

# *Diseño de Algoritmos*

2018

# *Diseñando Algoritmos*

**UN PROGRAMADOR ES ANTES QUE NADA  
UNA PERSONA QUE RESUELVE PROBLEMAS.**

**EL PROGRAMADOR PROCEDE A RESOLVER  
UN PROBLEMA, A PARTIR DE LA DEFINICIÓN  
DE UN ALGORITMO Y DE LA TRADUCCIÓN DE  
DICHO ALGORITMO A UN PROGRAMA QUE  
EJECUTARÁ LA COMPUTADORA.**

# *Diseñando Algoritmos*

## **PROFUNDIZANDO:**

**Algoritmo:** un algoritmo **es un método** para resolver un problema, que consiste en la realización de un conjunto de pasos lógicamente ordenados tal que, partiendo de ciertos datos de entrada, permite obtener ciertos resultados que conforman la **solución del problema**.

# *Diseñando Algoritmos*

## **EJEMPLO:**

Es decir, podríamos definir un algoritmo para atarnos los cordones de la siguiente forma:

1. Ponerme las zapatillas.
2. Agarrar los cordones con ambas manos.
3. Hacer el primer nudo.
4. Hacer un bucle con cada uno de los cordones.
5. Cruzar los dos bucles y ajustar.
6. Corroborar que al caminar los cordones no se sueltan y la zapatilla se encuentra correctamente atada.

# *Importante!!!*

Por otro lado, la complejidad de los distintos problemas que podamos abordar puede variar desde muy sencilla a muy compleja, dependiendo de la situación y la cantidad de elementos que intervienen.

En casos de mayor complejidad suele ser una buena solución dividir al problema en diferentes subproblemas que puedan ser resueltos de manera independiente.

De esta forma la solución final al problema inicial será determinada por las distintas soluciones de los problemas más pequeños cuya resolución es más sencilla.

# *Diseñando Algoritmos*

## **PROGRAMA**

Cuando traducimos el algoritmo en un conjunto de instrucciones, entendibles por la computadora, que le indican a la misma lo que debe hacer.

## **LENGUAJES DE PROGRAMACIÓN**

Para escribir un programa se utilizan **lenguajes de programación**, que son lenguajes que pueden ser entendidos y procesados por la computadora.

Un lenguaje de programación es tan sólo un medio para expresar un algoritmo y una computadora es sólo un procesador para ejecutarlo.

# *Pasos para la construcción de un programa*

- 01 DEFINICIÓN DEL PROBLEMA
- 02 ANÁLISIS DEL PROBLEMA
- 03 DISEÑO DEL ALGORITMO
- 04 CODIFICACIÓN
- 05 PRUEBA Y DEPURACIÓN

# *Pasos para la construcción de un programa*

## **1- DEFINICIÓN DEL PROBLEMA**

En este paso se determina la información inicial para la elaboración del programa. Es donde se determina qué es lo que debe resolverse, el cual requiere una definición clara y precisa.

Es importante que se conozca lo que se desea que realice la computadora; mientras la definición del problema no se conozca del todo, no tiene mucho caso continuar con la siguiente etapa.



# 1- DEFINICIÓN DEL PROBLEMA

Obtención del área de un rectángulo

# *Pasos para la construcción de un programa*

## **2 - ANÁLISIS DEL PROBLEMA**

Una vez que se ha comprendido lo que se desea hacer, es necesario definir:

- Los datos de entrada.
- Los datos de salida
- Los métodos y fórmulas que se necesitan para procesar los datos.

## 2 - ANÁLISIS DEL PROBLEMA

Datos de entrada:

*Altura*

*Base*

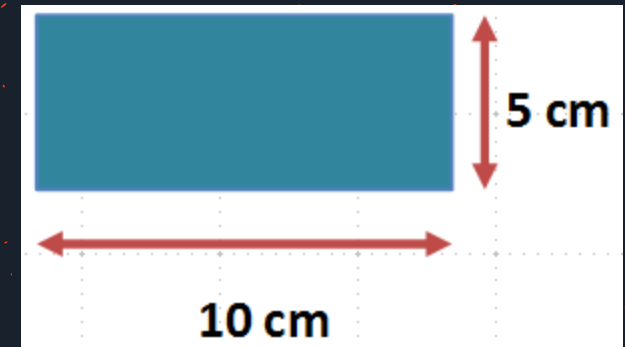
Proceso:

*Calculo del área del rectángulo:*

*Base \* Altura*

Salida:

*Área del rectángulo*



# *Pasos para la construcción de un programa*

## **3 - DISEÑO DEL ALGORITMO**

Se puede utilizar algunas de las herramientas de representación de algoritmos (Diagrama de flujos, Pseudocódigo, etc).

