



I-3

Introdução ao Matlab

Comunicações

(9 de Março de 2009)



Sumário

1. Matlab
2. Formas de Utilização
 - a) A Janela de Comandos
 - b) Scripts
 - c) Funções
3. Sintaxe, Variáveis, Matrizes e Vectores
4. Operações comuns
 - a) Definição de sinais
 - b) Desenho de gráficos
 - c) Operações sobre sinais
5. Controlo de Fluxo: if, for, while
6. Funções Úteis
7. Links



1. Matlab

- MATLAB - MATrix LABoratory

- Ambiente de execução
- Linguagem de programação
- Sistema interactivo para a execução de cálculos científicos e de engenharia
- Suportado por *software baseado em cálculo matricial*.

- Desenvolvido pela MathWorks

- <http://www.mathworks.com/>



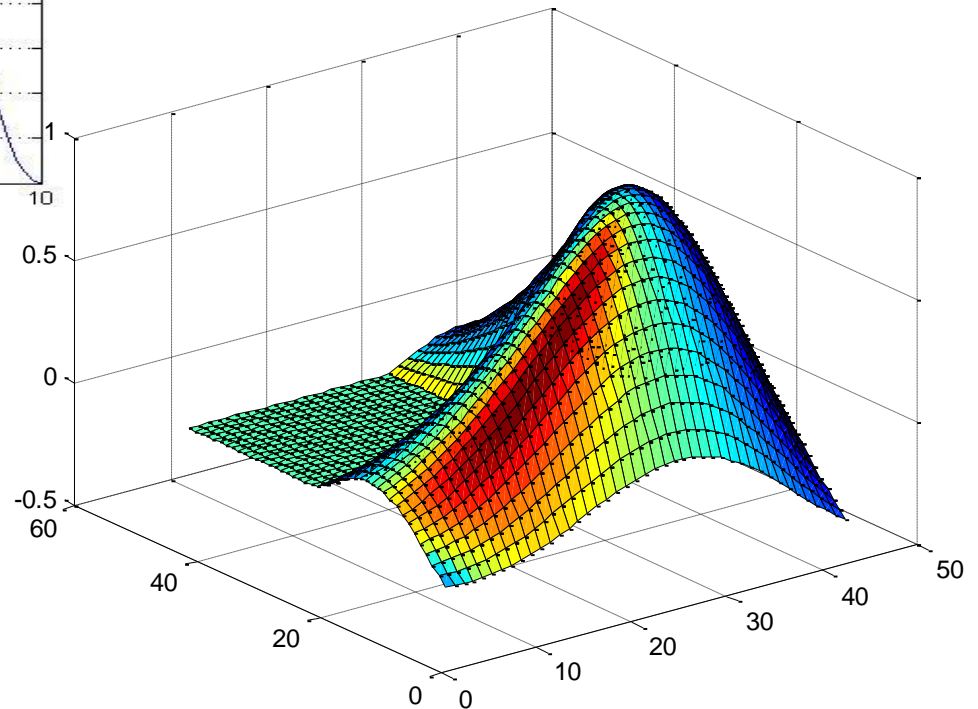
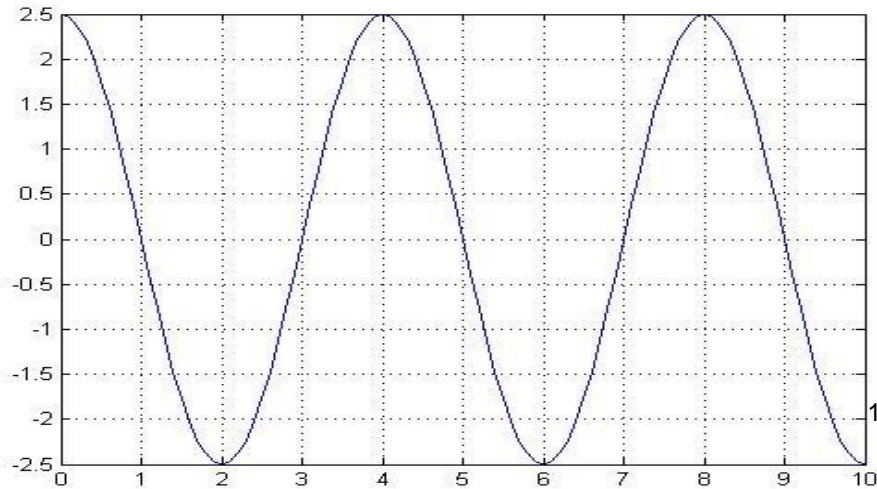
1. Matlab

- Manipulação dos dados na forma matricial
 - Vector linha $x=[10 \ 21 \ 13]$;
 - Vector coluna $y=[3 \ 5 \ 7 \ 12]'$;
 - Matriz $A=[1 \ 2 \ 3; 4 \ 5 \ 6; 7 \ 8 \ 9]$;
 - Cálculo da matriz inversa: $\text{inv}(A)$
 - Cálculo do determinante de A: $\text{det}(A)$
 - Soma de elementos:
 - $\text{sum}(x)$ e $\text{sum}(y)$, obtêm a soma de todos os elementos de x e y
 - $\text{sum}(A)$, soma das colunas de A
 - $\text{sum}(\text{sum}(A))$, soma de todos os elementos de A



