**RETO 4 – SEMANA 4**

**FUNDAMENTOS DE PYTHON**

**Deissy Cancino Joya**

**Bairon Gomez Peñuela**

**Universidad de Antioquia**

**Medellín**

**2023**

**RETO 4**

1. ¿Pueden existir funciones dentro de funciones?
2. ¿Se puede enviar como parámetro de una función otra función con retorno?
3. ¿Qué sucede si dentro de una función se hace un llamado a sí misma?
4. ¿Una variable creada como global dentro de una función, ¿sigue existiendo fuera de la función?
5. Después de indicar una variable en una función como global, ¿hay alguna manera de volver a usar la variable local?

**Respuestas:**

1. Sí es posible la definición de funciones dentro de otras funciones, son conocidas como funciones anidadas o funciones internas.
2. Sí puede realizarse y se les conoce como funciones de orden superior.
3. Sí es posible y se le llama “recursión”. Cuando pasa esto, debemos tener cuidado de lo que colocamos como return o la salida, porque podemos crear recusiones infinitas. Utilizamos esto en la función de factorial de la calculadora.
4. Si declaramos una variable como global dentro de una función, esa variable se convierte en una variable global y podríamos acceder a esa variable desde cualquier parte del código, tanto dentro como fuera de la función que definimos.
5. No es posible que después de declarar una variable como global, luego la utilicemos como local.

**Punto 1.** Tome el enunciado y la solución elaborada del reto 3, identifique si aplicó el paradigma funcional, si lo aplicó correctamente, si faltaron funciones o pudieron realizarse otras. Construya un documento que recopile las falencias, aciertos, dudas y dificultades que se le presentaron al aplicar los conceptos del paradigma funcional a la solución del reto 3.

En caso de no haber aplicado el paradigma funcional en el reto 3, analice cómo pudiera aplicarlo, qué funciones haría y qué pasos debería seguir para crear la solución bajo este paradigma, y construya un documento tipo texto con los resultados de este análisis y sugerencias, de tal forma que un programador pudiera realizar dicha implementación bajo el paradigma funcional.

**Respuesta:**

El código que hemos desarrollado no se adhiere completamente al paradigma funcional. A pesar de contar con algunas funciones definidas, también incorpora elementos como bucles for, condicionales if y un bucle while, que no están encapsulados dentro de funciones.

Aunque hemos creado algunas funciones para realizar cálculos y utilizamos argumentos y valores de “return” para obtener resultados, aún persisten operaciones de entrada a través de la función input(). Además, utilizamos variables como listas (gastos\_totales, costos\_totales, costos\_mdo\_total) que se modifican dentro de bucles for.

Para que un programa sea considerado completamente funcional, debería minimizar los efectos secundarios, como la modificación de variables mutables. Es importante centrarse en la composición de funciones puras. Sin embargo, en este caso, necesitamos la interacción con el usuario para obtener ciertos valores, lo que impide que el código sea completamente funcional en su totalidad.

Para mejorar el código y acercarlo a un estilo más orientado al paradigma funcional, podríamos considerar organizar los menús dentro de funciones específicas. De esta manera, podríamos llamar estas funciones en cualquier momento y evitar que se regrese al inicio del código en cada iteración. Además, siempre que sea posible, debemos evitar la modificación de variables después de su inicialización y preferir la creación de nuevas variables en lugar de modificar las existentes.

En el punto 2, crearemos la calculadora con menús con funciones definidas que sean llamadas por el menú principal, para ejemplificar este último punto del regreso al menú principal.