

## Problemes capítol 4

Realitzar cada exercici donant el pseudocodi, el diagrama de flux i el codi C.

Considerar sempre totes les possibilitats

1. Donat com a entrada un any, calcular si és de traspàs.
2. Fer un programa que doni l'estimació de la població mundial per un any posterior al 1990. L'estimació es farà en base a la fórmula  $em = 5,5 \cdot (1 + e^{0,02 \cdot (a-1990)})$ .
3. Calcular  $a^n$ ,  $n^a$
4. Calculeu l'àrea d'un triangle coneixent els tres costats (fórmula d'Heró):

$$A = \sqrt{p(p-a)(p-b)(p-c)}, \text{ on } p = \text{semiperímetre.}$$

Nota: Cal esbrinar, tot primer, si els tres costats formen un triangle: ( $p > a$ ,  $p > b$  i  $p > c$ )

5. Donat un capital C, posat a un rèdit r, calcular en quants anys es doblarà

$$C_i = \left( C_{i-1} + C_{i-1} \cdot \frac{r}{100} \right).$$

6. Escriure un programa que tregui per pantalla la següent seqüència: 0, 01, 012, 0123. ..., 0123456789.
7. Llegir un conjunt d'enters positius per pantalla. Donar el més gran i el nombre de cops que ha aparegut. Sortir en el moment d'entrar un 0.
8. Generar la sèrie:  $a, a + (d-a), a + (d-a) + (2d+a), a + (d-a) + (2d+a) + (3d-a) + \dots$
9. Calcular les mitges aritmètica, geomètrica i harmònica de n valors (enters i positius) que l'usuari entrat per teclat. La seqüència de nombres finalitza quan l'usuari entra un 0.

10. Calcular el valor de e a partir de la seva sèrie  $e = 1 + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \dots$  amb un error màxim de  $10^{-9}$ .

Comparar el resultat amb el seu valor real.

11. Aproximar el nombre  $\pi$ :  $\pi = 4 \cdot (1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots)$ . Comprovar la convergència molt lenta de la sèrie.

12. Calcular  $\sin(x)$  a partir de la sèrie  $\sin(x) = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$  vàlida fins a  $\pi/2$ , amb un error màxim de  $10^{-5}$ . Entrar els angles en graus, tenint en compte que x està expressat en radians. Comparar el resultat amb el valor real de  $\sin(x)$ . Cercar fins a quin angle és vàlida l'aproximació.

13. Calcular  $\sum_{i=1}^n i!$ . Controlar el valor n fins al qual és vàlid el valor (**long int**) calculat.

14. Donar el voltatge que veuran els borns d'una capacitat durant el transitori d'un circuit RC, alimentat a 10V. El circuit donarà una sortida en forma de taula  $v=v(t)$ . Com a dades d'entrada s'entraran R i C. Un factor temporal d'ajust RC en l'escala temporal donarà un resultat més ajustat per una taula més petita.

15. Fer el programa que calcula la taula del polinomi  $f(x)=2x^2-x-1$  des de  $x=-2$  fins a  $x=+2$ , amb passos de 0.25

16. Es demana trobar les arrels del polinomi  $f(x)=4 \cdot x^3 + 4 \cdot x^2 - x - 1$ . Se sap que les arrels es troben entre -2 i 2. Ajustar a error en els zeros del 1/1000.

Ajut: Tot i que aquest exemple és senzill, trobar les arrels d'una equació pot tenir la seva gràcia. En aquest cas, agafant increments de l'índex de 0.1 surt senzill.

**17.** Realitzar una calculadora. El programa ha d'admetre les 4 operacions elementals: +, -, \* i /. Les expressions s'entraran en el format natural: a + 4

**18.** Escriure tots els caràcters codi ASCII en una taula on hi apareguin:

codi hex	codi dec	codi oct	codi ASCII
----------	----------	----------	------------

Agrupar la sortida en grups de 22 files per a la seva visibilitat

**19.** Realitzar un programa que escrigui per pantalla tots els caràcters entrats per teclat fins que s'hagi entrat un \*. Després s'escriurà el nombre de caràcters escrits.

**20.** Imprimir un romb de n (senar) estrelles d'amplada

```

      *
     ***
    *****
   *********
  ***********
 * **********
*  **********
*   *******
*    *****
*     ****
*      ***
*       *
```

# Solucions problemes

## Exercici 4.1

Pseudocodi

Entrades: any // any

Sortides: Si traspàs

Enter any

Inici

Llegir (any)

Si (any%400=0) Escriure("Traspàs")

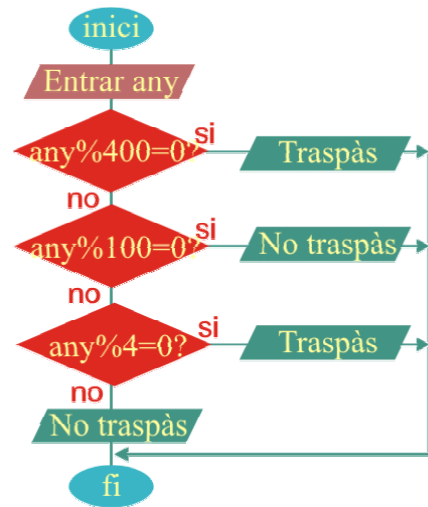
SiNo Si (any%100=0) Escriure("No traspàs")

SiNo Si (any%4=0) Escriure("Traspàs")

SiNo Escriure("No traspàs")

FiSi

Fi



## Codificació en C

```
/*
Calcul si any es de traspas.
*/
#include <stdio.h>
void main()
{
    int any;
    printf("CALCUL SI, DONAT UN ANY, ES DE TRASPAS\n");
    printf("Entra any: ");
    scanf("%u", &any);
    if (any%400==0) printf("...any de traspas!\n");
    else if (any%100==0) printf("...any no de traspas!\n");
    else if (any%4==0) printf("...any de traspas!\n");
    else printf("...any no de traspas!\n");
}
```

