



# ALGORITMIA Y PROGRAMACION

## INTRODUCCIÓN

# CONTENIDO



- 1. Conceptos preliminares.
- 2. Metodología.
  - 2.1 Análisis del problema.
  - 2.2 Diseño y pseudocódigo.
  - 2.3 Prueba de escritorio.
  - 2.4 Codificar el algoritmo.
  - 2.5 Probar el algoritmo.

# Conceptos preliminares



Se puede definir un **algoritmo** como una secuencia de pasos lógicos y ordenados con las cuales se puede dar solución a un problema determinado.

# Conceptos preliminares



Se puede definir un **algoritmo** como una secuencia de pasos lógicos y ordenados con las cuales se puede dar solución a un problema determinado.

Un **problema** es cualquier situación que requiera de una **solución**.

# Conceptos preliminares



Se puede definir un **algoritmo** como una secuencia de pasos lógicos y ordenados con las cuales se puede dar solución a un problema determinado.

Un **problema** es cualquier situación que requiera de una **solución**.

Para **solucionar un problema** éste se debe identificar como tal, debemos efectuar una serie de **acciones** para encontrarle una **solución** y describir dicha solución de la forma más clara posible para que los otros la entiendan.

# Conceptos preliminares

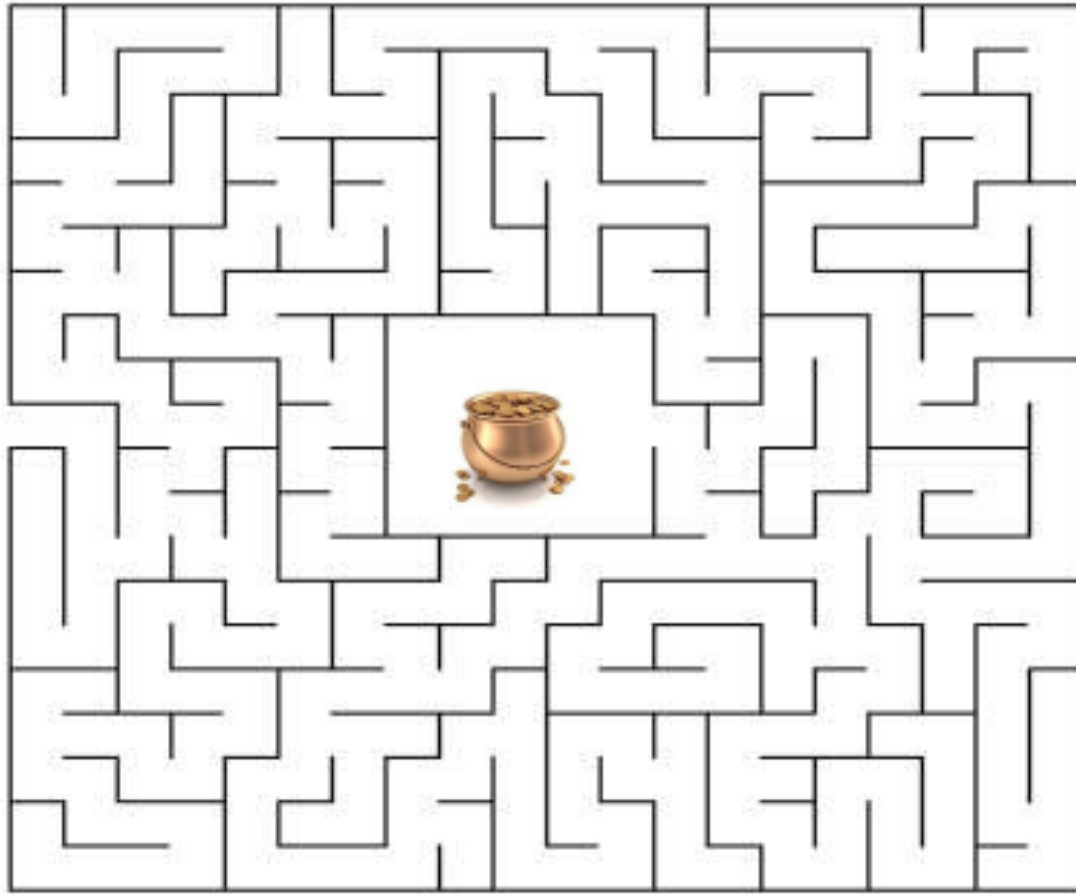


Para un problema pueden existir **diferentes maneras de expresar una solución** y por lo tanto diferentes algoritmos.

