

Control de programación en Matlab

Crea un archivo de texto chamado `datos_exame1.txt` co seguinte contido:

```
1 0 -1
0 -2 1
1 -1 0
2 -1 3
```

Escrebe un programa de Matlab chamado `exame1.m` que lea por teclado o nome do arquivo anterior e lea o arquivo a unha matriz **a**. Logo debe crear o vector **v** concatenando as filas 1 e 2 da matriz **a**, e o vector **w** concatenando as filas 3 e 4 da matriz **a**. O programa debe chamar a unha función `funcion_exame1(...)`, cos argumentos oportunos, que calcule repetidamente os valores $u_i = (\mathbf{v} \cdot \mathbf{w})^i$ con $i = 1 \dots n$, (\cdot representa o produto escalar) sendo n o número de repeticións necesarias para que $s = \sum_{i=1}^n u_i > 10^6$. En cada repetición, o programa debe mostrar por pantalla e almacenar no arquivo `resultados_exame1.txt` una liña cos valores de i e u_i . Logo de chamar a esta función, o programa principal debe mostrar por pantalla o vector **z** cos valores de u_i e a suma s . **NOTA:** debes obter $\mathbf{z}=[6, 36, 1296, 1679616]$, $s=1680948$.

SOLUCIÓN:

```
clear all

nf=input('archivo? ','s');
a=load(nf);
v=[a(1,:) a(2,:)];w=[a(3,:) a(4,:)];

[z s]=funcion_exame1(v,w);

disp('z=');disp(z)
disp('s=');disp(s)

% — archivo funcion_exame1.m —————
function [z s]=funcion_exame1(v,w)
f=fopen('resultados_exame1.dat','w');
if ~l==f
    error('erro fopen resultados_exame1')
end
s=0;i=1;u=v*w';z=[];
while s<le6
    fprintf(f,'%i %i\n',i,u);
    fprintf('%i %i\n',i,u);
    i=i+1;s=s+u;z=[z u];u=u*u;
end
fclose(f);
end
```

Control de programación en Matlab

Escrebe un programa chamado `exame2.m` que lea por teclado unha matriz e calcule o seu número de filas n e columnas m . Se a matriz non é cadrada e $n > m$, debe repetir as $n - m$ primeiras columnas á dereita da matriz (supón que a matriz ten $n - m$ columnas). Se, polo contrario, $m > n$, debe facer o mesmo pero coas filas. Logo, o programa debe chamar á función `calcula(...)`, cos argumentos que creas convintes, que percorra os elementos da matriz **a** e os vaia sumando. Naqueles elementos nos que a suma sexa par, o elemento debe ser almacenado nun vector **x**, e a suma debe multiplicarse nunha variábel p . Tanto **x** como p deben ser retornados pola función. Finalmente, o programa principal debe mostrar almacenar a matriz **a** (cada fila nunha liña)m o vector **x** (tódolos elementos na mesma liña) e o produto p no arquivo `resultados_exame2.txt`. Proba coa seguinte matriz **a**:

```
1 2 3
4 5 6
```

```
clear all
a=input('a[;]? ');
[n m]=size(a);
if n~=m
    dif=abs(n-m);
    if n>m
        a=[a a(:,1:dif)];
    else
        a=[a ; a(1:dif,:)];
    end
end
[x p]=calcula(a);
f=fopen('resultados_exame2.txt','w');
if ~i==f
    error('erro fopen resultados_exame2.txt')
end
fprintf(f,'a=\n');
for i=1:n
    fprintf(f,'%i ',a(i,:));fprintf(f,'\n');
end
fprintf(f,'x=');fprintf(f,'%i ',x);fprintf(f,'\np=%i\n',p);
fclose(f);

% ——— arquivo calcula.m ———
function [x p]=calcula(a)
disp(a)
n=size(a,1);s=0;x=[];p=1;
for i=1:n
    for j=1:n
        t=a(i,j);s=s+t;
        if rem(s,2)==0
            x=[x t];p=p*s;
        end
    end
end
end
end
```

Control de programación en Matlab

Escribe un programa en Matlab chamado `exame3.m` que lea un número t por teclado (usa $t = 5$) e chame á función `serie(...)`, cos argumentos axeitados. Esta función debe aproximar a suma da serie:

$$\sum_{n=0}^{\infty} \frac{n^2 + 1}{t^n} \quad (1)$$

considerando somentes os sumandos maiores de 10^{-4} , e debe retornar a suma s da serie e un vector \mathbf{x} cos sumandos considerados. Logo de chamar a esta función, o programa principal debe pedir por teclado o nome dun arquivo (usa `resultados_exame3.txt`) e almacenar nel a suma s e os sumandos (vector \mathbf{x}), todos na mesma liña do arquivo. Finalmente, se n é o número de sumandos, o programa principal debe crear e mostrar por pantalla unha matriz \mathbf{a} de orde n con elementos dados pola expresión:

$$a_{ij} = \begin{cases} x_i & i = j \\ x_i x_j & i > j \\ x_i + x_j & i < j \end{cases} \quad (2)$$

```
clear all
t=input('t? ');
[s x]=serie(t);
nf=input('arquivo? ','s');
f=fopen(nf,'w');
if ~l==f
    error('erro fopen %s',nf)
end
fprintf(f,'s=%g\n',s);
fprintf(f,'x=');fprintf(f,'%g ',x);fprintf(f,'\n');
fclose(f);

n=length(x);a=zeros(n);
for i=1:n
    for j=1:n
        if i==j
            a(i,j)=x(i);
        elseif i>j
            a(i,j)=x(i)*x(j);
        else
            a(i,j)=x(i)+x(j);
        end
    end
end
disp('a=');disp(a)

% --- arquivo serie.m ---
function [s x]=serie(t)
s=0;x=[];sumando=1;n=0;
while sumando>1e-4
    sumando=(n^2+1)/t^n;
    s=s+sumando;x=[x sumando];n=n+1;
end
end
```

Control de programación en Matlab

Escribe un programa de Matlab chamado `exame4.m` que defina un vector \mathbf{v} con $n=100$ elementos dados por $v_i = \frac{1}{1 + \exp\left(\frac{50-i}{10}\right)}$, con $i = 1, \dots, n$. O programa debe chamar a unha función `integral(...)`, pasándolle os argumentos axeitados, que retorne outro vector \mathbf{p} (primitiva de \mathbf{v}) coa integral indefinida do vector \mathbf{v} , de modo que $p_1 = 0$ e que $p_{i+1} = p_i + hv_i$, con $h = 1$ e $i = 1, \dots, n - 1$. Asemade, a función `integral(...)` debe retornar a integral definida y do cadrado de \mathbf{v} , dada por $y = h \sum_{i=1}^n v_i^2$ con $h = 1$. Logo de chamar á función, o programa principal debe atopar os valores máximo M e mínimo m do vector \mathbf{p} e calcular un novo vector \mathbf{x} cos índices dos elementos p_i do vector \mathbf{p} que verifican $m + r/3 < p_i < M - r/3$, sendo $r = M - m$. Finalmente, o programa debe almacenar no arquivo `resultados_exame4.txt` os vectores \mathbf{p} e \mathbf{x} , e a integral definida y .

```
clear all
n=100;i=1:n;v=1./(1+exp((50-i)/10));
[p y]=integral(v);
M=max(p);m=min(p);r=M-m;p1=m+r/3;p2=M+r/3;
x=find(p>p1 & p<p2);
f=fopen('resultados_exame4.txt','w');
if ~l==f
    error('erro fopen resultados_exame4.txt')
end
fprintf(f,'p=');fprintf(f,'%i ',p);
fprintf(f,'\nx=');fprintf(f,'%i ',x);
fprintf(f,'\ny= %g\n',y);
fclose(f);

% — arquivo integral.m —————
function [p,y]=integral(v)
n=length(v);n1=n-1;p=zeros(1,n);
for i=1:n1
    p(i+1)=p(i)+v(i);
end
y=sum(v.*v);
end
```

Control de programación en Matlab

Escribe un programa de Matlab chamado `exame5.m` que lea por teclado un número ε e a expresión analítica dunha sucesión x_n como unha cadea de caracteres. Usa $\varepsilon = 10^{-5}$ e $x_n = (n^2 + n)^{1/(2n+1)}$. O programa debe converter esta cadea nunha función `f` de Matlab. Logo, debe chamar á función `limite(...)`, pasándolle os argumentos axeitados, que calcule repetidamente x_n ata que $|x_{n+1} - x_n| < \varepsilon$. A función debe retornar un vector `x` cos termos x_n calculados e o límite $l = \lim_{n \rightarrow \infty} x_n$. O programa principal debe almacenar no arquivo `resultados_exame5.txt` o número de termos calculados, os seus valores (un en cada liña) e o límite con 6 decimais.

```
clear all
epsilon=input('epsilon? ');
xn=input('xn? ','s');
f=inline(xn);
[n x l]=limite(f,epsilon);
f=fopen('resultados_exame5.txt','w');
if ~l==f
    error('erro fopen resultados_exame5.txt')
end
fprintf(f,'numero de termos=%i\n',n);
fprintf(f,'x=\n');fprintf(f,'%g\n',x);
fprintf(f,'limite= % .6f\n',l);
fclose(f);

% -- arquivo limite.m -----
function [n x l]=limite(f,epsilon)
n=1;xn=f(1);dif=inf;x=[];
while dif>epsilon
    x=[x xn];n=n+1;xn1=f(n);dif=abs(xn1-xn);xn=xn1;
end
l=xn1;
end
```