

1 Instrucciones

- Generar un diagrama de flujo que bosqueje la solución a los siguientes problemas
- desarrollar un programa que resuelva los siguientes problemas
- Todos los programas deberán de subirse a la plataforma GitHub
- Seguir las rúbricas de Moodle.

2 Problemas

1. Desarrollar un programa que lea dos valores a y b , una vez que se hayan leído las variables, el programa debe de indicar cuál es la variable mayor.
2. Desarrollar un programa que lea una variable de tipo entera `value`, indique si es un número par o impar.
3. Escribir un programa que muestre el resultado de la suma de 456.98 y 231.4
4. Suponga que se requieren encontrar las raíces del polinomio de segundo orden

$$ax^2 + bx + c = 0 \quad (1)$$

desarrolle un programa que lea las tres variables y que obtenga el valor de las raíces

$$r_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (2)$$
$$r_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

Nota: Para la raíz cuadrada debe de incluir la librería `math.h`, y puede utilizar la función `sqrt(valor)`. Debe de tener en cuenta que si $b^2 - 4ac$ debe de ser positivo, sino lo es no debe de ejecutar la operación sino mostrar un mensaje de error.

5. Escriba un programa que lea los valores del radio de un círculo y que imprima los valores del área y el perímetro.
6. Escriba un programa que lea dos valores y que realice la división

$$\frac{a}{b} \quad (3)$$

en este caso la variable b no puede ser idéntica a cero, si lo es, debe de indicar error y no realizar la operación.

3 Preguntas

1. Determine cuál de los siguientes es un identificador válido

- A. record1 **VALIDO**
- B. 1record **VALIDO**
- C. file_3 **VALIDO**
- D. return **VALIDO**
- E. \$tax **VALIDO**
- F. Name **VALIDO**
- G. name and address **INVALIDO**
- H. name_and_address **VALIDO**
- I. name-and-address **INVALIDO**
- J. 123-45-6789 **INVALIDO**

2. Escriba las declaraciones apropiadas para cada una de las variables

Todas funcionan con:

#include <stdio.h>

Int main (){

- Variables enteras **p, q** **int p, q;**
 - Variables de tipo flotante: **x,y,z** **float x, y, z;**
 - Variables de tipo caracter: **a,b,c** **char a, b, c;**
- }**

3. Escriba las declaraciones apropiadas para cada una de las siguientes variables

Todas funcionan con:

#include <stdio.h>

Int main (){

- Variables de punto flotante: **root1, root2** **float root1, root2;**
 - Variables de para un entero largo: **counter** **long counter;**
 - Variable de entera corta: **flag** **short flag;**
- }**

4. Escriba las declaraciones apropiadas para cada una de las siguientes variables

Todas funcionan con:

#include <stdio.h>

Int main (){

- Variable entera: **index** **int index;**
 - Variable entera sin signo: **cust_no** **unsigned int cust_no;**
 - Variable de doble precisión: **gros, tax, net** **double gros, tax, net;**
 - Variables de tipo caracter **current, last** **char current, last;**
 - Variables de tipo punto flotante: **error** **float error;**
- }**

