

0101010  
0100101  
1101010

# UD1.- INTRODUCCIÓ AL DISSENY D'ALGORISMES

0485 - Programació  
1er DAW/DAM

# 0. CONTINGUTS

---

- Problemes, algorismes i programes
- Llenguatges de programació
- Disseny de la solució d'un problema
- Disseny d'algorismes
- Elements bàsics
- Diagrames de flux

# 1. PROBLEMES, ALGORISMES I PROGRAMES

## Què és la programació?

- Conjunt de **tècniques**, **processos** i **ferramentes** que ens permeten **definir el comportament** d'una **màquina** amb la fi de **resoldre un determinat problema**.



# 1.1 QUÈ ÉS UN ALGORISME?

---

- Conjunt **d'accions** o **seqüència d'operacions** que, executades en un ordre determinat, resolen un problema.
- Al llarg de les nostres vides hem après de forma **empírica** el **desenvolupament d'algorismes**.
  - *Tornar el canvi al pagar una compra*
  - *Prendre una decisió*

# 1.1 QUÈ ÉS UN ALGORISME?

---

Algorisme “pagar el pàrquing”

## INICIO

Traure el Ticket

Introduir ticket en el caixer

**SI** tenim *val de descompte*

Introduir val en el caixer

**FIN SI**

Veure import calculat

Pagar import

Agarrar Ticket

## FIN

Com seria l'algorisme de **preparació d'una  
tassa de te?**

# 1.1 QUÈ ÉS UN ALGORISME?

---

- Algorisme “preparar tassa de té”

## INICIO

Agafar la tetera  
Omplir-la d'aigua  
Encendre el foc  
Posar la tetera al foc  
Esperar a que bulla l'aigua  
Agafar la bossa de te  
Introduir-la en la tetera  
Esperar 1 minut  
Tirar el té dins de la tassa

## FIN

# 1.1 QUÈ ÉS UN ALGORISME?

---

## Característiques

- Un **algorisme** ha de ser:
  - **Concret**: Deu indicar amb exactitud l'operació a dur a terme i el seu ordre.
  - **Determinista**: Partint de les mateixes condicions de entrada el resultat ha de ser el mateix.
  - **Finit**: Ha de finalitzar en un nombre determinat de passos
  - Especificació **no ambigu**.
  - **Independent** del llenguatge de programació.

# 1.1 QUÈ ÉS UN ALGORISME?

## Algorismes computacionals

- Dissenyats per a ser executats per un computador
  - **PageRank** de Google.
  - **NewsFeed** de Facebook.
  - **Intel·ligència artificial** (Reconeixement facial, identificació de patologies,...).
  - **Moviments d'un joc.**

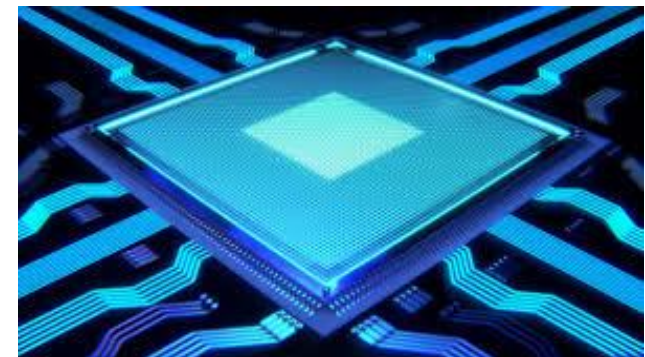




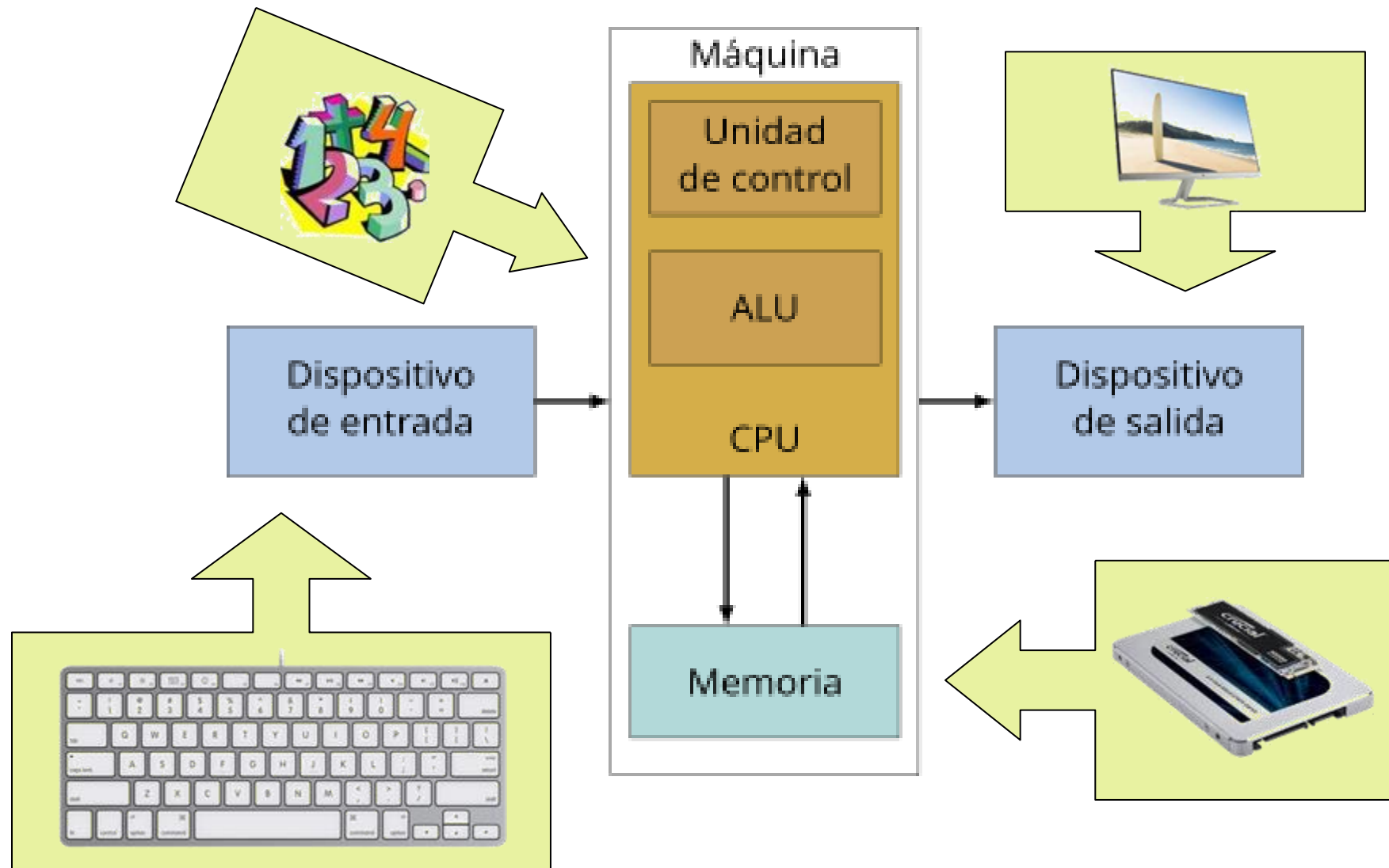
## 1.2 QUÈ ÉS UN PROGRAMA?

