

UF01. - FONAMENTS DE PROGRAMACIÓ

- Teoria -

PROGRAMACIÓ
CFGs DAW

Autor: José Manuel Martí Fenollosa

Revisat per: Guillermo Garrido Portes

g.garridoportes@edu.gva.es

2023/2024

FONAMENTS DE PROGRAMACIÓ

ÍNDIX DE CONTINGUT

- INTRODUCCIÓ
- ALGORITME
- CICLE DE VIDA D'UN PROGRAMA
- DOCUMENTACIÓ
- ELEMENTS D'UN PROGRAMA
- ESTRUCTURES ALTERNATIVES



GENERALITAT
VALENCIANA



FONAMENTS DE PROGRAMACIÓ

INTRODUCCIÓ



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

Utilitzem els **ordinadors**
per a **resoldre problemes**



Mitjançant programes
escrits per programadors/es.

PROGRAMA

Entrada de dades

+

**Processament de la informació
(Algoritme)**

+

Eixida de dades

ALGORITME  **Accions + Ordre + Dades = Solució**

Característiques:

- **Ordenat.**
- **Finit, limitat** (Si no acabara mai, no es resoldria el problema).
- **Concís i detallat.**
- **Exacte i precís**, sense ambigüitat.
- Pot tindre **diverses dades d'entrada**.

ALGORITME

EXEMPLE



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

Exemple: *algoritme per a fregir un ou.*

Dades d'entrada: Ou, oli, paella, foc.

Dades d'eixida: ou caigut.

Procediment:

1. Posar l'oli en la paella.
2. Posar la paella al foc.
3. Quan l'oli estiga calent, trencar l'ou i introduir-lo.
4. Cobrir l'ou d'oli.
5. Quan l'ou estiga fet, retirar-lo.



CICLE DE VIDA D'UN PROGRAMA

DEFINICIÓ



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

Per a crear qualsevol programa cal realitzar 3 passos genèrics:



Fase de definició



Fase de desenvolupament



Fase de manteniment

CICLE DE VIDA D'UN PROGRAMA

FASE DE DEFINICIÓ



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA



A la fase definició s'intentarà **caracteritzar el sistema que s'ha de construir:**

- Quina **informació** usarà?
- Quines **funcions** ha de realitzar?
- **Condicions?**
- **Interfícies** del sistema (mig comú perquè els objectes no relacionats es comuniquen entre si)
- Quins **criteris de validació** s'utilitzaran?

***“Si no sabem amb claredat què és el que hem de resoldre,
no podrem trobar una solució.”***

CICLE DE VIDA D'UN PROGRAMA

FASE DE DESENVOLUPAMENT



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA



En aquesta fase **es dissenyen estructures** de dades i dels programes, **s'escriuen i documenten** aquests, i **es prova** el software.

Utilitzarem auxiliars de disseny: **diagrames de flux, pseudocodi**, etc.

Diseny descendent (**Top-Down design**): Els **problemes complexos**, es descomponen en **subproblemes més simples** i a continuació dividir aquests en **uns altres més simples (mòduls)** que siguin més fàcils de solucionar que l'original.

CICLE DE VIDA D'UN PROGRAMA

FASE DE DESENVOLUPAMENT

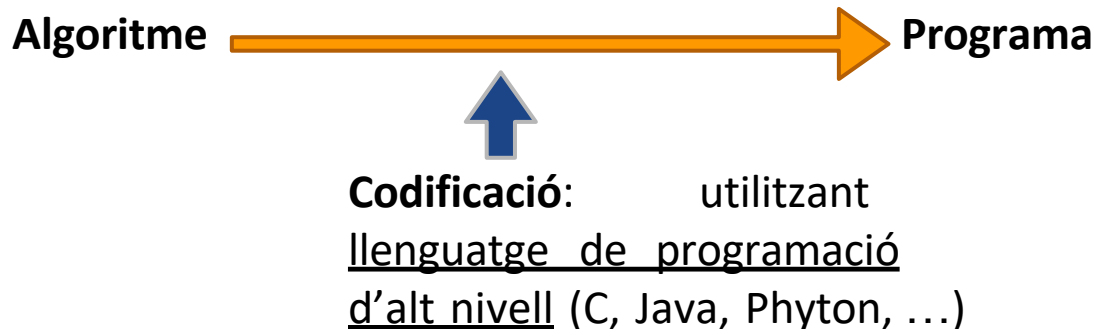


GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA



En aquesta fase **es converteix l'algoritme en programa.**



*“Si l’algoritme està **ben definit**, detallat i amb bona llegibilitat, la **codificació** serà una **simple tasca mecànica**.”*

CICLE DE VIDA D'UN PROGRAMA

FASE DE DESENVOLUPAMENT



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA



Algunes normes a tindre en compte:

- **Estructures acceptables** (programació estructurada).
- **Convencions de nominació**: maneres uniformes de designació d'arxius i variables.
- **Convencions de comentaris**.

