

INTRODUÇÃO AO OCTAVE



AULA 1: PARTE 1

HELAINÉ FURTADO



INTRODUÇÃO AO OCTAVE



- O que é o OCTAVE?
- **GNU Octave** é uma linguagem computacional, desenvolvida para computação matemática;
- Possui uma interface em linha de comando para a solução de problemas numéricos, lineares e não-lineares;
- Faz parte do projeto GNU, é um software livre sob os termos da licença GPL;
- Foi escrito por John W. Eaton.
- Possui compatibilidade com MATLAB, possuindo um grande número de funções semelhantes.





INTRODUÇÃO AO OCTAVE

- A estrutura de dado básica é um **vetor**.

Desta forma, todas as variáveis são matrizes:

Exemplos:

- um **escalar** é uma **matriz 1x1**;

```
>> x = 4
```

- um **vetor** é uma matriz **1xN**.

```
>> x = [ 3 5 4 6 ]
```

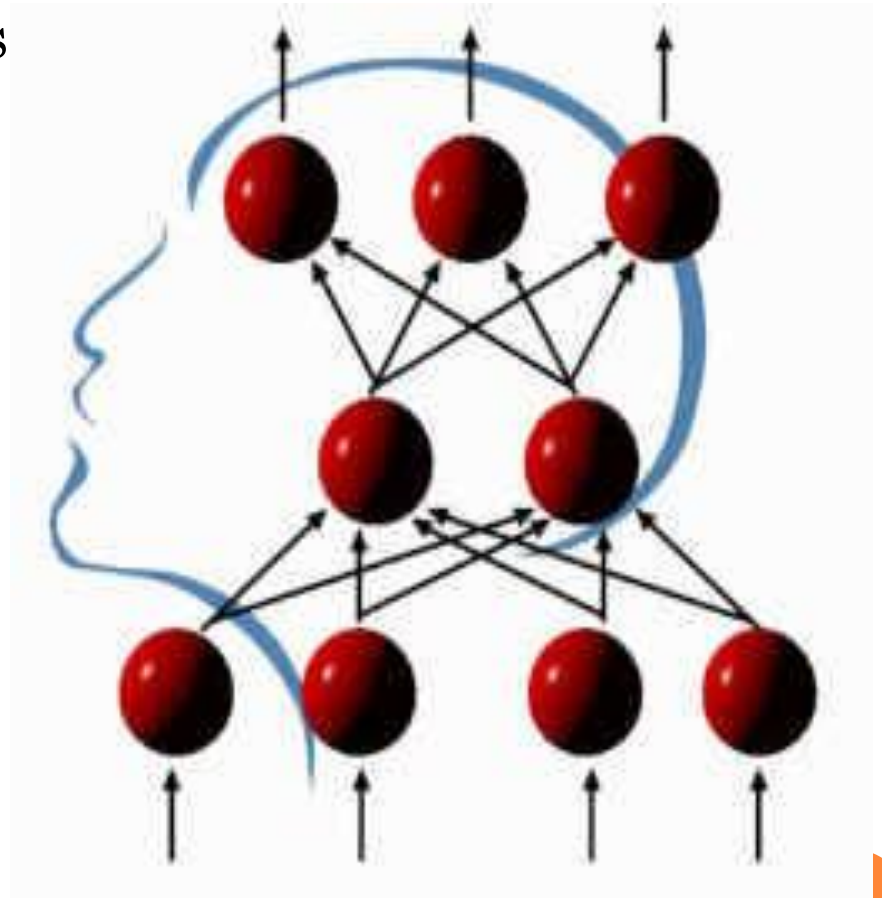
- Dois modo de utilização:

- Linha de comandos - janela de comandos;
- Linguagem de programação - janela de edição (com for, while, blocos if, etc.)



LISTA DE TOOLBOXES

- Identificação de Sistemas
- Lógica Fuzzy
- Análise Espectral
- Processamento Digital de Imagens
- Redes Neurais
- Otimização
- Equações Diferenciais Ordinárias e Parciais
- Processamento de Sinais
- Estatística
- Wavelets
- Matemática Simbólica, etc.