---

- **Conjunt d'instruccions** que se li proporcionen a un ordinador en un **llenguatge que ell pot entendre**, per a dir-li exactament què volem que faça.
- Els **processadors** sols son capaços d'executar una serie de tasques elementals.
  - Hem **d'implementar l'algorisme** fent ús d'aquestes **instruccions bàsiques**.
  - És el que anomenem **programar**.



# 1.3 INSTRUCCIONS



## 1.3 INSTRUCCIONS. Exemples

### Exemple d'ordres e instruccions

**Mostrar** una dada **per pantalla**

**Llegir** un número **del teclat**

**Guardar** un **resultat** en **memòria**

Fer una **operació aritmètica** (+,-,/,)

Fer una **comparació** (edat > 18)

## 2. LENGUATGES DE PROGRAMACIÓ

---

- Els **llenguatges de programació** ens permeten especificar les tasques que ha de seguir l'ordinador per executar un algoritme.
  - Possibiliten la **comunicació home-màquina**.
  - Es tracta d'un **llenguatge artificial**



## 2. LLENGUATGES DE PROGRAMACIÓ



C#



Objective-C

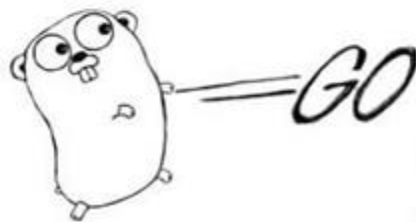
C++



python



Perl



JavaScript

THE  
C

PROGRAMMING  
LANGUAGE



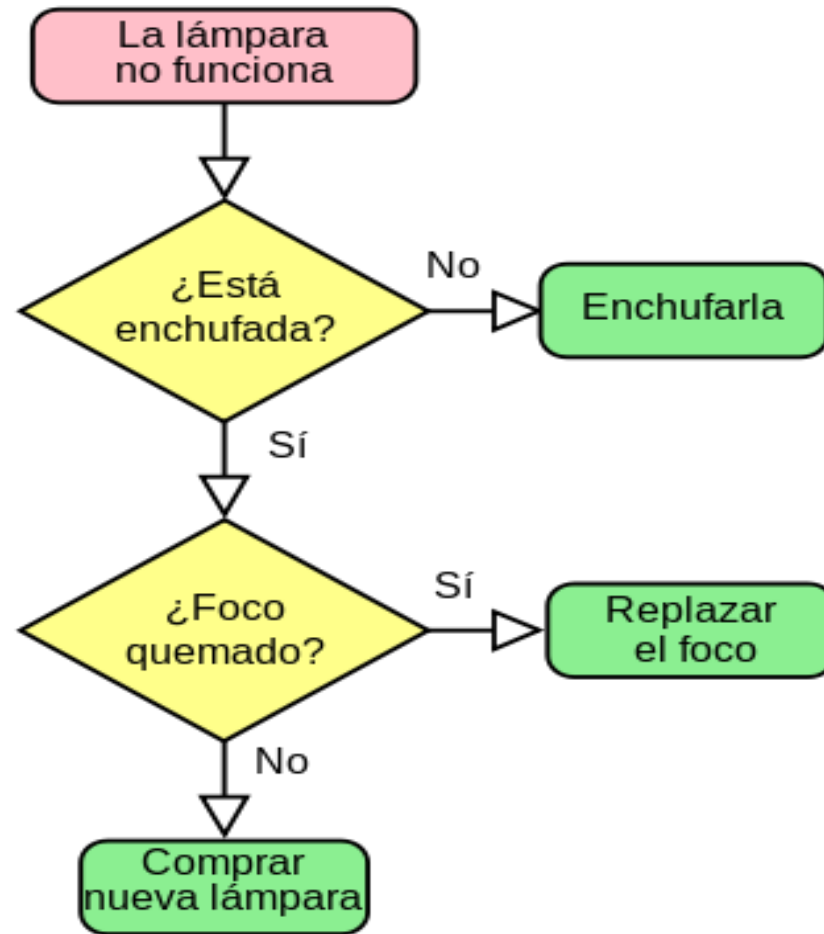
### 3. DISSENY DE LA SOLUCIÓ A UN PROBLEMA

1. **Entendre** l'enunciat del problema.
2. **Dissenyar** l'algorisme de funcionament.
  - **Tècniques de representació (diagrames de flux, pseudocodi....)**
3. **Codificar** amb un llenguatge de programació.
4. **Provar i tornar a redissenyar** (*si fora necessari*).