Este proceso consiste en definir la secuencia de pasos que se deben llevar a cabo para conseguir la salida identificada en el paso anterior.

### 3 - DISEÑO DEL ALGORITMO

```
INICIO CalculoDelArea
    ENTERO altura;
    ENTERO base;
    ENTERO area;
    IMPRIMIR "Ingrese la altura"
    IMPRIMIR "Ingrese la base"
    ENTERO area = altura*base
    IMPRIMIR "El resultado es" + area
FIN
```

# *Pasos para la construcción de un programa*

## 4 - CODIFICACIÓN

La codificación es la operación de escribir la solución del problema (de acuerdo a la lógica del diagrama de flujo o pseudocódigo), en una serie de instrucciones detalladas, en un código reconocible por la computadora.

La serie de instrucciones detalladas se conoce como **código fuente**, el cual se escribe en un **lenguaje de programación** o **lenguaje de alto nivel**.

## 4 - CODIFICACIÓN

```
public class AreaRectangulo {  
    public static void main(String[] args) {  
        int area;  
        Scanner sc = new Scanner(System.in);  
        System.out.println("ingrese la base del rectangulo");  
        int = sc.nextInt();  
        System.out.println("ingrese alto del rectangulo");  
        int b = sc.nextInt();  
        area = a * b;  
        System.out.println(area);  
    }  
}
```

# *Pasos para la construcción de un programa*

## **5 - PRUEBA Y DEPURACIÓN**

Se denomina prueba de escritorio a la comprobación que se hace de un algoritmo para saber si está bien realizado. Esta prueba consiste en tomar datos específicos como entrada y seguir la secuencia indicada en el algoritmo hasta obtener un resultado, el análisis de estos resultados indicará si el algoritmo está correcto o si por el contrario hay necesidad de corregirlo o hacerle ajustes.



## 5 - PRUEBA Y DEPURACIÓN

Obtención del área de un rectángulo:

Altura: 5 cm

Base: 10 cm

1. Salida: área

Fórmula del área:  $\text{base} \times \text{altura}$ .

2. Entradas de datos:

Dato de Entrada 1: altura: 5 cm

Dato de Entrada 2: base: 10 cm

3. Proceso:

$\text{área} = \text{base} * \text{altura} \rightarrow \text{área} = 50$

```
INICIO CalculoDelArea
    ENTERO altura;
    ENTERO base;
    ENTERO area;
    IMPRIMIR "Ingrese la altura"
    IMPRIMIR "Ingrese la base"
    ENTERO area = altura*base
    IMPRIMIR "El resultado es" + area
FIN
```

*Preguntas?*

# Ejercicios

## 1- Ejercicio – Secuencial

Escribir un algoritmo que permita realizar una suma de dos números enteros. El usuario deberá ingresar primero un número, luego el siguiente número, y el sistema arrojará el resultado correspondiente.

## 2- Ejercicio – Secuencial

Escribir un algoritmo junto con su prueba de escritorio que pregunte el nombre del usuario y su edad. Y posteriormente permita saludar al usuario y mostrar la edad que tiene.

# Ejercicios

## 3- Ejercicio – Secuencial

Escribir un algoritmo junto con su prueba de escritorio que permita realizar una resta de dos números enteros. El usuario deberá ingresar primero un número, luego el siguiente número, y el sistema arrojará el resultado correspondiente.

Tipo de Datos en Pseudocódigo	Explicación	Ejemplo	Correspondencia en Java
<b>Texto</b>	<p>Representa un carácter o una cadena de caracteres.</p> <p>Su valor debe ir encerrado entre comillas dobles.</p> <p>Si se desea concatenar texto con variables (que pueden ser de tipo: entero, decimal o booleano), se utiliza el signo más (+), de la siguiente manera:</p> <p>Entero suma = 20 + 50;</p> <p>"El valor de la suma es: " + suma</p>	<p>"Hoy es viernes",</p> <p>"b",</p> <p>"El precio del producto es \$20"</p>	String
<b>Entero</b>	Representa un número entero.	10,29,30	int
<b>Decimal</b>	Representa un número decimal.	21.3, 43.9, 23.564	float o double
<b>Booleano</b>	Define una bandera que puede tomar dos posibles valores: Verdadero o Falso.	Verdadero, falso	boolean

# Ejercicios

## 4- Ejercicio – Secuencial

Escribir un algoritmo junto con su prueba de escritorio que permita calcular una altura en pulgadas ( $1 \text{ pulgada} = 2.54 \text{ cm}$ ) y pies ( $1 \text{ pie} = 12 \text{ pulgadas}$ ), a partir de la altura en centímetros, que se introduce por el teclado.