## Exercici 4.2

### Pseudocodi

Entrades: any // any

Sortides: Estimació població

Enter any

Real em // Estimació població

Inici

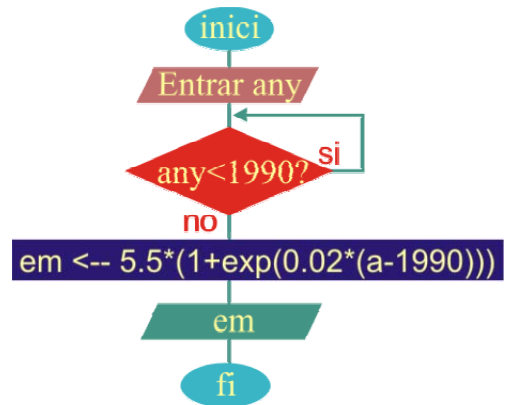
Fer

Llegir (any)

Mentre (any<1990)

Escriure(em)

Fi



### Codificació en C

```
/*
Estimació de la població mundial anual
No admet anys inferiors al 1990
*/
#include <stdio.h>
#include <math.h>
void main()
{
    int a;
    float em;
    printf("ESTIMACIO DE LA POBLACIO MUNDIAL (en milions d'habitants)\n");
    do
    {
        printf("Dona l'any (>=1990): ");
        scanf("%i", &a);
    }while (a<1990);
    em = 5.5*(1+exp(0.02*(a-1990)));
    printf("La poblacio mundial a l'any %i sera de %3.1f \n", a, em);
}
```

### Exercici 4.3

Pseudocodi

emprant for

Entrades: a i n // valors

Sortides:  $a^n$  i  $n^a$

Enter a, n

Inici

```
Llegir(a, n)
pot ← 1
Per (i=0; i<n) Fer
    pot ← pot*a
    i ← i+1
FiPer
pot ← 1
Per (i=0; i<a) Fer
    pot ← pot*n
    i ← i+1
FiPer
Escriure(pot)
```

Fi

Pseudocodi

emprant while

Entrades: a i n // valors

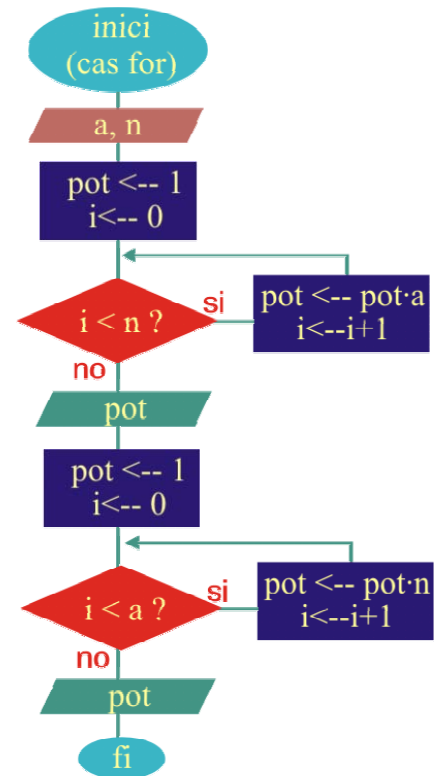
Sortides:  $a^n$  i  $n^a$

Enter a, n

Inici

```
Llegir(a, n)
pot ← 1
i ← 0
Mentre (i<n) Fer
    i ← i+1
    pot ← pot*a
FiMentre
Escriure(pot)
pot ← 1
i ← 0
Mentre (i<a) Fer
    i ← i+1
    pot ← pot*n
FiMentre
Escriure(pot)
```

Fi



### Codificació en C

```
/*Calcular a^n, n^a*/
#include <stdio.h>
void main()
{
    int a, n, pot, i;
    printf("Entra els nombres a i n: "); //Entrar a i n
    scanf("%i%i", &a, &n);
    printf("\nEmprant for\n"); //emprant instrucció for
    pot=1;
    for (i=0; i<n; i++) pot *= a; //a^n
    printf("\ta^n = %i\n", pot);

    pot=1;
    for (i=0; i<a; i++) pot *= n; //n^a
    printf("\tn^a = %i\n", pot);

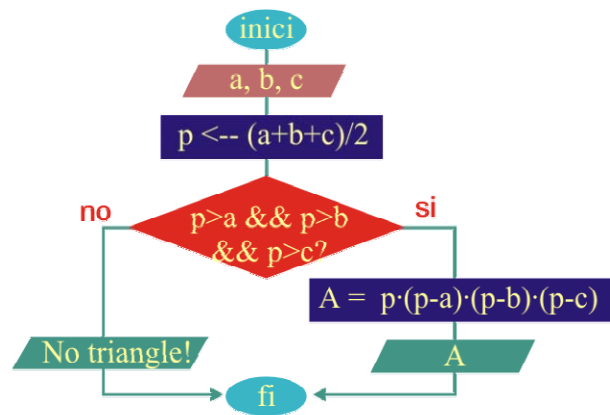
    printf("\nEmprant while\n"); //emprant instrucció while
    pot=1;
    i=0;
    while(i++<n) pot *= a; //a^n
    printf("\ta^n = %i\n", pot);

    pot=1;
    i=0;
    while(i++<a) pot *= n; //n^a
    printf("\tn^a = %i\n", pot);
}
```

## Exercici 4.4

### Pseudocodi

```
Entrades: a, b, c //Costats triangle
Sortides: A //Àrea triangle
Float a, b, c //Costats triangle
Float p, A //Semiperímetre, àrea
Inici
  Llegir (a, b, c)
   $p \leftarrow (a+b+c)/2$ 
  Si (p>a && p>b && p>c) Llavors
     $A = \sqrt{p \cdot (p-a) \cdot (p-b) \cdot (p-c)}$ 
    Escriure(A)
  Sino
    Escriure("No hi ha triangle!")
  FiSi
Fi
```



