

Unidad 1. Lógica de algoritmos, estructuras e introducción a Python

Taller entregable 1

2024-2

Es requerido que todos los códigos hechos utilicen ciclos y condicionales para resolver lo pedido, no está permitido utilizar funciones de librerías importadas.

El trabajo podrá ser enviado en parejas, al enviar el trabajo no olviden mencionar a su compañero, en caso contrario se le calificará como si hubiera sido entregado por la persona que subió el archivo únicamente.

Se recomienda hacer entrega del informe en un archivo formato ipynb utilizando el markdown como textos por facilidad de calificación, u opcionalmente mandar el informe en pdf y el código en .py.

Entregable

Objetivo: Practicar el uso de listas, condicionales, ciclos y claridad sobre los tipos de dato y conceptos de programación en Python.

Procedimiento:

Los puntos 1-3 son preguntas/análisis, el 4 es un código

- 1) Responder las siguientes preguntas (15%, 5% cada uno):
 - ¿Qué diferencia hay entre un ciclo *for* y uno *while*? De un ejemplo de un caso donde sea preferible usar el ciclo *for* y uno donde sea preferible usar el ciclo *while*.
 - ¿Qué es un booleano? De ejemplos en Python de booleanos (Al menos 3)
 - ¿Cuál es el tipo de dato que la función *input* que es nativa de Python entrega al ingresar por consola la información? Si se quiere cambiar este tipo de dato ¿Cómo se hace?

- 2) Verdadero o falso y justificar (No tiene que ser una justificación larga) (10%, 2.5% cada uno):
 - Para Python, un True puede ser cualquier String o número diferente de 0.
 - En Python, no se pueden sumar un entero con un flotante.
 - En Python el tipo de dato *Lista* es mutable y ordenado (Mantiene el orden de los datos ingresados).

- En Python, los strings son similar a una lista de caracteres, y utilizando la indexación puedes acceder a cada carácter por separado.

3) Observe el siguiente código y responda las preguntas preferiblemente sin correr el código en Python (25%, 5% por pregunta):

Parte 1

```
list = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
```

```
for i in list:
```

```
    square = i**2
```

```
    print(square)
```

Parte 2

```
list = [1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

```
count = 0
```

```
while count < len(list):
```

```
    squareRoot = list[count]**(1/2)
```

```
    print(squareRoot)
```

```
    count+=1
```

- ¿Qué función hace el operador “**”?
- ¿Qué imprime la parte 1? No los números, diga en sus palabras que imprimirá y el porqué.
- ¿Qué imprime la parte 2? No los números, diga en sus palabras que imprimirá y el porqué.
- En la parte 1 ¿Qué función hace la variable “i”?
- En la parte 2 ¿Cuándo y porque se detiene el ciclo while?

4) Realice un código que pueda realizar lo siguiente (50%):

Se requiere estimar la tasa de filtración glomerular [ml/min] de un grupo de pacientes, la formula de Cockcroft-Gault puede estimar el aclaramiento de creatinina, que a su vez estima la TFG de la siguiente manera:

Hombres:

$$TFG = \frac{(140 - \text{edad en años}) \times \text{peso en kg}}{72 \times \text{creatinina sérica en mg/dL}}$$

Mujeres:

$$TFG = \frac{(140 - \text{edad en años}) \times \text{peso en kg} \times 0.85}{72 \times \text{creatinina sérica en mg/dL}}$$

Se desea entonces que se pueda ingresar los datos de los pacientes y posteriormente estimar su tasa de filtración glomerular y guardar en una lista todos los valores obtenidos e indicar si se encuentra en un rango normal (Se tomara para este caso entre 75 y 130, no considerar esto real, pues el valor normal varia con la edad), para posteriormente imprimir todos los datos calculados junto con el nombre del paciente al final del código, tener en cuenta también las unidades de cada uno de los valores ingresados y resultantes.

Cualquier duda por favor escribir al monitor andres.banquez@udea.edu.co