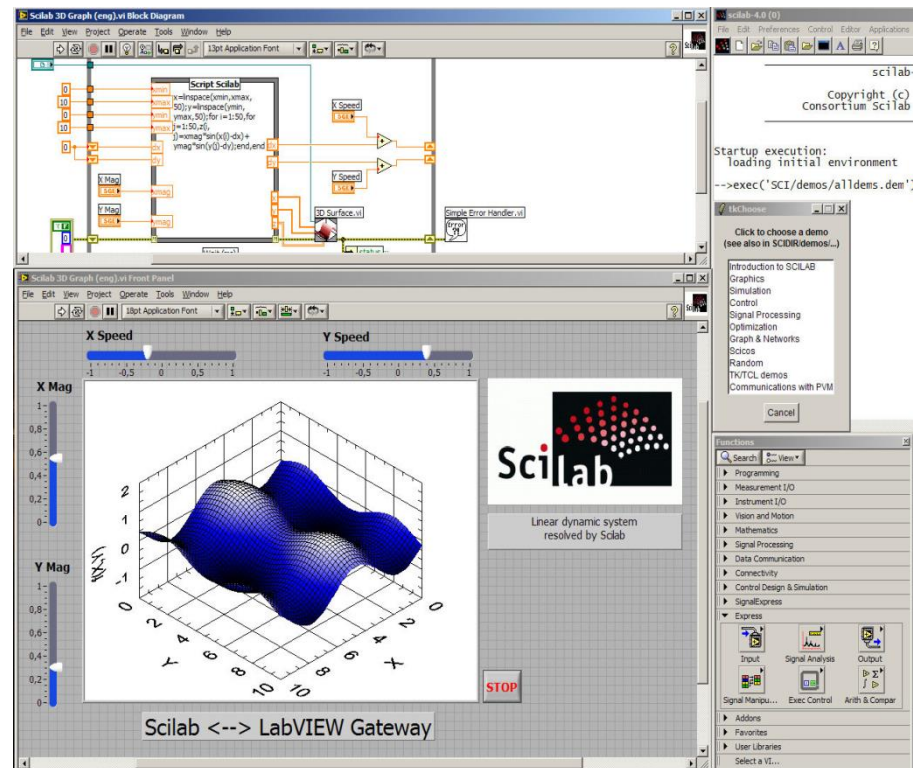


INTRODUÇÃO AO OCTAVE

o SciLab

- Software livre;
- Possui distribuições para Windows e Linux.

<http://www.scilab.org/>





INTRODUÇÃO AO OCTAVE

◦ Vantagens:

- O tempo de implementação é mais rápido se comparada com linguagens C e Fortran, por exemplo.

◦ Desvantagens:

- Elevado esforço computacional - por ser uma linguagem de programação interpretada;



O AMBIENTE DO OCTAVE



Octave

Arquivo Editar Depurar Janela Ajuda Novidades

Diretório Atual: C:\Users\Helaine Furtado

Navegador de Arquivos

C:/Users/Helaine Furtado

Nome

- > .conda
- > .config
- > .ipython
- > .matplotlib
- > .spyder-py3
- > .VirtualBox

Ambiente de Trabalho

Filtrar

Nome	Classe	Dimensão	Valor	Atributo
------	--------	----------	-------	----------

Histórico de Comandos

Filtrar

```
sqrt
sqrt(4)
sqrt(74)
sqrt(7)
sqrt(7)*5
(sqrt(7)*5)/14
clc
# Octave 5.2.0, Mon Sep 14 12:46:00 2020 GMT <unknown@DESKTOP-203F9CQ>
```

Janela de Comandos

GNU Octave, version 5.2.0

Copyright (C) 2020 John W. Eaton and others.

This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "x86_64-w64-mingw32".

Additional information about Octave is available at <https://www.octave.org>.

Please contribute if you find this software useful.

For more information, visit <https://www.octave.org/get-involved.html>

Read <https://www.octave.org/bugs.html> to learn how to submit bug reports.
For information about changes from previous versions, type 'news'.

>>

Janela de Comandos

Documentação

Editor

Editor de Variáveis



25°C Chuva 13:58

O AMBIENTE DO OCTAVE



Current directory: Exibe o conteúdo do diretório de trabalho

GNU Octave, version 5.2.0
Copyright (C) 2020 John W. Eaton and others.
This is free software; see the source code for copying conditions.
There is ABSOLUTELY NO WARRANTY; not even for MERCHANTABILITY or
FITNESS FOR A PARTICULAR PURPOSE. For details, type 'warranty'.

Octave was configured for "x86_64-w64-mingw32".

Additional information about Octave is available at <https://www.octave.org>.

sqrt
sqrt(4)
sqrt(74)
sqrt(7)
sqrt(7)*5
(sqrt(7)*5)/14
clc
Octave 5.2.0, Mon Sep 14 12:46:00 2020 GMT <unknown@DESKTOP-203F9CQ>



O AMBIENTE DO OCTAVE NA NUVEM






Caixa de entrada (9) - helaine x | (9) WhatsApp x | SIGAA - Sistema Integrado d... x | Octave Online - Cloud IDE co... x | GNU Octave - Wikipédia, a e... x

octave-online.net

Google | Entrada | CALLAN METHOD... | Livemocha: Learn It... | Freeletics - function... | Técnica Acadêmica... | Bruce Springsteen -... | Data assimilation c... | Outros marcadores

OctaveOnline

MENU

Files    my_script.m

Drop Files Here to Upload

Tips & Tricks

The files you make on Octave Online will be saved for the next time you visit. They will be deleted after 6 months of inactivity.