CICLE DE VIDA D'UN PROGRAMA

FASE DE MANTENIMENT



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

Una vegada obtingut el **programa font** , és necessària la seua **traducció al codi màquina (Sí executable per l'ordinador).**

CICLE DE VIDA D'UN PROGRAMA

FASE DE MANTENIMENT



GENERALITAT
VALENCIANA



2 tipus de llenguatges segons el traductor utilitzar:

- **Llenguatges Interpretats**: sistema tradueix i executa cada instrucció.
- **Llenguatges Compilats**:
 - **1er**: es tradueix el programa font complet → **Programa**.
 - **2on**: es fusiona amb llibreries (*linkat*) → **Executable**.
 - **Ràpida execució** → doncs posteriors execucions es faran de l'executable emmagatzemat.

Posada a punt:

- Detecció d'errors.
- Depuració d'errors.
- Prova del programa.

La documentació associada al programari:

- Documentació Interna**: la que s'inclou **dins del codi font dels programes**. Ens aclareixen aspectes de les pròpies instruccions del programa.
- Documentació Externa**: tots els documents relatius al **disseny de l'aplicació**, a la **descripció** de la mateixa i els seus mòduls corresponents, als **manuals d'usuari** i els **manuals de manteniment**.

ELEMENTS D'UN PROGRAMA

DEFINICIÓ



GENERALITAT
VALENCIANA



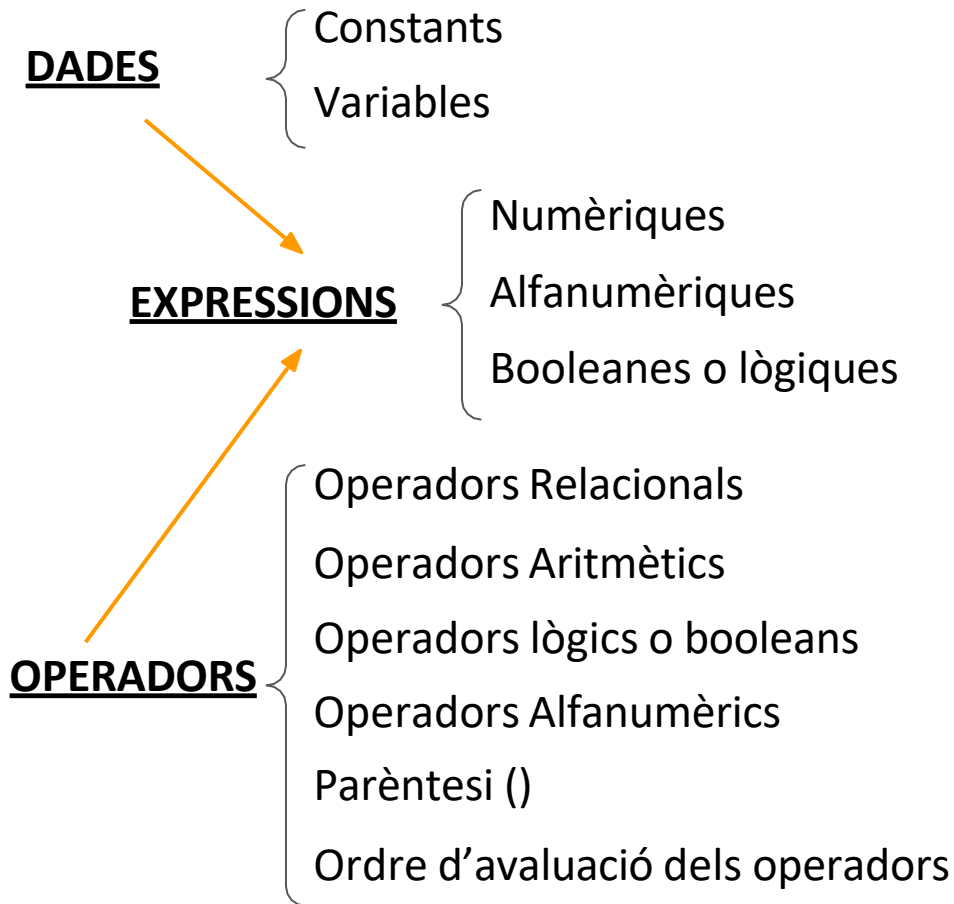
Les **dades** són la matèria primera de qualsevol programa informàtic i són utilitzades per realitzar operacions i generar resultats.