# Ejemplo



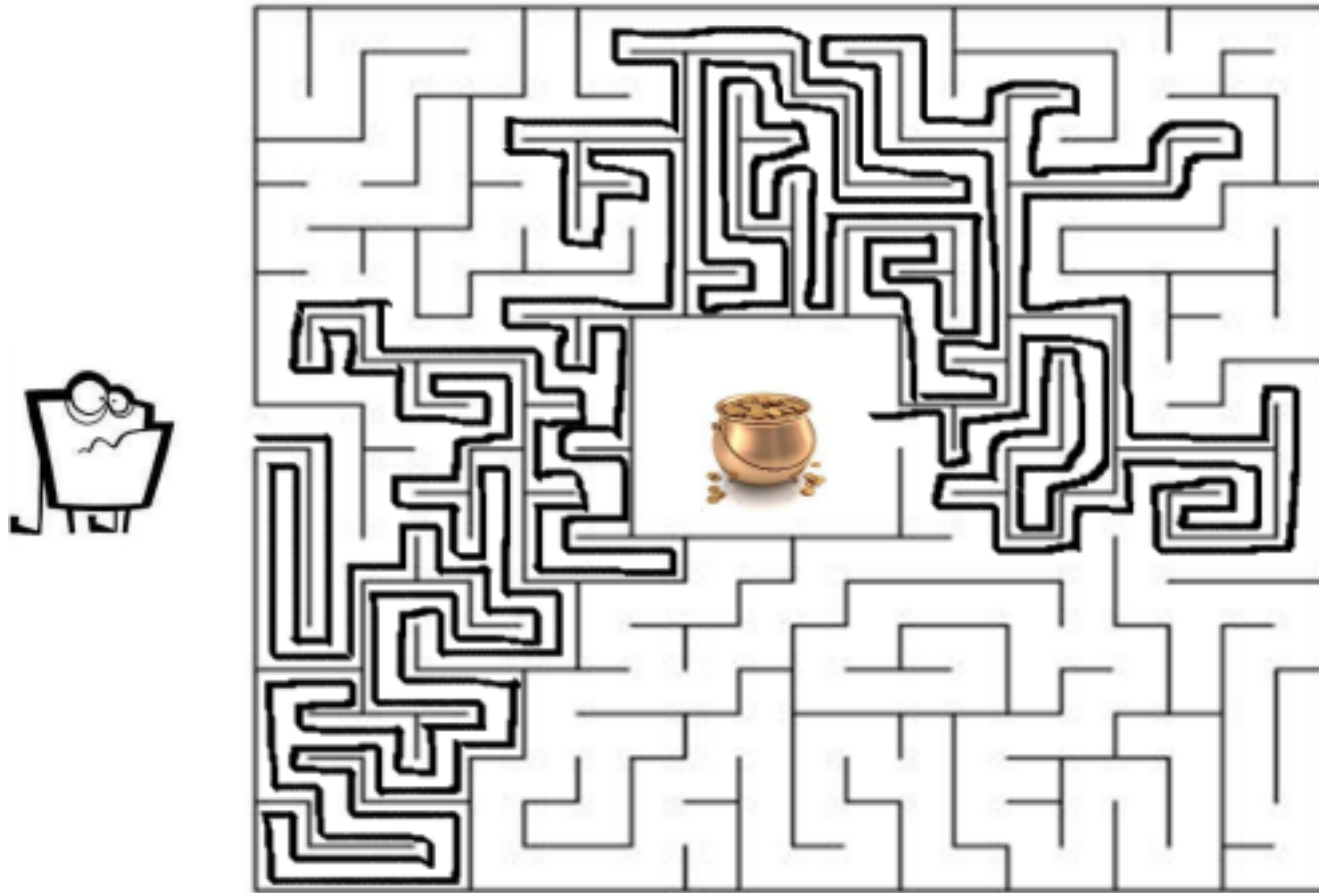
**Ejercicio:** ¿Qué procedimiento seguiría para solucionar el laberinto?



# Ejemplo



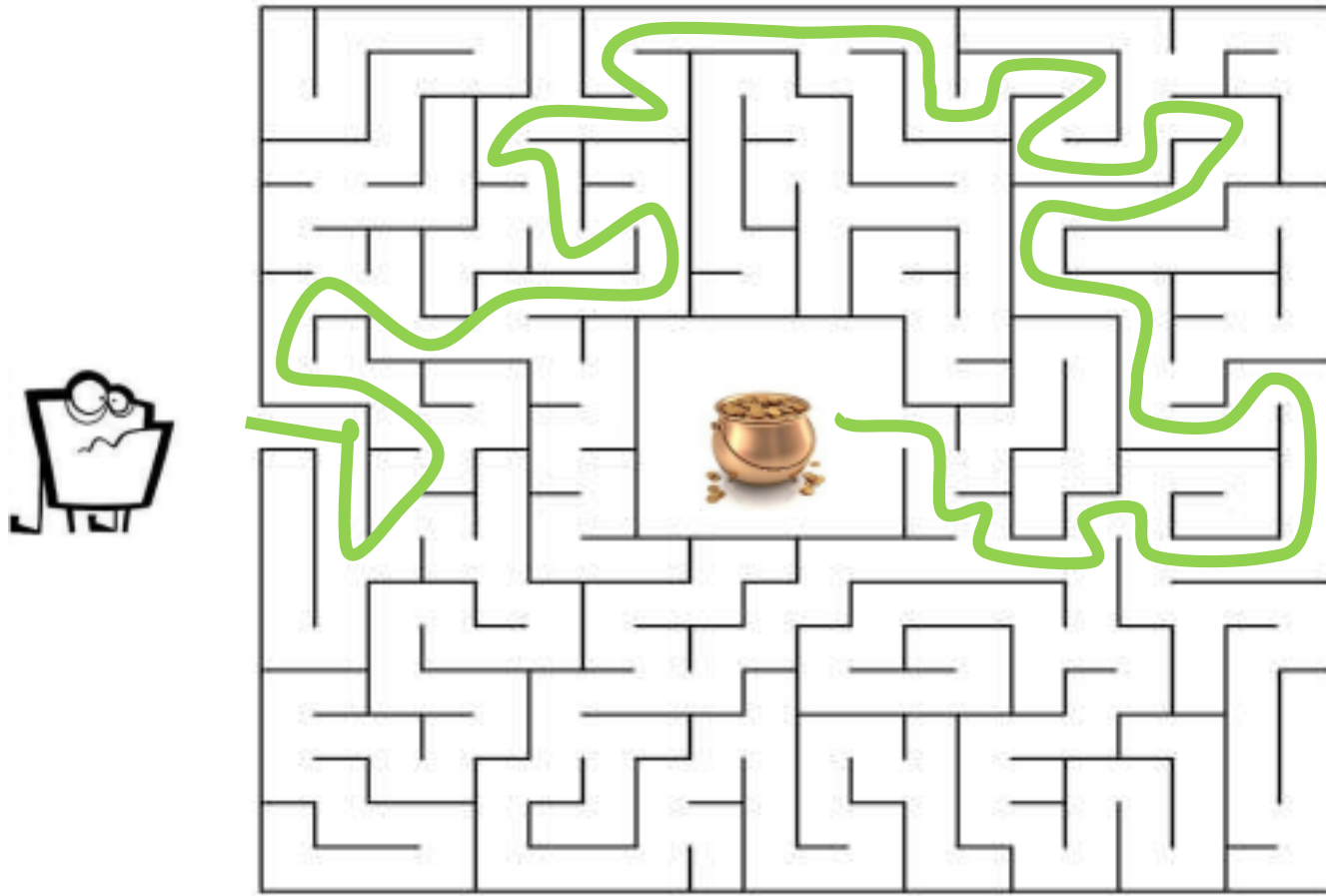
**Ejercicio:** ¿Qué procedimiento seguiría para solucionar el laberinto?