Keyboard Shortcuts

Common shortcuts:

- `Ctrl + Space`
→ Show the auto-completion menu
- `Cmd/Ctrl + S`
→ Save the file
- `Cmd/Ctrl/Win + R`
→ Run the file
- `Cmd/Ctrl/Win + E`
→ Set focus to the prompt

[Full List](#)

Vars

ans

NOTICE: Due to inactivity, your session will expire in five minutes.
Octave Exited. Message: Session Timeout
[Click Here to Reconnect](#)

Ads by Google

Stop seeing this ad

Why this ad?

25°C Chuva 14:00



JANELAS DE COMANDOS OU EDIÇÃO:

- As variáveis são criadas e redimensionadas em *run-time* :
 - não é necessário declarar variáveis.
- Declaração de uma variável no OCTAVE usa-se o operador atribuição “=”.

```
>> variavel = valor;
```

Tipos de valores podem ser numérico (real ou inteiro), complexo ou caractere.

- O OCTAVE é *case-sensitive*, ou seja, comandos e variáveis escritos em letras minúsculas são **diferentes** dos escritos em letras maiúsculas!





INTRODUÇÃO AO OCTAVE

Regras para nomes	Comentário e exemplos
<i>Case sensitive</i>	Custo, CustO, CuStO e CUSTO são variáveis diferentes no OCTAVE
Devem conter no máximo 31 caracteres	CustoTotalAnoAnterior = 10 CustoTotalEstimadoProximoAno = 4
Não deve começar com número e nem conter pontuações ou espaços	Valor_Estimado = 3 Valor1 = 9.2 X123 = 2.5





DECLARAÇÃO DE VARIÁVEIS

○ Exemplos:

- matriz 1x1;

```
>> x = 4
```

- um vetor é uma matriz 1xN;

```
>> x = [ 3 5 4 6 ]
```

- Matrizes (MxN);

```
>> x = [ 3 5 4 6; 2 3 4 5; 4 5 6 7 ]
```

- Os elementos devem ser declarados entre [];
- Os elementos de uma mesma linha numa matriz são separados por espaço(s) ou vírgula;
- ponto-e-vírgula(;) indica o final de uma linha de uma matriz ou final de uma expressão;





OPERADORES ARITMÉTICOS MATRICIAIS

Operação	Símbolo
Adição	+
Subtração	-
Multiplicação	*
Divisão	/
Potenciação	^
Matriz transposta	'





OPERADORES ARITMÉTICOS MATRICIAIS

○ Operações simples entre escalares

```
>> x = 4
```

```
x = 4
```

```
>> y = 3
```

```
y = 3
```

```
>> z = x + y
```

```
z = 7
```

```
>> w = x - y
```

```
w = 1
```

```
>> v = y / x
```

```
v = 0.7500
```

```
>> u = x^2 + y^2
```

```
u = 25
```





OPERADORES ARITMÉTICOS MATRICIAIS

```
>> A = [1 2; 3 4]
```

```
A =
```

```
1    2
3    4
```

```
>> B = [5 6; 7 8]
```

```
B =
```

```
5    6
7    8
```

```
>> C = A + B
```

```
C =
```

```
6    8
10   12
```

```
>> C1 = A - B
```

```
C1 =
```

```
-4   -4
-4   -4
```

```
>> C2 = A^(-1)  %=(inv(A))
```

```
C2 =
```

```
-2.0000    1.0000
 1.5000   -0.5000
```

```
>> C3 = A/B  %=(A*inv(B))
```

```
C3 =
```

```
3.0000   -2.0000
 2.0000   -1.0000
```

```
>> C4 = A^2 + B^2
```

```
C4 =
```

```
74    88
106   128
```

```
>> C5 = A'
```

```
C5 =
```

```
1    3
2    4
```





FUNÇÕES MATEMÁTICAS ELEMENTARES

- **abs(x)** - valor absoluto de x.
- **sqrt(x)** - raiz quadrada de x.
- **round(x)** - converte x para o valor inteiro mais próximo.
- **floor(x)** - maior valor inteiro, que é menor ou igual a x.
- **ceil(x)** - menor valor inteiro, que é maior ou igual a x.
- **sign(x)** - retorna -1 se $x < 0$ ou 1 se $x \geq 0$.
- **rem(x,y)** - resto da divisão de x por y.
- **exp(x)** - exponencial de x.
- **log(x)** - logaritmo natural de x na base e.
- **log10(x)** - logaritmo de x na base 10.



OPERADORES ARITMÉTICOS

PONTO-A-PONTO



- Operadores aritméticos ponto-a-ponto
 - Executa as operações entre (ou sobre) os elementos da matriz. Quando envolver duas matrizes, as suas dimensões devem ser as mesmas.