S'assignen a variables o constants que tindran els següents **atributs**:

- Nom: l'identificador.
- Tipus: conjunt de valors que pot prendre.
- Valor: element del tipus que se li assigna.

ELEMENTS D'UN PROGRAMA

DEFINICIÓ



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

DADES D'UN PROGRAMA

DADES



GENERALITAT
VALENCIANA



DADES

Constants: objectes el **valor dels quals roman invariable** al llarg de l'execució d'un programa.

Per exemple: $\pi = 3.14.1592$ $e = 2.718281$ $longitud = 25$

“Faciliten la modificabilitat del programa”

Variables: objectes el **valor dels quals pot ser modificat** al llarg de l'execució d'un programa

Per exemple: *una variable per a calcular l'àrea d'una circumferència determinada, una variable per a calcular una factura, etc.*

OBJECTES D'UN PROGRAMA

EXPRESSIONS



GENERALITAT
VALENCIANA



EXPRESSIONS

Numèriques: produeixen resultats numèrics.

Per exemple: $pi*sqr(x)$ $(2*x)/3$

Alfaumèriques: produeixen resultats alfanumèrics.

Per exemple: `"Don " + "José"`

Booleans o lògiques: produeixen resultats Vertader o Fals.

Per exemple: $a < 0$ $(a > 1) \text{ and } (b < 5)$

OBJECTES D'UN PROGRAMA

OPERADORS



GENERALITAT
VALENCIANA



OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

OPERADOR	DEFINICIÓ
+	Suma
-	Resta
*	Multiplicació
^	Potència
/	Divisió
%	Reste de la divisió

Per exemple:

Expressió	Resultat
$3 + 5 - 2$	6
$24 \% 3$	0
$8 * 3 - 7 / 2$	20.5

OBJECTES D'UN PROGRAMA

OPERADORS



GENERALITAT
VALENCIANA



OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

OPERADOR	DEFINICIÓ
<	Menor que
>	Major que
==	Igual que
>=	Major o igual que
<=	Menor o igual que
<>	Distints que

Per exemple:

Expressió	Resultat
$A' < b'$	Vertader, ja que en còdic ASCII la A està abans que la B
$1 < 6$	Vertader
$10 < 2$	Fals

OBJECTES D'UN PROGRAMA

OPERADORS

OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

OPERADOR	DEFINICIÓ
NO (not)	Negació
I (and)	Conjunció
O (or)	Disjunció

El comportament d'un operador lògic es defineix mitjançant la seua corresponent **taula de veritat**.

OBJECTES D'UN PROGRAMA

OPERADORS



GENERALITAT
VALENCIANA



OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

OPERADOR	DEFINICIÓ
NO (not)	Negació
I (and)	Conjunció
O (or)	Disjunció

NO lògic (**NOT**) o negació:

A	NOT A
V	F
F	V

- Taula de veritat del NOT -

L'operador NOT inverteix el valor: Si és vertader (V) retorna fals (F), i viceversa.

OBJECTES D'UN PROGRAMA

OPERADORS



GENERALITAT
VALENCIANA



OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

OPERADOR	DEFINICIÓ
NO (not)	Negació
I (and)	Conjunció
O (or)	Disjunció

O lògic (OR) o disjunció:

A	B	A OR B
V	V	V
V	F	V
F	V	V
F	F	F

- Taula de veritat del OR -

L'operador **OR** retorna **vertader (V)** si algun dels dos **valors és vertader**. En cas contrari, retorna Fals (F).

OBJECTES D'UN PROGRAMA

OPERADORS



GENERALITAT
VALENCIANA



OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

OPERADOR	DEFINICIÓ
NO (not)	Negació
I (and)	Conjunció
O (or)	Disjunció

I lògic (AND) o conjunció:

A	B	A AND B
V	V	V
V	F	F
F	V	F
F	F	F

- Taula de veritat del AND -

L'operador **AND** retorna **Vertader (V)** només si **tots dos valors són vertaders**. En qualsevol altre cas retorna **Fals (F)**.