## A simple illustration showing a right hand holding a black-handled magnifying glass. The lens of the magnifying glass is positioned over a large, empty white rectangular area, suggesting a search or inspection process. The background is light gray. A small signature "fotoka" is visible in the bottom right corner of the illustration.

10



# Conceptos preliminares



## Ejemplos:

¿Cuál es el algoritmo para cambiar un bombillo que ya no enciende?

¿Cuál es el algoritmo para cambiar una llanta que se ha pinchado?

# Conceptos preliminares

## Características de los Algoritmos:

1. Un algoritmo debe ser **preciso** e indicar el orden de realización de cada paso.
2. Un algoritmo debe estar **definido**. Si se sigue el algoritmo dos veces, con los mismos datos de entrada, se debe obtener el mismo resultado cada vez.
3. Un algoritmo debe ser **finito**. Si se sigue el algoritmo, se debe terminar en algún momento, o sea debe tener un número finito de pasos.

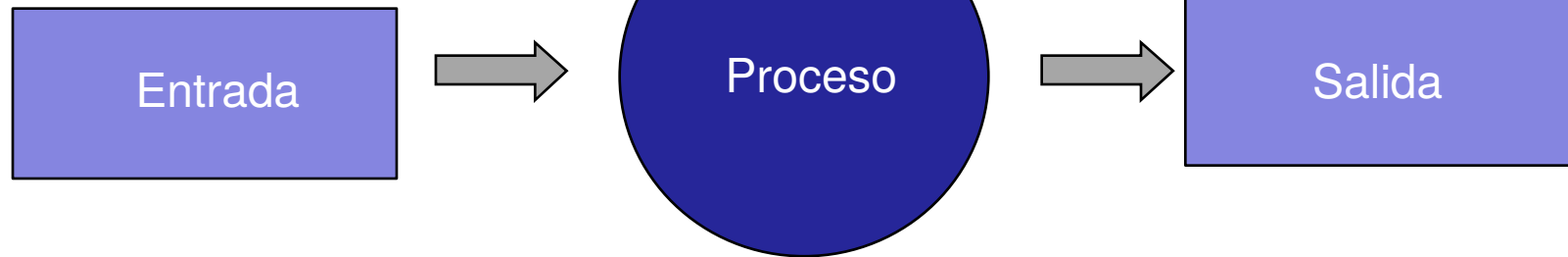
# Conceptos preliminares



## Componentes de un Algoritmo

Definida

Ejemplo sumar dos numeros

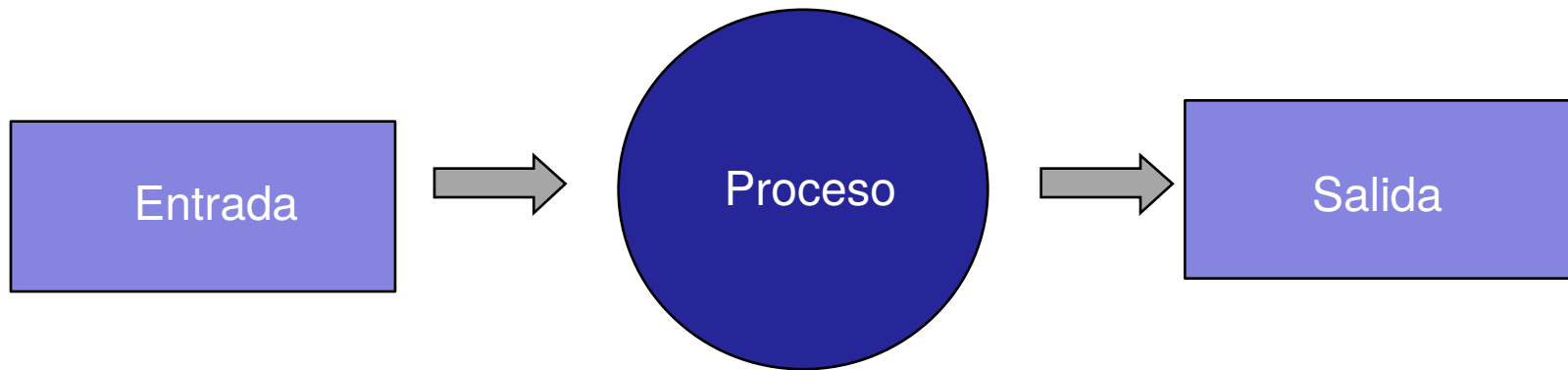


Es lo esperado de acuerdo a la entrada

# Conceptos preliminares



## Componentes de un Algoritmo



**Siempre que se diseña un algoritmo se deben verificar estos componentes**

# Metodología



Pasos que debemos seguir para resolver cualquier problema usando algoritmos.

- 1. Análisis del problema.**
- 2. Diseñar el algoritmo y escribirlo en pseudocódigo.**
- 3. Realizar una prueba de escritorio para el algoritmo.**
- 4. Codificar el algoritmo usando algún lenguaje de programación.**
- 5. Probar el algoritmo implementado.**

# Metodología



Pasos que debemos seguir para resolver cualquier problema usando algoritmos.

