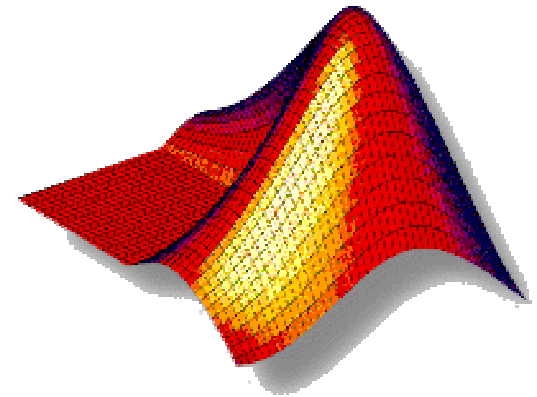


MatLab

Ficha Teórico-Prática #1



**Jorge Henriques,
Paulo Gil,
Rui Pedro Paiva**

MatLab	1
1. Operações Simples	3
1.1 Comandos Simples	3
1.2 Interpretação de alguns resultados	4
2. Manipulação de Vectores	7
3. Operações com Matrizes	14
4. Programação com Matlab	17

1. Operações Simples

1.1 Comandos Simples

- » `x=3+5`
- » `4-10`
- » `4-10;`
- » `x=3+5;`
- » `x=5*9-3`
- » `x=3/4+6-10`
- » `x=log(8)`
- » `x=exp(log(8))`
- » `x=log(exp(8))`
- » `x=cos(pi/4)`
- » `x=log(sqrt(4))`
- » `x=sqrt(8)`
- » `x=8^(1/2)`
- » `x=5^2`

```
» sin(pi)
» eps
» realmin
» realmax
» realmax*1.1
» realmax+10
» eps
» format long
```

1.2 Interpretação de alguns resultados

```
» a = 2500/20
» a = 2500/20;
» b = [1 2 3 4 5 6 7 8 9]
» c = [1 2 3 ; 4 5 6 ; 7 8 9]
» c = [c ; [10 11 12]]
» c(2,2) = 0
» l = length(b)
» [m,n] = size(b)
» [m,n] = size(c)
```

```
» who
» whos
» clear
» who
» b = 1 + 2 + 3 + 4 + ...
» 5 + 6 - 7
» x = 1 : 2 : 9
» x = (0.8 : 0.2 : 1.4);
» y = sin(x)
» dir
» a = 2^3
» a = 4/3
» format long
» a = 4/3
» format short
» clear
» a=[1 2 3 ; 4 5 6 ; 7 8 9];
» b = a'
» c = a + b
» c = a - b
» a(l,:) = [-1 -2 -3]
```

```
» c = a(:,2)
» c = a(2:3, 2:3)
» x = [- 1 0 2];
» y = [-2 -1 1]';
» x*y
» c = x + 2
» a = [1 0 2; 0 3 4 ; 5 6 0];
» size(a)
» b = inv(a);
» c = b*a
» c = b/a
» c = b\a
» clear a b x y
» whos
```

2. Manipulação de Vectores

%----- inicialização de vectores

```
» x= []
```

```
» x=[ 0  1  -1  3  4] ;
```

```
» x=[ 0  1  -1  3  4]
```

```
» x
```

```
» z
```

```
» z=x*3 ;
```

```
» z
```

```
» z=x.*y      %.. produto componente a componente.
```

```
» sum(z)      %.. soma das componentes do vector
```

```
» z=x./y ;
```

```
» z
```

```
» z=x*y ;      % erro ! devido à falta do "."
```

```
» x=[x y] ;
```

%----- criação de um vector a partir de outros já existentes.

» **x**

» **x=x.*y;** % erro ! produto de dois conjuntos de dimensões diferentes.

» **x=[x x x];**

» **x**

» **pi**

» **cos(pi/4)**

» **x=x(5:8);** % cria um vector com parte de outro.

» **x=[ones(1,4), [2:2:11], zeros(1,3)];** % cria um vector à custa de outros

» **x=[ones(1,4), 2:2:11, zeros(1,3)];**

» **x(3:7)**

» % não faz nada; serve para pôr comentários

» **r=rand(size(1:10))**

% Os números complexos são fáceis de tratar. Faça, sucessivamente:

```
» z = 2 + 2j
» conj(z)
» abs(z)
» angle(z)
» real(z)
» imag(z)
» exp(i*pi)
» exp(j*[ pi/4 -pi/4 ])
» exp(j*[ pi/4:0.1:-pi/4 ])
» z^r
```

Exercício 1

Considere o polinómio $x^4 + 0.47x^3 - 3.2x^2 + 12$

- a) Use a função **roots** para determinar as suas raízes.
- b) A partir das raízes, reconstrua o polinómio usando a função **poly**.
- c) Represente graficamente as raízes usando a função **polar**.
- d) Considere outro polinómio $2x^3 + 0.7x^2 + -5x + 3.4$. Repita as questões anteriores.
- e) Multiplique os dois polinómios. Para isso, use a função **conv**.
- f) Use a função **roots** para confirmar o resultado da alínea anterior.