Operação	Símbolo
Multiplicação	. *
Divisão	. /
Potenciação	. ^



OPERADORES ARITMÉTICOS

PONTO-A-PONTO



```
>> A = [1 2; 3 4]
```

```
A =
```

```
1    2  
3    4
```

```
>> B = [5 6; 7 8]
```

```
B =
```

```
5    6  
7    8
```

```
>> C1 = A * B
```

```
C =
```

```
19    22  
43    50
```

```
>> C1 = A .* B
```

```
C2 =
```

```
5    12  
21    32
```

```
>> D1 = A/B
```

```
C3 =
```

```
3.0000    -2.0000  
2.0000    -1.0000
```

```
>> D2 = A./B
```

```
D1 =
```

```
0.2000    0.3333  
0.4286    0.5000
```

```
>> E1 = A^3
```

```
E1 =
```

```
37    54  
81    118
```

```
>> E2 = A.^3
```

```
E2 =
```

```
1    8  
27   64
```



OPERADORES ARITMÉTICOS

PONTO-A-PONTO



○ Exercícios

(1) Use o OCTAVE para avaliar a seguinte função:

$$f(x, y) = \frac{x^2 y^3}{(x - y)^2}$$

para $x = 4$ e $y = 5$.

Resp.: 2000

(2) A distância percorrida por uma bola em queda livre é dada pela equação:

$$x = x_0 + v_0 t + \frac{1}{2} a t^2$$

Use o OCTAVE para calcular a posição da bola no tempo $t=6s$ se $x_0=8m$, $v_0=13m/s$ e $a=-9,8m/s^2$

Resp.: -90.4m



OPERADORES ARITMÉTICOS

PONTO-A-PONTO



○ Respostas dos Exercícios:

○ (1)

```
>> x=4;
```

```
>> y=5;
```

```
>> f=x^2*y^3/(x-y)^2
```

○ (2)

```
>> t=6;
```

```
>> x0=8;
```

```
>> v0=13;
```

```
>> a=-9.8;
```

```
>> x=x0+v0*t+0.5*a*t^2
```



OPERADOR DOIS-PONTOS (:)



- É um dos operadores mais úteis do OCTAVE - cria vetores.

```
>> vetor = valor_inicial:incremento:valor_final  
>> vetor = valor_inicial:valor_final (incremento=1)
```

```
>> C = 1:2:11
```

```
C =
```

```
1    3    5    7    9   11
```

```
>> D = 0 : 0.5 : 3
```

```
D =
```

```
0    0.5    1.0    1.5    2.0    2.5    3.0
```



OPERADOR DOIS-PONTOS (:)



```
>> A = [1 : 2 : 15]
```

```
A =
```

```
1 3 5 7 9 11 13 15
```

```
>> A(1,5)
```

```
ans =
```

```
9
```

```
>> B = [1:1:3;2:2:6;5:2:9]
```

```
B =
```

```
1 2 3
```

```
2 4 6
```

```
5 7 9
```

```
>> b = [10 20 30]
```

```
b =
```

```
10 20 30
```

```
>> B = [B;b]
```

```
B =
```

```
1 2 3
```

```
2 4 6
```

```
5 7 9
```

```
10 20 30
```

```
>> B(1:3,2)
```

```
ans =
```

```
2
```

```
4
```

```
7
```

```
>> B(3,:) 
```

```
ans =
```

```
5 7 9
```

```
>> B(end,:)
```

```
ans =
```

```
10 20 30
```





UTILITÁRIOS PARA MATRIZES

```
>>a=eye(n)                % Matriz identidade de ordem n

>>a=zeros(n)              % Matriz nula de ordem n

>>a=ones(n,n)              % Matriz unitária de ordem n

>>a=rand(n)                % Matriz aleatória de ordem n

>>d=det(a)                  % Determinante da matriz a

>> [l,c] = size(a)         % Retorna o número de linhas
                           (l) e colunas (c) da matriz a
```





UTILITÁRIOS PARA MATRIZES

```
>> A = ones(3,3)
```

```
A =
```

```
1 1 1
1 1 1
1 1 1
```

```
>> B = zeros(4,3)
```

```
B =
```

```
0 0 0
0 0 0
0 0 0
0 0 0
```

```
>> I = eye(4,4)
```

```
I =
```

```
1 0 0 0
0 1 0 0
0 0 1 0
0 0 0 1
```

```
>> C = rand(3,3)
```

```
C =
```

```
0.8147 0.9134 0.2785
0.9058 0.6324 0.5469
0.1270 0.0975 0.9575
```

```
>> rand('seed',0)
```

```
>> D = rand(3,3)
```

```
D =
```

```
0.2190 0.6793 0.5194
0.0470 0.9347 0.8310
0.6789 0.3835 0.0346
```

```
>> E = diag(D)
```

```
E =
```

```
0.2190
0.9347
0.0346
```