5. Escriba las declaraciones e inicializaciones de variables, conforme se muestra a continuación

Todas funcionan con:

```
#include <stdio.h>
```

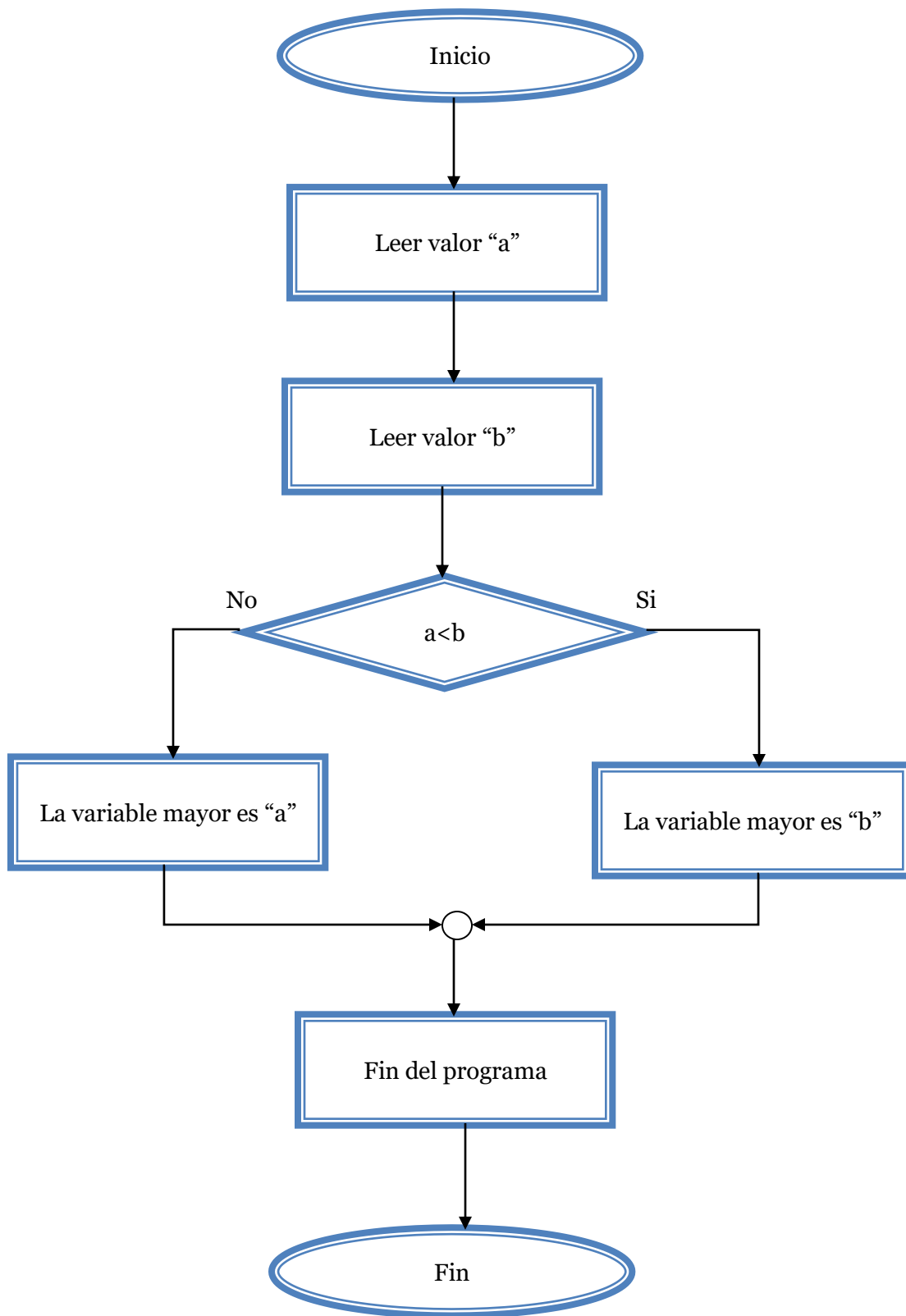
```
Int main (){
```

- Variables de punto flotante a=-8.2 y b=0.005 `float a=-8.2, b=0.005;`
 - Variables de tipo entero x = 129, y = 87 y z = -22 `int x=129, y=87, z=-22;`
 - Variables de tipo caracter c1 = 'w', c2 = '&' `char c1=w, c2=&;` (o utilizando ascii: `char c1=87, c2=38;`)
- ```
}
```

