



# Tendencias e Innovación en Tecnología Agrícola- TEA (CG2335-223E)

Laboratorio 04

## Capítulo 4 y 5 (Funciones e Iteración)

Estudiante: Valeria Araujo

Identificador de Github: valeriaraujo03

Directorio del Laboratorio en GitHub:  
<https://github.com/valeriaaraujo03/Tea.git>

Fecha: 29 de Septiembre del 2022.

## Parte 1: Funciones (Capítulo 4)

Sigan los ejercicios del libro.

### Ejercicio 4: ¿Cuál es la utilidad de la palabra clave “def” en Python?

- a) Es una jerga que significa “este código es realmente estupendo”
- b) Indica el comienzo de una función
- c) Indica que la siguiente sección de código indentado debe ser almacenada para usarla más tarde
- d) b y c son correctas ambas
- e) Ninguna de las anteriores

**Respuesta: b**

*Solo pongan la letra y/o coloquen un color a la respuesta correcta que se identifique fácil:*

*Ejemplo: respuesta.*

### Ejercicio 5: ¿Qué mostrará en pantalla el siguiente programa Python?

```
def fred():  
    print("Zap")
```

```
def jane():  
    print("ABC")
```

```
jane()  
fred()  
jane()
```

- a) Zap ABC jane fred jane
- b) Zap ABC Zap
- c) ABC Zap jane
- d) ABC Zap ABC
- e) Zap Zap Zap

**Respuesta: d**

*Solo pongan la letra y/o coloquen un color a la respuesta correcta que se identifique fácil:*

*Ejemplo: respuesta.*

### Ejercicio 6: Reescribe el programa de cálculo del salario, con tarifa-ymedia para las horas extras, y crea una función llamada calculo\_salario

**que reciba dos parámetros (horas y tarifa).**

Introduzca Horas: 45

Introduzca Tarifa: 10

Salario: 475.0

```

lab04 > Ejercicio5.py > ...
1 def calcularSalario(horas, tarifa):
2     horas_extras = horas - MAX_HORAS_SEMANALES
3     if(horas_extras>0):
4         pago= (MAX_HORAS_SEMANALES* tarifa)+ (horas_extras*(tarifa*1.5))
5     else:
6         pago= horas*tarifa
7     return pago
8
9
10 try:
11     MAX_HORAS_SEMANALES= 40
12     horas= int(input("Ingrese número de horas: "))
13     tarifa= float(input("Ingrese tarifa por hora: "))
14     salario= calcularSalario(horas, tarifa)
15     print(salario)
16 except:
17     print("Error, debe ingresar un valor numérico")
  
```

Terminal output:

```

PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB> & C:/Users/valeria.araujo/AppData/Local/Microsoft/WindowsApps/python3.10.exe "c:/Users/valeria.araujo/OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA/LAB/lab04/Ejercicio5.py"
Ingrese número de horas: 45
Ingrese tarifa por hora:10.60
503.5
PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB>
  
```

**Ejercicio 7: Reescribe el programa de calificaciones del capítulo anterior usando una función llamada `calcula_calificacion`, que reciba una puntuación como parámetro y devuelva una calificación como cadena.**

Puntuación Calificación

> 0.9 Sobresaliente

> 0.8 Notable

> 0.7 Bien

> 0.6 Suficiente

<= 0.6 Insuficiente

Introduzca puntuación: 0.95

Sobresaliente

Introduzca puntuación: perfecto

Puntuación incorrecta

Introduzca puntuación: 10.0

Puntuación incorrecta

Introduzca puntuación: 0.75

Bien

Introduzca puntuación: 0.5

Insuficiente

Ejecuta el programa repetidamente para probar con varios valores de entrada diferentes.

The screenshot shows the Visual Studio Code editor with the file explorer on the left displaying a project named 'LAB'. The file 'ejer.py' is selected. The main editor window shows the following Python code:

```
1 def calcular_calificacion(puntuacion):
2     if puntuacion > 1.00:
3         print("Puntuación incorrecta")
4     elif (puntuacion)>=0.90 and puntuacion <=1.0):
5         print("Sobresaliente")
6     elif (puntuacion)>=0.80 and puntuacion <0.9):
7         print("Notable")
8     elif (puntuacion)>=0.70 and puntuacion <0.8):
9         print("Bien")
10    elif (puntuacion)>=0.6 and puntuacion <0.7):
11        print("Suficiente")
12    elif (puntuacion)>=0 and puntuacion <0.6):
13        print("Insuficiente")
14    elif puntuacion >1.00:
15        print("Puntuación incorrecta")
16    else:
17        print("Calificación fuera de rango, no válida")
18    return puntuacion
19
20 try:
21     puntuacion= float(input("Introduzca puntuación: "))
22     calificación= calcular_calificacion(puntuacion)
23     print("Su puntuación es: ", puntuacion)
24 except:
25     print("Error, puntuación es solo numérica")
```

The bottom status bar shows the file path: PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB>.