# Resultados

Escribir un algoritmo que permita realizar una suma de dos números enteros. El usuario deberá ingresar primero un número, luego el siguiente número, y el sistema arrojará el resultado correspondiente.

Pseudocódigo

INICIO

entero primerNum;

entero segundoNum;

entero suma;

IMPRIMIR: "Ingrese el primer número:";

TOMAR: primerNumIngresado;

PrimerNum=primerNumIngresado;

IMPRIMIR: "Ingrese el segundo número:";

TOMAR: segundoNumIngresado;

SegundoNum=segundoNumIngresado;

Suma=primerNum+segundoNum;

IMPRIMIR: "El resultado de la suma es:" + suma;

FIN

# Resultados

2- Escribir un algoritmo junto con su prueba de escritorio que pregunte el nombre del usuario y su edad. Y posteriormente permita saludar al usuario y mostrar la edad que tiene.

Pseudocódigo

INICIO

string nombre;

string edad;

IMPRIMIR: "Ingrese su nombre"

TOMAR: nombreIngresado;

nombre=nombreIngresado;

edad: "Ingrese el su edad";

TOMAR: edadIngresado;

edad=edadIngresado;

IMPRIMIR: "Buenas tardes," + nombre + " su edad es:" + edad;

FIN



# Resultados

4- Escribir un algoritmo junto con su prueba de escritorio que permita calcular una altura en pulgadas (1 pulgada=2.54 cm) y pies (1 pie=12 pulgadas), a partir de la altura en centímetros, que se introduce por el teclado.

Pseudocódigo

INICIO

decimal alturaNum;

decimal alturaPulgadasNum;

decimal alturaPiesNum;

IMPRIMIR: "Ingrese la altura:";

TOMAR: alturaNumIngresado;

alturaNum=alturaNumIngresado;

// Calcular la altura en pulgadas

alturaPulgadasNum = alturaNum / 2.54;

// Calcular la altura en pies

alturaPiesNum = alturaPulgadasNum / 12;

IMPRIMIR: "El resultado en pulgadas es:" + alturaPulgadasNum;

IMPRIMIR: "El resultado en pies es:" + alturaPiesNum;

FIN

# Pruebas de Escritorio






Escribir un algoritmo que permita realizar una suma de dos números enteros. El usuario deberá ingresar primero un número, luego el siguiente número, y el sistema arrojará el resultado correspondiente.

*Identificación de nombres de variables, con su tipo de variable y tipo de dato.*

TIPO VARIABLE	TIPO DE DATO	NOMBRE
Entrada	Entero	primerNumIngresado
Entrada	Entero	segundoNumIngresado
Auxiliar	Entero	primerNum
Auxiliar	Entero	segundoNum
Salida	Entero	Suma