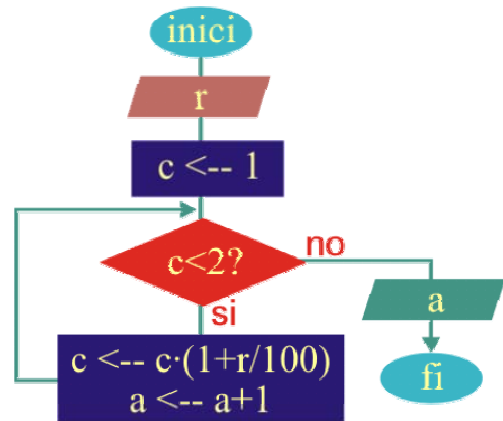
### Codificació en C

```
/*
Calcul de l'àrea d'un triangle a partir de la fórmula d'Heró
*/
#include <stdio.h>
#include <math.h>
void main()
{
    float a, b, c, p, A;
    printf("AREA D'UN TRIANGLE\n");
    printf(" Entra els costats (a, b, c): "); //Entrar costats triangle
    scanf("%f%f%f", &a, &b, &c);
    p=(a+b+c)/2; //Semiperímetre
    if (p>a && p>b && p>c) //Comprovació triangularitat
    {
        A = sqrt(p*(p-a)*(p-b)*(p-c)); //Càlcul àrea
        printf("\tArea = %3.1f\n", A);
    }
    else
        printf("Això no és un triangle\n");
}
```

## Exercici 4.5

### Pseudocodi

```
Entrades: r           // r dit
Sortides: a           // anys
Enter a               // anys
Float c, r            // capital, r dit
Inici
Llegir (r)            // Entrar el r dit
c ← 1                 // Capital inicial = 1
Fer
  c ← c · (1+r/100)   // Increment de capital
  a ← a+1             // Increment d'anys
Mentre (c<2)
  Escriure("Anys =", a)
Fi
```



### Codificaci  en C

```
/*
Calcul inter s capital C a un r dit r
*/
#include <stdio.h>
void main()
{
    int a=0;
    float c, r;
    printf("CALCUL ANYS PER DOBLAR CAPITAL\n");
    printf("Entra redit (en %): ");           // Entrar el r dit
    scanf("%f", &r);
    c=1;                                       // Capital inicial = 1
    do
    {
        c=c*(1+r/100);                       // Increment de capital per any
        a++;
    }while (c<2);
    printf("\tEl nombre d'anys es %i\n", a);
}
```

## Exercici 4.6

Pseudocodi

Entrades: cap

Sortides: seqüència

Enter a, n

Inici

Llegir (a, n)

$j \leftarrow 0$

Mentre ( $j < 10$ ) Fer //Bucle línia

$j \leftarrow j + 1$

Per ( $i = 0; i < j$ ) Fer //Bucle nums

Escriure (i)

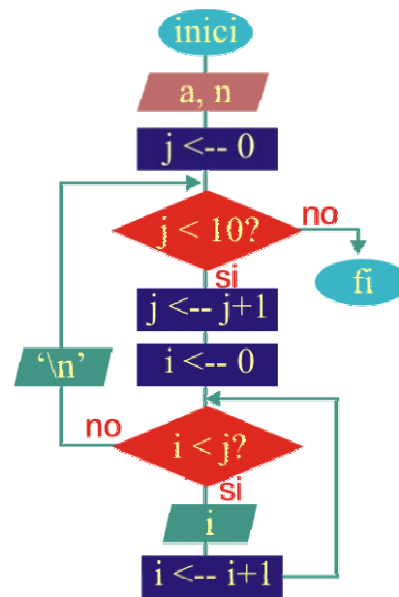
$i \leftarrow i + 1$

FiPer

Escriure ('\n')

FiMentre

Fi



## Codificació en C

/\*

Escriure un programa que tregui per pantalla el següent resultat  
(emprant estructures iteratives)

0

01

012

0123

...

0123456789

\*/

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    int i,j=0;
```

```
    while(j++<10) //Bucle que porta el compta de línia
```

```
    {
```

```
        for(i=0;i<j;i++) printf("%i", i); //Bucle que porta el compta de l'escriptura per línia
```

```
        printf("\n");
```

```
    }
```

```
}
```



## Exercici 4.7

Pseudocodi

Entrades: Nombres enters

Sortides: El nombre més gran i  
quants cops ha aparegut.

Enter i, num, rep

Inici

Fer

num ← 0

rep ← 0

Llegir (i) //Llegir nombre

Si (num==i) Llavors //num ja entrat

rep ← rep+1

SiNo Si (i>num) Llavors //num major

num ← i

rep ← 0

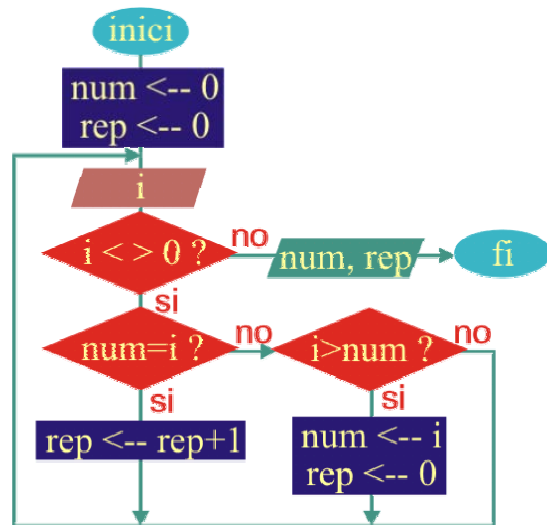
FiSi

FiSi

Mentre (i<>0)

Escriure (num, rep)

Fi



## Codificació en C

/\*

Llegir un conjunt d'enters positius per pantalla. Donar el mes gran i el nombre de cops que ha aparegut. Sortir en el moment d'entrar un 0

\*/

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    int i, num=0, rep=0;
```

```
    do
```

```
    {
```

```
        printf(" ?");
```

```
        scanf("%i", &i);
```

```
        if(num==i) rep++;
```

```
        else if (i>num)
```

```
        {
```

```
            num=i;
```

```
            rep=1;
```

```
        }
```

```
    }while (i!=0);
```

```
    printf("El numero mes gran es %i \n", num);
```

```
    printf("i ha aparegut %i cops\n", rep);
```

```
}
```

