

# TEMA 1.

## INTRODUCCIÓN A LA PROGRAMACIÓN

# CONTENIDOS

## 1. ALGORITMOS Y PROGRAMAS

### 1.1. CONCEPTOS DE ALGORITMO Y PROGRAMA

### 1.2. HERRAMIENTAS PARA EL DISEÑO DE ALGORITMOS

# Algoritmo

- Un **algoritmo** es una secuencia finita y precisa de instrucciones o pasos que permiten resolver un problema dado.
  - Transforma los datos iniciales en los resultados deseados.
- El algoritmo permite automatizar una operación
- Ejemplos
  - Algoritmo para ordenar una lista de números
  - Algoritmo para distribuir los escaños de un parlamento según los votos

# Programas

- **Programa**
  - Conjunto de órdenes que se le dan a un ordenador para que realice un determinado proceso
  - Es el resultado de traducir un algoritmo a un lenguaje que entiende el ordenador (lenguaje de programación).
  - Resuelve un problema mediante el uso de un ordenador
- Una instrucción es una orden lanzada al ordenador
  - Un programa es un conjunto de instrucciones

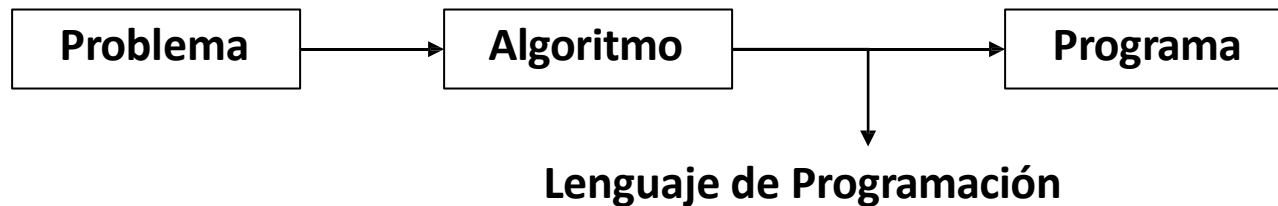
# Algoritmos y Programas

- Recapitulando:
  - Dado un problema , se necesita una información sobre el problema compuesta por datos; estos datos son procesados por el ordenador mediante un programa almacenado en él;
  - Éste programa ejecuta un conjunto de órdenes o instrucciones para implementar un algoritmo creado para resolver el problema.
  - El programa proporciona una salida que es la solución del problema, en forma de datos

# Datos, Expresiones e Instrucciones

- **Programa:** conjunto de órdenes (llamadas instrucciones o sentencias) escritas en un determinado lenguaje de programación, que se le dan a un ordenador para que realice un determinado proceso

- Un programa implementa un algoritmo



- Un programa está formado por:
  - Datos del programa:
    - Información que procesa el programa
  - Expresiones:
    - Operaciones sobre los datos, realizadas mediante operadores
  - Instrucciones:
    - Acciones a realizar , normalmente sobre los datos

# Algoritmos

- Un algoritmo debe ser:
  - **Preciso y unívoco**
  - **Finito**: número finito de pasos
  - **Correcto**: conducir a la solución
  - **Óptimo**: menor número de instrucciones posibles
- Hay herramientas que facilitan el desarrollo de algoritmos
  - Las de uso más común son
    - Diagramas de flujo (también llamados organigramas)
    - Pseudocódigo