## 4. DISSENY D'ALGORISMES: Diagrama de flux

Nom d'algoritme  
**reparaLlum**



## 4. DISSENY D'ALGORISMES

---

### Elements bàsics

Els **algorismes**, normalment, tenen els següents **elements**:

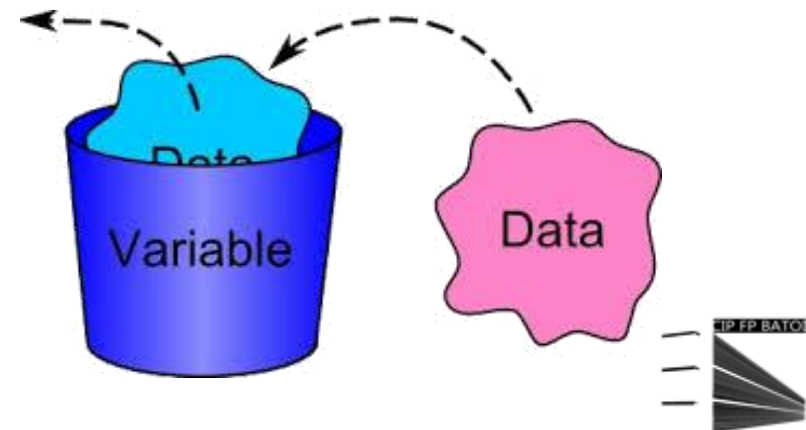
- **Identificador:** nom de l'algorisme.
- **Operacions de entrada de dades:** permet especificar les dades sobre les que operarà l'algorisme (Ex. *capturar dades per teclat*).
- **Operacions d'accés a memòria:** permeten guardar el resultat d'una instrucció per a utilitzar-lo en un moment posterior. (Ex. *guardar el resultat d'una operació aritmètica*).
- **Operacions d'eixida de dades:** permet mostrar dades a l'usuari (Ex. *Mostrar el resultat per pantalla*).



## 4. DISSENY D'ALGORISMES

### Elements bàsics

- **Dades:** elements sobre els que opera un algorisme:
  - **Literals:** lletres, números, cadenes de text.
    - 'a' , 1 , 'programació'
  - **Variables:** espais temporals on deixar dades.
- **Assignacions:** permeten guardar valors a les **variables**
  - `nomPersona <- 'pepe'`
  - `num <- 25`



# 4. DISSENY D'ALGORISMES

## Elements bàsics

### Operadors

- **Aritmètics:** +, -, \*, /, %
- **Comparació:** >, <, >=, <=, =
- **Assignació:** <-
- **Concatenació:** ,

En combinació amb les dades  
donen lloc a **expressions**

**resultat1 <- 1 + 10**

**resultat2 <- 2 - 40**

| Operador | Definició         |
|----------|-------------------|
| +        | suma              |
| -        | resta             |
| *        | multiplicació     |
| /        | divisió           |
| %        | mòdul             |
| <        | menor que         |
| >        | major que         |
| =        | igual que         |
| >=       | major o igual que |
| <=       | menor o igual que |
| <>       | diferent de       |

# 4. DISSENY D'ALGORISMES

## Ex. ús de literals

| Expressió             | Resultat        |
|-----------------------|-----------------|
| 1 + 2                 | 3               |
| 4 * 2                 | 8               |
| 20 / 4                | 5               |
| 1 = 2                 | FALS            |
| 2 < 3                 | VERDADER        |
| 2 > 3                 | FALS            |
| 3 < 3                 | FALS            |
| 3 <= 3                | VERDADER        |
| 3 >= 3                | VERDADER        |
| 15 = 15               | VERDADER        |
| 'Hola' , 'Mundo' , 66 | 'Hola Mundo 66' |

**Op. Aritmètic**  
El resultat és un  
**numèric**

**Op. Comparació**  
El resultat és un  
**Booleà**  
VERDADER  
/ FALS

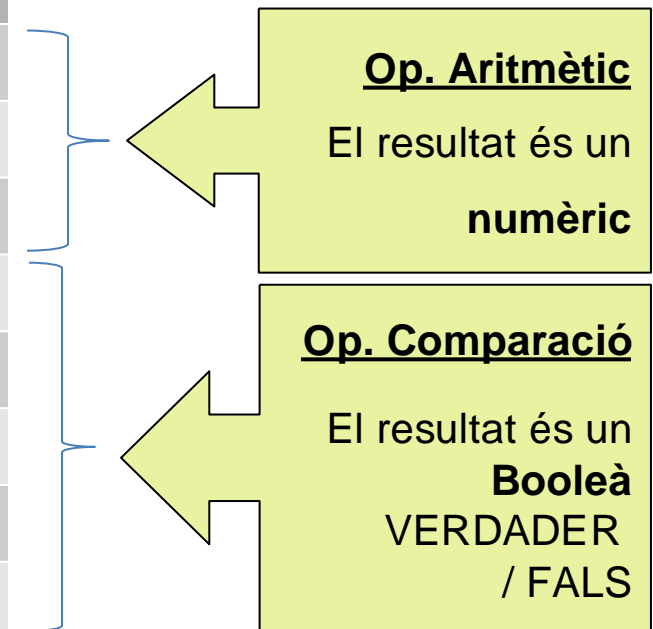
**Op. Concatenació**  
El resultat és  
una **cadena de text**

## 4. DISSENY D'ALGORISMES

### Ex. ús de variables

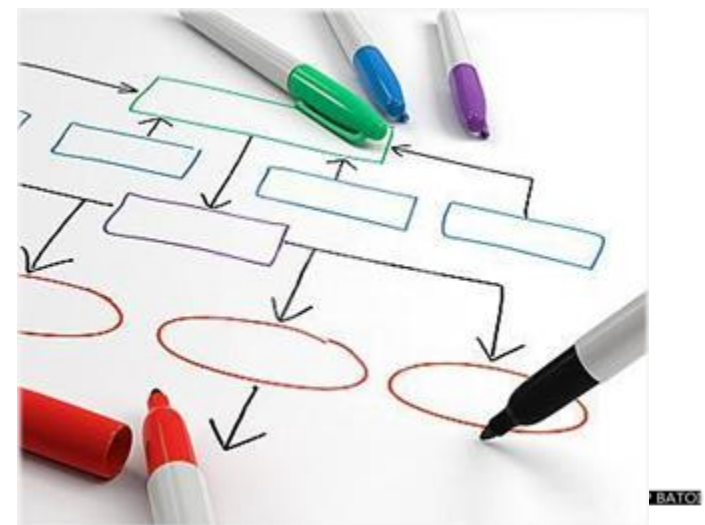
- Imaginem que tenim **2 variables A i B** amb valors **8 i 2** respectivament.

