

# Tema 2. Concepto de Algoritmo

---

## Objetivos

---

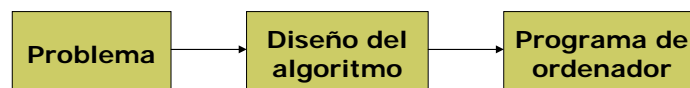
- ❑ Definir el concepto de algoritmo
- ❑ Resolver problemas sencillos mediante el diseño de un algoritmo
- ❑ Conocer los tipos de datos elementales y las operaciones sobre ellos

## Tema 2. Concepto de algoritmo

1. Resolución de problemas
2. Definición de algoritmo
3. Tipos de datos
4. Elementos de un algoritmo
5. Variables, constantes y expresiones
6. Precedencia de los operadores

## Resolución de problemas

- ¿Cómo se resuelve un problema mediante la programación?
  - Definir claramente el problema
    - Identificación de datos de entrada, de salida, de variables intermedias...
  - Separar entre
    - **Fase de solución general** al problema: diseño de un algoritmo
    - **Fase de solución específica**: programa de ordenador escrito en algún lenguaje de programación



## Pasos para resolver un problema

- ❑ Diseño del algoritmo
- ❑ Codificación del algoritmo mediante un programa escrito en lenguaje de programación adecuado
- ❑ Ejecución y validación de programa

## Definición de algoritmo

- ❑ La palabra algoritmo proviene del nombre del matemático persa del siglo IX Muhammad ibn Musa al-Jwarizmi
- ❑ Un algoritmo es una secuencia ordenada de pasos, exenta de ambigüedad, que permite la resolución de un problema determinado
- ❑ Características de un algoritmo
  - Preciso: exento de ambigüedad
  - Finito: si se sigue el algoritmo, se debe terminar en algún momento
  - Definido: si se sigue dos veces el algoritmo con los mismo datos de entrada, la salida debe ser la misma

## Algoritmo para cambiar una bombilla

---

1. Retirar la bombilla fundida
  1. Colocar una escalera debajo de la bombilla
  2. Subir la escalera
  3. Desenroscar la bombilla en el sentido contrario de las agujas del reloj
  4. Bajar la escalera
2. Poner una bombilla nueva
  1. Coger la bombilla nueva
  2. Subir la escalera
  3. Enroscar la bombilla en el sentido de las agujas del reloj
  4. Bajar la escalera
  5. Retirar la escalera

## Notaciones para describir algoritmos

---

- Método informal
  - Lenguaje natural
    - Ventajas: comprensible e intuitivo
    - Inconveniente: impreciso
- Métodos formales
  - Pseudocódigo
    - Lenguaje natural limitado y sin ambigüedad
  - Diagramas
    - Diagramas de flujo u organigramas
      - Utiliza un conjunto de símbolos para representar cada estructura de control y mediante líneas de flujo se indica el orden en que se realiza el flujo lógico del algoritmo
    - Diagramas de Nassi-Schneiderman o Diagramas N-S
      - Los pasos sucesivos se escriben en cajas con distintas formas según la estructura de control que representen

## Descripción de algoritmos

---

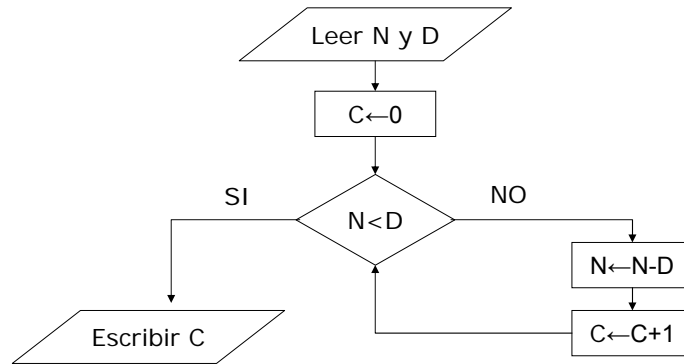
- Metodología para describir algoritmos
  - Entrada
  - Proceso
  - Salida

## Algoritmo para calcular el cociente de una división entera sin usar la operación de división (Lenguaje Natural)

---

1. Leer el numerador N y el denominador D
2. Inicializar el cociente a 0 ( $C \leftarrow 0$ )
3. Si  $N < D$  ir al paso 7
4. Restar a N el valor de D ( $N \leftarrow N - D$ )
5. Incrementar en uno C ( $C \leftarrow C + 1$ )
6. Si  $N \geq D$  volver al paso 4
7. Mostrar el valor de C

## Algoritmo para calcular el cociente de una división entera (Organigrama)



## Tipos de datos

- Conjunto de valores junto con las operaciones sobre esos valores
- Tipos elementales
  - Entero
  - Real
  - Carácter
  - Lógico

## Tipos de datos

---

- Tipo Entero
  - Conjunto de los números enteros positivos y negativos
  - En la práctica están limitados por el almacenamiento -> Overflow
  - Operaciones básicas: aritméticas, relacionales
  
- Tipo Real
  - Conjunto de los números reales positivos y negativos (parte entera y parte decimal separada por un punto)
  - En la práctica están limitados por el almacenamiento -> Overflow
  - Operaciones básicas: aritméticas, relacionales

## Tipos de datos

---

- Tipo carácter
  - Conjunto de elementos de un conjunto de caracteres predefinidos o alfabeto (normalmente se utiliza el conjunto de caracteres ASCII)
  - Generalmente se almacenan internamente como datos enteros (código ASCII)
  - Operaciones de conversión de enteros a caracteres y viceversa y relacionales
  