- 1. Análisis del problema.**
- 2. Diseñar el algoritmo y escribirlo en pseudocódigo.**
- 3. Realizar una prueba de escritorio para el algoritmo.**
- 4. Codificar el algoritmo usando algún lenguaje de programación.**
- 5. Probar el algoritmo implementado.**

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

Definir el problema con precisión

# Ejemplo



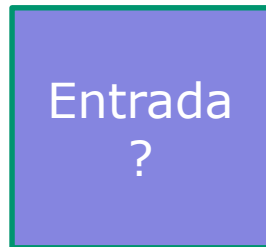
**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

Definir el problema con precisión

Identificar los datos de entrada

Numeros



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

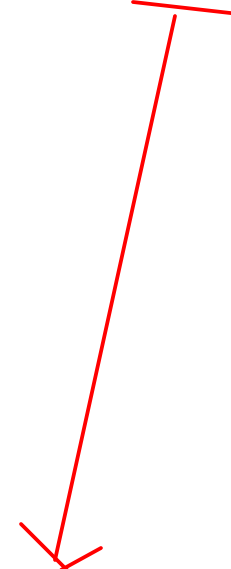
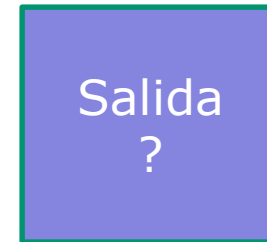
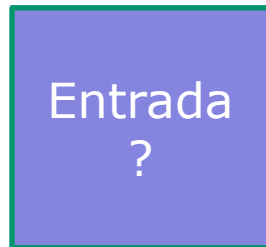
## 1. Análisis del problema

Definir el problema con precisión

Identificar los datos de entrada

Identificar los datos de salida

Numero



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

Definir el problema con precisión

Identificar los datos de entrada

Identificar los datos de salida

Identificar como transformar los datos de entrada en la salida



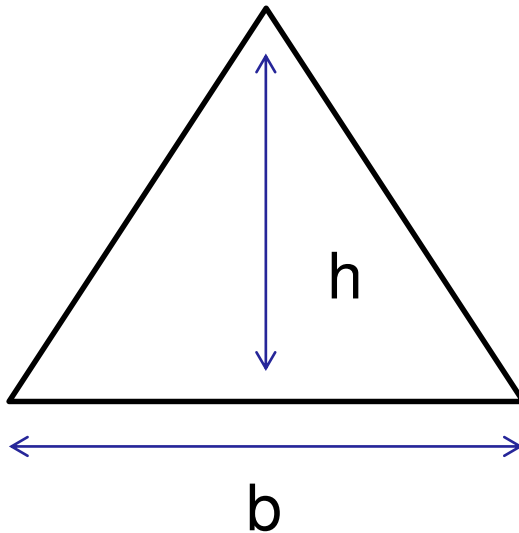
# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

**Entradas:** La base y la altura

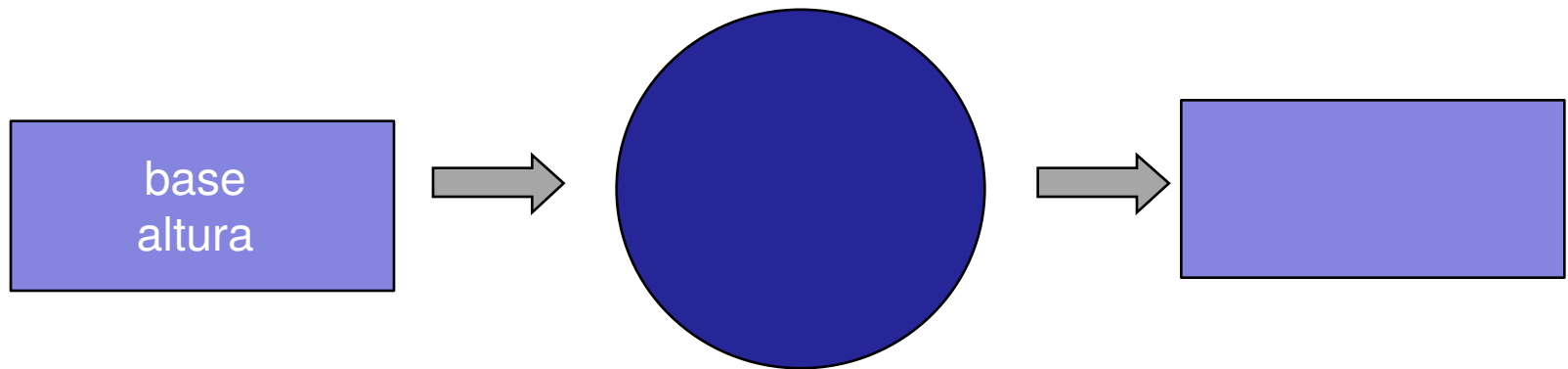


$$a = \frac{b \times h}{2}$$

# Conceptos preliminares



## Análisis del problema



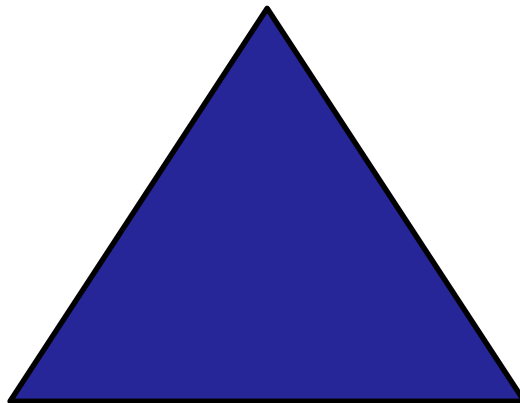
# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