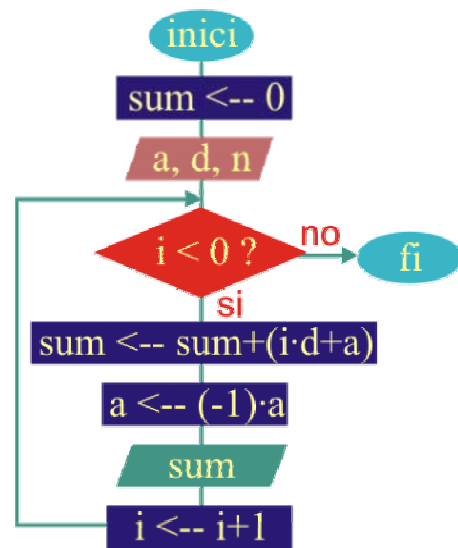
## Exercici 4.8

### Pseudocodi

Entrades: Nombres a i d

Sortides: Sèrie a, a+(d-a), a+(d-a)+(2d+a), ...

```
Inici
sum ← sum=0
Llegir (a, d, n)
Per (i=0; i<n) Fer //Bucle per n termes
    sum ← sum +(i*d+a)
    a ← (-1)*a
    Escriure (sum)
    i ← i+1
FiPer
Fi
```



### Codificació en C

```
/*
Generació de la serie
    a, a + (d-a), a + (d-a) + (2d+a), a + (d-a) + (2d+a) + (3d-a) + ...
*/
#include <stdio.h>
void main()
{
    int a, d, n, sum=0;
    printf(" Entra a i d: "); //Entrada de valors
    scanf("%i%i", &a, &d);
    printf(" Quants termes de la serie? ");
    scanf("%i", &n);
    for (int i=0;i<n;i++) //Bucle per n termes
    {
        if (i%10 == 0) printf("\n"); //Control per imprimir 10 termes per línia
        sum += i*d+a;
        a=(-1)*a;
        printf(" %3i", sum);
    }
    printf("\n\n");
}
```

## Exercici 4.9

### Pseudocodi

Entrades: Nombres

Sortides: Mitges aritmètica, geomètrica, harmònica

```
Real m_a, m_g, m_h //Mitges
Real t //Número que s'entra
Enter n //Control de nums entrats
Inici
```

```
n ← 0 //Inicialitzacions
```

```
m_a ← 0
```

```
m_g ← 0
```

```
m_h ← 0
```

```
Llegir (t)
```

```
Mentre (t>0) Fer //Càlculs intermitjos
```

```
n ← n+1
```

```
m_a ← m_a+t
```

```
m_g ← m_g*t
```

```
m_h ← m_h+1/t
```

```
Llegir (t)
```

```
FiMentre
```

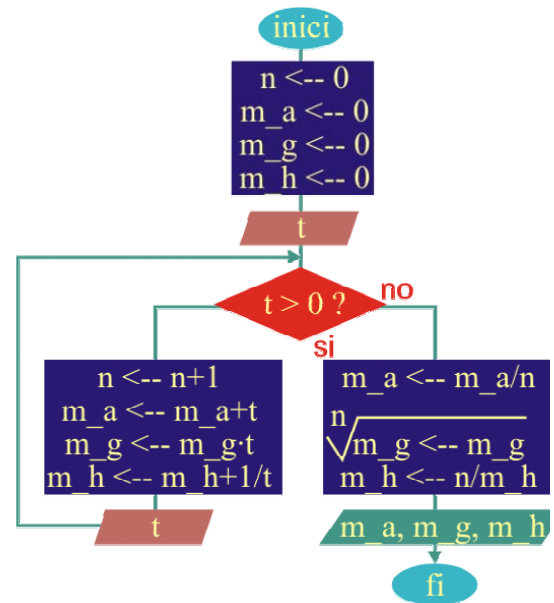
```
m_a ← m_a/n //Càlculs finals
```

```
m_g ← pow(m_g, 1.0/n)
```

```
m_h ← n/m_h
```

```
Escriure (m_a, m_g, m_h)
```

Fi



### Codificació en C

```
#include <stdio.h>
#include <math.h>
/*Mitges aritmètica, geomètrica i harmònica.*/
void main()
{
    int n=0; //Inicialitzacions
    float t, m_a=0, m_g=1, m_h=0;
    printf(" Entra terme: ");
    scanf("%f", &t);
    while (t>0) //Càlculs intermitjos
    {
        n++;
        m_a += t;
        m_g *= t;
        m_h += 1.0/t;
        printf(" Entra terme: ");
        scanf("%f", &t);
    }
    m_a /= n; //Càlculs finals
    m_g = pow(m_g, 1.0/n);
    m_h = n/m_a;
    printf("Mitja aritmetica = %5.2f\n", m_a);
    printf("Mitja geometrica = %5.2f\n", m_g);
    printf("Mitja harmonica = %5.2f\n", m_h);
}
```

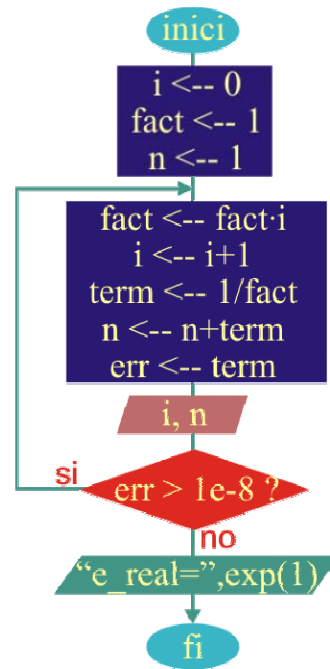
## Exercici 4.10

### Pseudocodi

Entrades: No n'hi ha

Sortides: Aproximació d'e

```
Enter i, fact      //Índex, factorial
Real n, err, term  //e, error, terme
Inici
  i ← 0            //Inicialiació
  fact ← 1
  n ← 1
  Fer
    fact ← fact*i  //factorial
    i ← i+1
    term ← 1.0/fact //terme suma
    n ← n+term     //e
    err ← term     //error
    Escriure (i, n)
  Mentre (err>1e-8) //Bucle fins error acomplert
  Escriure ("e_real=" exp(1))
Fi
```