UTILITÁRIOS PARA MATRIZES

```
>> A = ones(3,3)
```

```
A =
```

```
    1    1    1
    1    1    1
    1    1    1
```

```
>> dimA = size(A)
```

```
dimA =
```

```
    3    3
```

```
>> A1 = diag(A)
```

```
A1 =
```

```
    1
    1
    1
```

```
>> maxdimA = length(A)
```

```
maxdimA =
```

```
    3
```

```
>> B = diag(5 * A1)
```

```
B =
```

```
    5    0    0
    0    5    0
    0    0    5
```

```
>> rand('seed',10)
```

```
>> C = rand(size(A))
```

```
C =
```

```
    0.1291    0.9505    0.1597
    0.6048    0.3367    0.7808
    0.5040    0.0924    0.6925
```

```
>> C1 = triu(C)
```

```
C1 =
```

```
    0.1291    0.9505    0.1597
         0    0.3367    0.7808
         0         0    0.6925
```

```
>> C2 = tril(C)
```

```
C2 =
```

```
    0.1291         0         0
    0.6048    0.3367         0
    0.5040    0.0924    0.6925
```





UTILITÁRIOS PARA MATRIZES

○ Exercícios

(1) Sejam $A = \begin{bmatrix} 2 & 3 & 1 \\ 4 & 4 & 1 \\ 0 & 1 & 2 \end{bmatrix}$; $B = \begin{bmatrix} 1 & 2 & 0 \\ 3 & 2 & 1 \\ 0 & 0 & 6 \end{bmatrix}$; $C = \begin{bmatrix} 2 & 1 & 5 \\ 3 & 0 & 2 \end{bmatrix}$

Calcule:

- (a) $3A^3 + 2B^2$;
- (b) $A^2 * C^t$;
- (c) $5B^3 * C$. (É possível? Por que?)





UTILITÁRIOS PARA MATRIZES

Resposta:

```
>> A = [2 3 1; 4 4 1; 0 1 2]
```

```
A =
```

```
    2    3    1
    4    4    1
    0    1    2
```

```
>> B = [ 1 2 0; 3 2 1; 0 0 6]
```

```
B =
```

```
    1    2    0
    3    2    1
    0    0    6
```

```
>> C=[2 1 5 ; 3 0 2 ]
```

```
C =
```

```
    2    1    5
    3    0    2
```

(a)

```
>> 3.*A.^3+2.*B.^2
```

```
ans =
```

```
    26    89    3
   210   200    5
     0     3   96
```

(b)

```
>> A.^2*C'
```

```
ans =
```

```
    22    14
    53    50
    21     8
```

(c)

```
>> 5.*B^3.*C
```

```
??? Error using ==> times
Matrix dimensions must
agree.
```





OPERADORES LÓGICOS E RELACIONAIS

Operador lógico	Símbolo
E	&
Ou	
Não	~

1: True – Verdadeiro
0: False - Falso

Operador relacional	Símbolo
Igual	==
Diferente	~=
Maior	>
Menor	<
Maior ou igual	>=
Menor ou igual	<=



OPERADORES LÓGICOS E RELACIONAIS



```
>> 2 == 3
```

```
ans = 0
```

```
>> 2 <= 3
```

```
ans = 1
```

```
>> (2 == 3) & (2 <= 3)
```

```
ans = 0
```

```
>> ~(2 == 3) & (2 <= 3)
```

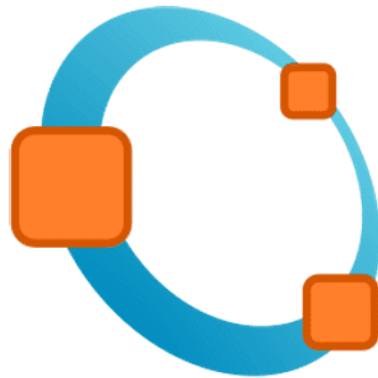
```
ans = 1
```

```
>> (2 ~= 3) & (2 <= 3)
```

```
ans = 1
```







GNU Octave

AULA 1: PARTE 2

HELAINÉ FURTADO



TRABALHANDO COM SCRIPTS

O que é um script?

- conjunto de instruções do MATLAB que podem ser executadas sistematicamente.

Para que serve?

