[[1]](#footnote-1)

**CALCULADORA DE FIGURAS GEOMÉTRICAS (octubre de 2024)**

Autores: Torres Pachón Juan Esteban, González Gonzáles Dinalut y Ruiz Rincon Jessica Jineth estudiantes de la fundación universitaria de San gil Unisangil (SEDE-CHIQUINQUIRA)

***Resumen*—En el presente documento hemos diseñado un algoritmo en Python, usando la programación modular para crear una calculadora de figuras geométricas que calcule el área de un cuadrado, un triángulo, un círculo, un pentágono, un trapecio, un romboide, un rombo y un rectángulo. Por lo que aplicamos una metodología en la que establecimos los requisitos y limitaciones del algoritmo. Además, utilizamos PSeInt para crear un pseudocódigo que luego implementamos en Python. Al ejecutar el programa logramos identificar que el algoritmo calcula de forma precisa el área de las figuras geométricas.**

**Palabras clave-programación modular, calculadora, área, figuras geométricas, Python. Programación**

**INTRODUCCIÓN**

 En este documento se busca diseñar un algoritmo que cree una calculadora usando programación modular, que permita calcular el área de las figuras geométricas teniendo en cuenta las diferentes formas de calcularla dependiendo de la figura y de los datos proporcionados por el usuario. Para ello es necesario aplicar una metodología que identifique y calcule los datos necesarios para la implementación de un lenguaje de programación como Python.

1. ***ANÁLISIS***

***Contexto***

Desarrollar un algoritmo para crear una calculadora usando la programación modular que halle el área de las figuras geométricas (cuadrado, triángulo, círculo, pentágono, trapecio, romboide, rombo y rectángulo).

*B. Población*

Usuarios que necesiten calcular el área de figuras geométricas.

***C. Limitaciones y alcance***

Las limitaciones y alcances que se pudieron identificar son las siguientes:

**Limitaciones**

* Datos necesarios para calcular el área de las diferentes figuras geométricas.

**Alcances**

* El usuario podrá ingresar los datos requeridos.
* Se calculará el área de las diferentes figuras geométricas.

**III. OBJETIVOS**

***Objetivo general***

Diseñar un algoritmo para crear una calculadora, usando programación modular, que halle el área de figuras geométricas donde el usuario pueda ingresar los datos necesarios.

***Objetivos específicos***

* Solicitar al usuario la figura geométrica de la que desea hallar el área.
* Solicitar al usuario los datos necesarios para hallar el área.
* Mostrar al usuario el área de la figura geométrica con precisión.
* Determinar el lenguaje de programación que se va a usar para crear el algoritmo.
* Presentar al usuario un programa que pueda manejar de forma comprensible.

IV. **ESPECIFICACIÓN DE REQUERIMIENTOS.**

***CON RESPECTO AL DISPOSITIVO***

A continuación, se indican los requerimientos mínimos:

* Mínimo 4 GB de RAM.
* Sistema operativo: Windows 8 en adelante.
* CPU: Intel Core i3.
* Disco: HDD o SSD.

1. ***CON RESPECTO AL USUARIO.***

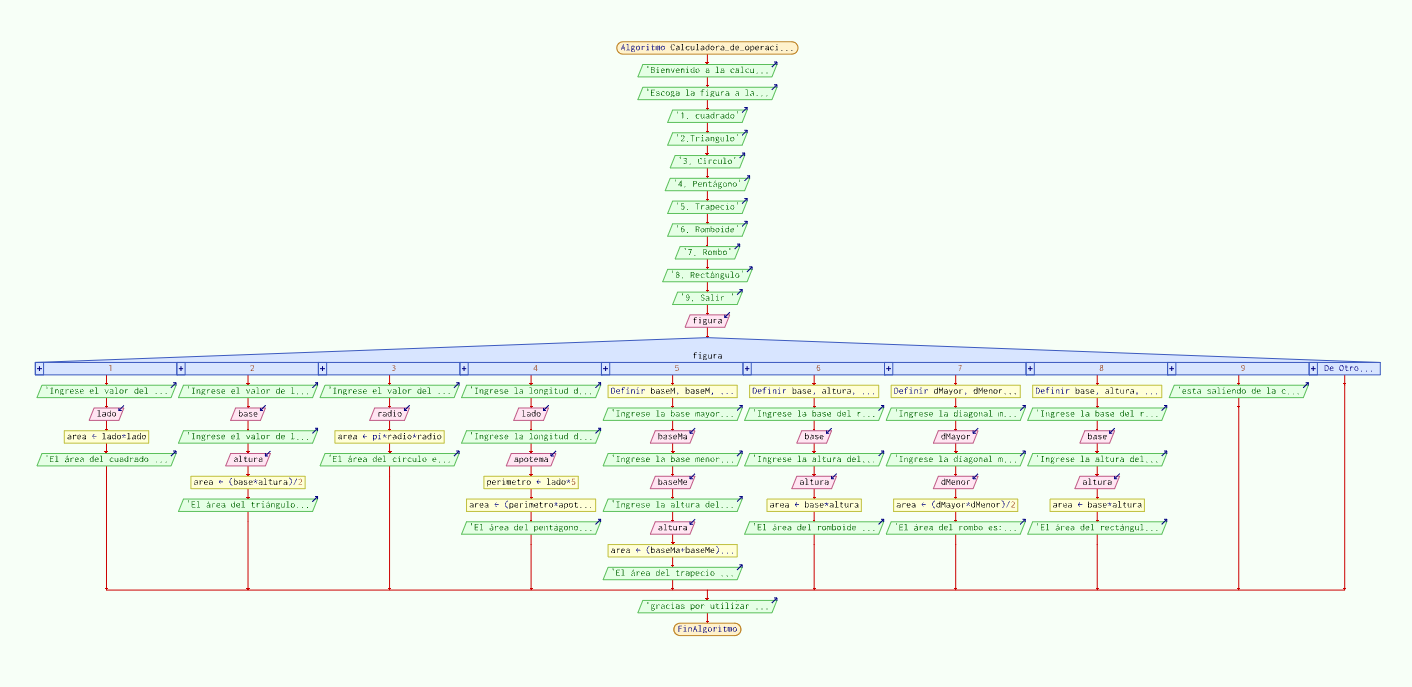
*Requisitos mínimos*

* *El usuario debe elegir la figura geométrica de la que desea calcular el área.*
* *El usuario debe ingresar los datos necesarios para hallar el área de la figura geométrica*
* *El usuario debe contar con una necesidad seleccionada, la cual es calcular el área de una figura geométrica.*