### Codificació en C

```
/*
Calcul de e a partir de la seva sèrie amb un error inferior a 10-9
*/
#include <stdio.h>
#include <math.h>

void main()
{
    int i=0, fact=1;           //Índex, factorial
    double n=1, err, term;    //e, error, terme
    printf("APROXIMACIO DE e...\n");
    do
    {
        fact *= ++i;          //factorial
        term=1.0/fact;        //terme suma
        n += term;            //e
        err = term;           //error
        printf("\ti=%i\te=%10.9f\n", i, n);
    }while (err>1e-8);
    printf("\te real = %10.9f\n", exp(1));
}
```

## Exercici 4.11

### Pseudocodi

Entrades: No n'hi ha

Sortides: Aproximació de  $\pi$

Enter i //Índex

Real pi, s, err, term // $\pi$ , signe, error, terme suma

Inici

pi  $\leftarrow$  0

s  $\leftarrow$  -1

i  $\leftarrow$  1

Fer

s  $\leftarrow$  s\*(-1) //signe a cada iteració

term  $\leftarrow$  s/i

pi  $\leftarrow$  pi+4\*term //pi a cada iteració

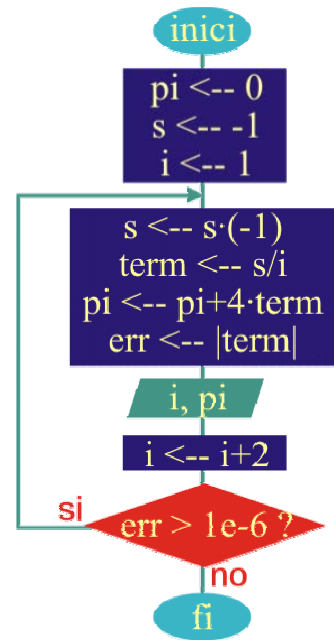
err  $\leftarrow$  |term| //error a cada iteració

Escriure (i, pi)

i  $\leftarrow$  i+2

Mentre (err>1e-6) //Bucle fins error acomplert

Fi



### Codificació en C

/\*

Aproximació de PI

\*/

#include <stdio.h>

#include <math.h>

void main()

{

double pi=0, s=-1, err, term; //Inicialització

int i=1;

printf("APROXIMACIO DE PI...\n");

do

{

s \*= -1.0; //signe de cada terme

term = s/i; //El fet que s sigui real fa que aquesta operació no sigui entera!

pi += 4\*term; //pi a cada iteració

err = term<0?-term:term; //error a cada iteració

printf("%i %8.7f \n", i, pi);

i+=2;

}while (err>1e-6); //Control de l'error

}

## Exercici 4.12

### Pseudocodi

Entrades: Angle x

Sortides: sin(x)

Enter i, fact //Índex, factorial  
 Real x, sinus, err //Entrada, sinus, error  
 Real xpot, t //Càlculs intermitjos  
 Inici

i ← fact ← 1

Llegir (x)

x ← x·π/180

xpot ← sinus ← x

fact ← 1

Fer

xpot ← (-1)·xpot·x·x

fact ← fact·(2·i)·(2·i+1)

t ← xpot/fact

sinus ← sinus+t

err ← |t|

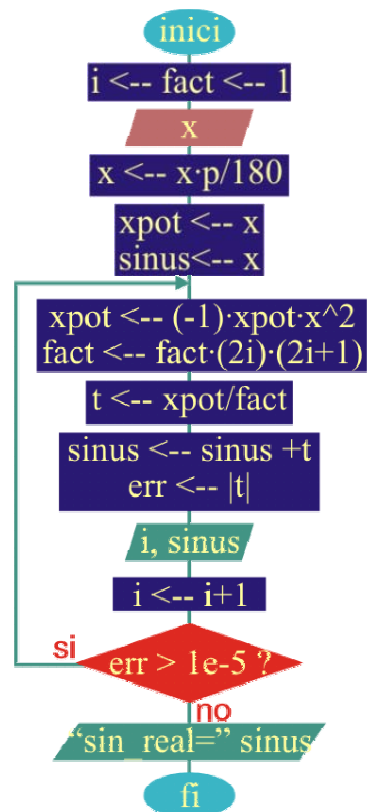
Escriure (i, sinus)

i ← i+1

Mentre (err>1e-5) //Bucle fins error acomplert

Escriure ("sinus\_real=", sin(x))

Fi



### Codificació en C

```
#include <stdio.h>
```

```
#include <math.h>
```

```
#define PI 3.14159
```

```
/*
```

Calcul de sin(x) a partir de la sèrie. Comparar el resultat amb el valor real de sin(x)

```
*/
```

```
void main()
```

```
{
```

```
int i=1, fact=1;
```

```
//Variables índex i factorial
```

```
float x;
```

```
//Variables entrada (en graus)
```

```
double sinus, xpot, t, err;
```

```
//Variables sinus, potència(x), terme intermig, error
```

```
printf("APROXIMACIO DEL sin(x)...\n");
```

```
printf("Entra angle: ");
```

```
//Entrada de l'angle
```

```
scanf("%f", &x);
```

```
x *= PI/180;
```

```
xpot = sinus = x;
```

```
do
```

```
{
```

```
    xpot *= (-1)*x*x;
```

```
    fact *= (2*i)*(2*i+1);
```

```
    t = xpot/fact;
```

```
    sinus += t;
```

```
    err = t<0?-t;t;
```

```
//Valor absolut del terme per calcular l'error
```

```
    printf("\ti=%i\tsin=%6.5f\n", i++, sinus);
```

```
//Impressió sinus sèrie
```

```
}while (err>1e-5);
```

```
printf("\tsinus real (x) = %f\n", sin(x));
```

```
//Impressió sinus real
```

```
}
```