%----- Gravação De Dados

» **help save**

» **help load**

» **a = [1 2 3 4 5 6 7 8];**

» **b = a*2;**

» **c = a - 1;**

» **save a b c**

» **dir**

» **clear**

» **whos**

» **load a b c**

» **whos**

%----- Recursos Gráficos 1

```
» y = [0 2 5 4 1 0];  
» plot(y)  
» t = 0:.4:4*pi  
» y = sin(t)  
» z = cos(t);  
» plot(t, y, \'.', t, z "-.")  
» title('Funções')  
» xlabel("t")  
» ylabel("Seno e Co-seno")  
» text(3, 0.5, 'Seno')
```

% Após o próximo comando, seleccione a posição que deseja colocar o texto 'Co-seno' com rato

```
» gtext('Co-seno')
```

Exercício 2

%----- Ajuste De Curvas De Dados Experimentais

- a) Faça a geração de 20 valores entre -1 e 1, usando a função **linspace**.
- b) Com aqueles valores, construa os valores da função $x=t^2$.
- c) Adicione ao sinal x , ruído uniforme no intervalo -0.1 a 0.1, i.e., $xr = x + r$ (utilize a função 'rand' para gerar o sinal de ruído).
- d) Desenhe o sinal xr .
- e) Use a função **polyfit** para ajustar um polinómio aos dados. Calcule os valores do polinómio nos pontos considerados, usando a função **polyval**.
- f) Com a função **ginput**, obtenha as coordenadas de 3 pontos seleccionados no gráfico com o rato

3. Operações com Matrizes

```
» v=1:4;  
» A = [1 2 3; 3 -5 6; 6 7 9]  
» eye(3)  
» zero(3)  
» zero(3,3)  
» zeros(3,3)  
» B = A'  
» A*B  
» A*B  
» A.*B  
» A+B  
» A(2,3)  
» A  
» A(1,:)   
» A([1,2],:)  
» A([3 2],[2 1])  
» A
```

- » $A \cdot v$
- » A
- » v
- » `size(A)`
- » `length(A)`
- » `sin(A)`
- » `sqrtm(A)`
- » `M=sqrtm(A)`
- » $M \cdot M$
- » `real(M*M)`
- » A
- » A^2
- » $A \cdot A$
- » $A.^2$
- » `inv(A)`
- » `det(A)`
- » `rank(A)`
- » `eig(A)`
- » `[V,D]=eig(A)`
- » `eig(A)`

```
» a = eye(4,5)
» r = rand(size(a))
» b = [2 0 0;0 3 0;0 0 -1];
» d = det(b)
» bi=inv(b)
» l = eig(b)
» c=[b b]
» d=[b;b]
» ci=pinv(c)
» di=pinv(d)
» c*ci
» ci*c
» di*d
» d*di
```


4. Programação com Matlab

% Abra um ficheiro a partir do Matlab (File, New, M-File)

% Digite os seguintes comandos e grave o ficheiro com o nome teste1.m no seu directório de trabalho.

```
n = 3 ;  
m = 3;  
for i = 1: m  
    for j= 1 : n  
        a(i, j) = i + j;  
    end;  
end  
disp('Matriz A')  
disp(a)
```

%final do programa teste1.m

Acabou de criar um "script" que pode chamar sempre que queira mesmo a partir de outros programas.

% Funções

% Abra outro ficheiro, guardando-o com nome de teste2.m

% Digite os seguintes comandos neste ficheiro

```
v = 1:1:10;  
m = media(v) ;  
s = sprintf('\n A média é: %4.2f' , m) ;  
disp(s) ;
```

% final do programa teste2.m

Agora crie o seguinte ficheiro, com o nome de media.m

```
function x = media(u)
% function x = media(u) calcula a média do vector u, colocando o resultado em x
x = sum(u)/length(u);
% final da função media.m
```

*Nota existe uma função que faz isto: a função **mean***

*Para calcular a média de um dado vector **b**, basta fazer **bm=media(b)** ou **bm=mean(b)**.*

%----- Recursos Gráficos 1

% Programa exemplo de gráfico 3d

% abra outro ficheiro, guardando-o com nome de teste3.m e digite os seguintes comandos neste ficheiro

```
clear
n = 30;
m = 30;
for i = 1:m
    for j = 1:n
```

```
        a(i,j) = sqrt(i+j);  
    end  
end  
b = [a+0.5 a'-0.5;  
    (a.^2)/5 ((a'-0.1).^2)/2];  
mesh(b)
```