**Salidas:** El área del triangulo

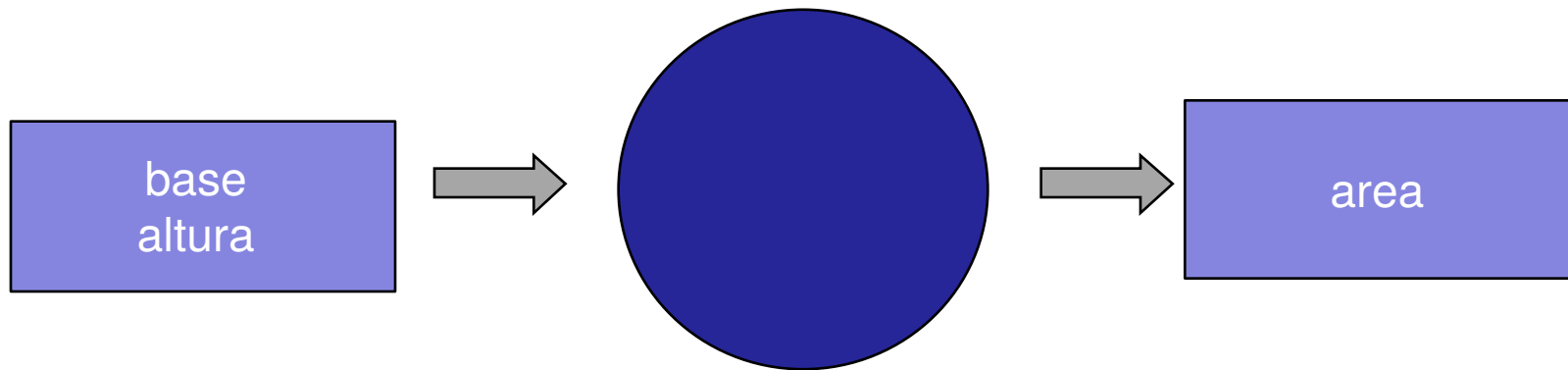


área

# Conceptos preliminares



## Análisis del problema





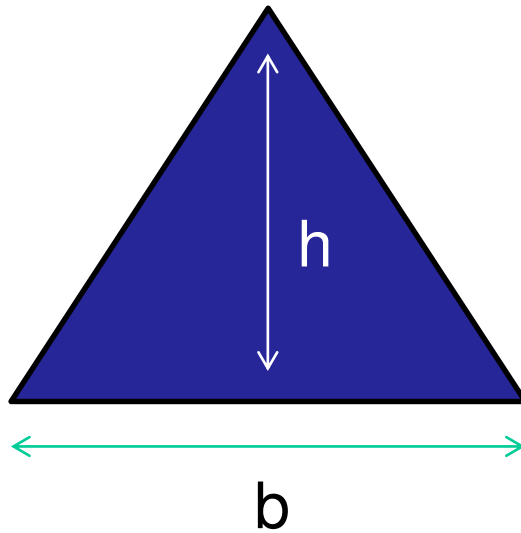
# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

**Proceso:** Fórmula para transformar la entrada(s) en la salida(s)

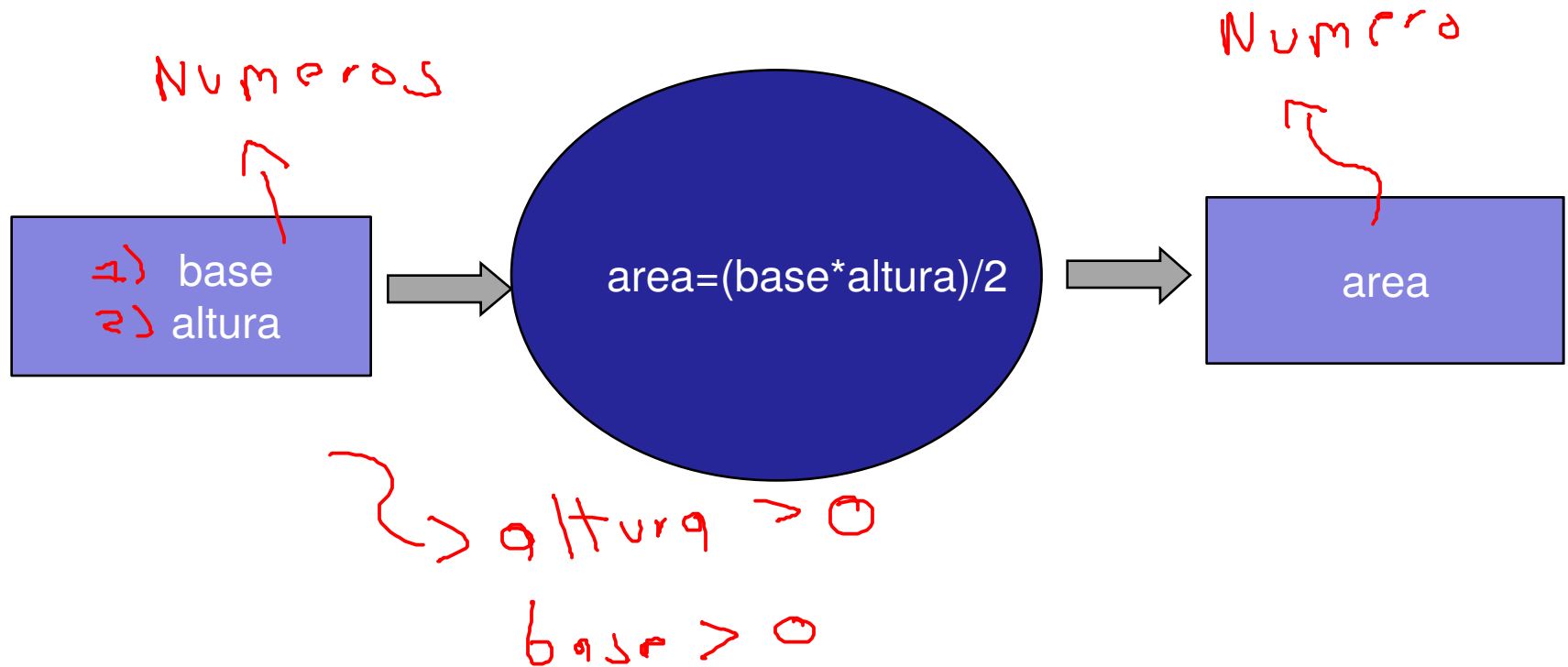


$$\text{area} = (\text{base} * \text{altura}) / 2$$

# Conceptos preliminares



## Análisis del problema



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

Al conjunto de entradas y salidas se les conoce como las **variables del algoritmo**

### Variables

base, altura

area

Pueden cambiar

Almacenan información

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

Las variables pueden ser **numéricas (entero, real)** o también pueden contener **texto**

Variables

base, altura: ?

area: ?