## Ejemplo de un algoritmo para resolver el problema 'desayunar':

Inicio

Sentarse

Servirse café con leche

Servirse azúcar

Si tengo tiempo entonces

    Mientras tenga apetito Hacer

        Untar una tostada con mantequilla

        Añadir mermelada

        Comérmela

    Fin Mientras

Fin Si

Beberse el café

Levantarse

Fin



# CONTENIDOS

## 1. ALGORITMOS Y PROGRAMAS

1.1. CONCEPTOS DE ALGORITMO Y PROGRAMA

1.2. **HERRAMIENTAS PARA EL DISEÑO DE ALGORITMOS**

# Herramientas para el Diseño de Algoritmos

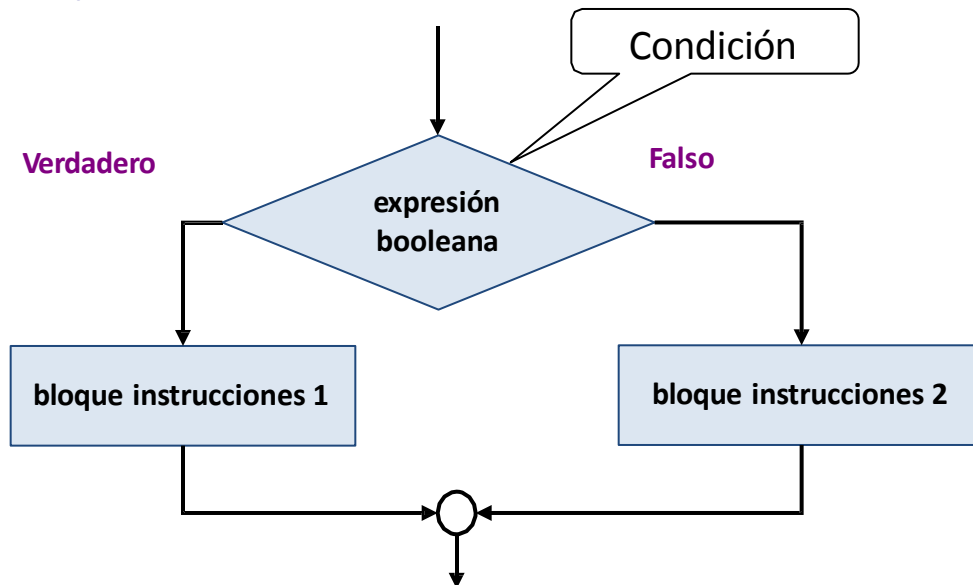
- Diagramas de flujo
  - Representación gráfica que permite detallar el flujo de un algoritmo
- Pseudocódigo
  - Expresión del algoritmo en lenguaje natural

# Diagramas de Flujo

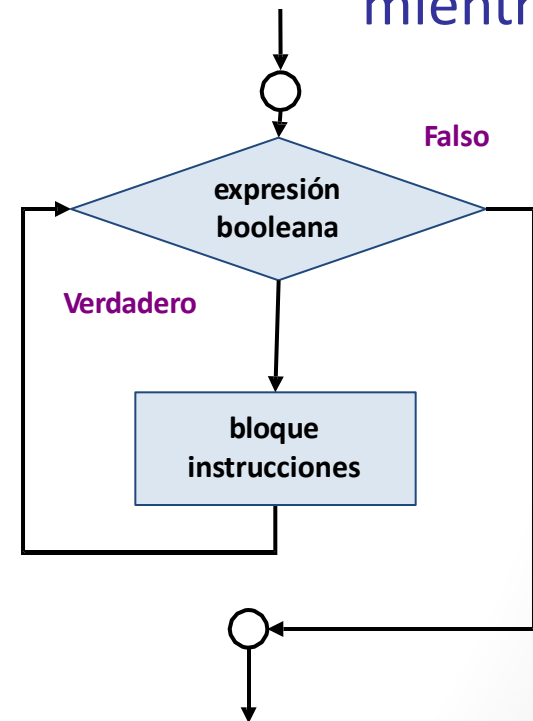
- **Diagramas de flujo u organigramas:**

- representaciones gráficas que permiten detallar el flujo de un programa (orden de las instrucciones)
- símbolos normalizados