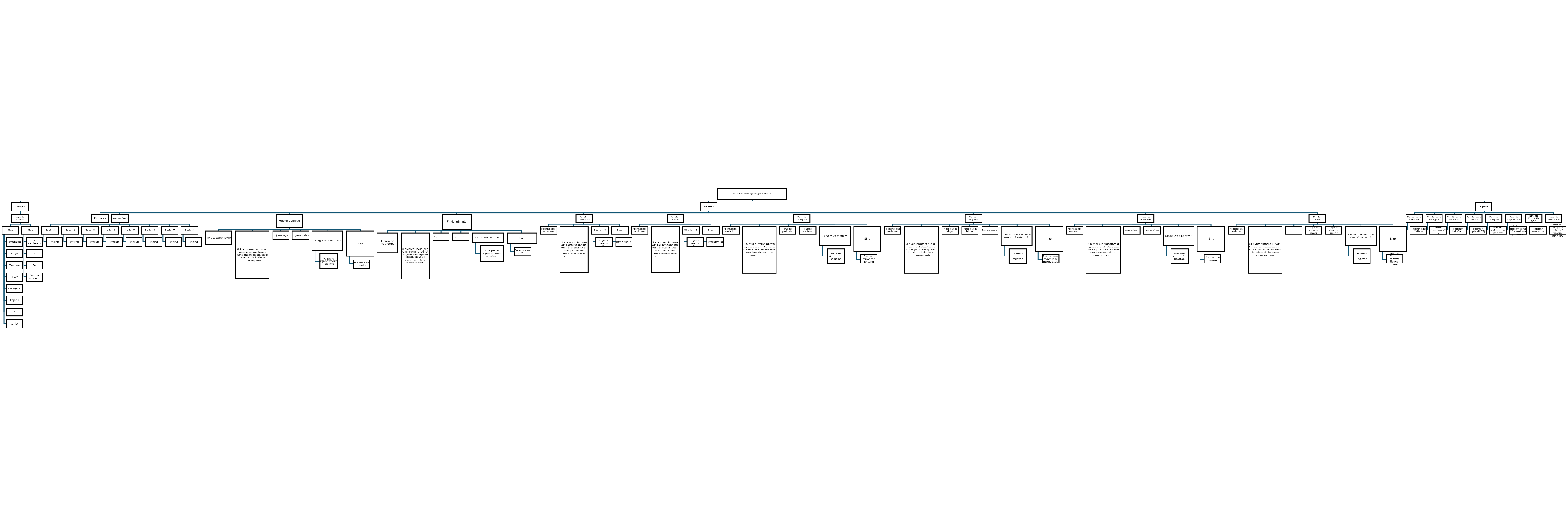
**V. DISEÑO DEL ALGORITMO**

Utilizamos un programa generador de pseudocódigo denominado PSeInt, el cual utilizamos como guía, para luego usar el lenguaje de programación Python, siendo este el estructurador del software requerido.

*Imagen 1. Pseudocódigo.*



**Programación modular:**



*En esta imagen. Se muestra el prototipo principal para tener en cuenta para el diseño del código fuente*

VI.CODIFICACIÓN.

 Para la codificación del algoritmo utilizamos los siguientes programas:

* PSeInt.
* Visual Studio Code.
* Python
* Git hub.

***A. Instrucciones del código fuente***

* 1. Crear una carpeta “calculadora de figuras geométricas”. que contendrá los archivos del código de la calculadora. (principal, interfaz y figuras).
  2. En interfaz creamos el código fuente de la calculadora.
  3. Código: la función def para crear funciones
  4. La variable “menú” contiene la lista de las figuras geométricas a elegir.
  5. While para crear ciclos y poder retornar algunas funciones
  6. La variable “opción” definirá la opción digitada por el usuario mediante un input (), el cual permite capturar datos de la consola.
  7. try para procesar la opción ingresada por el usuario
  8. if y elif para el proceso de hallar el área dependiendo de las diferentes formas de hallarla en cada opción ingresada por el usuario.
  9. import math necesaria para el uso de la constante PI
  10. print para mostrar el resultado de cada área

***B. Código fuente***

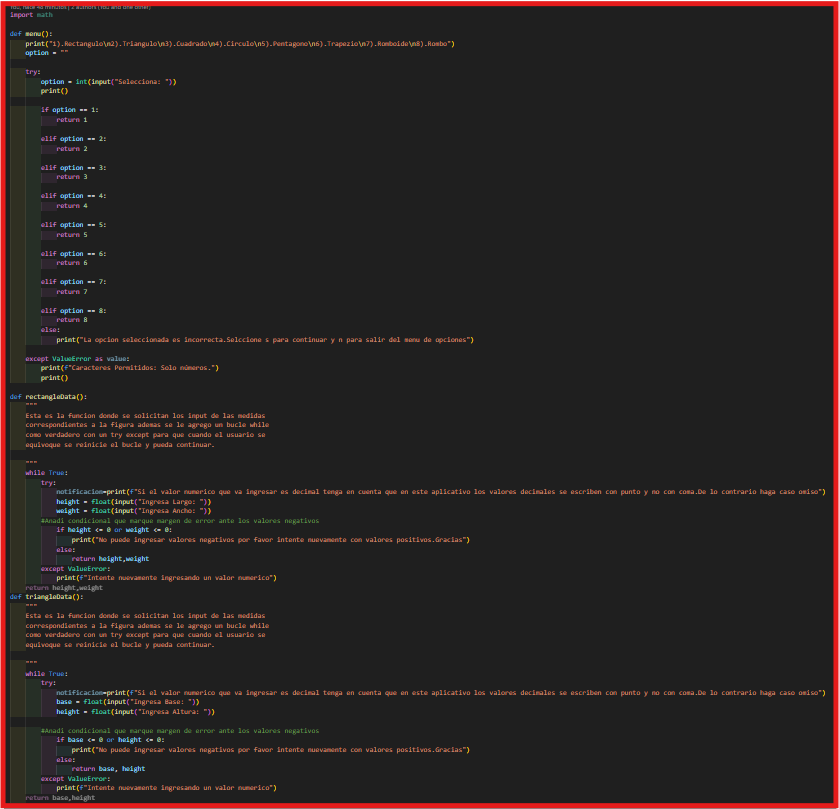
***VERSION 3.0***

**Imagen. 2. Código fuente.**

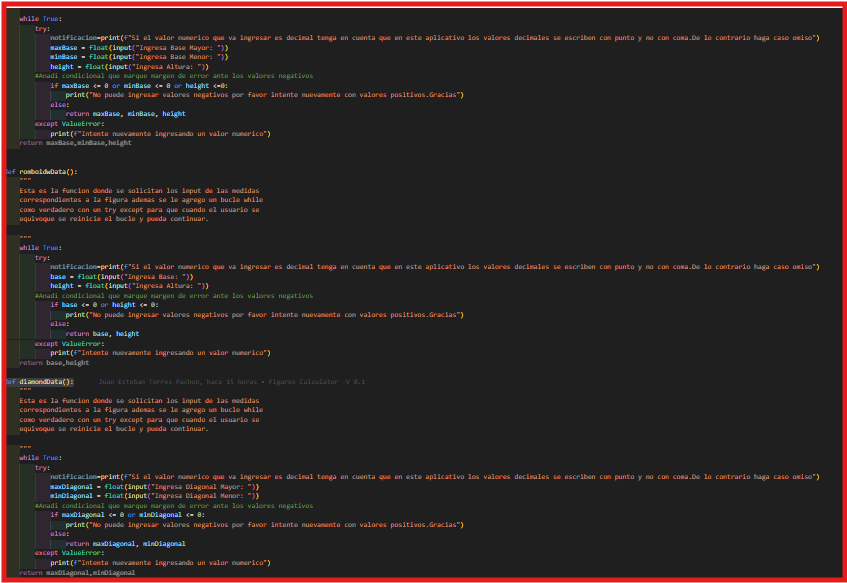
**Principal**



**Interfaz:**

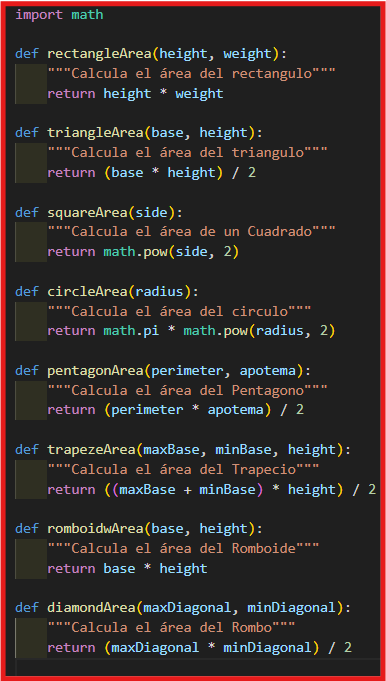






*Imagen. 2: En esta imagen se muestra el diseño del algoritmo donde se utiliza el lenguaje de programación Python para ejecutarlo.*