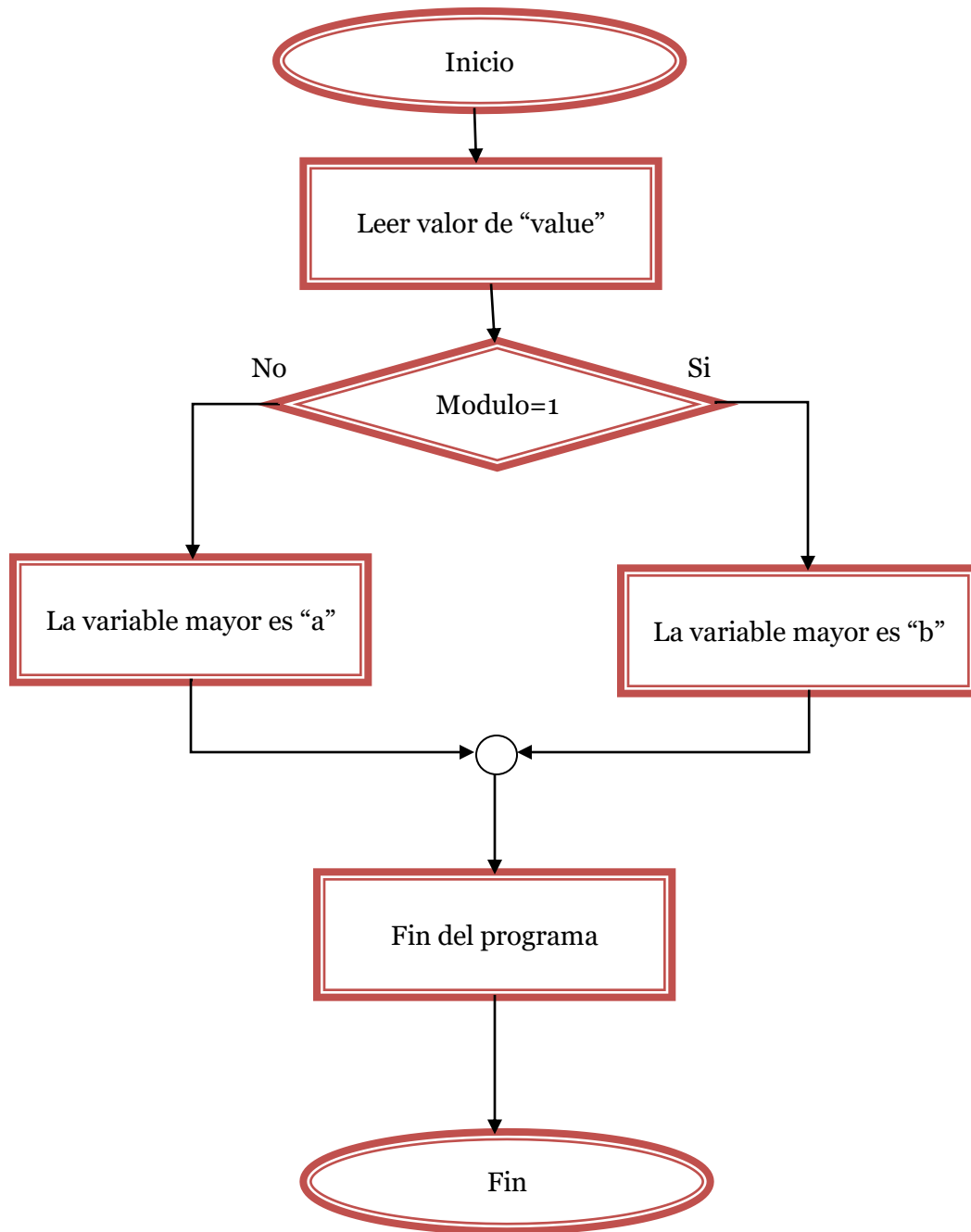
6. Explique que el objetivod de cada expresión

- `a — b` Se resta el valor de “a” menos el valor de “b”
- `a * (b + c)` Primero se hace la suma de “b” mas “c” y el resultado se multiplica por “a”
- `d = a * (b + c)` La letra “d” será igual al resultado de hacer la suma de “b” mas “c” multiplicado por “a”
- `a >= b` El valor de la letra “a” es mayor o igual al valor de la letra “b”
- `(a % 5) == 0` “a” modulo cinco debe ser exactamente igual a cero, en otras palabras; el residuo de la división del valor de “a” entre cinco debe ser exactamente igual a cero

