

# UD1 . Fundamentos de programación

Verónica Mascarós

**Curs 23-24**



```
e = m(b, " ");  
-1 < e && b.splice(e, 1);  
e = m(b, void 0);  
-1 < e && b.splice(e, 1);  
e = m(b, "");  
-1 < e && b.splice(e, 1);  
for (c = 0; c < d && c < b.length;  
    a += b[c].b + ", ", n.push(  
}  
for (g = 0; g < f;) {  
    e = Math.floor(b.length *  
    d.c + "</span></li>"), b[e]  
}  
for (; c < b.length; c++) {  
    void 0 !== b[c] && ("para  
}  
function(b);  
    "single").h("mode_9  
    ent(
```

# Algoritmos

- Un ALGORITMO es un conjunto ordenado y finito de operaciones que permiten resolver un problema.
- Es independiente del lenguaje de programación y de los dispositivos en el que se ejecuta.

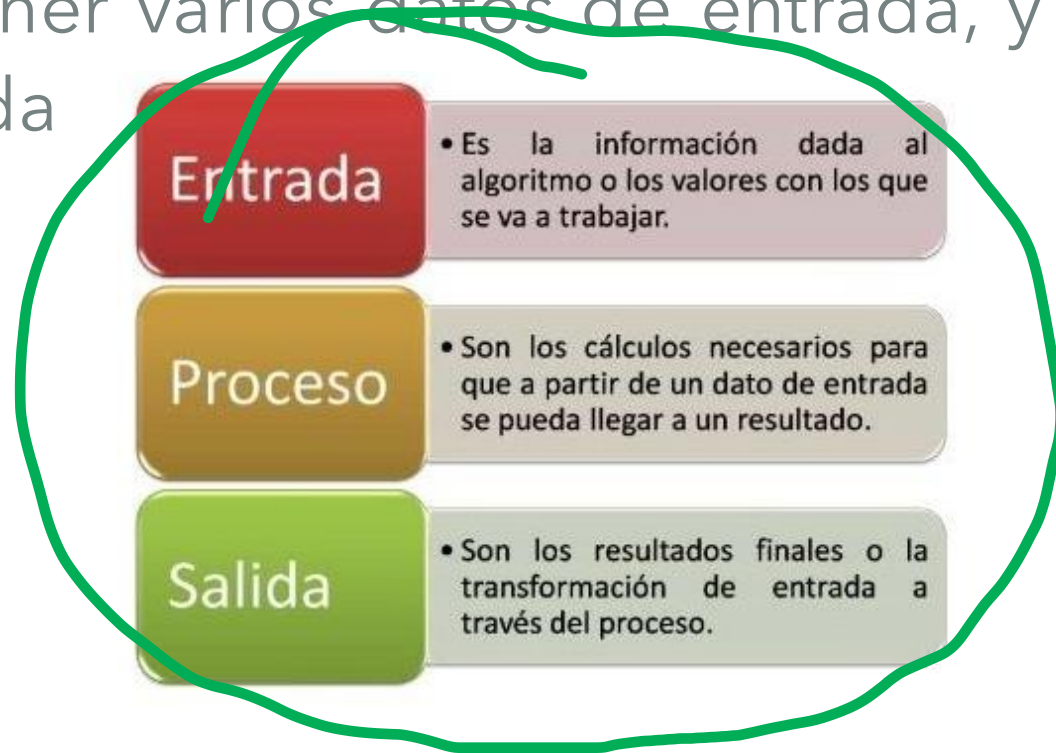


# Algoritmos

- Un algoritmo debe ser PRECISO, FINITO y DEFINIDO.
- **PRECISO**: debe tener instrucciones claras e indicar el orden de realización de cada paso
- **FINITO**: tiene un número finito de pasos, debe terminar en un tiempo finito. Si no terminara nunca, no se resolvería el problema.
- **DEFINIDO**: Todas las operaciones deben estar definidas de forma precisa y sin ambigüedad, de modo que si se sigue el algoritmo dos veces, se debe obtener el mismo resultado.

# Algoritmos

- Para definir un algoritmo debemos tener en cuenta que consta de 3 partes: la ENTRADA, el PROCESO y la SALIDA.
- Un algoritmo puede tener varios datos de entrada, y como mínimo un dato de salida



# Algoritmos

- Un claro ejemplo de algoritmo es una **receta de cocina**, donde tenemos unos pasos a seguir en un orden y deben estar bien definidos, tiene un tiempo finito y tiene unos datos de entrada (ingredientes) y una salida (el plato).
- Por ejemplo, el algoritmo para freír un huevo podría ser el siguiente:
- **Datos de entrada:** Huevo, aceite, sartén, fuego.
- **Datos de salida:** huevo frito.
- **Procedimiento:**
  1. Poner el aceite en la sartén.
  2. Poner la sartén al fuego.
  3. Cuando el aceite esté caliente, romper el huevo e introducirlo.
  4. Cubrir el huevo de aceite.
  5. Cuando el huevo esté hecho, retirarlo.



# Programa

- La codificación de un ALGORITMO mediante un lenguaje de programación se llama PROGRAMA.
- Por tanto, un programa informático no es más que un conjunto de instrucciones para un ordenador.
- Los lenguajes de programación son un medio para expresar el algoritmo.

# Elementos de un programa

- En un programa podemos encontrar principalmente:
- Constantes.
- Variables.
- Expresiones.
- Operadores.
  - Relacionales,
  - Aritméticos,
  - Lógicos o Booleanos,
  - Alfanuméricos y
  - Paréntesis.