OBJECTES D'UN PROGRAMA

OPERADORS



GENERALITAT
VALENCIANA



OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

Per exemple:

Expressió	Resultat
$9 == (3 * 3)$	Vertader
$3 <> 2$	Vertader
$9 == (3 * 3) \text{ Y } 3 <> 2$	Vertader
$3 > 2 \text{ Y } b < a$	Vertader Y Fals \rightarrow Fals
$3 > 2 \text{ O } b < a$	Vertader O Fals \rightarrow Vertader
$\text{NO } (a < b)$	NO Vertader \rightarrow Fals
$5 > 1 \text{ Y } \text{NO}(b < a)$	Vertader Y NO Fals \rightarrow Vertader

OBJECTES D'UN PROGRAMA

OPERADORS



GENERALITAT
VALENCIANA



OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

Uneix dades de tipus alfanumèric. També dit **concatenació**.

Per exemple:

Expressió	Resultat
"Ana " + "Ferrandis"	Ana Ferrandis
"boca" + "badat"	bocabadat

OBJECTES D'UN PROGRAMA

OPERADORS

OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

Nien expressions.

Per exemple:

*Operació $(3*2) + (6/2) \rightarrow$ Resultat 9*

OBJECTES D'UN PROGRAMA

OPERADORS

OPERADORS

Aritmètics

Relacionals

Lògics o booleans

Alfanumèrics

Parèntesi ()

Ordre d'avaluació

1. Parèntesi ()
2. Potència ^
3. Multiplicació, divisió i resto * / %
4. Sumes i restes + -
5. Concatenació +
6. Relacionals < <= > >= == <>
7. Negació NOT
8. Conjunció AND
9. Disjunció OR



Paréntesis ()

Exponents x^3

Multiply \times

Divide \div

Addition +

Subtraction -

Operadors d'igual ordre ➡ D'esquerra a dreta.

ESTRUCTURES ALTERNATIVES

DEFINICIÓ



GENERALITAT
VALENCIANA



Les **formes** més importants **de representació d'algoritmes** son:

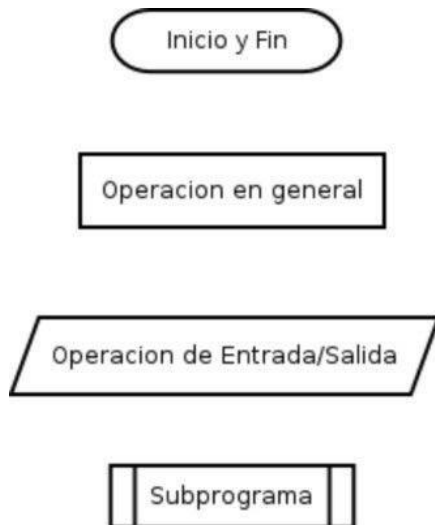
- Diagrames de flux (**ordinogrames**).
- **Pseudocodi**.

ESTRUCTURES ALTERNATIVES

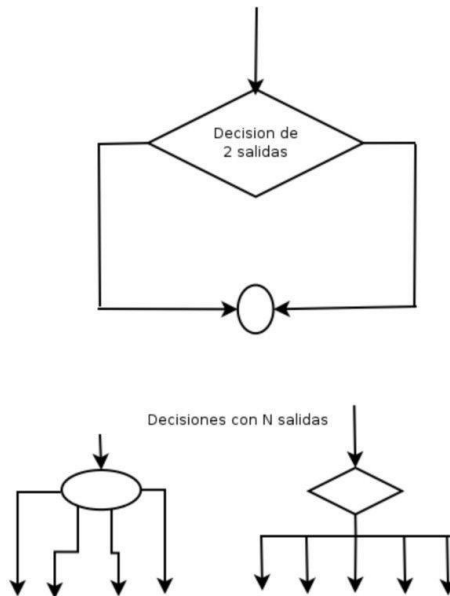
DIAGRAMES DE FLUX (Ordinogrames)

Representació gràfica dels fluxos que seguiran les dades i seqüència lògica dels operadors per a resoldre el problema.