### Exercici 4.13

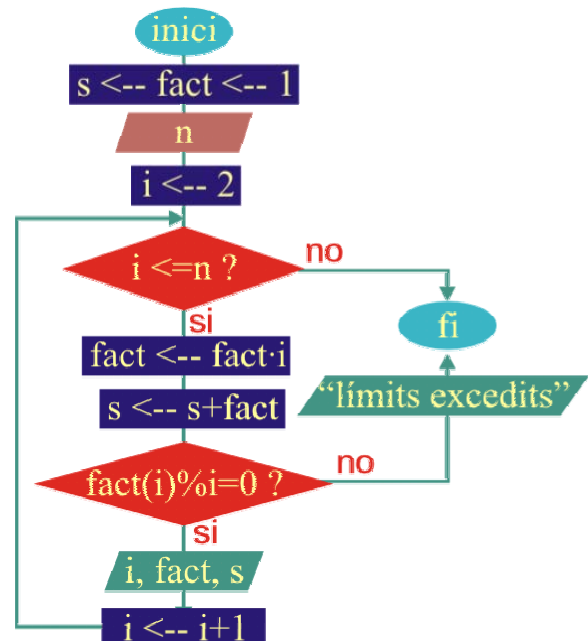
Pseudocodi

emprant for

Entrades: n // Valor fins on es calcula

Sortides:  $s = \sum_{i=1}^n i!$

```
Enter i, n //Índex, valor n
Enter fact, s //Factorial, suma factorial
Inici
fact ← 1
s ← 1
Llegir (n) //Entrada de n
Per (i=2; i<=n) Fer
    fact ← fact*i //Càlcul fact(i)
    s ← s+fact //Suma de factorials
    Si ( fact(i)%i=0 ) Llavors //Control error
        Escriure(i, fact, s)
    SiNo
        Escriure("Limits excedits!")
        Fi //És 'break' → sortida ràpida
    FiSi
    i←i+1
FiPer
Fi
```



### Codificació en C

```
/*
Calcul de S(i!), per i=1 fins a n.
*/
#include <stdio.h>
void main()
{
    int i, n;
    unsigned long fact=1, s=1;
    printf("CALCUL DE S(i!)\n");
    printf(" Entra n: "); //Entrada de n
    scanf("%i", &n);
    printf("\t n\t\t n!\t\t S(n!)\n");
    for (i=2; i<=n; i++)
    {
        fact *= i; //Càlcul fact(i)
        s += fact; //Suma de factorials
        if (!(fact%i)) //Control correctitud factorial calculat
            printf("\t %2i\t %11u\t %11u\n", i, fact, s);
        else
        {
            printf("\n A partir de %i ...s'excedeix la capacitat de representacio!\n", i);
            break;
        }
    }
}
```

## Exercici 4.14

### Pseudocodi

Entrades: Valors x

Sortides: Per a cada x, f(x)

Enter Constant  $V=10$

Real r, c, t, v //Índex, f(índex)

Inici

Llegir (r, c)

Escriure("--t(mseg)---v(mV)--")

Per (i=0;i<20) Fer //taula 20 valors

t ← r·c·i/10 //valor t escalat

v ← (1-e<sup>-t/(r·c)</sup>)·V/r //valor v(t)

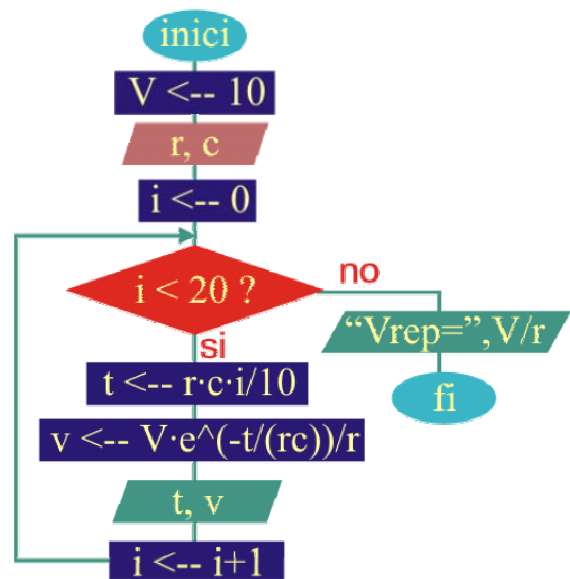
Escriure (t, v)

i ← i+1

FiPer

Escriure("Vrepos = ", V/r)

Fi



### Codificació en C

/\*

Càlcul del transitori (voltatge entre els borns d'una capacitat) d'un circuit RC, alimentat a 10V  
Posar un factor d'escala (rc) per ajustar els valors de l'escala temporal

\*/

#include <stdio.h>

#include <math.h>

const int V=10;

void main()

{

float r, c, t, v;

printf("Entra r(kohm) i c (uF): ");

scanf("%f%f", &r, &c);

printf(" t(mseg) v(mV)\n");

printf("-----\n");

for (int i=0; i<20; i++)

{

t = r\*c\*i/10;

v =(1-exp(-t/(r\*c)))\*V/r;

printf(" %5.2f %5.2g\n", t, v);

}

printf("\n Valor en repos: %3.2g mV\n", V/r);

}

//Voltatge entre borns circuit rc

//resistència, capacitat, temps i voltatge

//Entrada: r, c

//Imprimir capçalera

//Taula de 20 valors

//Factor d'escala

//Càlcul v(t)

//Imprimir (t,v(t))