| Expressió ( A<-8, B<-2 ) | Resultat |
|--------------------------|----------|
| <b>A + B</b>             | 10       |
| <b>A * B</b>             | 16       |
| <b>A / B</b>             | 4        |
| <b>A = B</b>             | FALS     |
| <b>A &lt; B</b>          | FALS     |
| <b>A &gt; B</b>          | VERDADER |
| <b>A &lt;= B</b>         | FALS     |
| <b>A &gt;= B</b>         | VERDADER |



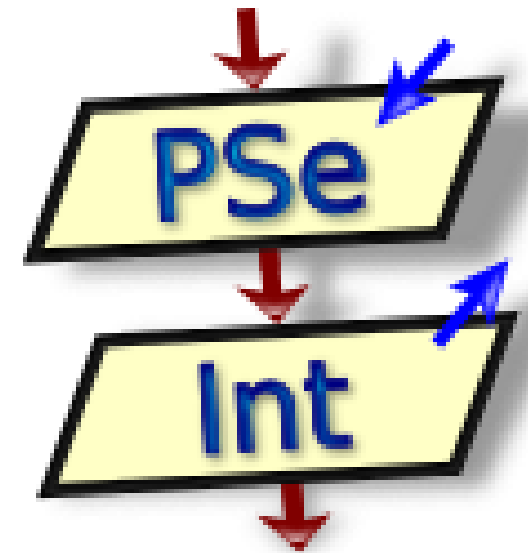
# 4.1 REPRESENTACIÓ D'ALGORISMES

- **Diagrama de flux / ordinograma**
  - Ens permet representar gràficament la lògica de un algorisme mitjançant
    - **Símbols normalitzats** (ANSI): Junt al text especifica la operació a realitzar.
    - **Fletxes**: Especifiquen el flux d'execució



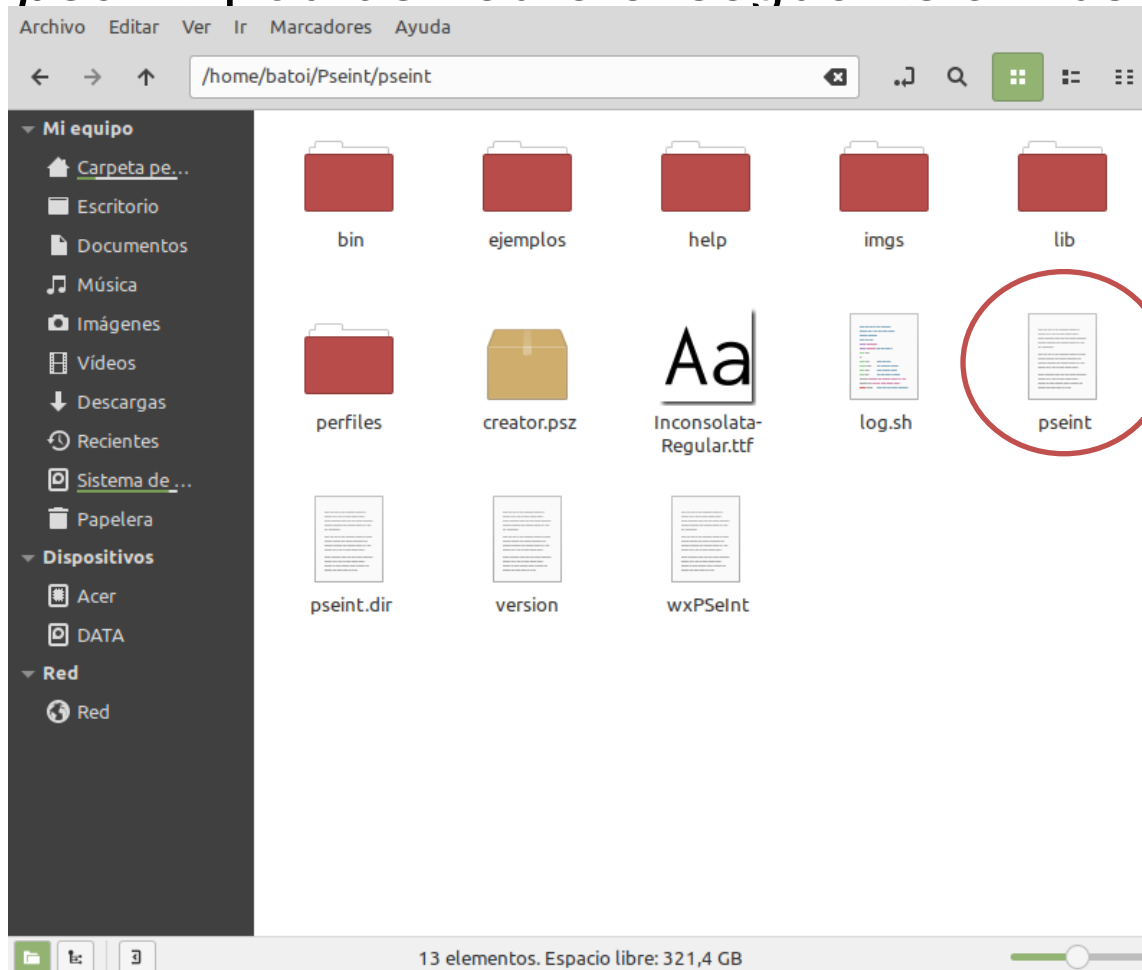
## 4.1 REPRESENTACIÓ D'ALGORISMES

- **Diagrama de flux / ordinograma**
  - En aquesta unitat, farem servir una eina molt senzilla per a l'elaboració de diagrames de flux anomenada ***Pseint que fa servir els símbols ISO normalitzats.***
  - També ens dona la possibilitat de conèixer el pseudocodi associat al diagrama i viceversa.
  - Pots descarregar-ho des d'aquest [enllaç](#).