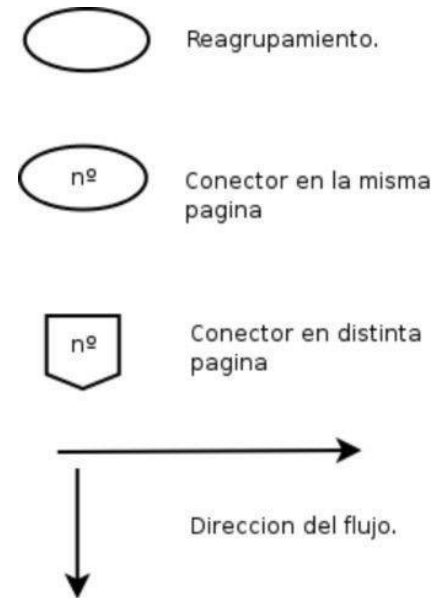
Símbols d'operació



Símbols de decisió



Símbols de connexió



ESTRUCTURES ALTERNATIVES

DIAGRAMES DE FLUX (Ordinogrames)



GENERALITAT
VALENCIANA



A tindre en compte:

1. **Senzillesa** en la seua construcció.
2. **Claredat** en la seua compressió.
3. **Normalització** en el seu disseny.
4. **Flexibilitat** en les seues modificacions.

Es convenient:

- El **començament** del programa figurarà **en la part superior** de l'ordinograma.
- El **símbol de començament** haurà d'aparéixer **una sola vegada** en l'ordinograma.
- El **flux de les operacions** serà, sempre que siga possible de **dalt a baix i d'esquerra a dreta (Top-Down)**
- S'evitaran sempre els **encreuaments de línies** utilitzant connectors.

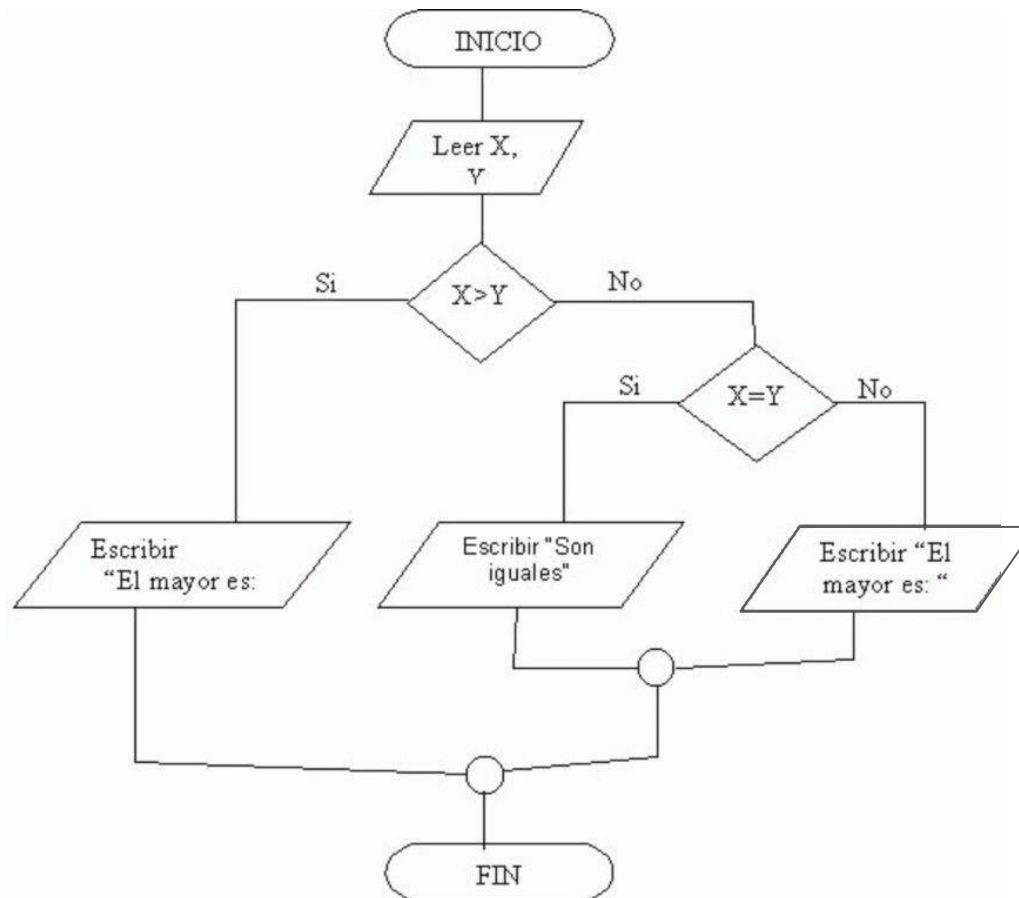
ESTRUCTURES ALTERNATIVES

DIAGRAMES DE FLUX (Ordinogrames)



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA



Llenguatge intermedi entre el llenguatge natural i el llenguatge de programació.