//Vrepos



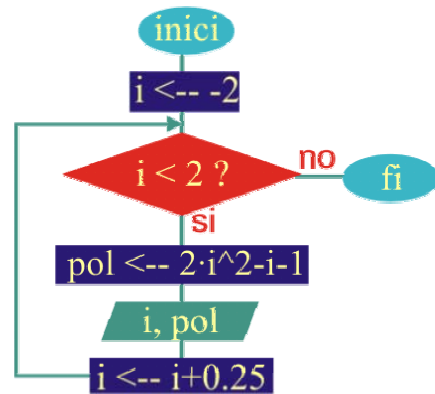
### Exercici 4.15

#### Pseudocodi

Entrades: Valors x

Sortides: Per a cada x, f(x)

```
Real i, pol //Índex, f(índex)
Inici
  Escriure("----i-----pol---")
  Per (i=-2;i<=2) Fer
    pol ← 2·i2-i-1 //valors f(x)
    Escriure (i, pol)
    i ← i+0.25
  FiPer
Fi
```



#### Codificació en C

```
/*
Taula del polinomi f(x)=2x2-x-1
des de x=-2 fins a x=+2, amb passos de 0.25
*/
#include <stdio.h>
void main()
{
    float pol;
    printf("----i-----pol---\n"); //Format presentació
    for (float i=-2; i<=2; i+=0.25) //Bucle de valors
    {
        pol=2*i*i-i-1;
        printf("\t%4.2ft%4.2f\n", i, pol); //Imprimir x, f(x)
    }
}
```

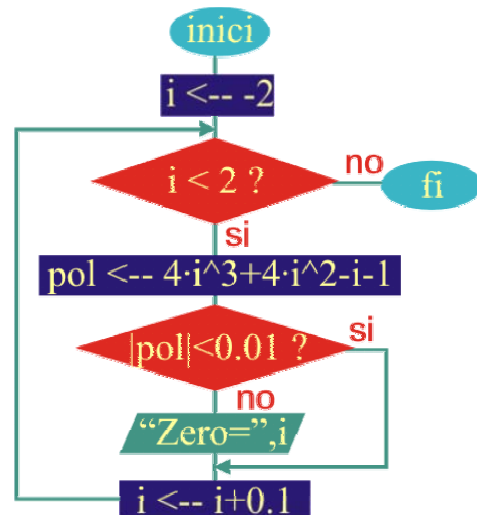
### Exercici 4.16

Pseudocodi

Entrades: No n'hi ha d'externes

Sortides: Per a cada x,  $f(x)=pol$

```
Real i, pol //Índex, f(índex)
Inici
Per (i=-2;i<=2) Fer
    pol ←  $4 \cdot i^3 + 4 \cdot i^2 - i - 1$  //valors f(x)
    Si ( $|pol| < 0.001$ ) Llavors //Comprovació zero
        Escriure ("Zero a ",i)
    FiSi
    i ← i+0.1
FiPer
Fi
```



### Codificació en C

```
/*
Trobar les arrels del polinomi  $f(x)=6x^3+5x^2-2x-1$ . Ajustar a error (en el zero) del 1/1000.
*/
```

```
#include <stdio.h>
#include <math.h>
void main()
{
    float pol;
    for (float i=-2; i<=2; i+=0.1)
    {
        pol=4*i*i*i + 4*i*i - i - 1; //Valor de la funció per i actual
        if (fabs(pol)<=0.001) //Càlcul error
            printf("\t\tZero a %5.3f\n", i);
    }
}
```

## Exercici 4.17

### Pseudocodi

Entrades: Expressió: (2 nombres + 1 caràcter)

Sortides: \*, -, \*, / ò %(mòdul)

Real a, b, res //Operands i resultat

Caràcter car

Inici

Fer

Llegir ("Operació = ", a, car, b)

EnCasDe (car) Fer

'+": res=a+b //Suma

Fi+

'-': res=a-b //Resta

Fi-

'\*': res=a\*b //Multiplicació

Fi\*

!': res=a/b //Divisió

Fi/

'%': res=a%b //Mòdul

Fi%

FiCas

Escriure (res) //Escriure resultat

Llegir ("Altre operació?", car)

Mentre (car<>'Salt de línia')

Fi

### Codificació en C

```
/* Calculadora de +,-,*,/,%.
```

```
Format d'entrada: 3+4 */
```

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
float a, b, res;
```

```
char car;
```

```
do
```

```
printf("Entra operacio (ex: 3 + 4): ");
```

```
scanf("%f%c%f", &a, &car, &b);
```

```
switch(car)
```

```
{
```

```
case '+': res= a+b;
```

```
break;
```

```
case '-': res= a-b;
```

```
break;
```

```
case '*': res= a*b;
```

```
break;
```

```
case '!': res= a/b;
```

```
break;
```

```
case '%':res= int(a)%int(b);
```

```
}
```

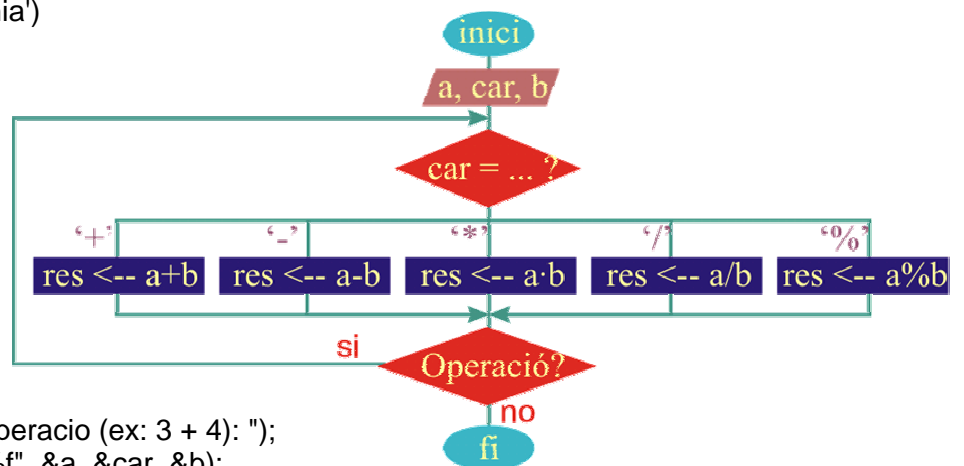
```
printf(" --> =%4.2f\n", res);
```