## 4.1 REPRESENTACIÓ D'ALGORISMES Pseint

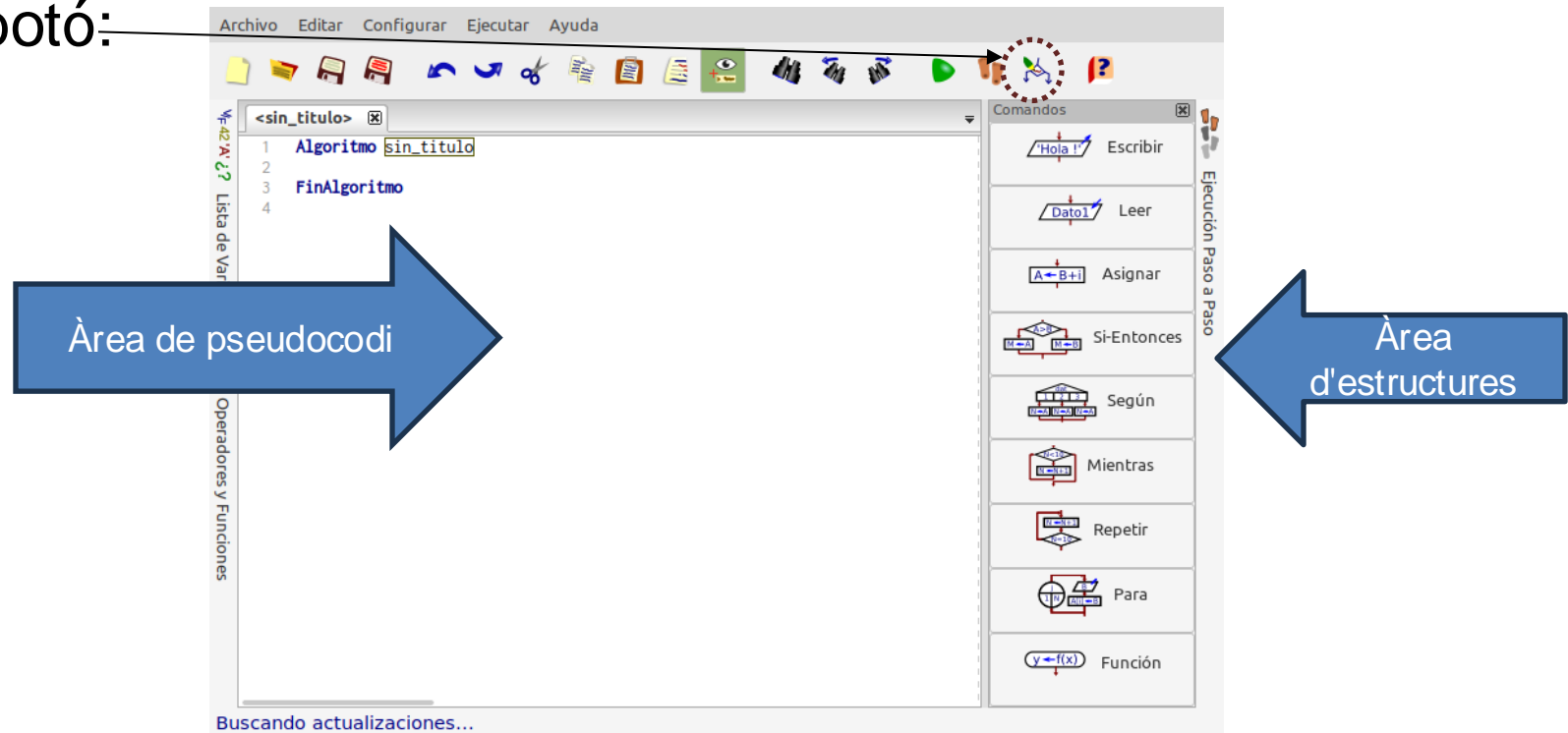
- Una vegada descomprimit i extret l'arxiu, dins de la carpeta *pseint* podràs veure el següents arxius:



Executable que  
obrirà el  
maquinari Pseint

## 4.1 REPRESENTACIÓ D'ALGORISMES:Pseint

- Per a la elaboració de diagrames, en aquesta unitat, accedirem sempre a l'editor de diagrames de flux que s'obrirà al fer click al següent botó:



**Important:** Abans de obrir el editor de diagrames, guarda el disseny actual, fent click en el botó "Guardar" (icona de disquette). Dona un nom i ubicació al arxiu .psc que es crearà.



## 4.1 REPRESENTACIÓ D'ALGORISMES:Pseint

- L'editor de diagrames començarà sempre d'aquesta manera.



**Important:** Si volem guardar els canvis haurem de fer-ho sempre a la finestra d'edició de pseudocodi (el disquette d'aquesta no guarda el diagrama).

## 4.1.1 DIAGRAMES DE FLUXE

### • Símbols normalitzats

- Encara que ANSI (American National Standard Institute) defineix una gran **quantitat de símbols (Consultar recurs)**
  - En la pràctica **utilitzarem 6:**

Algoritmo Inici/Fin

- **Inici:** indica l'inici de l'algorisme (ací s'escriu el **nom de l'algoritme**)

Algoritmo Inicio/Fin

FinAlgoritmo

- **Fi:** indica la fi del l'algorisme

FinAlgoritmo

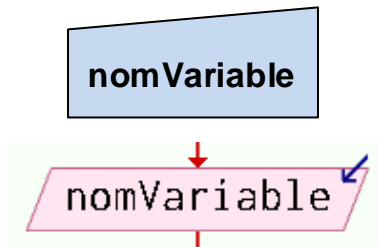
A <- a + 10

- **Procés:** indica que s'està executant una expressió (que pot incloure diferents operacions, normalment sempre una assignació)

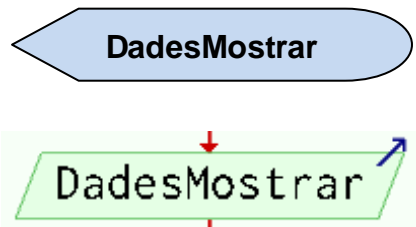
a ← a + 10

## 4.1.1 DIAGRAMES DE FLUXE

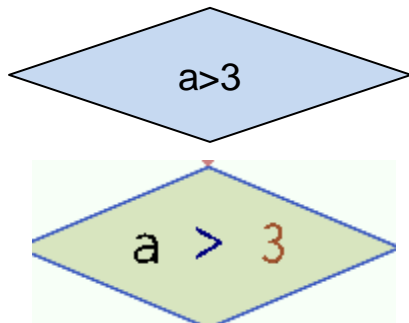
### • Símbols normalitzats



- **Entrada de dades:** Permet agafar dades introduïdes per teclat i assignar-les a una **variable** (a l'exemple la variable s'anomena `nomVariable`)