# Elementos de un programa

- Las constantes y variables podemos decir que son objetos.
- Llamamos OBJETO de un programa a todo lo que se puede manipular por diferentes instrucciones que forman parte del programa.
- En ellos se almacenarán tanto los datos de entrada como los de salida (resultados).



# Elementos de un programa

- Todos los objetos tienen los siguientes atributos:
  - **Nombre:** el identificador del objeto.
  - **Tipo:** conjunto de valores que puede tomar. Por ejemplo: números enteros.
  - **Valor:** elemento del Tipo que se le asigna.
- Por ejemplo: `int miVariable = 2`
- (tipo) (nombre) (valor)

# Constantes

- Su **valor no varía** a lo largo de la ejecución del programa.
- Una constante es como ponerle un nombre a un valor concreto, por lo que se utiliza su nombre cada vez que queremos referenciar ese valor.
- Por ejemplo:
- $\text{IVA} = 21$
- $\text{pi} = 3,14$  --> Si queremos calcular la longitud de una circunferencia de  $\text{radio}=4$ , calcularíamos:  $2 \cdot \text{pi} \cdot \text{radio}$

# Constantes

- Las constantes se utilizan sobre todo para simplificar las modificaciones y mantenimientos de los programas.
- Imagina que estamos escribiendo un programa contable o financiero, donde utilizamos muchas veces operaciones con el IVA.
- Si en un futuro el importe cambia, sólo tendríamos que modificar el valor de la constante IVA una vez, en cambio, si utilizamos cada vez 0,21 debemos modificar ese valor en cada operación.

# Variables

- Son objetos cuyo **valor puede ser modificado** a lo largo de la ejecución de un programa.
- Por ejemplo: una variable para calcular el área de una determinada circunferencia, una variable para calcular el importe de una factura, etc.

# Operadores

- Son los símbolos que hacen de enlace entre los argumentos de una expresión:
- 1.- Relacionales.
- 2.- Aritméticos.
- 3.- Lógicos o Booleanos.
- 4.- Operador Alfanumérico (+).
- 5.- Paréntesis ().

# Operadores relacionales

- Se utilizan para formar expresiones que al ser evaluadas devuelven un valor booleano (Verdadero o Falso).

**X=5; Y=3**

Operación	Operador	Ejemplo	Comparación	Resultado/Binario
Mayor que	>	X > Y	¿Es X mayor que Y?	Verdadero (1)
Menor que	<	X < Y	¿Es X menor que Y?	Falso (0)
Mayor o igual que	>=	X >= Y	¿Es X mayor o igual que Y?	Verdadero (1)
Menor o igual que	<=	X <= Y	¿Es X menor o igual que Y?	Falso (0)
Igual a	==	X == Y	¿Es X igual a Y?	Falso (0)
Diferente a	!=	X != Y	¿Es X diferente a Y?	Verdadero (1)

# Operadores aritméticos

- Se utilizan para realizar operaciones aritméticas:



Operador	Definición
+	Suma
-	Resta
*	Multiplicación
^	Potencia
/	División
%	Resto de la división

- Ejemplos:

Expresión	Resultado
3 + 5 - 2	6
24 % 3	0
8 * 3 - 7 / 2	26

# Operadores lógicos o booleanos

- Se utilizan con expresiones booleanas (V o F). Los más habituales:
  - Operador NOT (No lógico o negación): invierte el valor
  - Operador AND (Y lógica o conjunción): devuelve Verdadero si todos los valores son verdaderos, en caso contrario devuelve Falso
  - Operador OR (O lógica o disyunción): devuelve Verdadero si algún valor es verdadero





# Operadores lógicos o booleanos

Los operadores lógicos NOT, AND y OR se evalúan según las tablas de verdad, siendo X e Y expresiones booleanas cuyo resultado de las operaciones lógicas viene determinado por dichas tablas:

Operador NOT	
X	NOT X
F	V
V	F

Operador AND		
X	Y	X AND Y
F	F	F
F	V	F
V	F	F
V	V	V

Operador OR		
X	Y	X OR Y
F	F	F
F	V	V
V	F	V
V	V	V

- Ejemplos

Expresión	Resultado
$9 = (3 * 3)$	Verdadero
$3 <> 2$	Verdadero
$9 = (3 * 3) \text{ Y } 3 <> 2$	Verdadero
$3 > 2 \text{ Y } b < a$	Verdadero Y Falso = Falso
$3 > 2 \text{ O } b < a$	Verdadero O Falso = Verdadero
$\text{no}(a < b)$	No Verdadero = Falso
$5 > 1 \text{ Y NO}(b < a)$	Verdadero Y no Falso = Verdadero

# Operador alfanumérico

- Une datos de tipo alfanumérico (texto). También se llama concatenación.

*COMILLAS*

- Ejemplos:

Expresión	Resultado
"Ana " + "López"	Ana López
"saca" + "puntas"	sacapuntas

## Paréntesis

- Sirve para agrupar o anidar expresiones. Ejemplo:
- $(3*2) + (6/2) \rightarrow$  Resultado: 9

# Orden de evaluación de los operadores

- A la hora de resolver una expresión, el orden a seguir es el siguiente:
  1. Paréntesis (comenzando por los más internos)
  2. Signo
  3. Potencias
  4. Producto y divisiones
  5. Sumas y Restas
  6. Concatenación
  7. Relacionales
  8. Negación
  9. Conjunción
  10. Disyunción
- La evaluación de operadores de igual orden se realiza de izquierda a derecha