- Sintetiza os comandos frequentemente digitados em um único comando (o script com os comandos é executado)
- Aplicável a problemas mais extensos onde é exigido mais comandos, com extensão **.m**



EDITANDO SCRIPTS



Octave

Arquivo Editar Depurar Janela Ajuda Novidades

Diretório Atual: C:\Users\Helaine Furtado

Navegador de Arquivos

C:\Users\Helaine Furtado

Nome

- > .conda
- > .config
- > .ipython
- > .matplotlib
- > .spyder-py3
- > .VirtualBox

>> edit

Ambiente de Trabalho

Filtrar

Nome	Classe	Dimensão	Valor	Atributo
------	--------	----------	-------	----------

Histórico de Comandos

Filtrar

```
sqrt(74)
sqrt(7)
sqrt(7)*5
(sqrt(7)*5)/14
clc
# Octave 5.2.0, Mon Sep 14 12:46:00 2020 GMT <unknown@DESKTOP-203F9CQ>
clc
edit
```

Editor

Arquivo Editar Visualizar Depurar Executar Ajuda

<sem_nome>

1

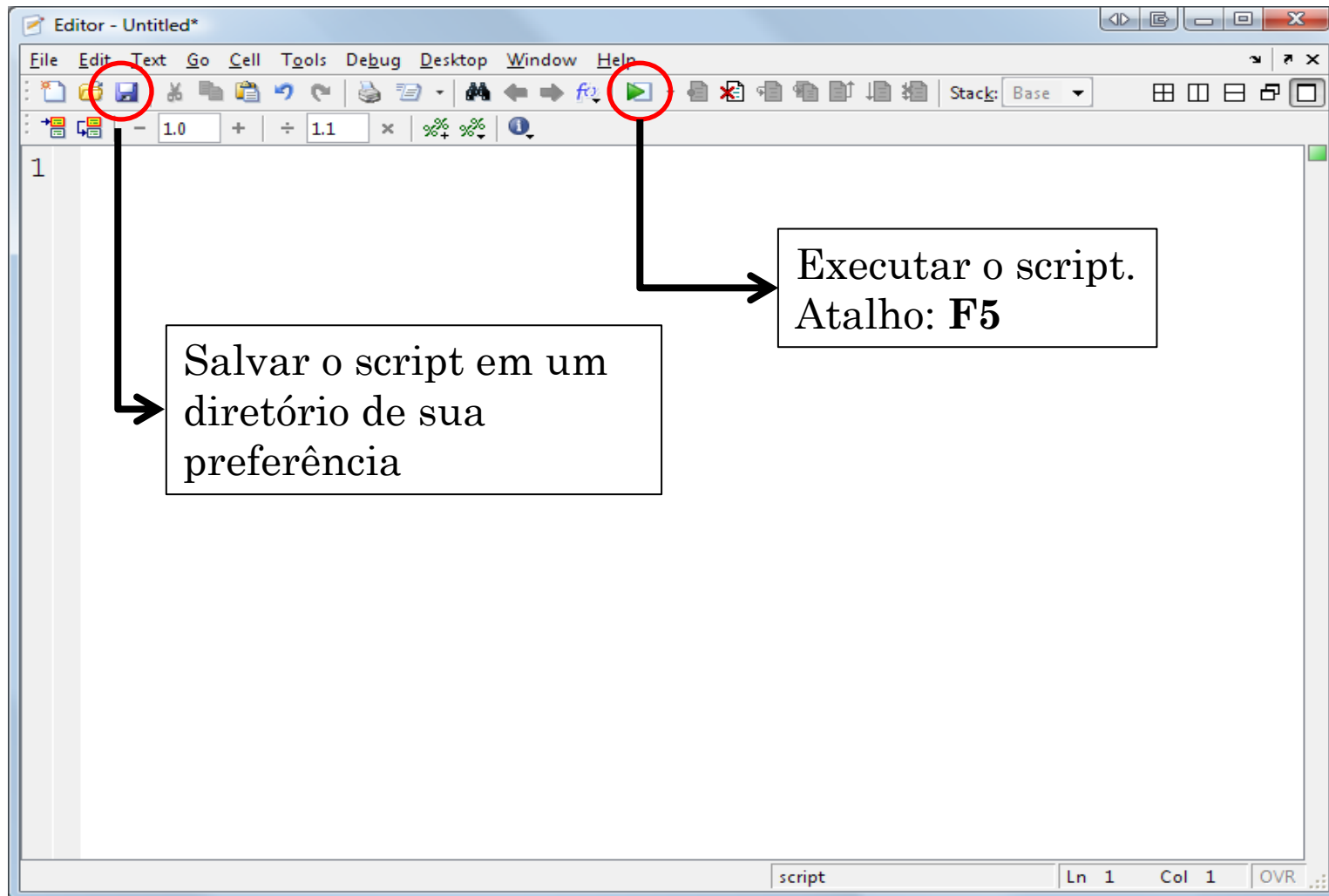
linha: 1 col: 1 codificação: SYSTEM fdl: CRLF

Janela de Comandos Documentação Editor Editor de Variáveis

Outra opção:
File > New M-File

25°C Chuva 14:05

PROGRAMANDO EM OCTAVE



COMANDOS DE ENTRADA E SAÍDA NA TELA.



