se ejecutarán 10 test de los 100 creados en los que pueden o no estar algunos que se les han facilitado.



# Consideraciones

- Con esto se le desea remarcar que su solución debe funcionar para la generalidad del problema y no para el caso específico de los 10 test proporcionados.
- Finalmente, programas que intenten realizar “hardcoding” o prácticas similares serán sancionados con nota mínima.
- Esto significa hacer un programa no con el objetivo de resolver el problema, sino de entregar salidas forzadas para que superar el test.
- Ejemplo:
  - Que para el problema de “cree un programa que determine si un laberinto tiene o no solución” su programa siempre entregue “sí tiene solución” sin intentar solucionar el problema.



# Recordatorio

Leer y aceptar el  
código de honor

Desde abril los ejercicios de laboratorio son con nota y recuerde que si no ha leído y firmado el código de honor tendrá nota mínima en las evaluaciones del curso.





# Ejercicios

(Sí, son sin nota)



# Ejercicio 0.1

- Utilizando el **revisor - estudiante 0.1**
- Cree un .py el directorio “**estudiantes**” usando su RUN como nombre para el a programa. Ejemplo **12345678-9.py**
- Escriba el código necesario para que el programa muestre por pantalla el mensaje “hello world.”
- Vuelva a al directorio de **revisor - estudiante 0.1** y ejecute el programa **revisor 2000 - estudiante.py**
- Compruebe si ha superado los 10 test por lo indicado en la consola.
- Compruebe el archivo **Resultados.xlsx**



# Ejercicio 0.2

- Utilizando el **revisor - estudiante 0.2**
- Se le facilita la imagen de un código.
- Este utiliza contenidos que iremos aprendiendo durante el semestre.
- El objetivo del ejercicio no es que entienda cada instrucción que contiene, sino que comience a familiarizarse con la forma de escritura en Python.
- Cuando termine, compruebe si logra aprobar los test de **revisor - estudiante 0.2**



```
import random
random.seed(21)
lista = []
for i in range(1,22):
    num = str(i)
    lista.append("0"*(2-len(num))+num)

random.shuffle(lista)

print("Elija un número entero del 1 al 21.")

i = 0
while i < 3:
    lista1 = []
    lista2 = []
    lista3 = []

    while len(lista) > 0:
        lista1.append(lista.pop(0))
        lista2.append(lista.pop(0))
        lista3.append(lista.pop(0))

    print()
    print("Grupo 1:",lista1)
    print("Grupo 2:",lista2)
    print("Grupo 3:",lista3)
    op = input("Indique en qué grupo está su número (1, 2 o 3): ")

    if op == "1":
        lista = lista2 + lista1 + lista3
    elif op == "2":
        lista = lista1 + lista2 + lista3
    else:
        lista = lista2 + lista3 + lista1
    i+=1

print("Su número es:",lista[10])
```

La imagen se incluye  
en el **revisor -  
estudiante 0.2**





# ¿CONSULTAS?