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

Las variables pueden ser **numéricas (entero, real)** o también pueden contener **texto**

Variables

base, altura: **entero**

area: **?**

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y su altura

## 1. Análisis del problema

Las variables pueden ser **numéricas (entero, real)** o también pueden contener **texto**

Variables

base, altura: **entero**

area: **real** ← Debe ser real, ya que si es entero al tener decimales estos se pierden

## Análisis del problema

### 1) Identificar entradas

-- Variables (Numeros o texto)

### 2) Identificar salidas

-- Variables (Numero o texto)

3) EL proceso (La secuencia de pasos para transformar la entrada en la salida)

# Metodología



Pasos que debemos seguir para resolver cualquier problema usando algoritmos.

1. Análisis del problema.

**2. Diseñar el algoritmo y escribirlo en pseudocódigo.**

3. Realizar una prueba de escritorio para el algoritmo.

4. Codificar el algoritmo usando algún lenguaje de programación.

5. Probar el algoritmo implementado.



# Metodología

**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## **2. Diseñar el algoritmo y escribirlo en pseudocódigo**

El **pseudocódigo** es utilizado para describir algoritmos en un lenguaje **humano** simplificado que no es dependiente de ningún lenguaje de programación



# Metodología

**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

**2. Diseñar el algoritmo y escribirlo en pseudocódigo**

**Inicio**

**Fin**



# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

**2. Diseñar el algoritmo y escribirlo en pseudocódigo**

**Inicio**

```
base, altura: entero  
area: real
```

**Fin**

# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## 2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

**Fin**

# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## 2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

**Fin**

# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## 2. Diseñar el algoritmo y escribirlo en pseudocódigo

### Inicio

```
base, altura: entero
```

```
area: real
```

```
base = leer ("Digite la base")
```

```
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2
```

```
imprimir (area)
```

### Fin

Cuáles son las entradas, proceso y salidas en este algoritmo?

# Metodología



Pasos que debemos seguir para resolver cualquier problema usando algoritmos.

1. Análisis del problema.
2. Diseñar el algoritmo y escribirlo en pseudocódigo.
- 3. Realizar una prueba de escritorio para el algoritmo.**
4. Codificar el algoritmo usando algún lenguaje de programación.
5. Probar el algoritmo implementado.

# Metodología

**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### **3. Realizar una prueba de escritorio para el algoritmo**

La prueba de escritorio es una herramienta para entender qué hace un determinado algoritmo, o para verificar que la solución al problema es la correcta





# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### **3. Realizar una prueba de escritorio para el algoritmo**

En esta prueba, se hace una **ejecución manual del algoritmo** y se hace antes de implementarlo en el computador. Se continúa con el proceso hasta verificar si el algoritmo cumple con los criterios de salida esperados.

# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

**3. Realizar una prueba de escritorio para el algoritmo**

**Inicio**

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

**Fin**

Variables

base	altura	área

# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### 3. Realizar una prueba de escritorio para el algoritmo

#### Inicio

```
base, altura: entero  
area: real
```

```
leer (base)  
leer (altura)
```

```
area = (base*altura)/2  
imprimir (area)
```

#### Fin

#### Variables

base	altura	área
2		

# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### 3. Realizar una prueba de escritorio para el algoritmo

#### Inicio

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

#### Variables

base	altura	área
2	4	

#### Fin

# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## 3. Realizar una prueba de escritorio para el algoritmo

**Inicio**

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

**Fin**

Variables

base	altura	área
2	4	4.0

# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### 3. Realizar una prueba de escritorio para el algoritmo

#### Inicio

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

#### Fin



#### Variables

base	altura	área
2	4	4.0

# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### 3. Realizar una prueba de escritorio para el algoritmo

#### Inicio

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la  
altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

#### Variables



base	altura	área
2	4	4.0
3	1	1.5

Fin

# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## 3. Realizar una prueba de escritorio para el algoritmo

**Inicio**

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

**Fin**

Variables



base	altura	área
2	4	4.0
3	1	1.5



# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### 3. Realizar una prueba de escritorio para el algoritmo

#### Inicio

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

#### Fin

#### Variables

base	altura	área
2	4	4.0
3	1	1.5



# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

### 3. Realizar una prueba de escritorio para el algoritmo

#### Inicio

```
base, altura: entero  
area: real
```

```
base = leer ("Digite la base")  
altura = leer ("Digite la altura")
```

```
area = (base*altura)/2  
imprimir (area)
```

#### Fin

#### Variables

base	altura	área
2	4	4.0
3	1	1.5



**El algoritmo funciona correctamente, según la prueba de escritorio**

# Metodología



Pasos que debemos seguir para resolver cualquier problema usando algoritmos.

1. Análisis del problema.
2. Diseñar el algoritmo y escribirlo en pseudocódigo.
3. Realizar una prueba de escritorio para el algoritmo.
- 4. Codificar el algoritmo usando algún lenguaje de programación.**
5. Probar el algoritmo implementado.

# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## **4. Codificar el algoritmo usando algún lenguaje de programación**

Para **codificar (implementar) los algoritmos** se usan los lenguajes de programación. Un lenguaje de programación permite escribir instrucciones que pueden ser interpretadas por una computadora.

# Metodología

**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## **4. Codificar el algoritmo usando algún lenguaje de programación**

Los algoritmos escritos en pseudocódigo se deben traducir a un lenguaje de programación



# Metodología



**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

**4. Codificar el algoritmo usando algún lenguaje de programación**

**Nota:** Se recomienda siempre, antes de codificar un algoritmo realizar su respectiva **prueba de escritorio**.

# Metodología



## Ejemplo.

### Problema:

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## 4. Codificar el algoritmo usando algún lenguaje de programación

En nuestro caso usaremos un lenguaje de programación llamado **Python**, pero existen muchos otros como:

- El lenguaje C
- C++
- Ruby
- C#
- PHP



# Metodología

**Ejemplo.**

**Problema:**

Desarrollar un algoritmo que permita calcular el área de un triángulo a partir de su base y altura.