- Comando de entrada: **input**

```
R = input('Forneca o valor de entrada:')
```

```
>>
```

```
Forneca o valor de entrada: <digitar valor>  
                             <enter>
```

- Comando de saída: **disp**

```
disp(X)
```

Mostra o valor de X na janela de comandos.

```
disp('Ola Mundo!')
```

```
>>
```

```
Ola Mundo!
```



COMANDOS DE ENTRADA E SAÍDA NA TELA.



- Exercício usando os comandos de entrada e saída na tela:

```
clc
A = input('Forneca uma matriz 2 x 2: ');
B = input('Forneca uma matriz 2 x 2: ');
C1 = A + B;
C2 = A * B;
disp('Soma entre as duas matrizes: ');
disp(C1)
disp('Produto entre as duas matrizes: ');
disp(C2)
```





ENTRADA E SAÍDA (ARQUIVOS)

○ Entrada e Saída (arquivos)

- Comando de entrada: **load**

Formato ASCII:

```
>> x = load('arquivo.txt');
```

Formato binário do MATLAB:

```
>> load arquivo.mat
```

- Comando de saída: **save**

Formato ASCII:

```
>> save 'arquivo.txt' variaveis -ascii
```

Formato binário do MATLAB:

```
>> save 'arquivo.mat' variaveis
```





FLUXOS DE CONTROLE

for: permite que um grupo de instruções se repitam por um número especificado de vezes.

```
for variavel = valor_inicial:passo:valor_final  
    instrucoes;  
end
```

```
>> for i=1:5  
    v(i) = i;  
    w(i) = 2*v(i);  
End
```

Obs: Para digitar múltiplas linhas antes de executá-las, digite a linha e pressione SHIFT+ENTER.





FLUXOS DE CONTROLE

○ Variantes do **loop for**:

- Incremento não-unitário:

```
>> for i=1:2:10  
    v(i) = i;  
end
```

Saída: v = 1 0 3 0 5 0 7 0 9

- Decremento:

```
>> for i =5:-1:1  
    v ( i ) = i ;  
end
```

Saída: v = 1 2 3 4 5





FLUXOS DE CONTROLE

Loop *while*: permite que uma ou mais linhas de comando sejam executadas um número indefinido de vezes, através do controle de uma condição lógica.

```
>> x = 1 ;  
>> y = 5 ;  
>> while x<3  
    y = y*x ;  
    x = x+1;  
end
```

Saída: $x = 3$; $y = 10$.





FLUXOS DE CONTROLE

Comando *if*: se uma expressão lógica é satisfeita uma seqüência de comandos é executada; caso contrário uma outra seqüência será executada.

```
>> for i = 1:3
    for j = 1:3
        if i == j
            A(i,j) = i;
        elseif i > j
            A(i,j) = j;
        else
            A(i,j) = 0;
        end
    end
end
end
```

```
>> A
```

```
A =
```

1	0	0
1	2	0
1	2	3





VISUALIZANDO FIGURAS

Comando **plot**: gera um gráfico 2D

```
>> plot(x,y)
```

x: vetor linha ou vetor coluna

y: vetor linha ou vetor coluna

Exemplo:

```
x = -pi:0.001:pi;
```

```
y = sin(x);
```

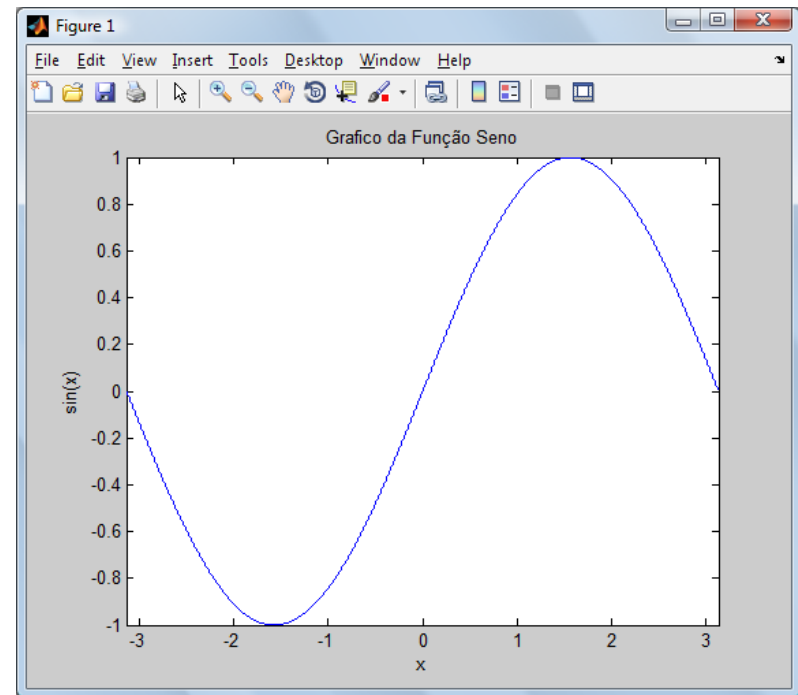
```
plot(x,y)
```

```
axis([-pi pi -1 1])
```

```
xlabel('x')
```

```
ylabel('sin(x)')
```

```
title('Gráfico da Função Seno')
```