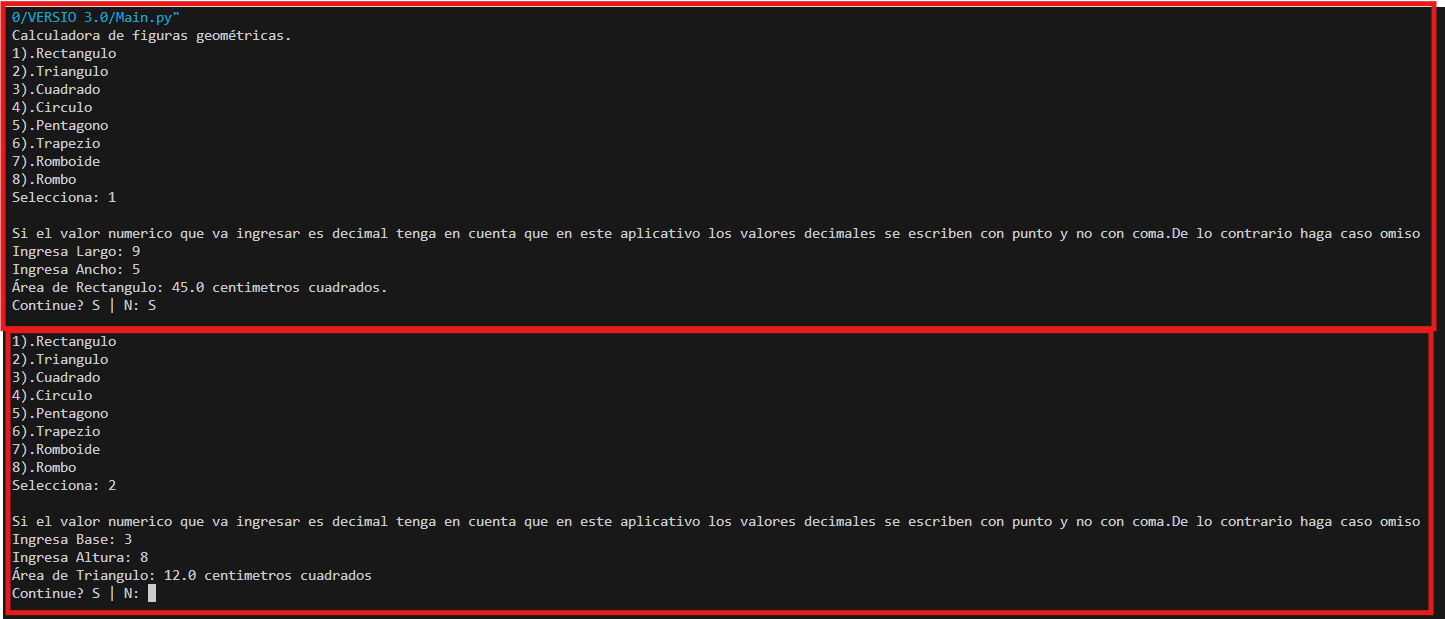
**Figuras**



***VII. EJECUCIÓN DEL PROGRAMA.***

En este caso comenzamos con la apertura de la terminal de Git Bash. Si ya se creó el repositorio remoto y se sincronizan las cuentas de los desarrolladores con el repositorio local, se envía el archivo al Staging área con el comando git add.  Luego ejecutamos el git commit -m “Comentario” para confirmar los cambios, donde, posteriormente usamos git push origin o pb master o main dependiendo la rama que se esté usando con el fin de almacenar la información en el repositorio remoto de GitHub.

I*magen. 3. Código ejecutado.*



*Nota: en esta imagen se muestra que efectivamente el algoritmo funciona cuando el usuario digita los datos requeridos*

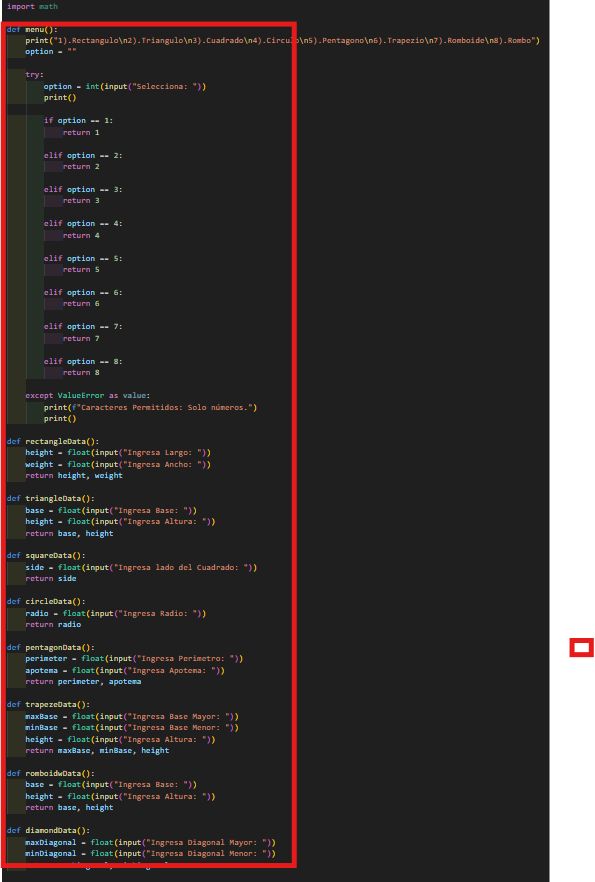
**IX VERIFICACIÓN Y DEPURACIÓN.**

En primer lugar, adjunto la versión 1.0 del algoritmo al cual se la harán las verificaciones y depuraciones:

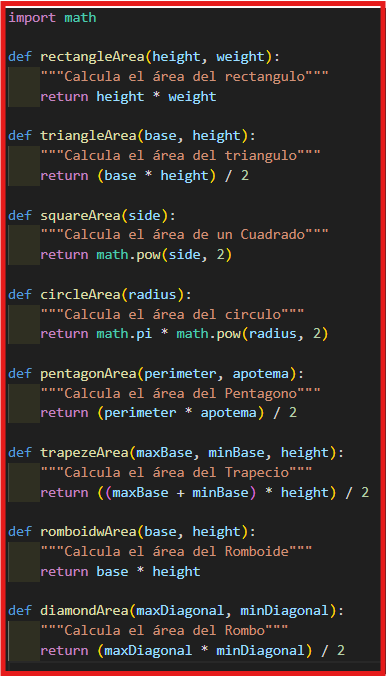
El código es un script el cual está conformado por una principal, interfaz y figuras:



**Principal**



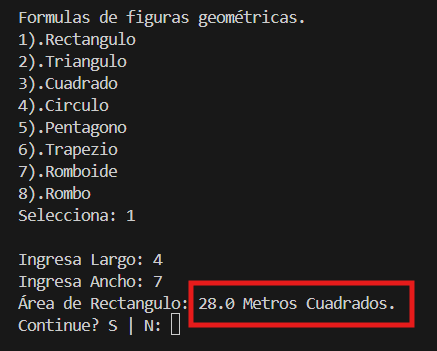
**Interfaz.**

****

**Figuras.**

**Primera prueba parcial:**

Al compilar el código y ejecutar el código con los valores se identifica que este funciona correctamente, pero a la hora de indicar la magnitud nos muestra como metros cuadrados como se muestra en la imagen en las demás no hay una magnitud indicada, cuando la calculadora está diseñada para centímetros cúbicos.

****

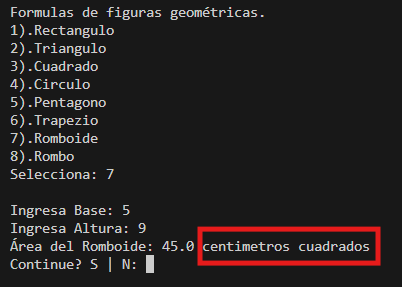
Por lo que tuvimos que modificar en el código principal del script, el print y en vez de metros cuadrados pone centímetros a la primera figura y las restantes.



por lo que se hizo el cambio respectivo con respecto a las magnitudes:

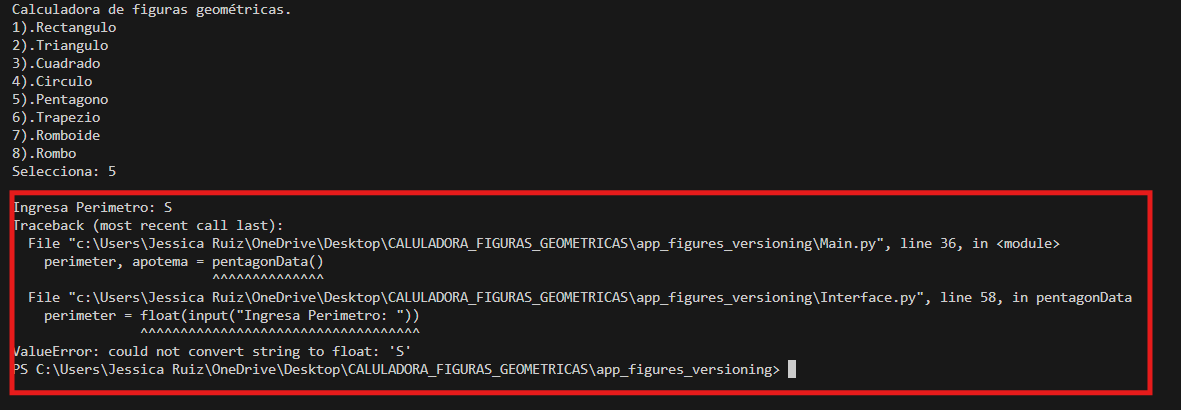
****

**Ejecución:**

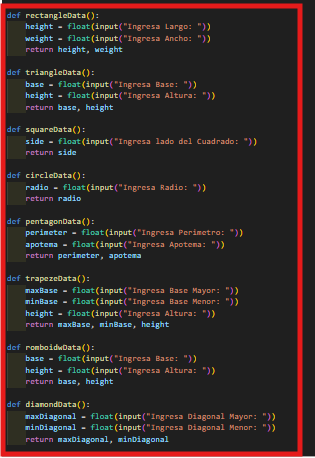


**Segunda prueba parcial:**

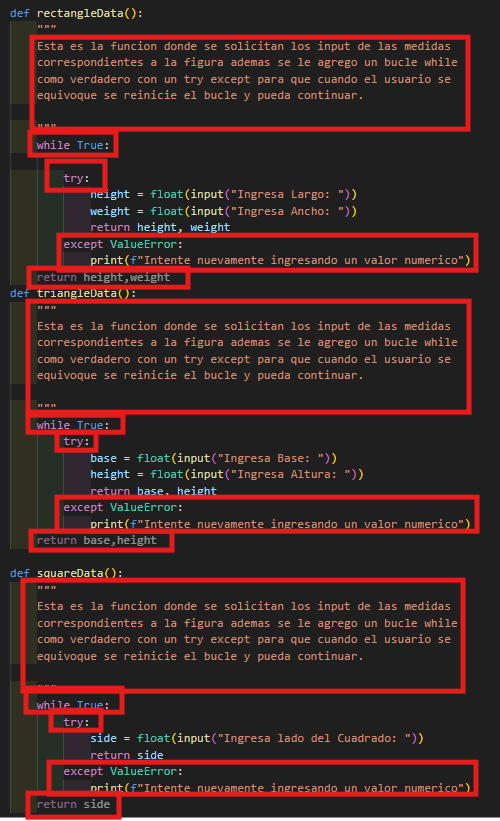
El usuario selecciono la opción de la figura a la cual le desea hallar el área, pero al momento de poner la medida por accidente colocolo una letra y no un numero por lo que genero un margen de error dentro del código:



en este caso identificamos que es necesario poner un condicional en el que si el usuario comete este error se pueda retractar de tal forma que pueda continuar el respectivo proceso por lo que se hicieron modificaciones en la interfaz del código, por lo que a continuación se muestra en la imaginen como se tenía el código y en la siguiente las modificaciones



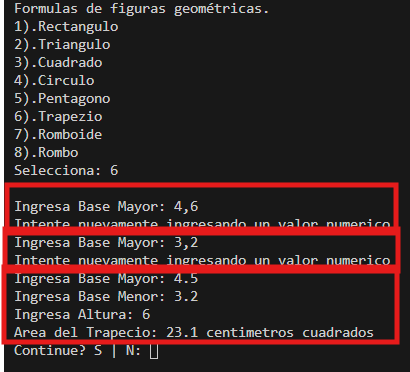
En este caso tomamos cada función como se muestra en la imagen y como alternativa pusimos un bucle while dentro del cual se llamó el try-except donde el try intenta convertir los valores en un numero por lo que si el usuario ingresa un valor numérico el código funciona correctamente, por otra parte, si el usuario ingresa letras o textos excepto generará una excepción con el valué error y volverá a reiniciar el bucle.