## **4. Codificar el algoritmo usando algún lenguaje de programación**

Durante las siguientes clases veremos el proceso para codificar los algoritmos en el lenguaje de programación **Python**





# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema

**Entradas:** Los coeficientes  $a$ ,  $b$ ,  $c$ .

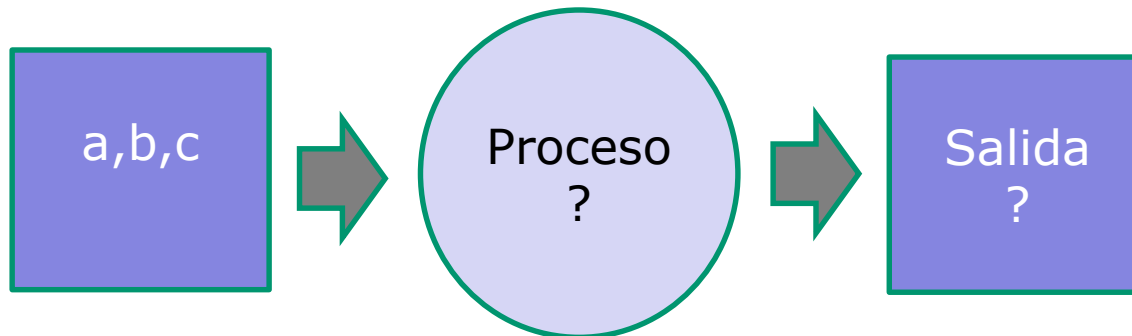
**Nota:**  $a$ ,  $b$ ,  $c$  son números reales y  $a$  es un número diferente de cero

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema

**Salidas:** los valores de  $x_1$  y  $x_2$ , correspondiente a las raíz de la ecuación:  $ax^2 + bx + c = 0$ .

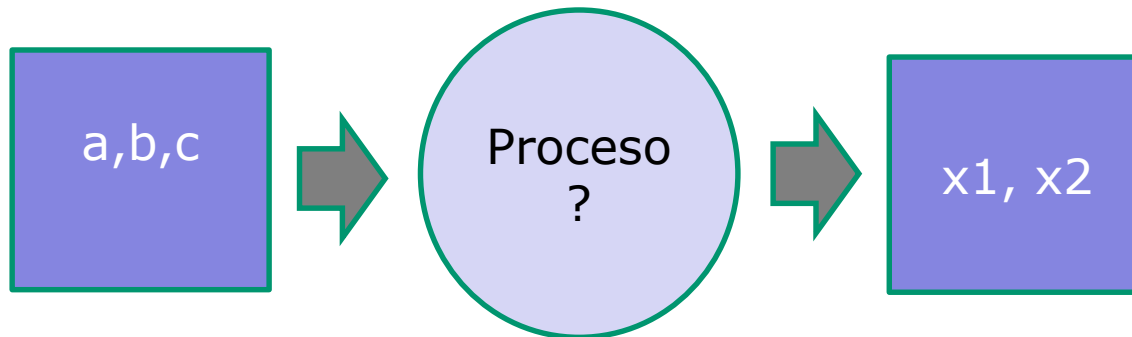
# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema

**Proceso:** Formula para transformar la entrada(s) en la salida(s)

$$x1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

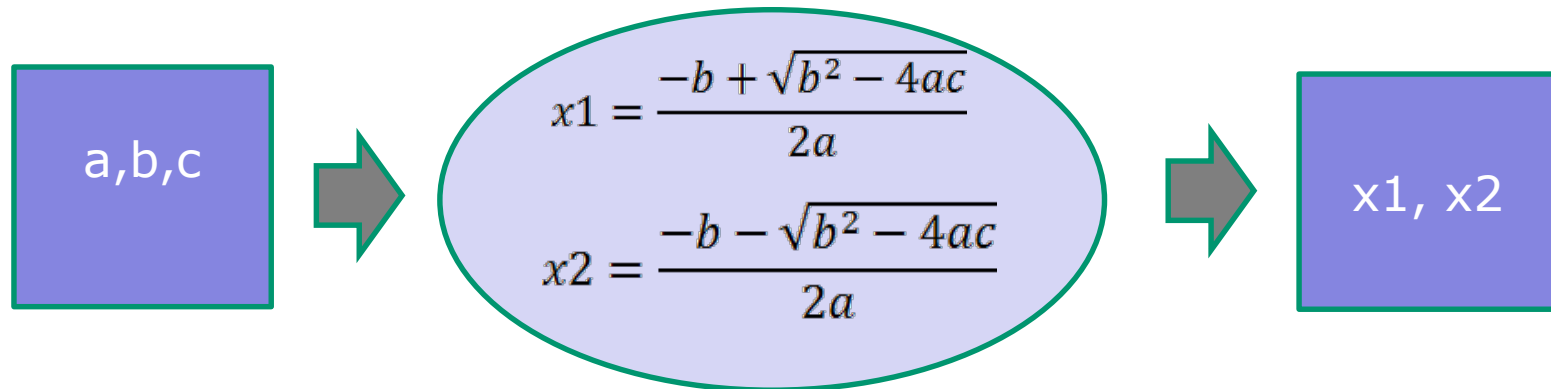
$$x2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema





# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema

Las variables del algoritmo

$a, b, c$

$x_1, x_2$

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema

**De qué tipo (entero, real, texto) son las variables a utilizar en este caso?**

Variables

a, b, c:?

x1, x2:?

