

# Taller de Programación

## 02 - Elementos para crear algoritmos

Mgs. Lic. Marcos Prunello

2018-07-30

Hemos definido a un algoritmo como una lista de instrucciones que serán traducidas con un lenguaje de programación para ser ejecutadas por computadora. En este sentido, el concepto físico de la máquina o computadora hace referencia a la necesidad de contar con un **procesador** para resolver el problema. Se define como

- **Procesador:** todo agente capaz de entender las órdenes del algoritmo y ejecutarlas.
- **Ambiente o Entorno:** recursos que forman parte del sistema en el cual se ejecuta el algoritmo disponibles para ser utilizados por el procesador.
- **Acciones:** cada una de las instrucciones que componen el algoritmo y modifican el entorno de alguna manera.

# Ejemplo 1

- **Problema:** preparar una tortilla de 6 huevos.
- **Entorno:** una mesa, una hornalla, una sartén, un plato, un tenedor, aceite, una fuente con huevos, un tarro de basura.
- **Procesador:** un adulto.
- **Acciones comprensibles por el procesador:** agarrar un huevo, romper un huevo en un plato, batir huevos, poner aceite, poner en la sartén, poner al fuego, retirar del fuego, tirar las cáscaras, encender el fuego.

¿Cuál es un algoritmo adecuado para solucionar este problema?

## Ejemplo 2

- **Problema:** calcular el factorial del número 5.
- **Entorno:** se dispone de una calculadora común.
- **Procesador:** un adulto.
- **Acciones comprensibles por el procesador:** pulsar teclas de la calculadora.

¿Cuál es un algoritmo adecuado para solucionar este problema?

# Acciones primitivas o compuestas

- **Primitiva:** acción sencilla directamente realizable por el procesador.
- **Compuesta:** compuestas por una sucesión de acciones primitivas.

- **Objeto:** cada pieza de información que compone el entorno y es manipulada por el procesador.
- Tipos básicos de objetos:
  - **Numérico**
  - **Caracter**
  - **Lógicos**
- Al nombre de un objeto se le dice **identificador**.

# Variables y constantes

**Variables:** objetos que almacenan temporalmente un valor durante la ejecución de un proceso y su contenido puede cambiar mientras corre el programa. Una variable es como una caja que contiene una etiqueta con su identificador y un valor:

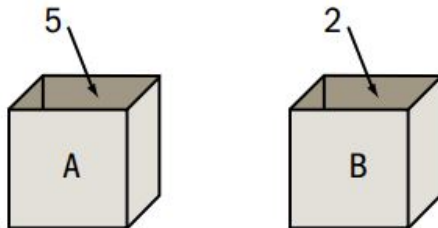


Figure 1: Las variables A y B contienen a los valores 5 y 2 respectivamente.

**Constantes:** objetos que tienen siempre necesariamente un valor fijo. Pueden usarse literalmente o también a través de un objeto que asocie un identificador al valor constante.

- Al expresar nuestros algoritmos en pseudocódigo declaramos (informamos) al inicio las variables y constantes necesarias para resolver el problema.



# Acción de asignación

- Para hacer que una variable guarde un determinado valor se recurre a una **acción de asignación**.
- En pseudocódigo:

```
lugarNacimiento <- "Bombal"  
pesoNacimiento <- 3.400
```

- En la asignación se pueden usar valores de otras variables. ¿Cuál es el valor de estas variables al finalizar?:

```
meses <- 2  
dias <- meses  
meses <- 7  
saldo <- meses  
saldo <- saldo + 10
```

- Los algoritmos determinan operaciones a efectuar entre objetos, las cuales pueden ser de distinto tipo: suma, resta, concatenación, procesos lógicos, etc.
- Los elementos que describen el tipo de operación a realizar entre dos objetos se denominan **operadores**.

# Operadores aritméticos

- Permiten realizar operaciones matemáticas con objetos de tipo numérico.
- Signos para operadores aritméticos que se pueden utilizar en el pseudocódigo:

Signo	Significado
+	Suma
-	Resta
	Multiplicación
/	División
	Potenciación
MOD	Resto de la división entera

# Operadores aritméticos

- El orden de prioridad de los operadores aritméticos es tal como estamos acostumbrados en matemática.
- Las expresiones entre paréntesis se evalúan primero; si hay paréntesis anidados se evalúan desde adentro hacia afuera.
- Dentro de una misma expresión, los operadores se evalúan en este orden:
  - 1 Potenciación
  - 2 Multiplicación, división, módulo
  - 3 Suma, resta
- Si la expresión presenta operadores con igual nivel de prioridad, se evalúan de izquierda a derecha.
- Ejemplos:

$$3 + 5 * (10 - (2 + 4)) =$$

$$8 * (7 - 6 + 5) \text{ MOD } (1 + 8 / 2) - 1 =$$

# Operadores relacionales

- Sirven para comparar dos valores de cualquier tipo y dan como resultado un valor lógico: *verdadero* ("V") o *falso* ("F").
- Signos para operadores lógicos que se pueden utilizar en el pseudocódigo:

Signo	Significado
>	Mayor que
<	Menor que
>=	Mayor o igual que
<=	Menor o igual que
=	Igual a
^=	Distinto a

# Operadores relacionales

Ejemplos de la utilización de operadores relacionales:

Operación relacional	Resultado
$5 > 3$	verdadero
$5 \leq 3$	falso
$3 * 4 = 10 + 2$	verdadero
$3 * 4 \neq 15 - 4$	verdadero

- Sólo toman operandos de tipo lógico y producen también un resultado lógico.
  - El operador lógico **Y** devuelve un valor **verdadero** sólo si son verdaderas las dos expresiones que vincula. Ejemplo:  $(3 > 2) \text{ Y } (3 > 5) \equiv V \text{ Y } F \equiv F$ .
  - El operador lógico **O** devuelve un valor **verdadero** si al menos una de las dos expresiones que vincula es verdadera. Ejemplo:  $(3 > 2) \text{ O } (3 > 5) \equiv V \text{ O } F \equiv V$ .
  - El operador lógico **NO**, niega un valor lógico, es decir, devuelve el opuesto. Ejemplo:  $\text{NO } (3 > 2) \equiv \text{NO } V \equiv F$ .

# Operadores lógicos

Las **tablas de la verdad** se utilizan para mostrar el resultado de los distintos tipos de operaciones lógicas:

Valor1	Operador	Valor2	Resultado
F	Y	F	F
F	Y	V	F
V	Y	F	F
V	Y	V	V
F	O	F	F
F	O	V	V
V	O	F	V
V	O	V	V
	NO	F	V
	NO	V	F



- Acción LEER: para requerir que el usuario del programa provea cierta información y registrar la misma como un valor que debe ser asignado a una variable.
- Acción ESCRIBIR: se utiliza cuando el algoritmo debe mostrar un resultado calculado.

# Ejemplo de declaraciones, asignación, operaciones, entrada y salida

ALGORITMO: "Calcular área de un círculo"

COMENZAR

CONSTANTE numérica pi

VARIABLE numérica radio, area

pi <- 3.1416

LEER radio

area <- pi \* radio \*\* 2

FIN