

Exame de Informática, 1º Matemáticas, Febreiro, 2012. Grupo E1

NOTA: Debes acadar alomenos 1 punto en cada apartado.

Apartado de Maple

1. (0.5 PUNTOS) Dada $f(x) = \frac{1}{1+x}$, escribe os comandos que calculan $\lim_{x \rightarrow -1^+} f(x)$ e tamén $f'(1)$.

SOLUCIÓN:

```
f := x -> 1/(1 + x): limit(f(x),x=-1,right); D(f)[1](1) # con funcions
f := 1/(1 + x): limit(f,x=-1,right); subs(x=1,diff(f,x)) # con expresions
```

2. (0.5 PUNTOS) Escribe o comando que calcula a suma $\sum_{n=1}^{\infty} \frac{1}{1+k^2n^2}$.

SOLUCIÓN:

```
sum(1/(1 + k^2 n^2),n=1..infinity) => (pi*cotanh(pi(k) - k)/2k
```

3. (0.5 PUNTOS) Escribe o comando que resolva o sistema formado polas ecuacións $x + \operatorname{sen}(y) = 0$ e $2x = \operatorname{cos}(y)$.

SOLUCIÓN:

```
solve({2*x = cos(y), x+sin(y) = 0}, {x, y}) => x=sqrt(5)/5, y=-arctan(1/2)
```

4. (0.5 PUNTOS) Representa gráficamente o lugar xeométrico dos puntos dados pola ecuación $x^2e^{-y^2} - y^2\operatorname{sen}(xy) = 1$ no intervalo $x, y \in [-10, 10]$

SOLUCIÓN:

```
implicitplot(x^2*exp(-y^2)-y^2*sin(x*y) = 1, x = -10 .. 10, y = -10 .. 10)
```

Apartado de Fortran

(4 PUNTOS) Escribe un programa en Fortran que lea un número enteiro n por teclado e reserve memoria dinamicamente para un vector \mathbf{v} e unha matriz cadrada \mathbf{a} , ambos de orde n . O programa debe darlle valores aos elementos v_i do vector de modo que $v_i = ie^{i-2}$, $i = 1, \dots, n$. Logo, o programa debe chamar ao subprograma `calcula_matriz(...)` (debes decidir o seu tipo e argumentos), que lle dea valores aos elementos a_{ij} da matriz a partir dos elementos v_i do vector, de modo que a_{ij} sexa:

$$a_{ij} = \begin{cases} v_{i+j} & i+j < n \\ v_i v_j & i+j = n \\ v_i - v_j & i+j > n \end{cases}$$

Finalmente, dende o programa principal débese crear o arquivo `saida.dat` no cal almacenar o vector \mathbf{v} (nunha única liña) e a matriz \mathbf{a} (unha fila en cada liña da terminal).

NOTA: Proba con $n = 5$. Debes obter o vector $\mathbf{v} = (0.36787945, 2.0000000, 8.1548452, 29.556225, 100.42769)$ e a matriz \mathbf{a} seguinte:

2.0000000	8.1548452	29.556225	10.873128	-100.05981
8.1548452	29.556225	16.309690	-27.556225	-98.427689
29.556225	16.309690	0.0000000	-21.401379	-92.272842
10.873128	27.556225	21.401379	0.0000000	-70.871460
100.05981	98.427689	92.272842	70.871460	0.0000000

SOLUCIÓN:

```

program xaneiro1
real, dimension(:), allocatable::v
real, dimension(:,:), allocatable::a
print *, "Introduce n: "
read *, n
allocate(v(n),a(n,n))
do i=1,n
  v(i)=i*exp(i-2.0)
end do
call calcula_matriz(a, v, n)
open(1, file="saida.dat", status="new", err=3)
write(1, *) "Vector v: "
write(1,*) (v(i), i=1,n)
write(1,*) "Matriz a: "
do i=1,n
  write(1,*) (a(i,j), j=1,n)
end do
close(1)
deallocate(v, a)
stop
3 print *, "Erro abrindo o arquivo saida.dat"
end program xaneiro1

```

```

! Definicion de calcula_matriz
subroutine calcula_matriz(b, x, m)
real, dimension(m), intent(in)::x
real, dimension(m,m), intent(out)::b
integer, intent(in)::m
do i=1,m
  do j=1,m
    k=i+j
    if(k<m) then
      b(i,j)=x(k)
    else if (k>m) then
      b(i,j)=x(i)-x(j)
    else
      b(i,j)=x(i)*x(j)
    end if
  end do
end do
return
end subroutine calcula_matriz

```

Apartado de Matlab

1. (0.5 PUNTOS) Escribe os comandos que permiten integrar a función $f(x) = e^{-x^2/2} \tanh(x/2)$ no intervalo $[0, 1]$, e calcular o seu polinomio de Taylor de orde 7 no punto $x = 0$.

SOLUCIÓN:

```

quad('exp(-x.^2/2).*tanh(x/2)',0,1)
syms x; taylor(exp(-x^2/2)*tanh(x/2),7,0)

```

2. (0.5 PUNTOS) Representa gráficamente o lugar xeométrico dos puntos dados polas ecuacións $x(t) = e^{-t} \text{sen}(10t)$, $y(t) = t^2$, $z(t) = 1/t$, $t = 1, \dots, 100$.

SOLUCIÓN:

```

ezplot3('exp(-t)*sen(10*t)', 't^2', '1/t', [1 100])

```

3. (3 PUNTOS) Crea un arquivo de texto chamado `entrada.dat` co seguinte contido:

```
1 2 3
4 5 6
7 8 9
```

Escribe un programa que lea este arquivo e almacene o seu contido na matriz **a**. Logo debe chamar á función de Matlab `calcula_vector(...)` (debes decidir os seus argumentos), que retorne o vector **v**, de orde n^2 , con elementos dados por $v_k = \max(a_{ij}, a_{ji}), k = 1, \dots, n^2$, sendo $i = \lfloor \sqrt{k} \rfloor$ (parte enteira por defecto) e $j = \lceil \frac{k}{n} \rceil$ (parte enteira por exceso). Finalmente, o programa principal debe engadir ao final do arquivo `entrada.dat` o vector **v** (tódalas compoñentes na mesma fila do arquivo).

NOTA: Debes obter $\mathbf{v} = (1, 1, 1, 5, 5, 5, 8, 8, 9)$

SOLUCIÓN:

```
clear all
a=load('entrada.dat')
v=calcula_vector(a)
fid=fopen('entrada.dat', 'a');
if fid < 0
    fprintf('Erro abrindo entrada.dat');
    break;
end
fprintf(fid, '\n Vector v: \n');
fprintf(fid, '%d ', v);
fclose(fid);

% arquivo calcula_vector.m
function v=calcula_vector(a)
    [fil col]=size(a);
    n=fil*fil;
    v=zeros(1, n);
    for k=1:n
        i=floor(sqrt(k));
        j=ceil(k/fil);
        v(k)=max(a(i,j), a(j,i));
    end
end
```