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

## 1. Análisis del problema

**De que tipo (entero, real, texto) son las variables a utilizar en este caso?**

Variables

a, b, c: **real**

x1, x2: **real**

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

**Fin**

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

$a, b, c$ : **real**

$x_1, x_2$ : **real**

**Fin**

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

**a, b, c: real**

**x1, x2: real**

**a = leer** ("Digite el valor de a")

**b = leer** ("Digite el valor de b")

**c = leer** ("Digite el valor de c")

Definición de variables

Solicitando los valores de las variables de entrada  
(Les estoy dando un valor)

**Fin**

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

**a, b, c: real**

**x1, x2: real**

**a = leer** ("Digite el valor de a")

**b = leer** ("Digite el valor de b")

**c = leer** ("Digite el valor de c")

**x1 =**  $(-b + \sqrt{(b*b) - (4*a*c)}) / 2*a$  **)**

**x2 =**  $(-b - \sqrt{(b*b) - (4*a*c)}) / 2*a$  **)**

**Fin**

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

$a, b, c$ : **real**

$x1, x2$ : **real**

**a** = **leer** ("Digite el valor de a")

**b** = **leer** ("Digite el valor de b")

**c** = **leer** ("Digite el valor de c")

$x1 = \frac{-b + \sqrt{(b*b) - (4*a*c)}}{2*a}$

$x2 = \frac{-b - \sqrt{(b*b) - (4*a*c)}}{2*a}$

**imprimir** ( $x1$ )      **imprimir** ( $x2$ )

**Fin**



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

Inicio

$a, b, c$ : **real**

$x1, x2$ : **real**

$a = \text{leer}$  ("Digite el valor de  $a$ ")

$b = \text{leer}$  ("Digite el valor de  $b$ ")

$c = \text{leer}$  ("Digite el valor de  $c$ ")

$x1 = \frac{-b + \sqrt{(b*b) - (4*a*c)}}{2*a}$

$x2 = \frac{-b - \sqrt{(b*b) - (4*a*c)}}{2*a}$

**imprimir** ( $x1$ )      **imprimir** ( $x2$ )

Fin

a	b	c	x1	X2

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

**Inicio**

**a, b, c:** **real**

**x1, x2:** **real**

**a = leer** ("Digite el valor de a")

**b = leer** ("Digite el valor de b")

**c = leer** ("Digite el valor de c")

**x1 =**  $(-b + \sqrt{(b*b) - (4*a*c)}) / 2*a$

**x2 =**  $(-b - \sqrt{(b*b) - (4*a*c)}) / 2*a$

**imprimir** (x1)      **imprimir** (x2)

**Fin**

A	b	c	x1	x2
1.0	5.0	6.0		

# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

Inicio

$a, b, c$ : **real**

$x1, x2$ : **real**

$a = leer$  ("Digite el valor de  $a$ ")

$b = leer$  ("Digite el valor de  $b$ ")

$c = leer$  ("Digite el valor de  $c$ ")

$x1 = (-b + \sqrt{(b*b) - (4*a*c)}) / 2*a$

$x2 = (-b - \sqrt{(b*b) - (4*a*c)}) / 2*a$

**imprimir** ( $x1$ )      **imprimir** ( $x2$ )

Fin

$a$	$b$	$c$	$x1$	$x2$
1.0	5.0	6.0	-2.0	-3.0



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

Inicio

$a, b, c$ : **real**

$x1, x2$ : **real**

$a = \text{leer}$  ("Digite el valor de  $a$ ")

$b = \text{leer}$  ("Digite el valor de  $b$ ")

$c = \text{leer}$  ("Digite el valor de  $c$ ")

$x1 = \frac{-b + \sqrt{(b*b) - (4*a*c)}}{2*a}$

$x2 = \frac{-b - \sqrt{(b*b) - (4*a*c)}}{2*a}$

**imprimir** ( $x1$ )      **imprimir** ( $x2$ )

Fin

a	b	c	x1	x2
1.0	5.0	6.0	-2.0	-3.0
-3.5	4.0	0.0		



# Ejemplo



**Problema:** Desarrollar un algoritmo que permita calcular el valor de  $x$ , que da solución a la ecuación cuadrática  $ax^2 + bx + c = 0$ .

2. Diseñar el algoritmo y escribirlo en pseudocódigo

Inicio

**a, b, c: real**

**x1, x2: real**

**a = leer** ("Digite el valor de a")

**b = leer** ("Digite el valor de b")

**c = leer** ("Digite el valor de c")

**x1 =**  $(-b + \sqrt{(b*b) - (4*a*c)}) / 2*a$

**x2 =**  $(-b - \sqrt{(b*b) - (4*a*c)}) / 2*a$

**imprimir** (x1)      **imprimir** (x2)

Fin

a	b	c	x1	x2
1.0	5.0	6.0	-2.0	-3.0
-3.5	4.0	0.0	<b>0.0</b>	<b>1.14</b>



# Metodología



## Ejercicios

Siguiendo la metodología propuesta anteriormente, resuelva los siguientes problemas.

- Desarrolle un algoritmo que lea el monto de una factura y muestre el valor que se debe pagar por concepto del impuesto del IVA (16%) y el valor total a pagar.
- Una casa de cambio necesita un programa que lea un valor en pesos muestre la cantidad correspondiente en dólares. (Asuma que el dólar está a \$ 2386)
- Desarrolle un algoritmo que permita calcular el área de un círculo a partir del radio.
- Desarrolle un algoritmo que permita convertir de grados centígrados a grados Fahrenheit

# Metodología



## Ejercicios.

- Suponga que una persona desea invertir un capital en un banco, y desea saber cuánto dinero ganará después de un mes, si el banco paga a razón de 2.8% mensual. Realice el algoritmo que lea el valor a invertir y muestre el valor a ganar.
- Un vendedor recibe un sueldo base de \$420.000 más un 10% extra por comisión de cada una de sus ventas durante el mes. El vendedor desea saber cual será el total de la comisión si hizo 3 ventas en el mes, y cuánto será el total del dinero que recibirá en su sueldo. Se debe leer el valor de cada una de las ventas que realizó el vendedor.

# Metodologia

## 1) Análisis del problema

--> Entradas --> Salidas --> Proceso

Identifico las variables y su tipo (numero, texto, etc)

## 2) Pseudocodigo

Secuencia de pasos (proceso) en lenguaje humano

## 3) Prueba de escritorio

Darle valores a las entradas y probar si la salida es correcta de acuerdo al problema

## 4) La codificación (Lenguaje de programación -- Ruby)

## 5) Pruebas en el código (Implementación)