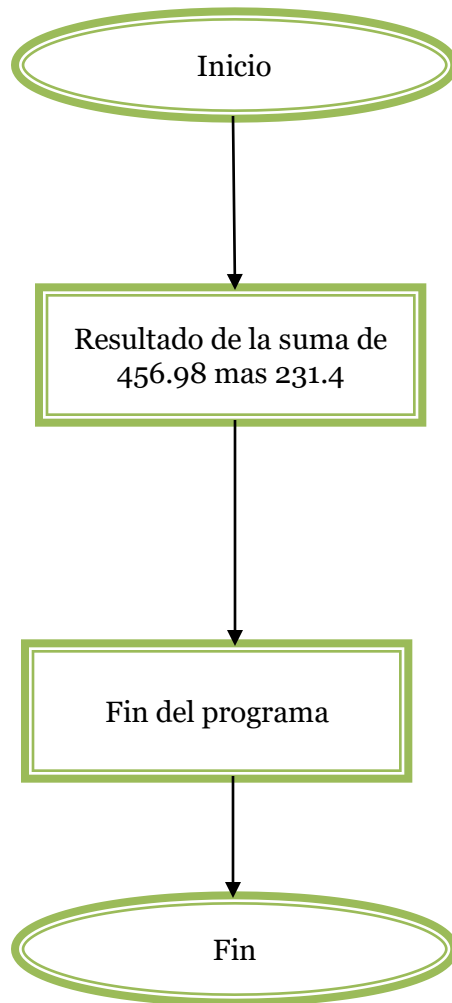
# Tarea1\_1



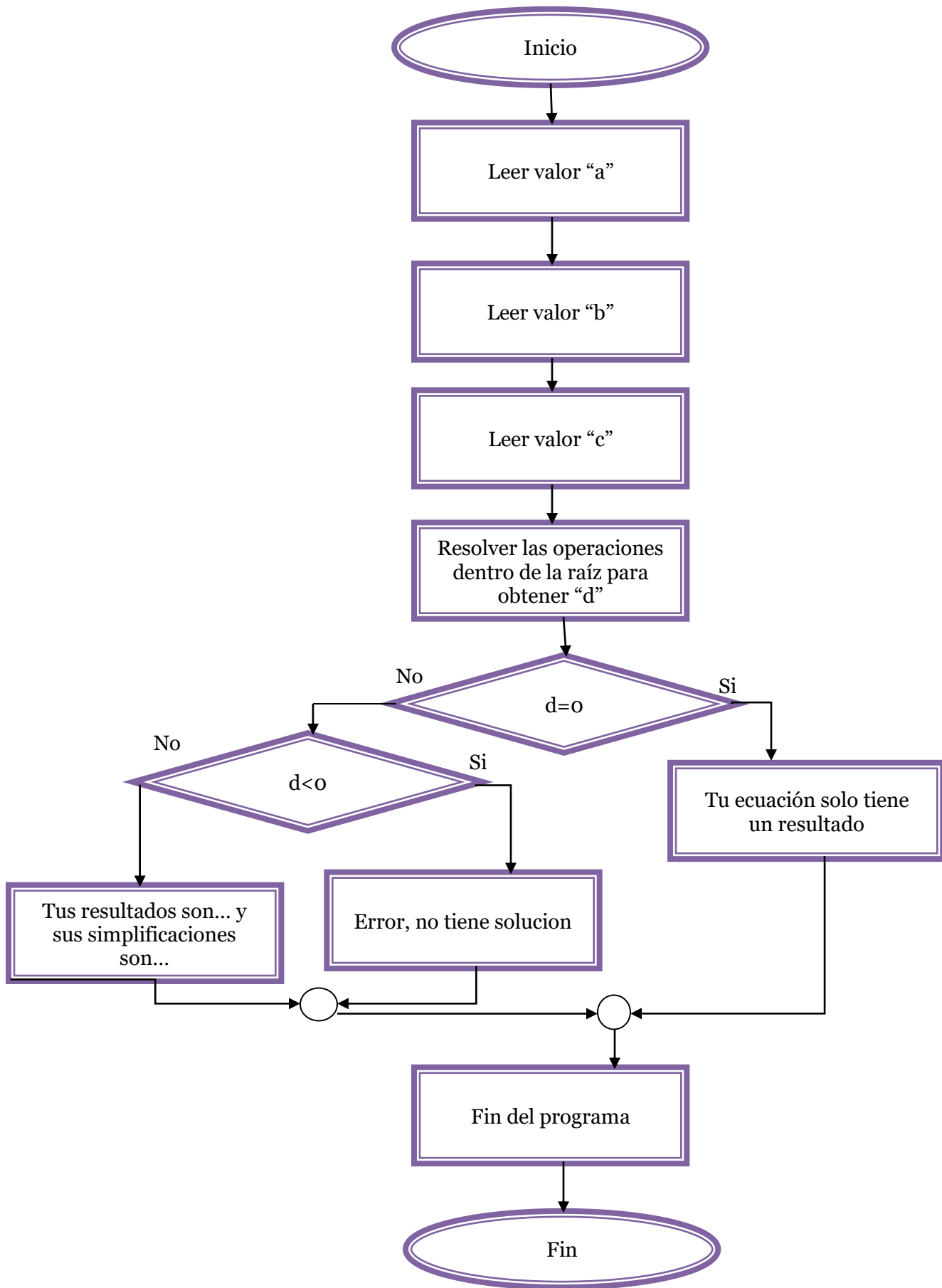