```
printf("Operacio (s/n)? ");
```

```
fflush(stdin);
```

```
}while(getchar()!='\n');
```

```
}
```



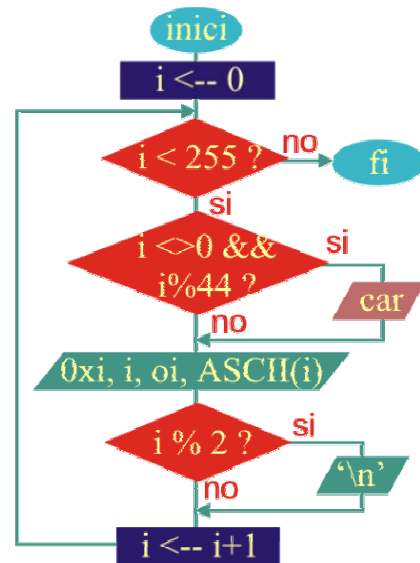
## Exercici 4.18

### Pseudocodi

Entrades: No n'hi ha d'externes

Sortides: Taula ASCII → Caràcters

```
Enter i //Índex
Inici
  Per (i=0;i<255) Fer
    Si (i<>0 && i%44=0) Llavors
      Llegir (caràcter) //Esperar
    FiSi
    Escriure (0xi, i, oi, ASCII(i)) //caràcters
    Si (i%2) Llavors
      Escriure('\n') //dos valors per línia
    FiSi
  FiPer
Fi
```



### Codificació en C

```
/*
Escriure tots els caràcters codi ASCII en una taula:
    codi hex      codi dec      codi oct      ASCII
Agrupar en grups de 22 files per a la seva visibilitat
*/
#include <stdio.h>

void main()
{
    int i;
    printf(" -hex---dec----oct---car---  -hex---dec----oct---car---\n");
    for (i=0; i<255; i++)
    {
        fflush(stdin); //Netejar buffer d'entrada
        if((i!=0)&&(i%44==0)) getchar(); //Esperar '\n' per altres 22 files
        printf(" 0x%2x\t%3d\t%2o\t%c\t", i, i, i, i); //Imprimir caràcter
        if (i%2) printf("\n"); //Imprimir 2 caràcters per línia
    }
}
```

### Exercici 4.19

Pseudocodi

Entrades: Caràcters

Sortides: Els caràcters entrats

Caracter ch //Caràcter llegit  
Enter n //Nombre caràcters entrats

Inici

n ← 0

Fer

Llegir (ch) //valors f(x)

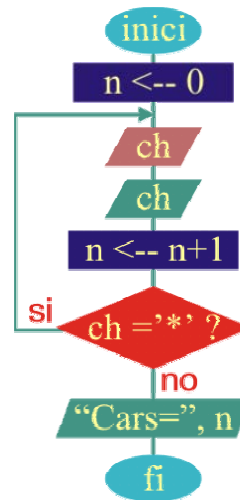
Escriure (ch)

n ← n+1

Mentre (ch<>'\*')

Escriure ("Nombre de caràcters = ",n)

Fi



### Codificació en C

/\*

Realitzar un programa que tregui per pantalla tots els caràcters entrats per teclat fins que s'entri \*  
Després s'escriurà el nombre de caràcters escrits.

\*/

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
    char ch;
```

```
    int n=0;
```

```
    do
```

```
    {
```

```
        ch=getchar();
```

```
        putchar(ch);
```

```
        n++;
```

```
    }while (ch != '*'); //es compta l'asterisc
```

```
    printf("\nNombre de caracters = %2i\n\nto", n);
```

```
}
```

### Exercici 4.20

Per a realitzar l'exercici és convenient entendre el que s'ha de mostrar per pantalla. Si agafem una entrada de 7 asteriscs, per exemple, es pot veure que el nombre d'espais a deixar i el nombre d'asteriscs a dibuixar estan relacionats amb l'índex que agafarem:

Cas	n_espais	índex	n_*
*	2	-2	1
***	1	-1	3
*****	0	0	5
***	1	1	3
*	2	2	1

$$n\_espais = |\text{índex}|$$

$$n\_* = 1 + 2 \cdot (n/2 - |\text{índex}|),$$

$$-n/2 < \text{índex} < n/2$$

Per tant, l'algorisme queda:

#### Pseudocodi

Enter a, absi, ast, i, j //Entrada, |a|, num\_\*, índexs  
Inici

```

Fer
  Llegir (a) //Entrada
Mentre (a%2=0)
  Per (i=-a/2; i<=a/2) Fer
    Si (i<0) Llavors absi=-i
    SiNo absi=i
  FiSi
  Per (j=0; j<absi) Fer
    Escriure(" ")
    j ← j+1
  FiPer
  ast= 1+2·(a/2 - absi);
  Per (j=0; j<ast) Fer
    Escriure("***")
    j ← j+1
  FiPer
  i ← i+1
FiPer
Fi
  
```

#### Codificació en C

/\*Fer un romb de n (senar) estrelles d'amplada.\*/

```
#include <stdio.h>
void main()
```

```

{
  int a, absi, ast, i, j; //Valor d'entrada, |a|, num asteriscs a dibuixar, índexs
  do
  {
    printf("Entra amplada de romb (senar): ");
    scanf("%i", &a);
  }while (a%2==0); //S'exigeix entrar un número senar
  for (i=-a/2; i<=a/2; i++)
  {
    absi=i<0?-i; //Càlcul |índex|
    for(j=0; j<absi; j++) printf(" "); //num_espais
    ast= 1+2*(a/2 - absi); //num_*
    for(j=0; j<ast;j++) printf("***");
    printf("\n");
  }
}
  
```