- **Eixida de dades:** Permet mostrar dades per pantalla al usuari (a l'exemple es mostra la informació de la variable `DadesMostrar`)

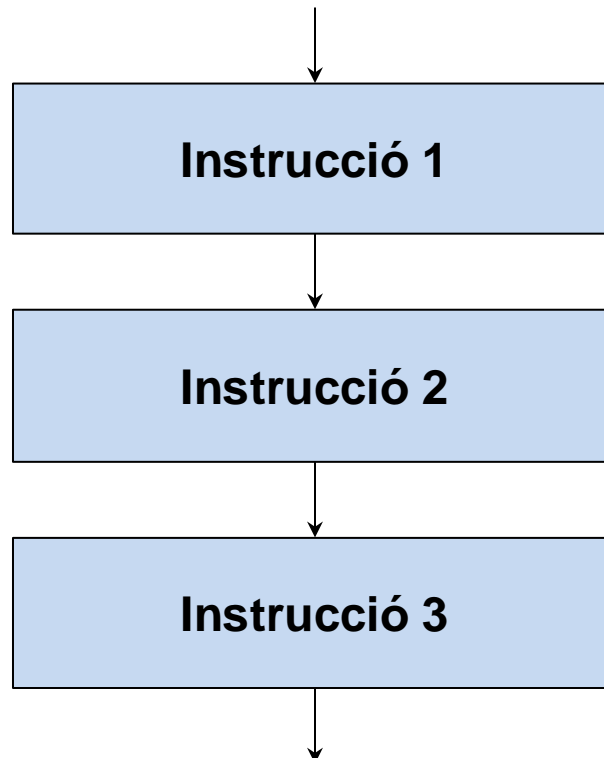


- **Decisió:** Permet executar una sentència o altra en funció de l'avaluació d'una expressió lògica.

## 4. DISSENY D'ALGORISMES

- **Estructures de control**

- **Seqüència:** Execució de **0 o N sentències** en l'ordre en el que han sigut escrites.



## 4. DISSENY D'ALGORISMES

- Ex. Algorisme seqüencial

Algoritmo calculaPrecioZapatillasRebajadas

precioZapatillas  $\leftarrow$  85

Guardem a una **variable** el valor de les **esportives**

Guardem a una **variable** la **rebaixa** a aplicar

descuento  $\leftarrow$   $85 * 0.15$

precioFinal  $\leftarrow$  precioZapatillas - descuento

Calculem i guardem el **preu final**

Mostrem per **pantalla** el **preu** al'usuari

'Precio rebajado ', precioFinal

FinAlgoritmo

## 4. DISSENY D'ALGORISMES

- Versió millorada:** Perquè és millor? Podria millorar-se més encara?

Algoritmo calculaPrecioZapatillasRebajadas

precioZapatillas  $\leftarrow$  85

descuento  $\leftarrow$  precioZapatillas  $\times$  0.15

precioFinal  $\leftarrow$  precioZapatillas - descuento

'Precio rebajado ', precioFinal

FinAlgoritmo

## 4. DISSENY D'ALGORISMES

---

### Activitats Prèvies

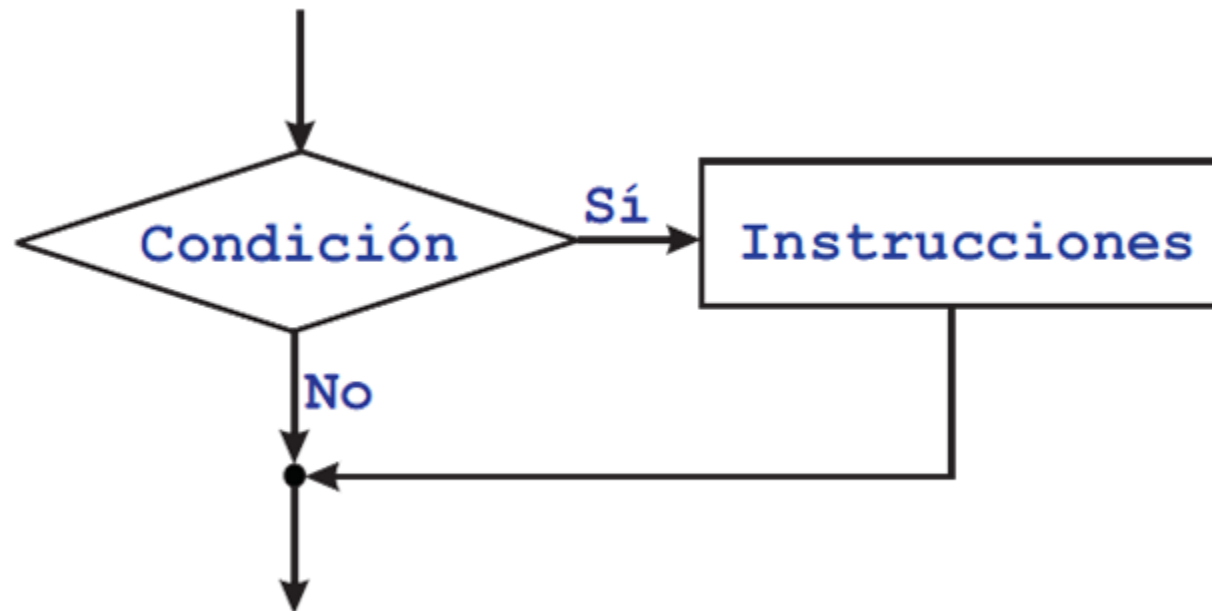
- **Activitat A.-** Dissenya un algorisme que calcule l'àrea d'un **quadrat** de costat 4 i el mostre per la pantalla.
- **Activitat B.-** Dissenya un algorisme que **lliça les dades** d'un **usuari (nom i cognoms)** i les mostre per pantalla.