# Tarea1\_2



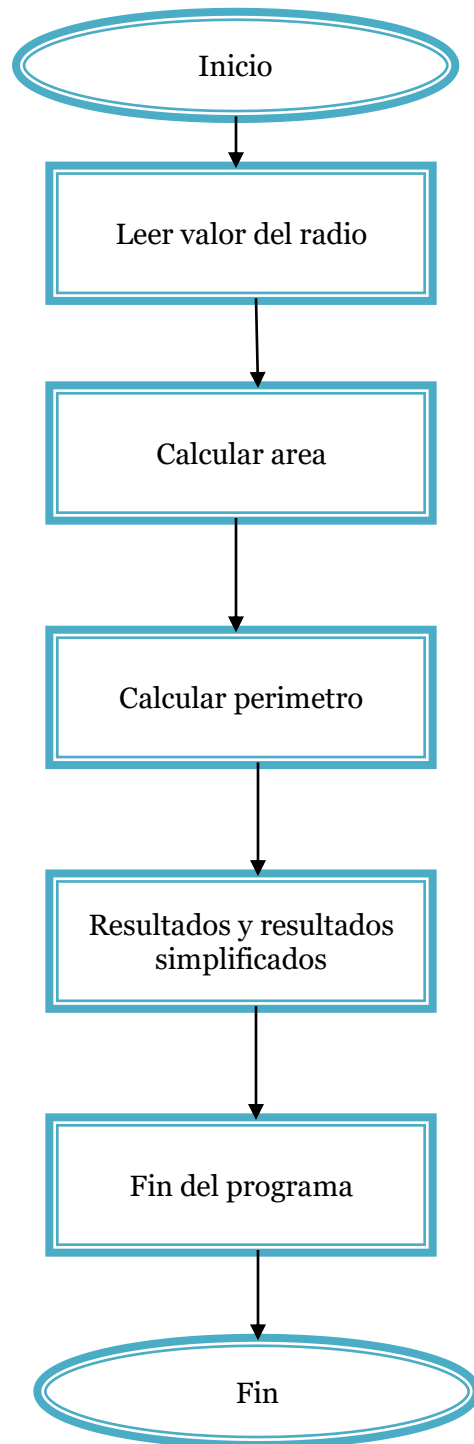
# Tarea1\_3



# Tarea1\_4



# Tarea1\_5





# Tarea1\_6