Quan s'escriu un algoritme mitjançant la utilització de pseudocodi, s'ha de **"sagnar" el text respecte al marge esquerre**, amb la finalitat que es comprenga més fàcilment el disseny que s'està realitzant.

****** equival a una línia de comentari

ESTRUCTURES ALTERNATIVES

PSEUDOCODI



GENERALITAT
VALENCIANA

ceedcv
CENTRE ESPECÍFIC
D'EDUCACIÓ A DISTÀNCIA DE
LA COMUNITAT VALENCIANA

CAPÇALERA

Programa: Nom del programa

Mòdul: Nom del mòdul

COS

INICI

DADES:

PARÀMETRES

Definició de paràmetres

CONSTANTS

Definició de constants

VARIABLES

Definició de variables

ALGORITME:

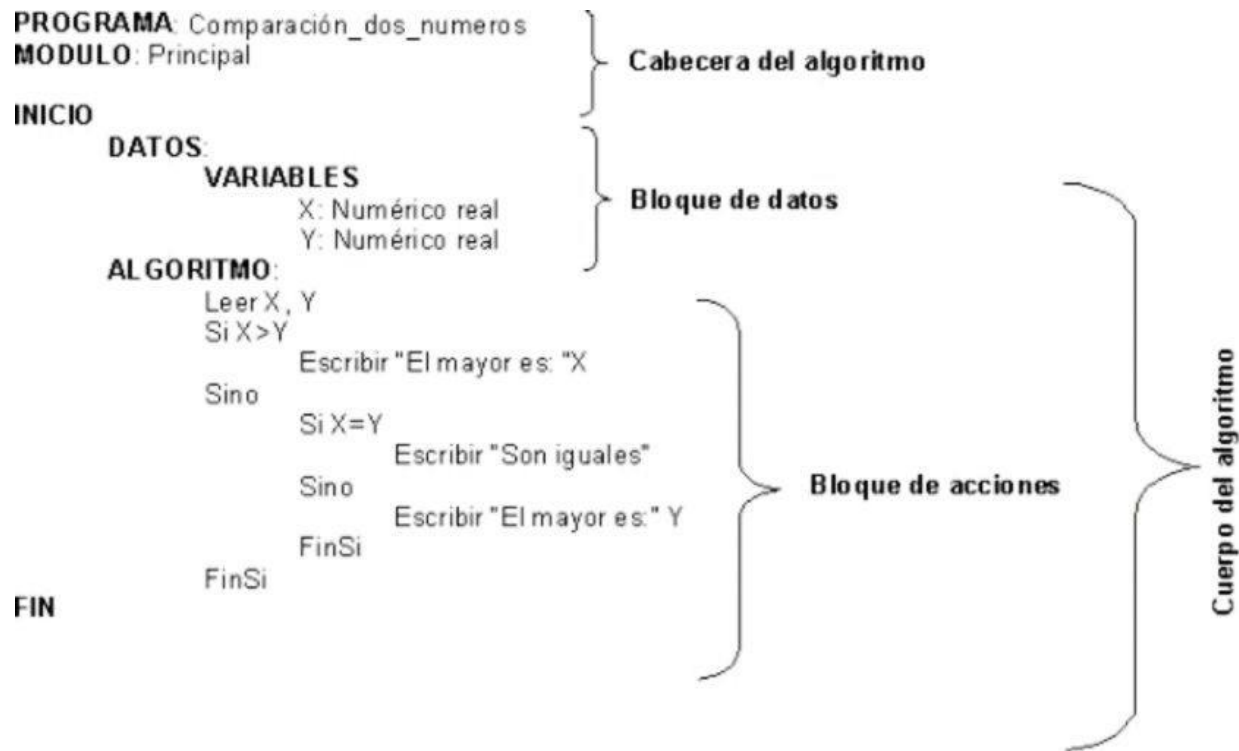
Descripció detallada d'ordres

FI

ESTRUCTURES ALTERNATIVES

PSEUDOCODI

Per exemple: Algoritme que llig dos números “X” e “Y”, determina si són iguals, i en cas de no ser-ho, indica quin d'ells és el major.



EXERCICIS PROPOSATS