VISUALIZANDO FIGURAS

Comando **plot3**: gera um gráfico 3D

```
>> plot3(x,y,z)
```

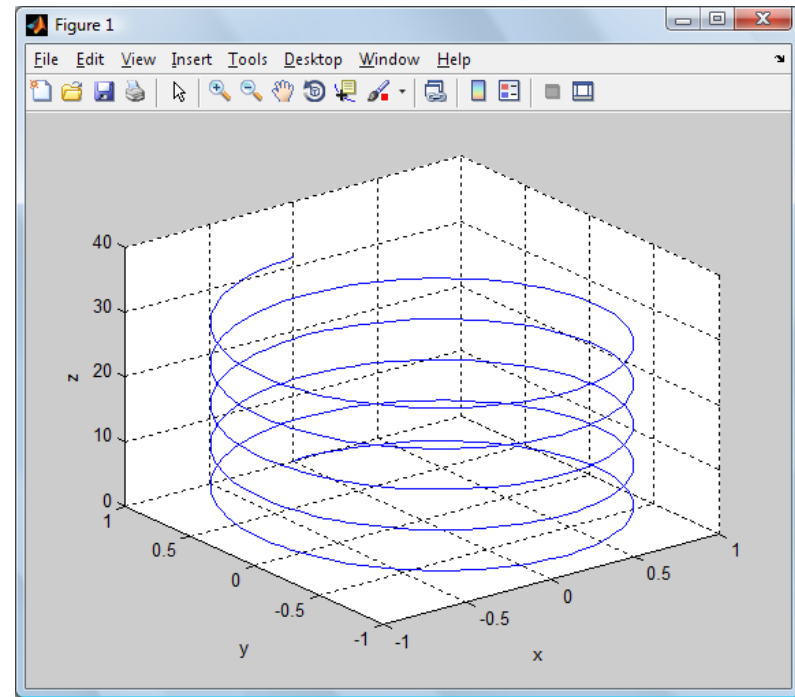
x: vetor linha ou vetor coluna

y: vetor linha ou vetor coluna

z: vetor linha ou vetor coluna

Exemplo:

```
t = 0 : pi/50 : 10*pi;  
plot3(sin(t) ,cos(t) ,t)  
axis([-1 1 -1 1 0 40])  
xlabel('x')  
ylabel('y')  
zlabel('z')  
grid on
```





VISUALIZANDO FIGURAS

Comando **print**:

```
Print(formato,nome_da_figura)
```

Formatos mais comuns:

- dpng**: salva figura em PNG
- deps**: salva em Encapsulated PostScript
- depssc**: salva Encapsulated Color PostScript

```
>> help plot
```





FUNÇÕES EM OCTAVE

- Uma função é um trecho de código que admite parâmetros de entrada e retorna valores.
- Arquivos “.m” podem ser tanto scripts quanto funções.
- Os comandos nativos do OCTAVE são funções criadas na sua própria linguagem.

```
function variavel_retorno = nome_da_funcao(variaveis_entrada)  
instrucoes para obter variavel_retorno;
```





CRIANDO UMA FUNÇÃO NO OCTAVE

- Exemplo:

Função que calcula a distância euclidiana entre dois vetores do R_n . Crie e salve o seguinte arquivo “.m” (calcula_distancia.m):

$$\sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

```
function distancia = calcula_distancia(x,y)
diferenca = x - y;
distancia = sqrt(diferenca*transpose(diferenca));
```





CRIANDO UMA FUNÇÃO NO OCTAVE

- Para executar essa função, vá a janela de comando e proceda da seguinte forma:

```
>> p = [1  0  0];  
>> q = [0  1  0];  
>> d = calcula_distancia(p, q)
```

- Ou:

```
>> d = calcula_distancia([1 0 0], [0 1 0])
```



COMO APRENDER MAIS E MAIS SOBRE OCTAVE?



- Acessando sua documentação:

Comando **doc**:

```
>> doc
```

- Acessando scripts demonstrativos:

Comando **demo**:

```
>> demo
```

- Maiores esclarecimentos sobre algum comando:

```
>> help comando
```

Exemplo:

```
>> help plot
```

```
>> doc plot
```