- Lógicos
  - Valores entre 2 posibles, es decir, valores binarios (verdad/falso, sí/no)
  - En algunos lenguajes no se implementan sino que se consideran como verdad cualquier valor distinto de 0
  - Operaciones: operadores lógicos (AND, OR y NOT)
  - Fundamentales para evaluar condiciones

## Operadores lógicos

		AND	OR	NOT	
V	V	V	V	F	F
V	F	F	V	F	V
F	V	F	V	V	F
F	F	F	F	V	V

- $(3 < 4)$  OR  $(3 = 4)$
- $(\text{Respuesta} = 'S')$  AND  $(\text{salir} = 'N')$
- NOT  $((3 < 4)$  OR  $(3 = 4))$
- $(\text{caracter} = 'A')$  AND  $(\text{caracter} = 'B')$  es siempre FALSO
- $(N \leq 10)$  OR  $(N > 10)$  es siempre VERDAD

## Elementos de un algoritmo

- Variables, constantes y expresiones
- Estructuras de control
  - Secuencial
    - Operación de asignación
    - Operación de entrada
    - Operación de salida
  - Condicional
  - Repetitiva

## Variables

---

- ❑ Una variable es un objeto de datos que posee un valor conocido por su nombre (identificador)
- ❑ Toda variable tiene un tipo asociado
- ❑ Su valor puede ser modificado pero en un instante dado sólo contiene un valor determinado
- ❑ Es necesario declarar las variables al principio del algoritmo
  - Elegir el nombre mediante un identificador
  - Determinar el tipo de datos que va a contener
- ❑ Ejemplos

```
Sueldo: real
Edad: entero
Letra_nif: caracter
```

## Constantes

---

- ❑ Objeto de datos que contiene un valor invariable conocido por su identificador
- ❑ Es necesario inicializar las constantes al principio del algoritmo
  - Esto determina el tipo de la constante
- ❑ Ejemplos

```
PI=3.14
SALUDO="Hola mundo"
MAXIMO=500
RESPUESTA='S'
```

## Identificadores

---

- ❑ Un identificador es una secuencia de caracteres que permite identificar de forma única a cada elemento/objeto de un algoritmo
- ❑ No todos los caracteres son válidos por lo que normalmente se usan letras, dígitos y subrayado (  )
- ❑ El primer carácter debe ser una letra o subrayado
- ❑ Pueden tener cualquier longitud dentro del límite que imponga el compilador
- ❑ Las palabras reservadas del lenguaje no pueden utilizarse como identificadores
- ❑ Algunos lenguajes no distinguen entre caracteres en mayúsculas y minúsculas (Pascal, Modula-2) y otros sí (C, Java)
- ❑ Reglas para crear identificadores
  - Usar solo letras, dígitos y subrayado
  - Usar minúsculas para identificadores de variables
  - Usar mayúsculas para identificadores de constantes
  - Para identificadores de funciones con más de una palabra usar subrayado para separar las palabras o mayúsculas para la primera letra de cada palabra (Ej. CalcularFechaNacimiento)

## Valores constantes

---

- ❑ También llamados valores literales
- ❑ Se trata de valores que aparecen explícitamente en un algoritmo y que no tienen identificador asociado (sólo pueden ser referenciados por su propio valor)
- ❑ Ejemplos
  - 0
  - "Introduzca el dato de entrada"
  - 100
  - 7.5
  - 'x'

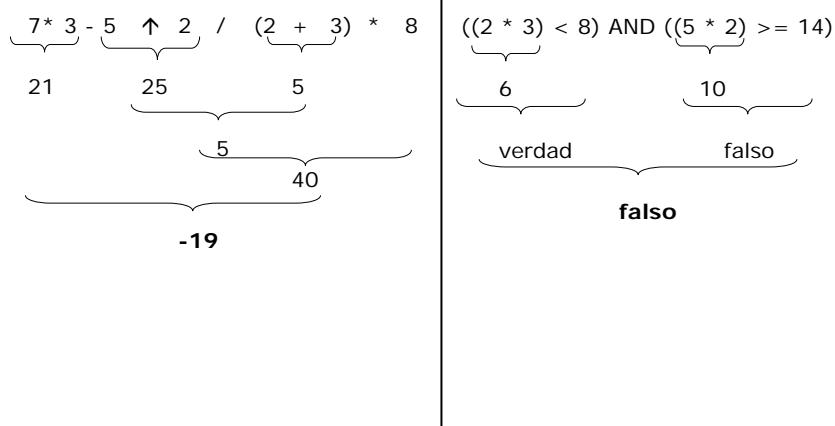
## Expresiones

- Una expresión es la combinación de distintas variables, constantes, valores constantes, operadores, paréntesis y nombres de funciones
- Toda expresión tiene en todo momento un valor concreto que es el resultado de evaluarla de izquierda a derecha, siempre teniendo en cuenta la precedencia de los operadores y los paréntesis que aparecen en la expresión
- El valor devuelto tras la evaluación de una expresión determina el tipo de la expresión (entera, lógica ...)
- Ejemplos
  - $5+6-5*2$
  - $(5+6-5)*2$
  - $(A<0) \text{ OR } B \text{ AND } C$

## Precedencia de los operadores

Operadores Lógicos	Operadores Relacionales	Operadores Aritméticos
NOT	<, >, =, <=, >=, <>	Operadores de signo (+, -)
AND		Multiplicación y división (*, /, %)
OR		Suma y resta (+, -)

## Ejemplo de árboles de precedencia



## Resumen

- Resolución de problemas mediante el diseño de algoritmos y la posterior codificación de un programa en un lenguaje de alto nivel
- Algoritmo
  - Definición
  - Características
  - Etapas para describir algoritmos
  - Elementos básicos
- Tipos de datos
- Identificadores, variables, constantes y expresiones
- Precedencia de los operadores