És el moment de fer les primeres  
activitats **A1.1-5**

## 4. DISSENY D'ALGORISMES

- Estructures de control

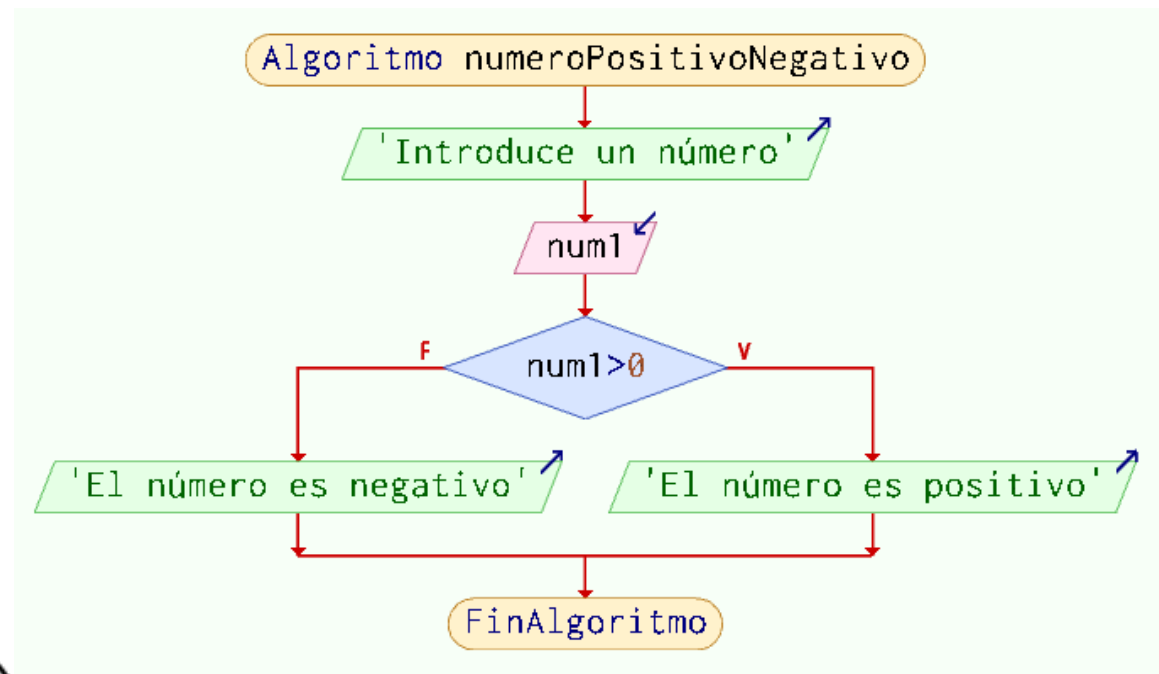
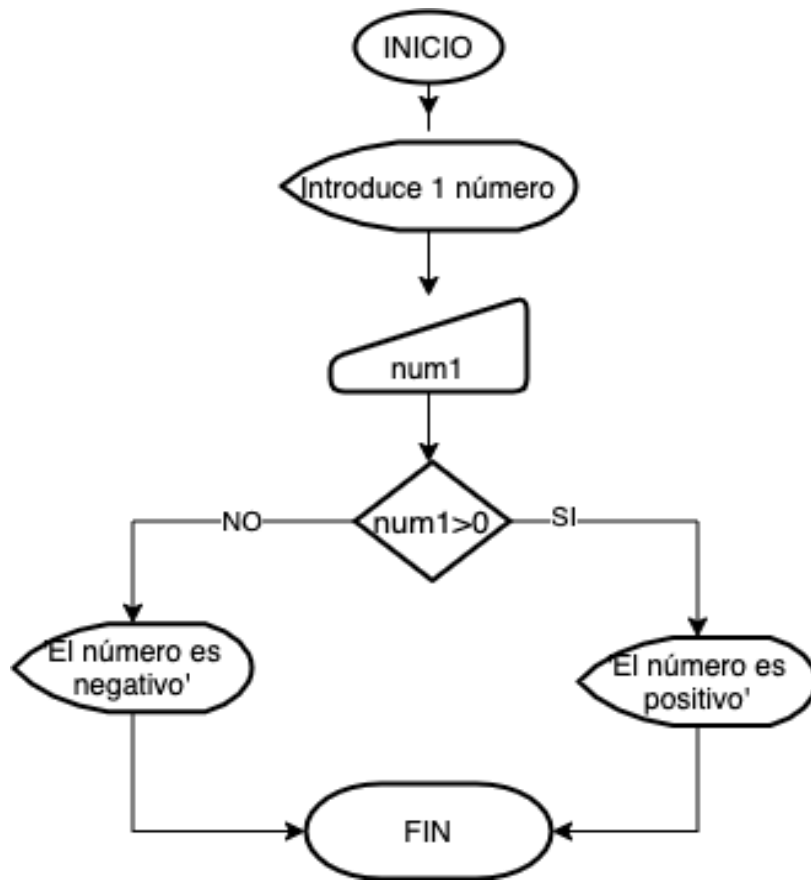
- **Selecció:** Execució d'una instrucció o grup d'elles en funció de la avaluació d'un predicat





## 4. DISSENY D'ALGORISMES

- Ex. Algorisme numeroPositivoNegativo



## 4. DISSENY D'ALGORISMES

---

### Activitats Prèvies

**Activitat C.-** Dissenya un algorisme que demane l'edat d'un usuari i ens mostre un missatge de "**Eres menor d'edat**" o eres "**major de d'edat**".

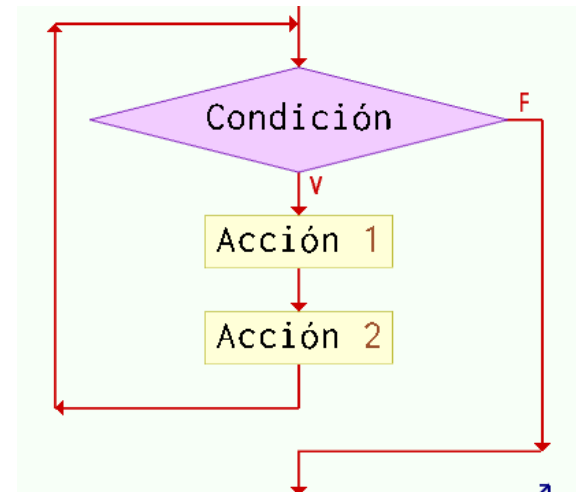
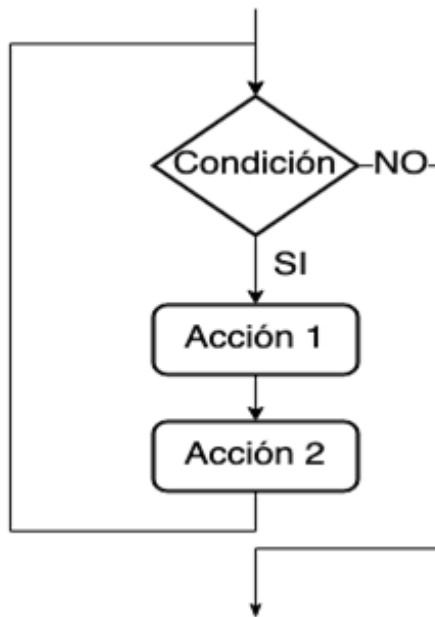
**Activitat D.-** Dissenya un algorisme que demane 2 números i diga quin és el **major**. Afegeix, a continuació, que diga si son **iguals**.