The screenshot shows the same Visual Studio Code editor with the 'ejer.py' file. The terminal window at the bottom displays the output of the program:

```
PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB> python3.10.exe "c:/Users/valeria.araujo/OneDrive - Zamorano/Docs/SEGUNDO AÑO/TERCER PERÍODO/ELECTIVA- TEA/LAB/lab04/ejer.py"
Introduzca puntuación: 0.98
Sobresaliente
Su puntuación es: 0.98
PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB> python3.10.exe "c:/Users/valeria.araujo/OneDrive - Zamorano/Docs/SEGUNDO AÑO/TERCER PERÍODO/ELECTIVA- TEA/LAB/lab04/ejer.py"
Introduzca puntuación: 50
Puntuación incorrecta
Su puntuación es: 50.0
PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB>
```

The status bar at the bottom shows the file path: PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB>.

The screenshot shows the Visual Studio Code interface. The Explorer pane on the left lists files in a 'LAB' folder, including 'ejer.py'. The main editor displays the code for 'ejer.py', which defines a function 'calcular\_calificacion' that takes a score and prints a grade based on ranges. The bottom panel shows the 'TERMINAL' output, where the script is executed with a score of 0.10, resulting in the output 'Insuficiente'. The status bar at the bottom indicates the file is 'ejer.py' and the Python version is 3.10.7.

```

1 def calcular_calificacion(puntuacion):
2     if puntuacion > 1.00:
3         print("Puntuación incorrecta")
4     elif (puntuacion >= 0.90 and puntuacion <= 1.0):
5         print("Sobresaliente")
6     elif (puntuacion >= 0.80 and puntuacion < 0.9):
7         print("Notable")
8     elif (puntuacion >= 0.70 and puntuacion < 0.8):
9         print("Bien")
10    elif (puntuacion >= 0.6 and puntuacion < 0.7):
11        print("Suficiente")
12    elif (puntuacion >= 0 and puntuacion < 0.6):
13        print("Insuficiente")
14    elif puntuacion > 1.00:
15        print("Puntuación incorrecta")
  
```

```

PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB> &
C:\Users\valeria.araujo\AppData\Local\Microsoft\WindowsApps\python3.10.exe "c:/Users/valeria.araujo/OneDr
ive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB\lab04\ejer.py"
Introduzca puntuación: 0.10
Insuficiente
Su puntuación es: 0.1
PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB> &
C:\Users\valeria.araujo\AppData\Local\Microsoft\WindowsApps\python3.10.exe "c:/Users/valeria.araujo/OneDr
ive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB\lab04\ejer.py"
Introduzca puntuación: Perfecta
Error, puntuación es solo numérica
PS C:\Users\valeria.araujo\OneDrive - Zamorano\Documents\SEGUNDO AÑO\TERCER PERÍODO\ELECTIVA- TEA\LAB>
  
```

## Parte 2: Iteración (Capítulo 5)

**Ejercicio 1:** Escribe un programa que lea repetidamente números hasta que el usuario introduzca “fin”. Una vez se haya introducido “fin”, muestra por pantalla el total, la cantidad de números y la media de esos números. Si el usuario introduce cualquier otra cosa que no sea un número, detecta su fallo usando try y except, muestra un mensaje de error y pasa al número siguiente.

```

Introduzca un número: 4
Introduzca un número: 5
Introduzca un número: dato erróneo
Entrada inválida
Introduzca un número: 7
Introduzca un número: fin
16 3 5.333333333333
  
```

*Incluya la captura de pantalla acá. La captura de pantalla **debe mostrar el código y el programa corriendo en VSC (Visual Studio Code)** ya sea utilizando el debugger o la terminal.*

**Ejercicio 2: Escribe otro programa que pida una lista de números como la anterior y al final muestre por pantalla el máximo y mínimo de los números, en vez de la media.**

### Parte 3: Laboratorio

¿Cuál es el url/link/enlace del directorio creado en Github? ¿A dónde se subieron los archivos de este laboratorio?

Ejemplo: <https://github.com/valeriaaraujo03/Tea.git>

Suba únicamente este documento a Blackboard.