****

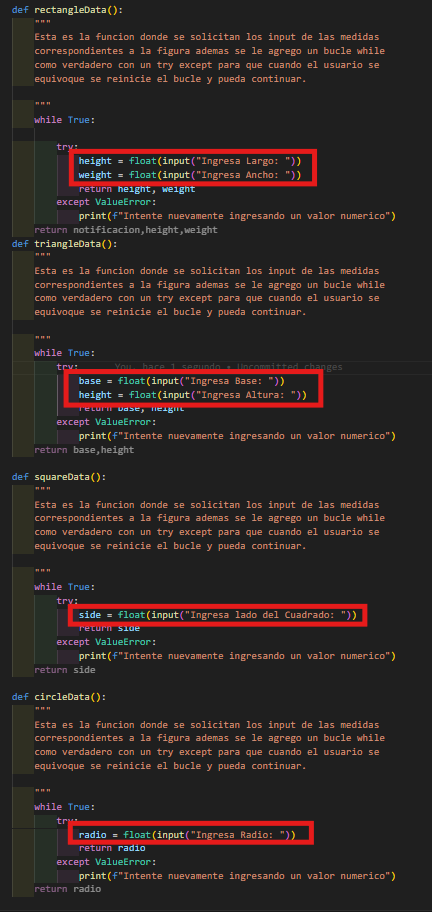
**Tercera prueba parcial:**

Generalmente en Colombia las personas utilizamos la coma para indicar un valor decimal en Python para representar un valor decimal se utiliza el punto por lo que si una persona ingresa un valor decimal con la coma se le pedirá que ingrese nuevamente un valor numérico ya que no lo reconoce como tal en este caso la solución que es indicarle al usuario por medio de un print que si la medida que desea ingresar es decimal en vez de usar una coma ponga un punto.

A continuación, se muestra el problema identificado:



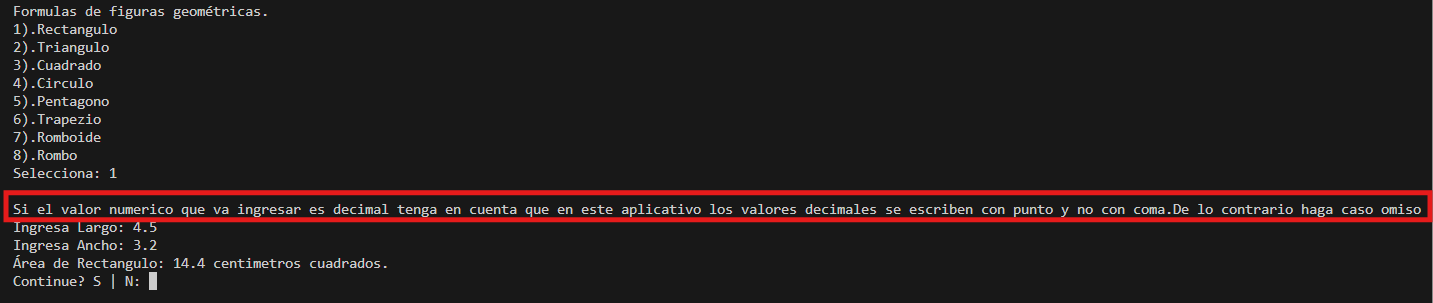
A continuación, se muestra la versión anterior del código:



por lo que creamos una variable dentro de cada función la cual tenemos que llamar desde la parte principal en este caso la variable se denomina notificación:

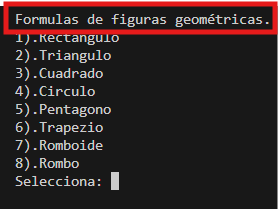


**Ejecución:**

****

**Cuarta prueba parcial:**

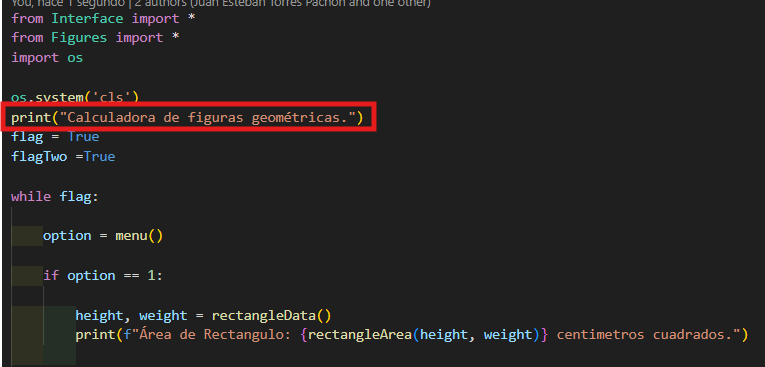
También se identificó que al compilar los datos en el primer print dice formula de figuras geométricas cuando debería decir calculadora de figuras geométricas:



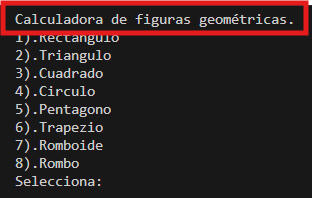
A continuación, se muestra la versión anterior del código:



Modificación del print en la escritura de la frase:

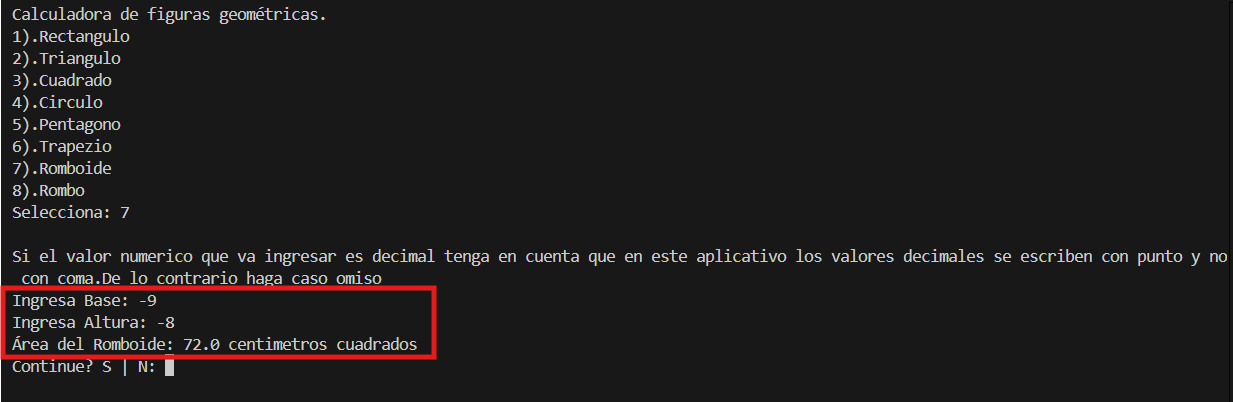


**Ejecución:**

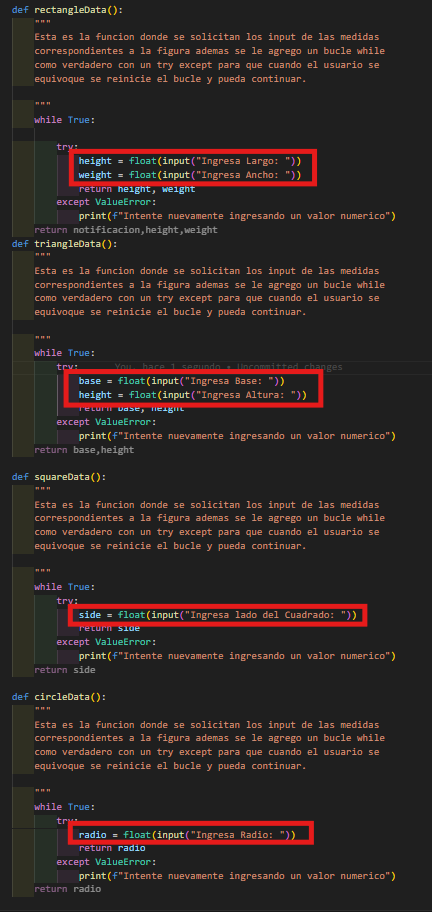
****

**Quinta prueba parcial:**

Generalmente para hallar el área de una figura geométrica se insertan valores positivos teniendo en cuenta las operaciones básicas. En caso de que el usuario quiera ingresar valores negativos la aplicación debería denegarles la opción ya que si el usuario quiere hallar el área con valores negativos seria aplicado en calculo integral o el plano cartesiano por lo que la calculadora está diseñada solamente para hallar el área de forma básica.