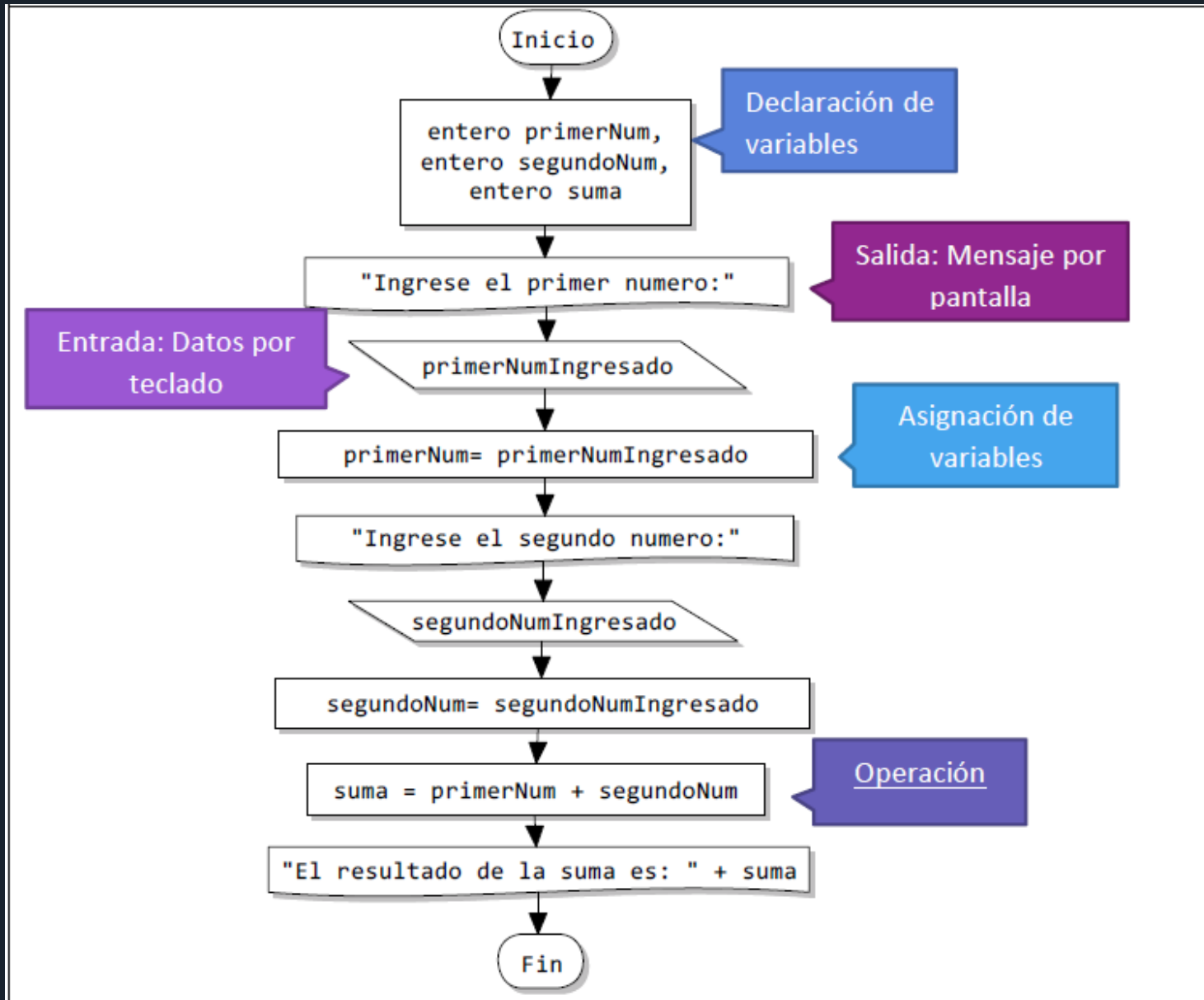
*Ejecución de Pruebas*

N° Prueba	Entrada		Asignación		Operación	Salida
	Primer Num Ingresado	Segundo Num Ingresado	primer Num	segundo Num	suma	Mensaje
1	20	30	20	30	20 + 30=50	"El resultado de la suma es:" + 50
2	15	150	15	150	15 + 150 =165	"El resultado de la suma es:" + 165
3	130	300	130	300	130 + 300=430	"El resultado de la suma es:" + 430

Nombre del Símbolo	Representación del Símbolo	Función
Inicio/Final		Representa el comienzo o la finalización de un algoritmo.
Entrada		Representa la entrada de datos por medio de un dispositivo de entrada, en general representa entrada de datos por teclado
Proceso		Representa una operación como puede ser una operación matemática (suma, resta, multiplicación, etc), una declaración de una variable o una asignación de una variable a un valor, etc.
Decisión		Representa una condición: realiza comparaciones entre distintos valores. Al evaluar dicha condición puede dar como resultado dos posibles valores: verdadero o falso, en función de que si esa condición se cumple o no.
Salida		Representa la salida de datos o resultados. Sirve para representar los mensajes que el sistema le muestra al usuario por pantalla.

# Recordemos el primer ejercicio

Escribir un algoritmo que permita realizar una suma de dos números enteros. El usuario deberá ingresar primero un número, luego el siguiente número, y el sistema arrojará el resultado correspondiente.



## Ejercicios:

Realizar los diagramas de flujos de los anteriores ejercicios

5 - Escriba un algoritmo junto a su prueba de escritorio y su diagrama de flujo, que permita conocer el área de un triángulo a partir de la base y la altura.

(Base por altura) / 2

# Ejercicios:

## Pseudocódigo:

INICIO

Base, Altura: ENTERO

ESCRIBA "Diga la Base: "

LEA Base

ESCRIBA "Diga la Altura"

LEA Altura

ESCRIBA "Area del Triangulo = ",  
(BASE\*ALTURA)/2

FIN

## Diagrama de flujo:

