



Tema 2

Algorismes i programes Exercicis

Informàtica
Grau en Física
Universitat de València

Francisco.Grimaldo@uv.es
Ariadna.Fuertes@uv.es





Programa: Nota final (1/4)

- Realitzeu un programa que sol·licite les notes obtingudes per un estudiant en cadascuna de les parts d'una assignatura i en calcule **la nota final** segons els percentatges següents:
 - a) Nota d'examen final: 40%.
 - b) Nota d'exàmens parcials: 40%.
 - c) Nota de pràctiques: 20%.
- La nota final de l'estudiant es calcula realitzant la **mitjana ponderada** si ha obtingut una nota mínima de 4 en cada una de les parts, o prenent el **mínim** entre 4 i la mitjana ponderada en cas contrari.



Programa: Nota final (2/4)

- 1r. Anàlisi del problema:
 - ◆ Dades d'entrada: 3 notes (necessite 3 variables reals per a emmagatzemar-les).
 - ◆ Dades d'eixida: nota final (una altra variable real).
- 2n. Disseny / cerca de l'algorisme:
 - ◆ Processament de la informació: veure si la nota és la mitjana ponderada o si és el mínim entre la mitjana i 4.
- 3r. Programació de l'algorisme:
 - ◆ Pseudocodi o diagrama de flux.
- 4t. Traducció / execució / comprovació (al lab.).



Programa: Nota final (3/4)

Pseudocodi:

Variables:

a, b, c, m, n : reals

Inici

Llegir(a)

Llegir(b)

Llegir(c)

$m \leftarrow 0.4*a + 0.4*b + 0.2*c$

Si₁ a>4 AND b>4 AND c>4 aleshores

n ← m

si no₁

Si₂ m<4 aleshores

n ← m

si no₂

n ← 4

Final_si₂

Final_si₁

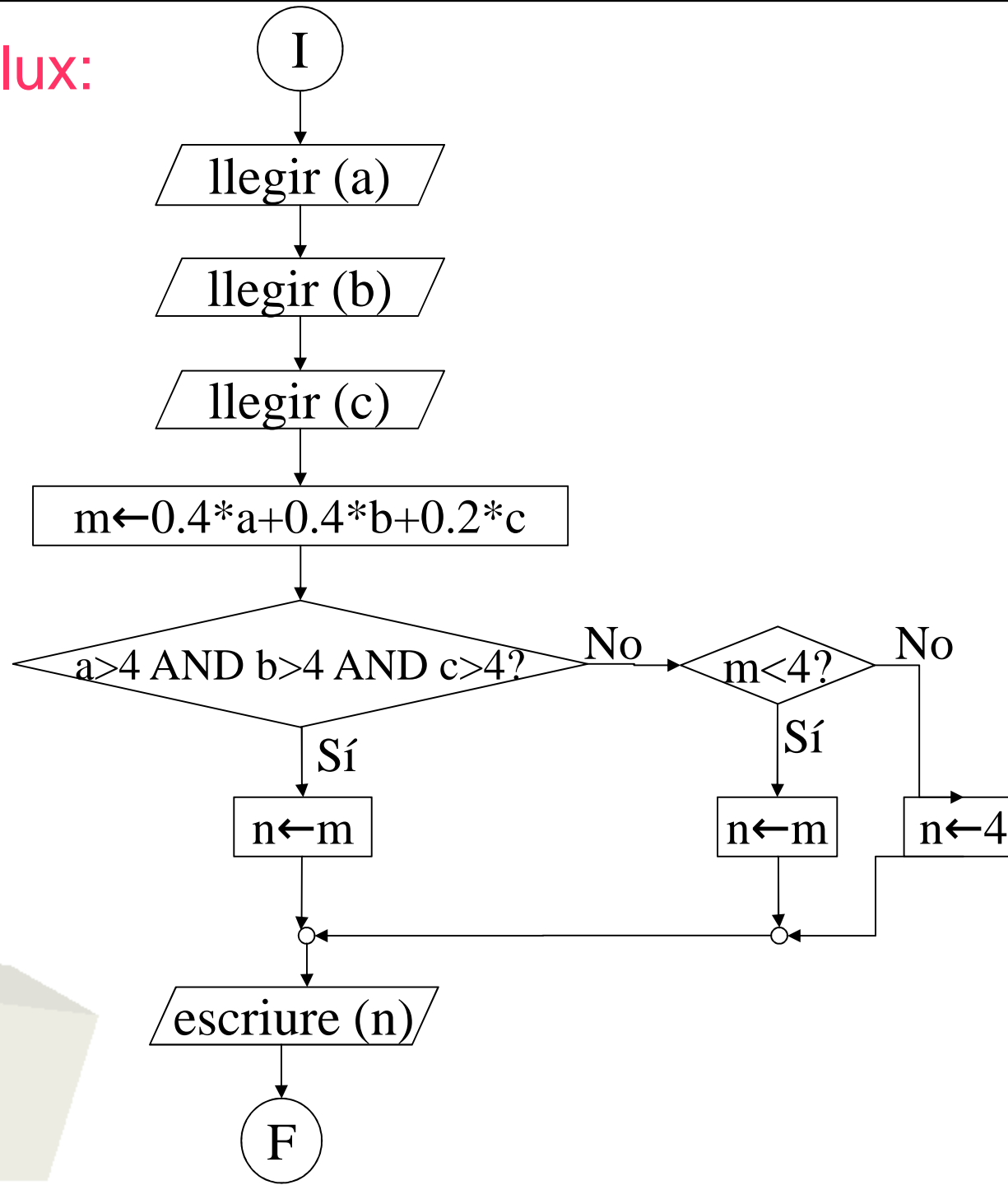
Escriure(n)

FINAL



Programa: Nota final (4/4)

Diagrama de flux:





Programa: Qualificació (1/3)

- Escriuiu un programa que demane a l'usuari la nota de teoria i de pràctiques d'un alumne i que en mostre per pantalla la **nota final i la qualificació**, tenint en compte que:
 - ◆ La nota final correspon al 60% de la nota de teoria i al 40% de la nota de pràctiques.
 - ◆ La qualificació podrà ser:
 - “Excel·lent”, que és 9 o més,
 - “Notable”, que va des del 7 fins al 9,
 - “Aprovat”, que és major que 5 i menor que 7,
 - “Suspens”, quan siga menor que 5.



Programa: Qualificació (2/3)

Pseudocodi:

Variables:

a, b, n : reals

Inici

Llegir(a)

Llegir(b)

$n \leftarrow 0.6 * a + 0.4 * b$

Escriure(n)

Si₁ $n \geq 9$ aleshores

Escriure("Excel·lent")

si no₁

Si₂ $n \geq 7$ aleshores

Escriure("Notable")

si no₂

Si₃ $n \geq 5$ aleshores

Escriure("Aprovat")

si no₃

Escriure("Suspens")

Final_si₃

Final_si₂

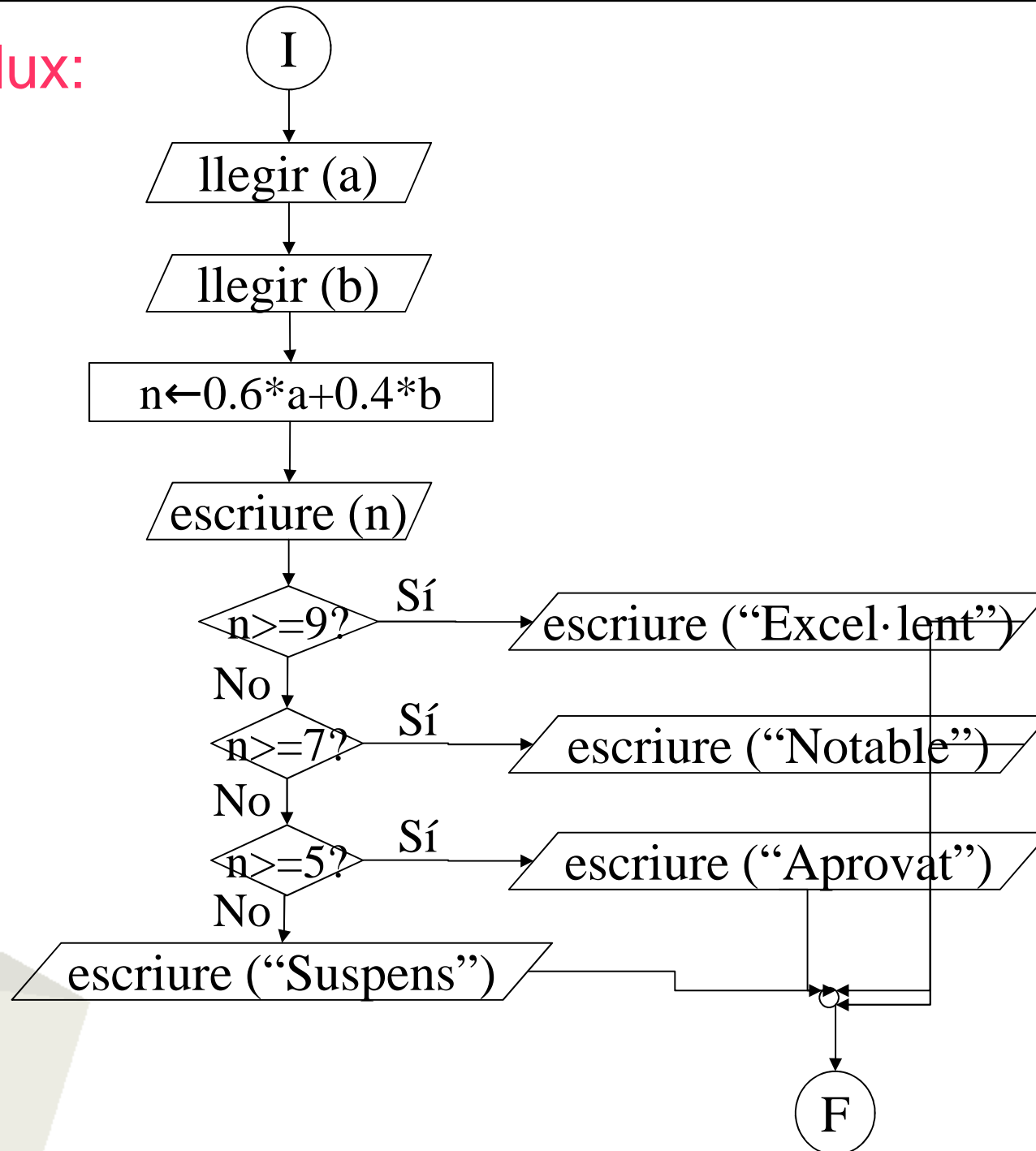
Final_si₁

FINAL



Programa: Qualificació (3/3)

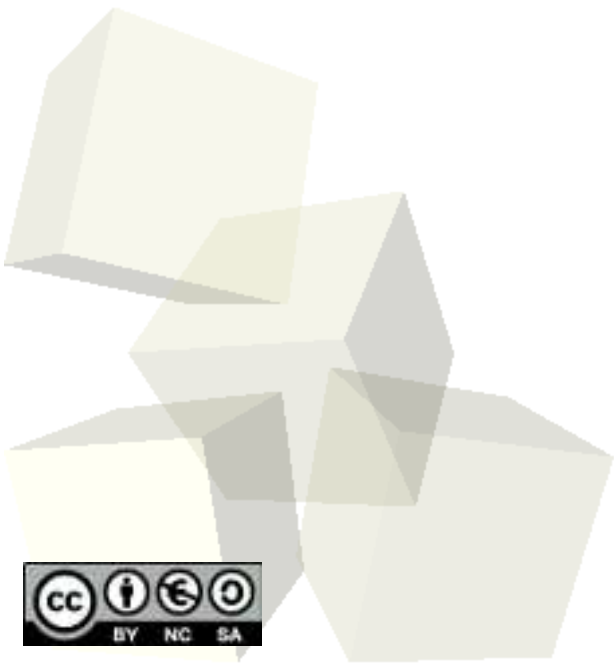
Diagrama de flux:





Programa: Ordenació (1/3)

- Escriuiu un programa que demane tres nombres pel teclat i diga:
 - ◆ si estan “Ordenats de manera ascendent”,
 - ◆ si estan “Ordenats de manera descendent”,
 - ◆ o si estan “Desordenats”.





Programa: Ordenació (2/3)

Pseudocodi:

Variables:

a, b, c : reals

Nota: Pel fet de ser reals no podem comparar amb l'operador =, és a dir, no són "exactament iguals".

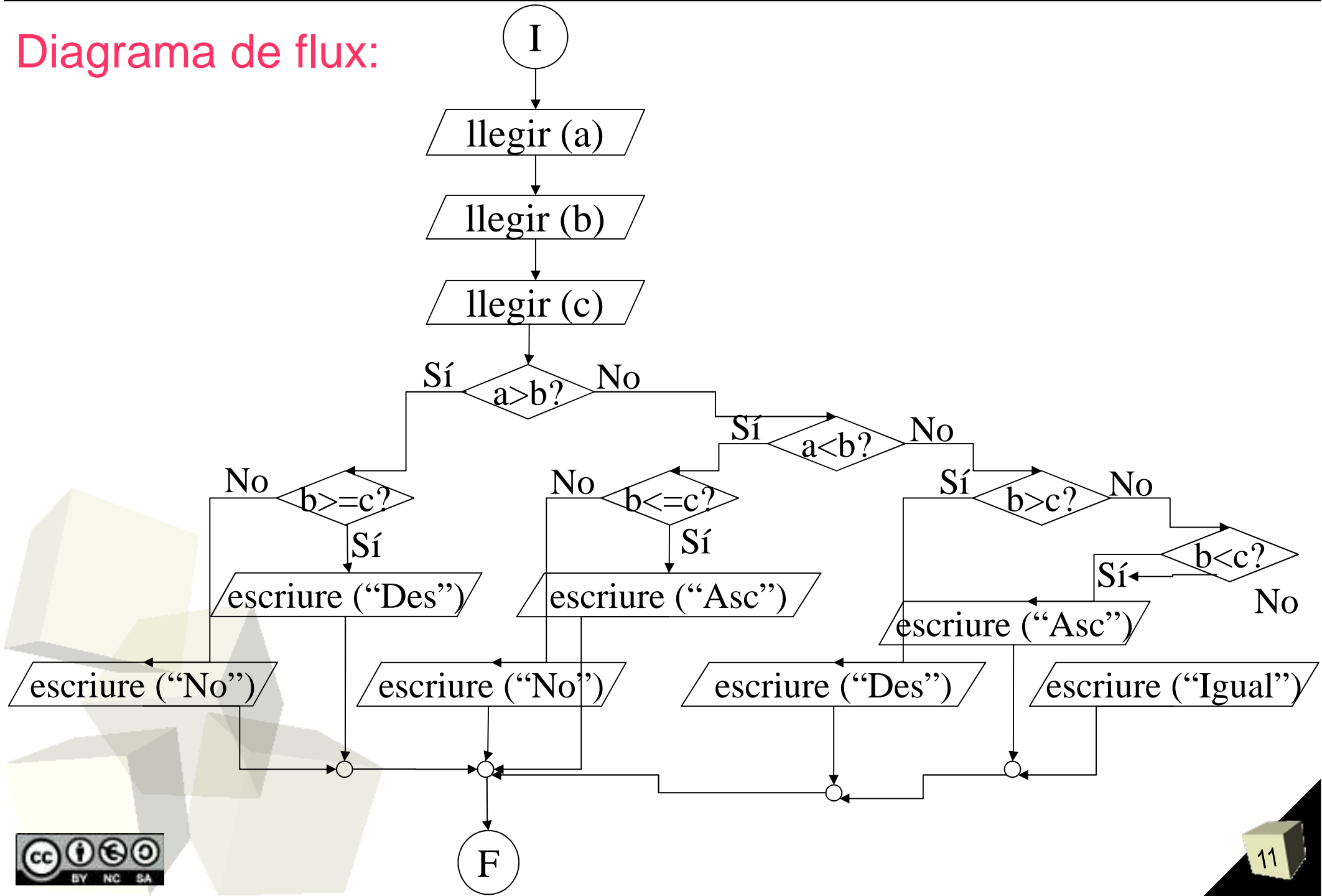
Inici

```
Llegir(a)
Llegir(b)
Llegir(c)
Si1 a>b aleshores
    Si2 b>=c aleshores
        Escriure("Descendent")
    si no2
        Escriure("No ordenats")
    Final_si2
si no1
    Si3 a<b aleshores
        Si4 b<=c aleshores
            Escriure("Ascendent")
        si no4
            Escriure("No ordenats")
        Final_si4
    si no3
        Si5 b>c aleshores
            Escriure("Descendent")
        si no5
            Si6 b< c aleshores
                Escriure("Ascendent")
            si no6
                Escriure("Són tots iguals")
            Final_si6
        Final_si5
    Final_si3
Final_si1
FINAL
```



Programa: Ordenació (3/3)

Diagrama de flux:





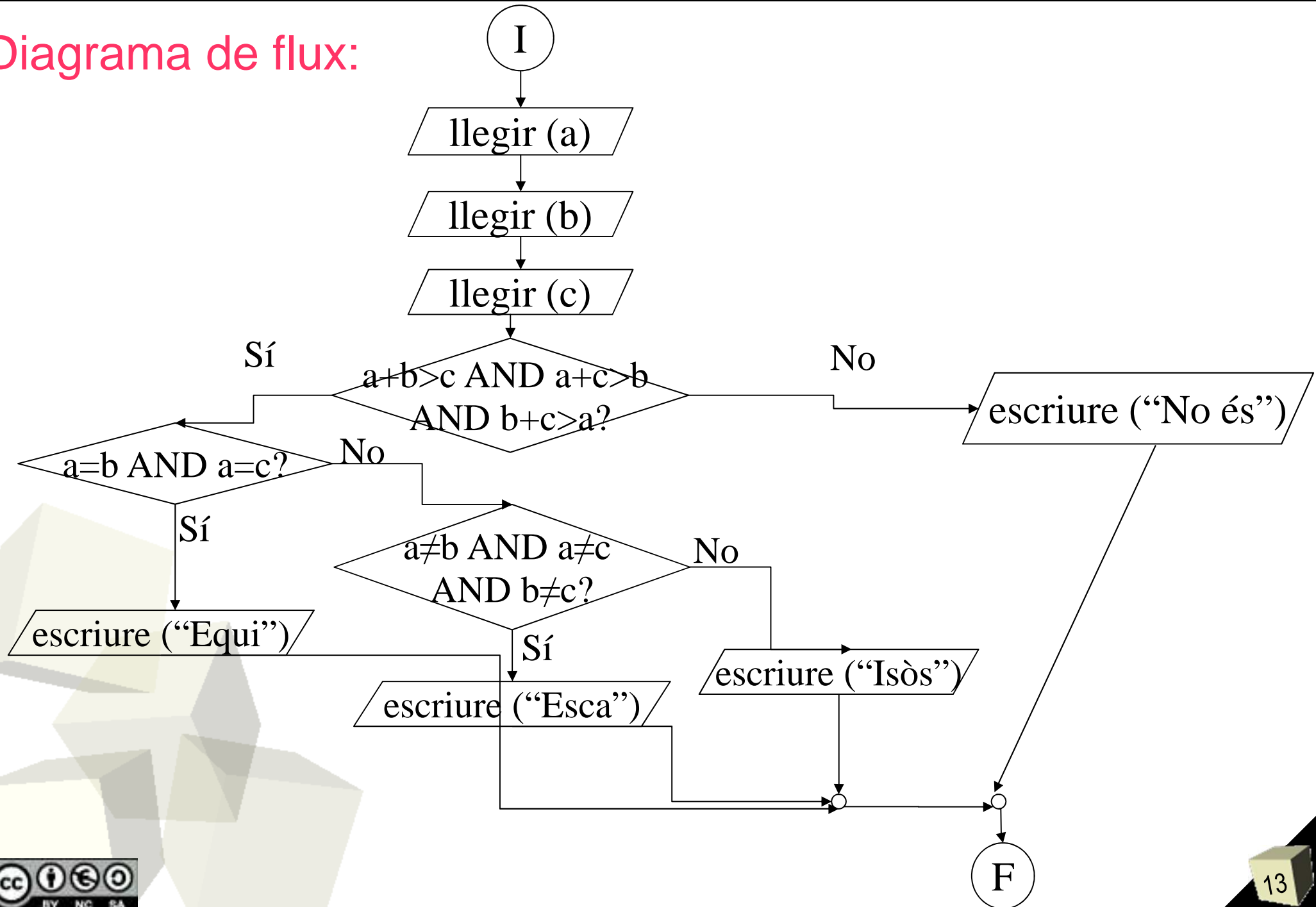
Programa: Triangle (1/3)

- Un triangle compleix sempre la regla següent: “La suma de les llargàries de dos costats qualssevol és estrictament major que la llargària del tercer costat”.
- Quan els tres costats són diferents, diem que el triangle és escalè, quan els tres costats són iguals l'anomenem equilàter i quan almenys dos costats són iguals, el triangle és isòsceles.
- Creeu un programa que demane a l'usuari la llargària de tres costats i comprove si a partir d'ells es pot **formar un triangle i de quin tipus seria.**



Programa: Triangle (2/3)

Diagrama de flux:





Programa: Triangle (3/3)

Pseudocodi:

Variables:

a, b, c : enters

Inici

Llegir(a)

Llegir(b)

Llegir(c)

Si₁ $a+b>c$ AND $a+c>b$ AND $b+c>a$ aleshores

Si₂ $a=b$ AND $a=c$ aleshores

Escriure("Equilàter")

si no₂

Si₃ $a\neq b$ AND $a\neq c$ AND $b\neq c$ aleshores

Escriure("Escalè")

si no₃

Escriure("Isòsceles")

Final_si₃

Final_si₂

si no₁

Escriure("No és un triangle")

Final_si₁

FINAL



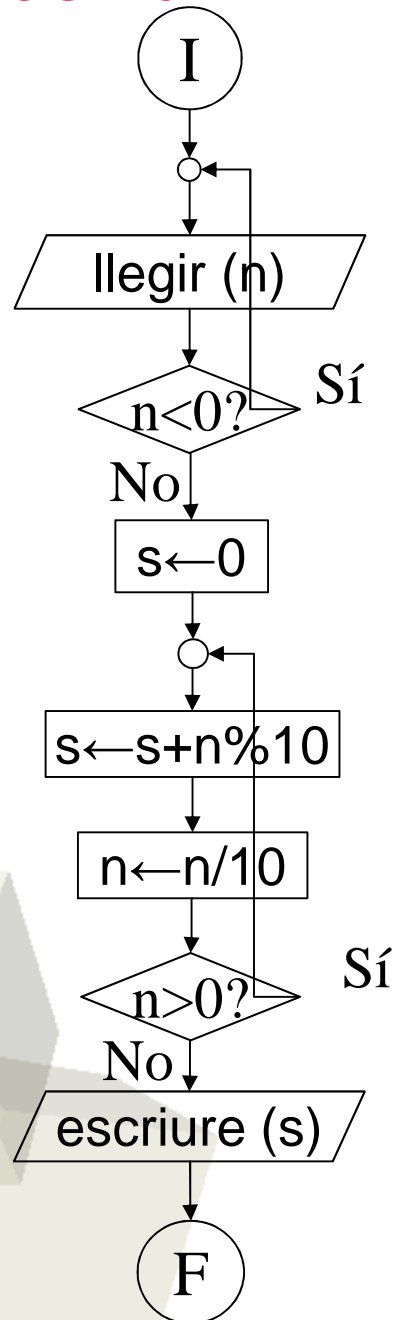
Programa: Suma xifres (1/2)

- Realitzeu un programa que calcule la **suma de les xifres d'un nombre enter positiu**.
- **1r. Anàlisi del problema:**
 - ◆ Dades d'entrada: 1 nombre enter positiu (necessite 1 variable entera per a emmagatzemar-lo).
 - ◆ Dades d'eixida: suma de les xifres (una altra variable entera).
- **2n. Disseny / cerca de l'algorisme:**
 - ◆ No permetre que s'introduïska un nombre negatiu. Quan siga positiu, extraure cadascuna de les xifres mitjançant divisions successives per 10 i agafar la resta (que és la xifra que busquem) fins que no es puga dividir més.
- **3r. Programació de l'algorisme.**



Programa: Suma xifres (2/2)

Diagrama de flux:



Pseudocodi:

Variables:

n, r, s : enters

Inici

Fer

Llegir(n)

Mentre₁ (n < 0)

s ← 0

Fer

s ← s + n % 10

n ← n / 10

Mentre₂ (n > 0)

Escriure(s)

FINAL



Programa: Seqüència numèrica (1/5)

- Escriu un programa que sol·licite a l'usuari un **nombre enter positiu n** (si hi introdueix un valor negatiu, tornarà a demanar un altre nombre fins que aquest siga positiu) i que mostre una **seqüència d'eixida** similar a (per a $n=6$):

123456

12345

1234

123

12

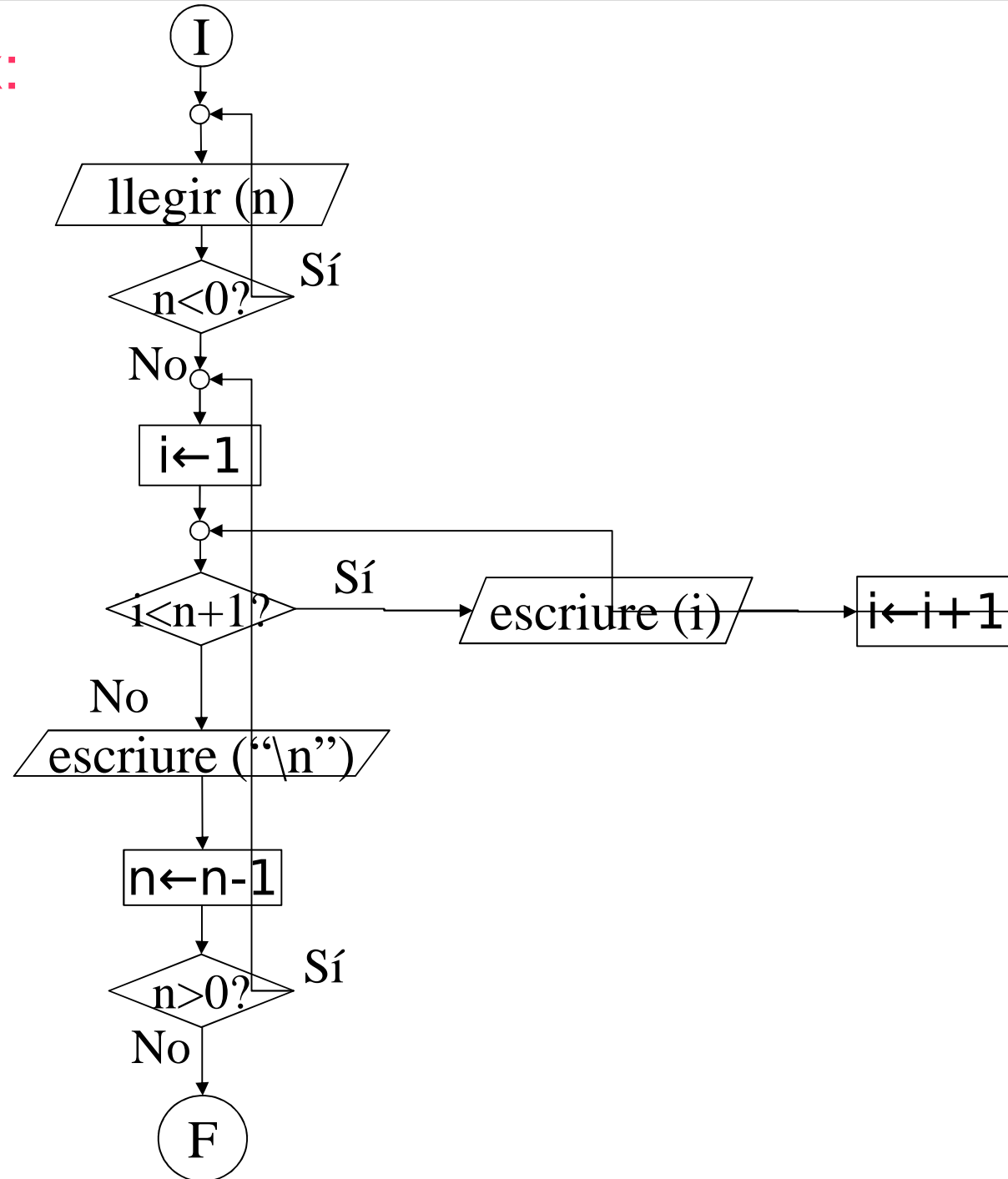
1



Programa: Seqüència numèrica (2/5)

Diagrama de flux:

fer...mentre





Programa: Seqüència numèrica (3/5)

Pseudocodi:

Fer...Mentre

Variables:

n, i : enters

Inici

Fer

Llegir(n)

Mentre₁ ($n < 0$)

Fer

Des de $i \leftarrow 1$ fins a n fer

 Escriure(i)

$i \leftarrow i + 1$

final_des_de

 Escriure("\n")

$n \leftarrow n - 1$

Mentre₂ ($n > 0$)

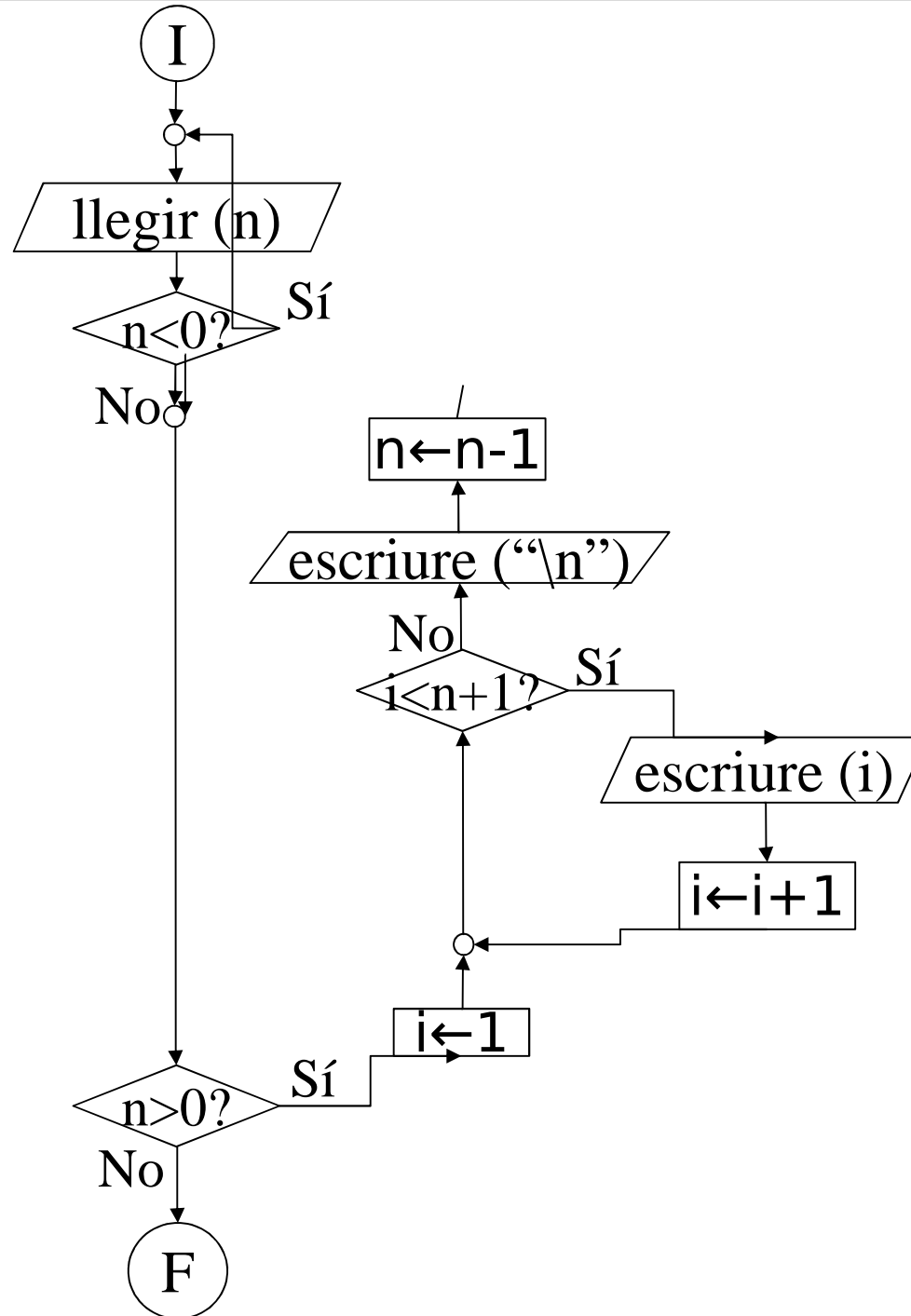
FINAL



Programa: Seqüència numèrica (4/5)

Diagrama de flux:

Mentre...Fer





Programa: Seqüència numèrica (5/5)

Pseudocodi:

Mentre...Fer

Variables:

n, i : enters

Inici

Fer

Llegir(n)

Mentre₁ ($n < 0$)

Mentre₂ ($n > 0$) fer

Des de $i \leftarrow 1$ fins a n fer

Escriure(i)

$i \leftarrow i + 1$

final_des_de

Escriure("\n")

$n \leftarrow n - 1$

Final_mentre₂

FINAL



Programa: Compta nombres (1/5)

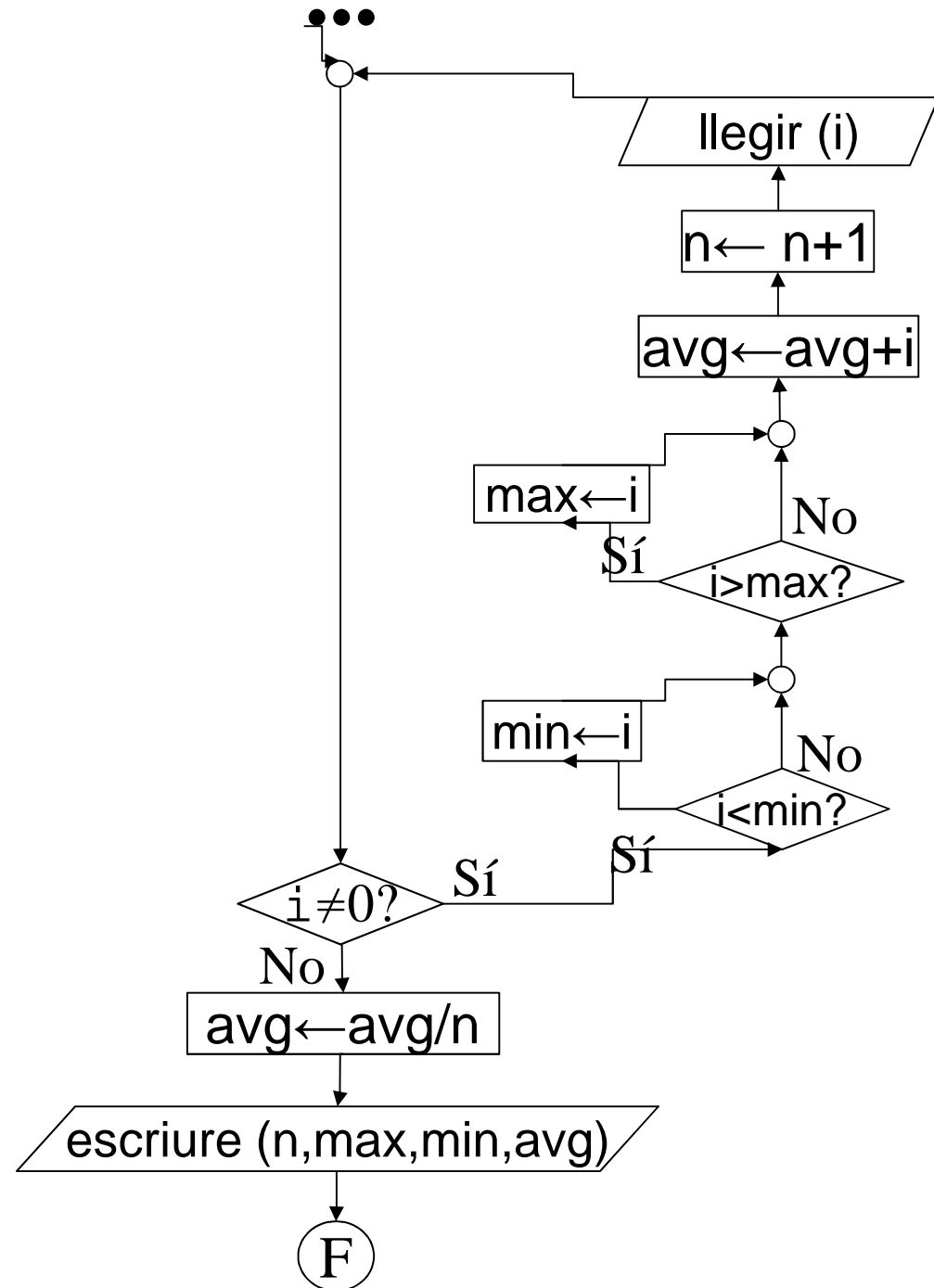
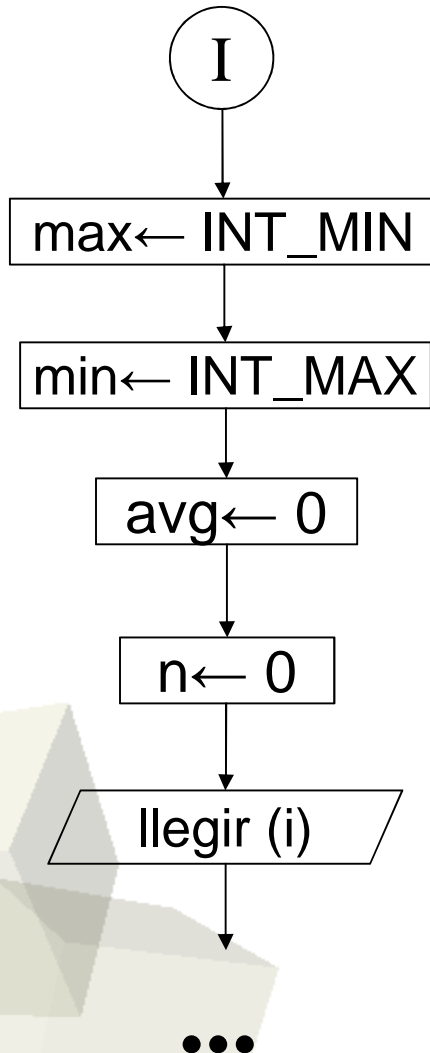
- Escriu un programa que llija nombres enters positius fins que l'usuari introdueixca un 0 (“valor sentinella”).
- Llavors, el programa mostrarà per pantalla la quantitat de nombres llegits, el major, el menor i la mitjana dels nombres llegits.



Programa: Compta nombres (2/5)

Diagrama de flux:

1a aproximació: Inicialitzar el mínim a un valor molt petit i el màxim, a un valor molt gran.





Programa: Compta nombres (3/5)

Pseudocodi:

Variables:

n, max, min, i : enters
avg : real

Inici

max ← INT_MIN

min ← INT_MAX

avg ← 0

n ← 0

Llegir(i)

Mentre (i ≠ 0) fer

Si₁ i < min aleshores

min ← i

final_si₁

Si₂ i > max aleshores

max ← i

final_si₂

avg ← avg + i

n ← n + 1

Llegir(i)

Final_mentre

avg ← avg / n

Escriure(n, max, min, avg)

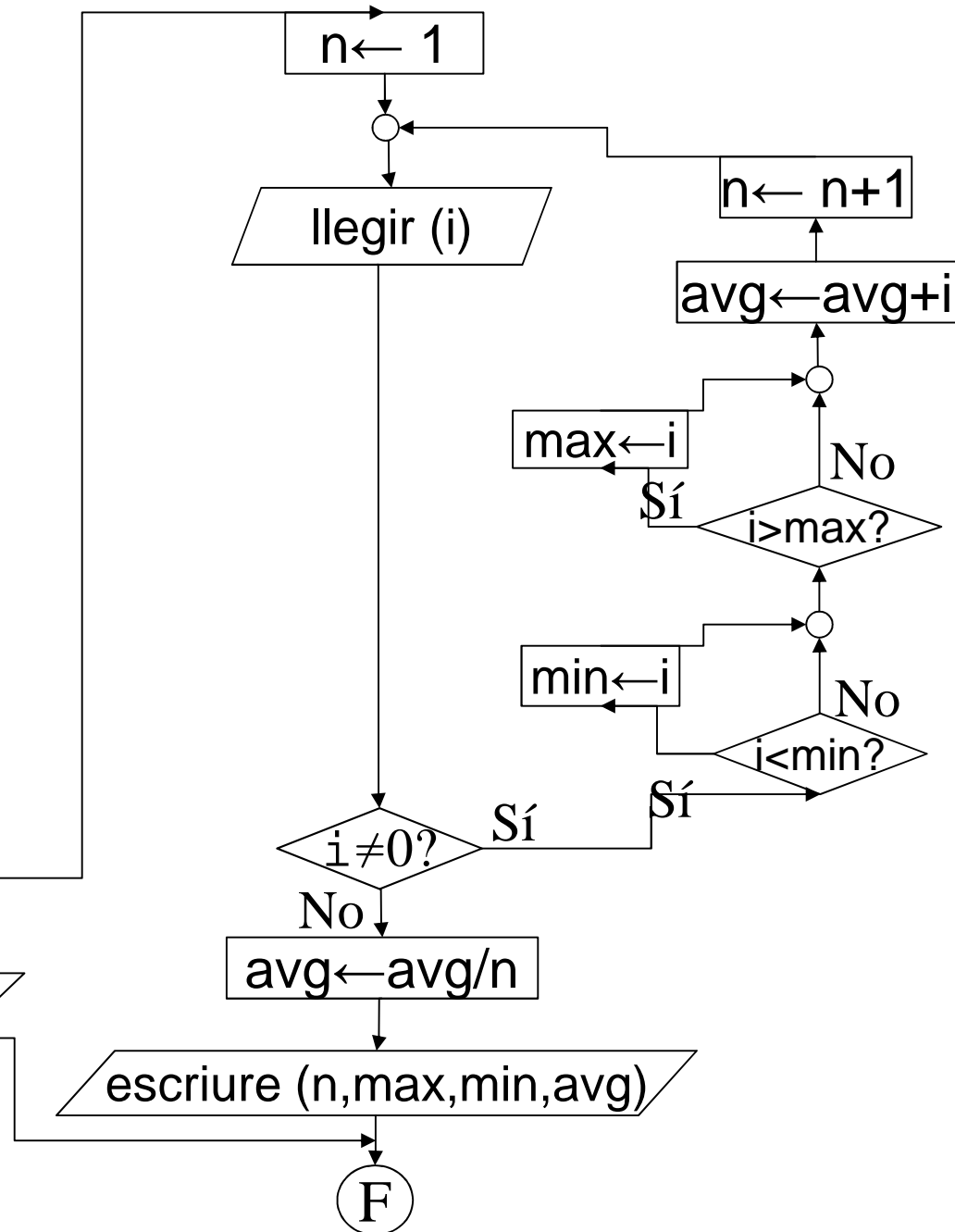
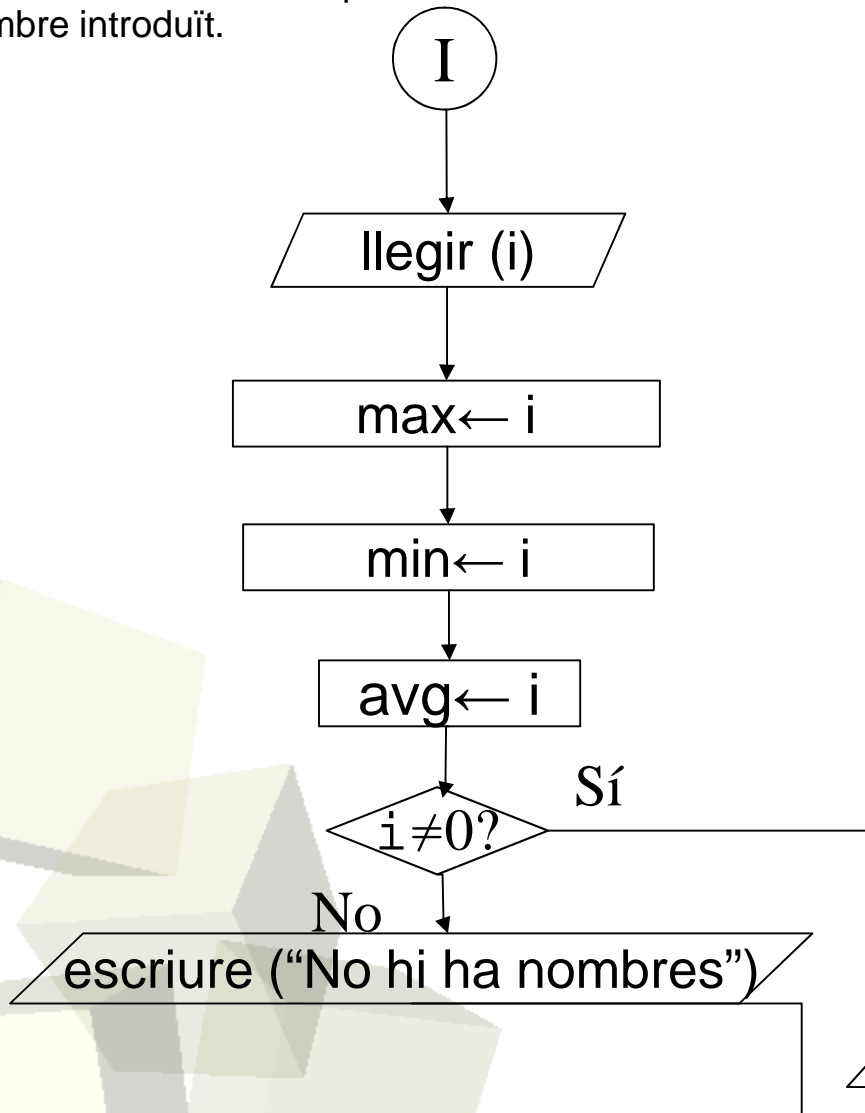
FINAL



Programa: Compta nombres (4/5)

Diagrama de flux:

2a aproximació: Inicialitzar els valors màxim i mínim al primer nombre introduït.





Programa: Compta nombres (5/5)

Pseudocodi:

Variables:

n, max, min, i : enters
avg : real

Inici

```
Llegir(i)
max ← i
min ← i
avg ← i
Si i ≠ 0 aleshores
    n ← 1
    Llegir(i)
    Mentre (i ≠ 0) fer
        Si i < min aleshores
            min ← i
        final_si_1
        Si i > max aleshores
            max ← i
        final_si_2
        avg ← avg + i
        n ← n + 1
        Llegir(i)
    Final_mentre
    avg ← avg / n
    Escriure(n, max, min, avg)
Si no
    Escriure("No hi ha nombres")
Final_si
FINAL
```



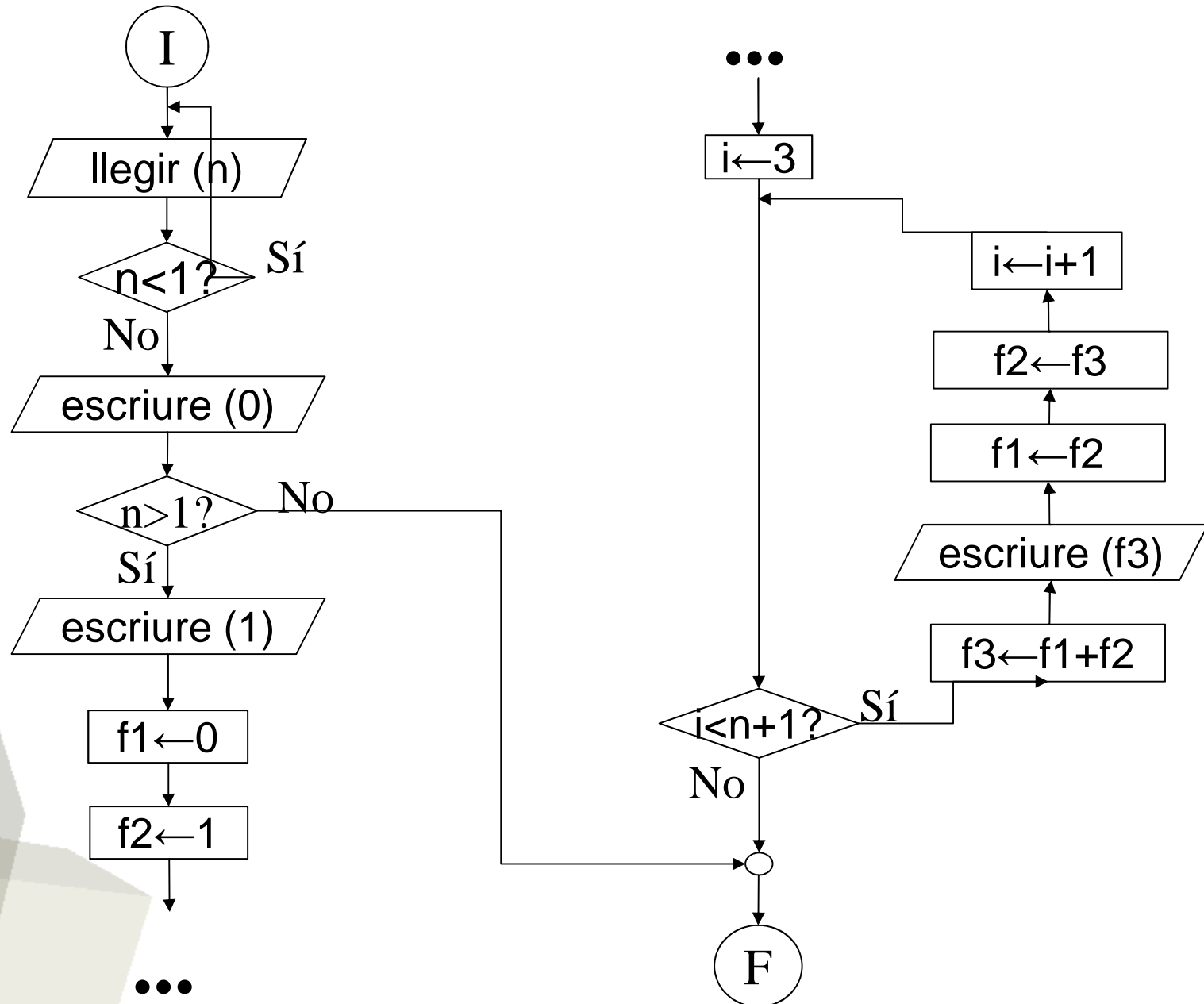
Programa: Fibonacci (1/3)

- La successió de Fibonacci és una seqüència infinita en la qual el primer element és un 0, el segon és un 1 i la resta d'elements es poden expressar com la suma dels dos elements anteriors.
- Escriviu un programa que mostre per pantalla els **primers n termes de la successió de Fibonacci**, on n és un nombre obtingut des del teclat.



Programa: Fibonacci (2/3)

Diagrama de flux:





Programa: Fibonacci (3/3)

Pseudocodi:

Variables:

$n, i, f1, f2, f3$: enters

Inici

Fer

Llegir(n)

Mentre₁ ($n < 1$)

Escriure(0)

Si $n > 1$ aleshores

Escriure(1)

$f1 \leftarrow 0$

$f2 \leftarrow 1$

Des de $i \leftarrow 3$ fins a n fer

$f3 \leftarrow f2 + f1$

Escriure($f3$)

$f1 \leftarrow f2$

$f2 \leftarrow f3$

$i \leftarrow i + 1$

Final_des_de

Final_si

FINAL