És el moment de fer les activitats  
**A1.6-9**

## 4. DISSENY D'ALGORISMES

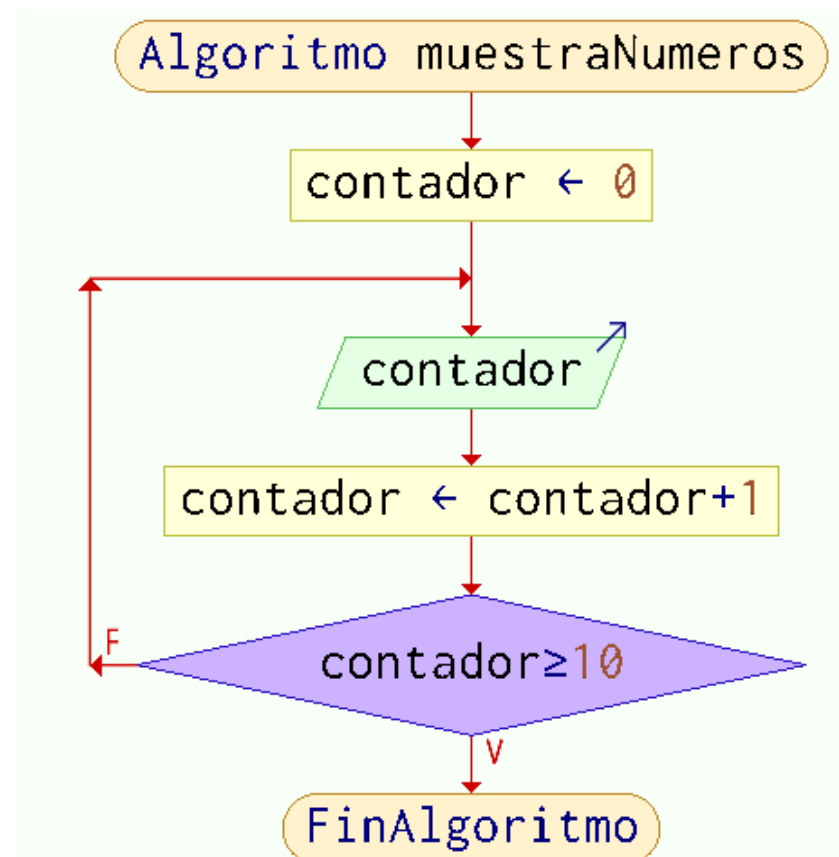
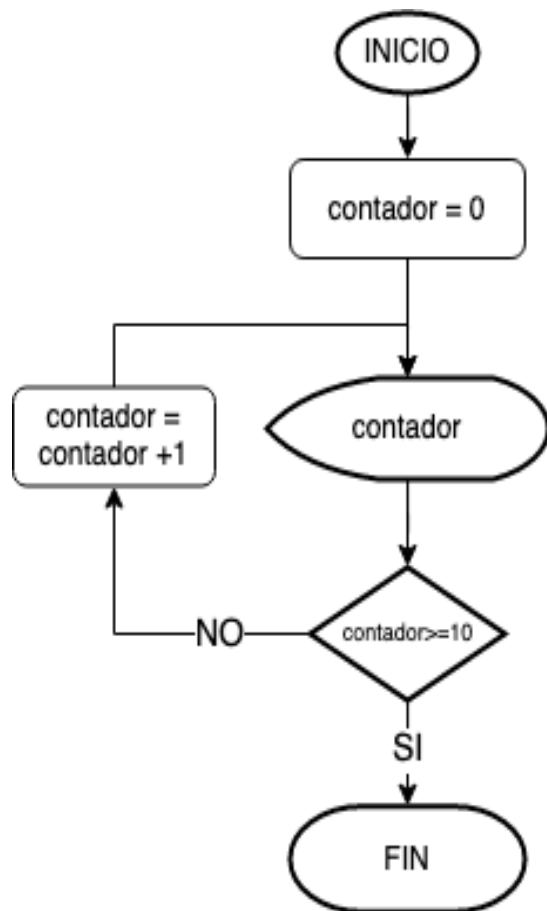
- Estructures de control

- **Repetició:** Permet l'execució d'una instrucció o un conjunt d'aquestes **de forma repetitiva** mentre es compleixi una condició.



## 4. DISSENY D'ALGORISMES

- Ex. Algorisme muestraNumeros



## 4. DISSENY D'ALGORISMES

### • Activitats Prèvies

- **Activitat E.-** Dissenya un algorisme que **mostre per pantalla els 100 primers números** naturals.

```
0 1 2 3 4 5 6 7 8 9 10 ... 99
```

- **Activitat F.-** Dissenya un algorisme que demane un número positiu a l'usuari, el guardi en una variable i mostri un **contador desde 0 al número introduït** respectant el següent format:

```
Introdueix un número: 10
```

```
0,1,2,3,4,5,6,7,8,9,10
```

És el moment de fer les activitats **A1.10-13**

## 4. DISSENY D'ALGORISMES

- *Teorema de Böhm i Jacopini*
- *"Qualsevol algorisme pot desenvolupar-se mitjançant l'ús de tres sentències de control; seqüència, selecció e iteració"*

**Programes amb una i només una entrada i eixida!**



## 4.2 EXERCICIS PREVIS

---

Això és tot... de moment :)

FinAlgoritmo