si /si-sino



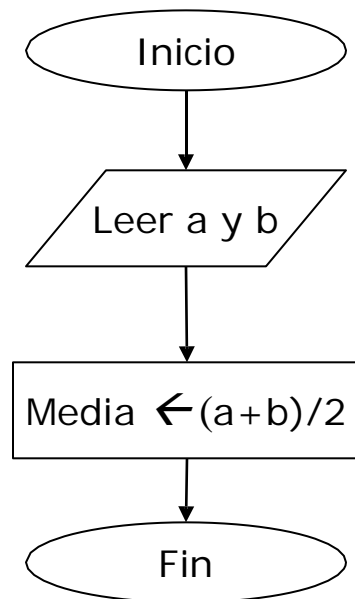
mientras



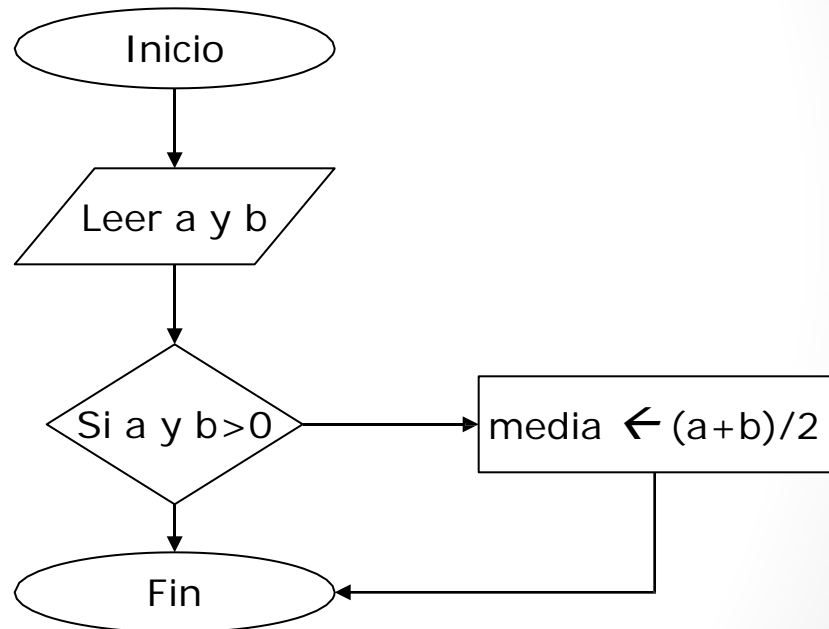
Nota: expresión booleana es una expresión cuyo resultado es o verdadero o falso

# Diagramas de Flujo. Ejemplos

**Calcular la media de dos números**



**Calcular la media de dos números si son positivos**



# Pseudocódigo

- la solución del problema se escribe utilizando palabras del lenguaje natural.
- Paso intermedio entre la solución del problema y su codificación en un determinado lenguaje de programación
- Los programas en pseudocódigo suelen incluir:
  - Nombre del programa
  - Entorno: descripción de los elementos a utilizar (variables, constantes), indicando su tipo.
  - Instrucciones necesarias para resolver el problema (en lenguaje natural)

```
Programa1  
  
ENTORNO  
    variables  
  
INICIO  
    instruc_1  
    instruc_n  
FIN
```

```
Media  
  
ENTORNO  
    a,b,media -- son números reales  
  
INICIO  
    Leer a y b  
    media  $\leftarrow (a+b)/2$   
    escribir media  
FIN
```

# Ejemplos de Pseudocódigo

- Estructuras alternativas: si

**Si** (expresión) **entonces**

I1,I2,...

**Fin si**

**Si** (expresión) **entonces**

I1,I2,...

**Sino**

S1,S2,...

**Fin si**

# Ejemplos de Pseudocódigo

- Estructuras repetitivas
  - Muchas veces es necesario repetir una misma instrucción o un bloque de instrucciones
  - Ejemplos
    - Algoritmo que escriba 100 veces hola
    - Algoritmo que escriba los números del uno al 100
    - Algoritmo que sume 100 números introducidos por teclado
  - Dos tipos: 'para' y 'mientras'

**Mientras** (expresión) hacer

l1,l2,...

**Fin mientras**

**Para** variable **variando desde** inicio **hasta** fin **de** cantidad **en** cantidad **hacer**

l1,l2,....

**Fin para**

# Ejemplo de Pseudocódigo

Escribir un algoritmo que dados dos números diga cuál es mayor, el primero o el segundo

Programa Mayor

Entorno

VAL1: Primer valor introducido por teclado, real

VAL2: Segundo valor, real

Inicio Programa

Leer Val1

Leer Val2

Si ( VAL1>VAL2) entonces

escribir ' El mayor valor es el primero'

sino

Si ( VAL1<VAL2) entonces

escribir 'El mayor valor es el segundo'

sino

escribir 'Ambos valores son iguales'

Fin Si

Fin Si

Fin Programa



# Ejemplo de Organigrama