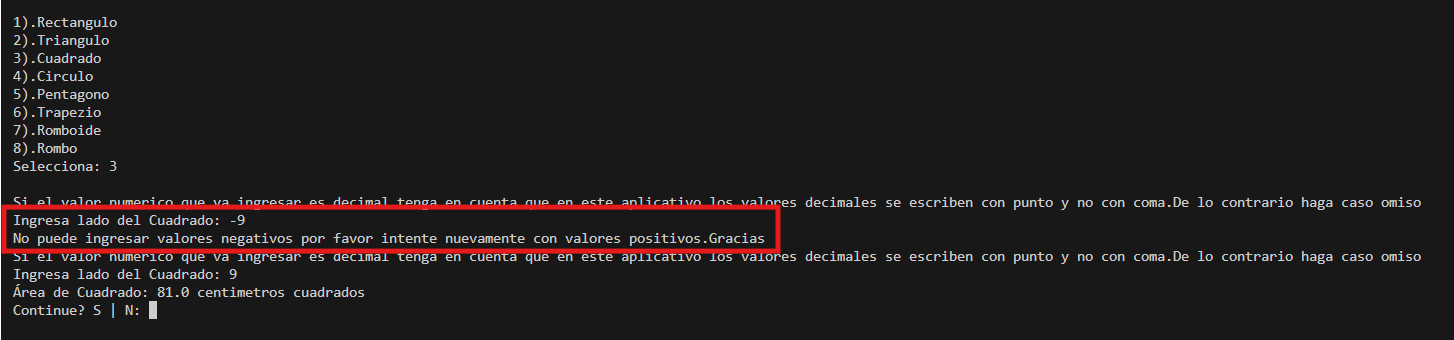
A continuación, se muestra la versión anterior del código



Se le añadió al código un condicional if para que no permita la ejecución de la operación si el valor es negativo.

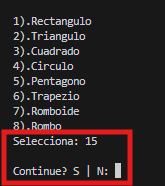


Ejecución:



**Sexta prueba parcial**

Se ingreso un valor que no corresponde a los valores correspondientes del menú para seleccionar una opción por lo que el código retorna el si el usuario desea continuar o no sin indicar in mensaje donde le diga al usuario que la opción seleccionada es incorrecta:

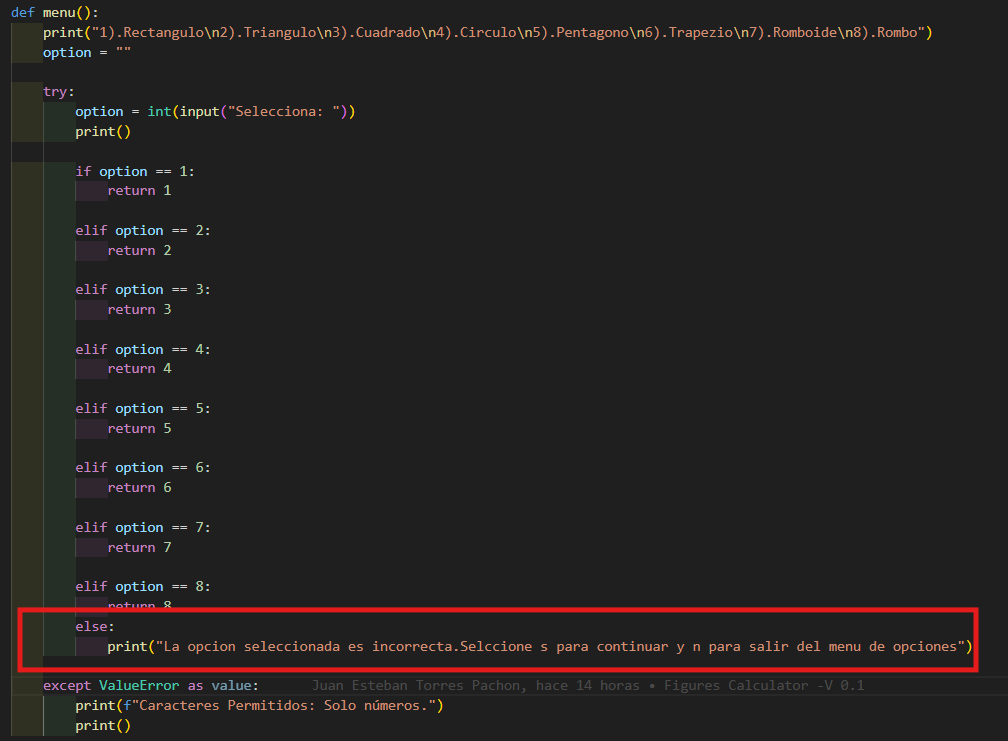


A continuación, se muestra la versión anterior del código:

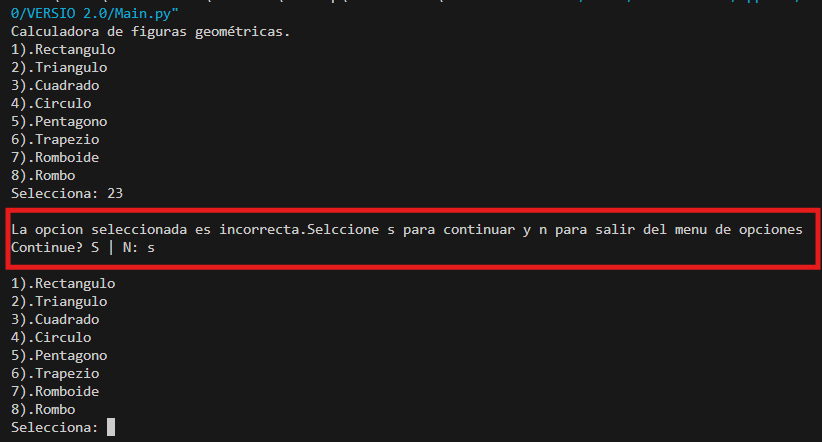
Texto

Descripción generada automáticamente

Por lo se le añadió un else al final del if para que cuando el usuario ingrese una opción no correspondiente le indique:

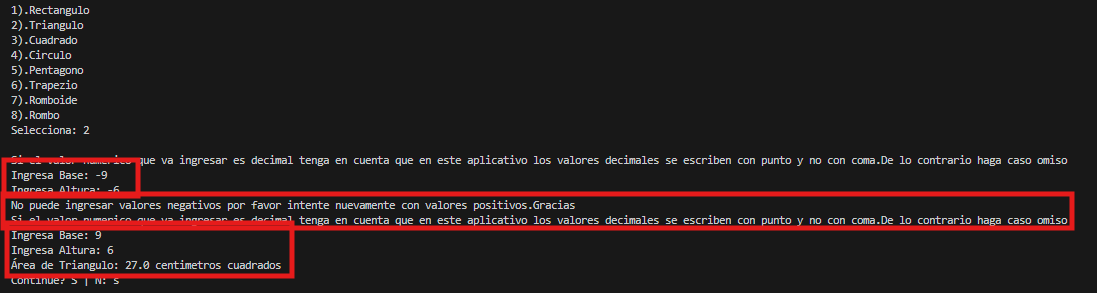


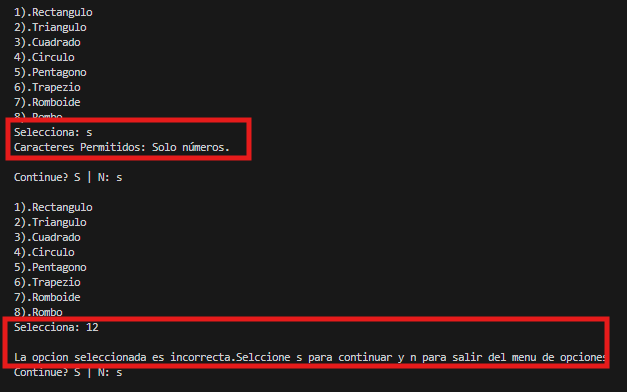
**Ejecución:**

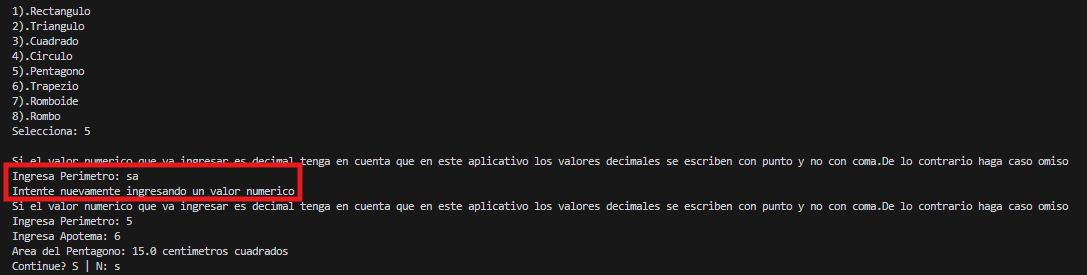
****

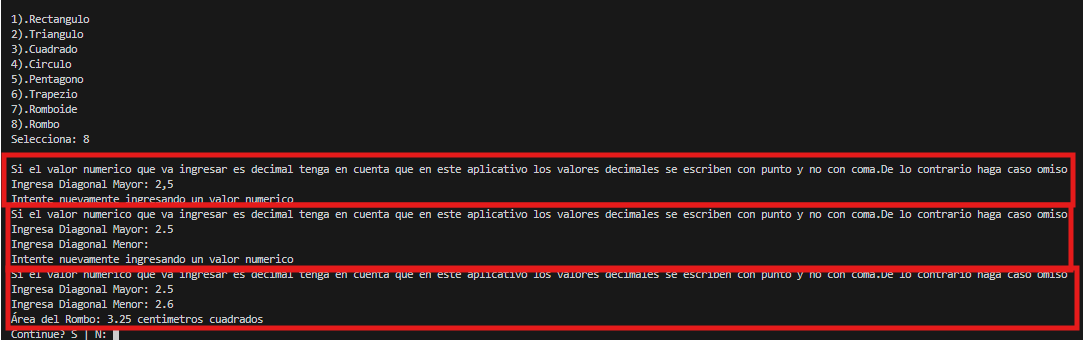
**Séptima, Octava, Novena y Décima prueba parcial**

Se hicieron varias pruebas buscando posibles errores, pero ante las correcciones anteriores se considera que ya no es necesario hacer más depuraciones.



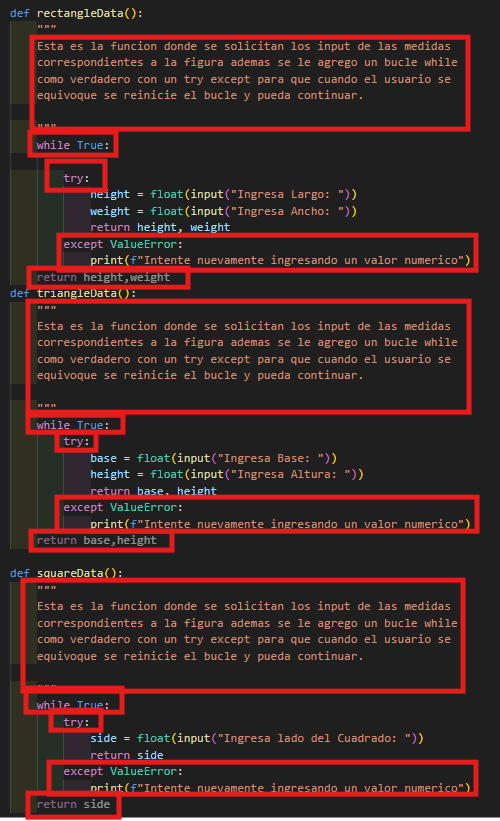




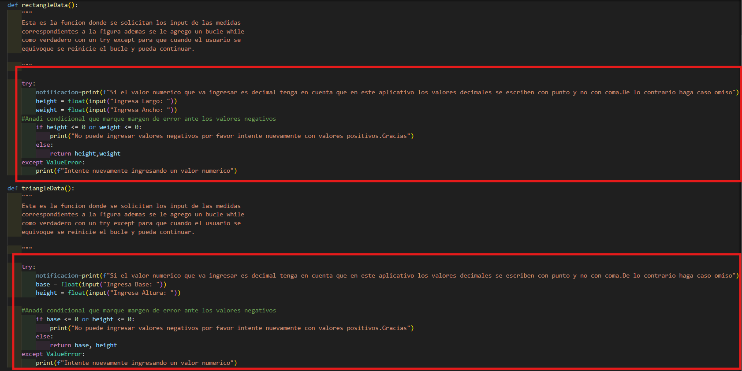
****

**Prueba parcial 11**

**Se identifico que es innecesario colocar un While dentro de cada función ya que esto no optimiza su proceso y se puede hacer en el código principal por lo que omitimos el While**

****

**MODIFICACION**

****

**X DOCUMENTACIÓN.**

1. ***Interna.***

*Toda la información relacionada con el desarrollo del código se encuentra almacenada en un repositorio en GitHub.*

*Enlace:* [*https://github.com/Zheilon/app\_figures\_versioning.git*](https://github.com/Zheilon/app_figures_versioning.git)

1. ***Técnica.***

***Diccionario de datos:***

1. *Funciones*
   1. *Figuras donde se almacena las formulas*
      1. *def rectangleArea (height, weight)*
      2. *def triangleArea (base, height)*
      3. *def squareArea(side)*
      4. *def circleArea(radius)*
      5. *def pentagonArea (perimeter, apotema)*
      6. *def trapezeArea (maxBase, minBase, height)*
      7. *def romboidwArea (base, height)*
      8. *def diamondArea (maxDiagonal, minDiagonal)*
   2. *Principal Se imprimetn las funciones delcodigo.*
      1. *Option cada opción imprime una figura.*
   3. *Interfaz se almacena la lógica de la calculadora*
      1. *def menu ()*
      2. *def rectangleData ()*
      3. *def triangleData ()*
      4. *def squareData ()*
      5. *def circleData ()*
      6. *def pentagonData ()*
      7. *def trapezeData ()*
      8. *def romboidwData ()*
      9. *def diamondData ():*

***Descripción del problema:***

Se identifico l necesidad de diseñar un algoritmo donde se pueda calcular el área de diversas figuras geométricas de manera eficiente y precisa, adaptada a las diferentes fórmulas según la figura y los datos proporcionados por el usuario.

***Modelos de análisis:***

Como modelo de análisis podemos identificar que se utilizó el pseudocódigo en pseint como se muestra en la Imagen1. pseudocódigo

1. ***Manual de usuario***

*Este manual va dirigido para que los usuarios puedan usar correctamente a la hora de calcular el área de las figuras geométricas mostradas en el menú.*

***Descripción del funcionamiento del programa:***

En este caso al usuario se le muestra un menú donde él puede seleccionar una de ocho figuras geométricas que se encuentran en el código le quiere halla el área entre ellas tenemos:

1. **Cuadrado.** si el usuario selecciona la opción de cuadrado oprimiendo el número que le corresponde dentro del menú que el usuario está visualizando y luego oprimir enter del teclado que está utilizando para ingresar los datos correspondientes para la utilización del código, el código le solicitara que ingrese los valores necesarios para halla el área de determinada figura. La fórmula para hallar el área de un cuadrado es en este caso se le pide al usuario ingresar la medida de un solo lado para determinar el area por lo que despues de que el usuario inserte la magnitud del lado del cuadrado se le proporcionara el area.
2. **Triangulo:** si el usuario selecciona la opción de triangulo oprimiendo el número que le corresponde dentro del menú que el usuario está visualizando y luego oprimir enter del teclado que está utilizando para ingresar los datos correspondientes para la utilización del código, el código le solicitara que ingrese los valores necesarios para halla el área de determinada figura. La fórmula para hallar el área de un triángulo es en este caso se le pide al usuario ingresar la medida de la base y la altura para determinar el area por lo que despues de que el usuario inserte las magnitudes de la base y la altura del cuadrado se le proporcionara el area.
3. **Circulo:** si el usuario selecciona la opción de circulo oprimiendo el número que le corresponde dentro del menú que el usuario está visualizando y luego oprimir enter del teclado que está utilizando para ingresar los datos correspondientes para la utilización del código, el código le solicitara que ingrese los valores necesarios para halla el área de determinada figura. La fórmula para hallar el área de un círculo esen este caso se le pide al usuario ingresar la medida de un solo lado para determinar el area por lo que despues de que el usuario inserte la magnitud del radio del circulo se le proporcionara el area.
4. **Pentágono:**  si el usuario selecciona la opción de pentágono oprimiendo el número que le corresponde dentro del menú que el usuario está visualizando y luego oprimir enter del teclado que está utilizando para ingresar los datos correspondientes para la utilización del código, el código le solicitara que ingrese los valores necesarios para halla el área de determinada figura. La fórmula para hallar el área de un pentágono esperímetro=lado\*5 y área(perímetro\*apotema) /2 estos cálculos los hace el algorimo internamente por lo que únicamente le pide al usuario ingresar la medida de un solo lado para determinar el area por lo que despues de que el usuario inserte la magnitud del lado del pentagono, se le proporcionara el area.
5. **Trapecio:**  si el usuario selecciona la opción de trapecio oprimiendo el número que le corresponde dentro del menú que el usuario está visualizando y luego oprimir enter del teclado que está utilizando para ingresar los datos correspondientes para la utilización del código, el código le solicitara que ingrese los valores necesarios para halla el área de determinada figura. La fórmula para hallar el área de un trapecio es(baseSuperior+baseInferior) \*altura/2 estos cálculos los hace el algoritmo internamente por lo que únicamente le pide al usuario ingresar la medida de la base superior y inferior para determinar el area por lo que despues de que el usuario inserte la magnitud solicitada del trapecio, se le proporcionara el area.
6. **Romboide:** Si el usuario selecciona la opción de romboide oprimiendo el número correspondiente en el menú y luego presiona Enter, el código solicitará que ingrese los valores necesarios para calcular el área del romboide. La fórmula para hallar el área de un romboide es base altura\text{base} \times \text{altura}base×altura. El algoritmo realizará estos cálculos internamente, por lo que solo se le pedirá al usuario ingresar la medida de la base y la altura. Después de que el usuario inserte estas magnitudes, se le proporcionará el área del romboide.
7. **Rombo:** Si el usuario selecciona la opción de rombo oprimiendo el número correspondiente en el menú y luego presiona Enter, el código solicitará que ingrese los valores necesarios para calcular el área del rombo. La fórmula para hallar el área de un rombo es diagonal1×diagonal2/2\text{diagonal1} \times \text{diagonal2} / 2diagonal1×diagonal2/2. El algoritmo realizará estos cálculos internamente, por lo que solo se le pedirá al usuario ingresar las medidas de las dos diagonales. Después de que el usuario inserte estas magnitudes, se le proporcionará el área del rombo.
8. **Rectángulo:** Si el usuario selecciona la opción de rectángulo oprimiendo el número correspondiente en el menú y luego presiona Enter, el código solicitará que ingrese los valores necesarios para calcular el área del rectángulo. La fórmula para hallar el área de un rectángulo es base altura\text{base} \times \text{altura}base×altura. El algoritmo realizará estos cálculos internamente, por lo que solo se le pedirá al usuario ingresar las medidas de la base y la altura. Después de que el usuario inserte estas magnitudes, se le proporcionará el área del rectángulo.

**XI MANTENIMIENTO.**

Por el momento esta sería la versión final de la calculadora porque a futuro se pueden plantear nuevas modificaciones y actualizaciones ya en un futuro se podría hacer un mantenimiento en base a la experiencia de los usuarios con el aplicativo

**VIII CONCLUSIONES.**

* En el presente trabajo logramos describir el diseño y codificación de un algoritmo el cual permite crear una calculadora de figuras geométricas que permite al usuario calcular el área de figuras geométricas.

* Implementamos un análisis situacional específico para captar con mayor detalle los requisitos, lo cual pudimos utilizar para escribir el código de manera correcta
* Aprendimos que es necesario aplicar una metodología que hace más comprensible y efectivo el proceso a la hora de desarrollar un algoritmo.

* Empleamos un generador de pseudocódigo denominado PSeInt como guía para estructurar de una forma clara el algoritmo en Python.

* Para realizar la codificación del código utilizamos varios programas como Visual Studio Code y GitHub los cuales facilitan el trabajo en grupo y la gestión de información.

1. Documento elaborado septiembre 2024 [↑](#footnote-ref-1)