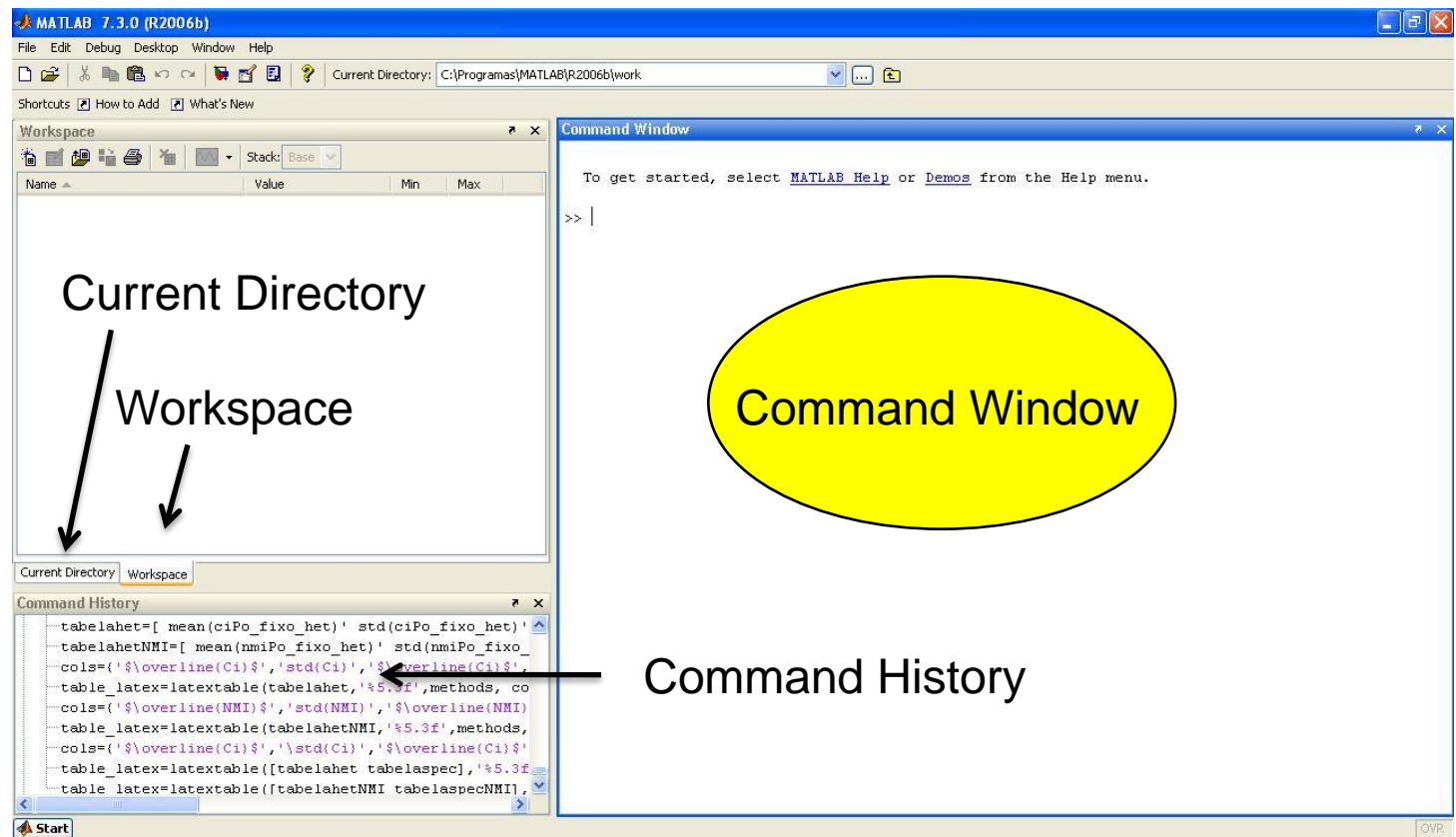
1. Matlab

Facilidade de desenho



1. Matlab

■ Ambiente de desenvolvimento



1. Matlab

■ Ambiente de desenvolvimento

The screenshot displays the MATLAB 7.6.0 (R2008a) interface. The main window is the Editor, showing a script named 'plot_periodic.m'. The script contains the following MATLAB code:

```
12 % plot_periodic.m
13 % Script para executar o desenho de um sinal periódico.
14 %
15
16 % Criar o vector de pontos.
17 t = 0 : 0.1 : 10;
18
19 % Definir a amplitude
20 A = 2.5;
21
22 % Aplicar a expressão do sinal.
23 x = A * cos( (pi/2)*t );
24
25 % Desenhar o sinal.
26 plot( t, x );
27
28 % Colocar uma grelha sobre o desenho.
29 grid on;
```

The Command Window shows the following commands and output:

```
>> cd 'D:\Docencia\Disciplinas\LEIC\Com\Bibliografia\Slides\Capítulo 1\MATLAB'
>> plot_periodic
K>>
```

The Workspace window shows the following variables:

Name	Value
A	2.5
t	0 : 0.1 : 10
x	A * cos((pi/2)*t)

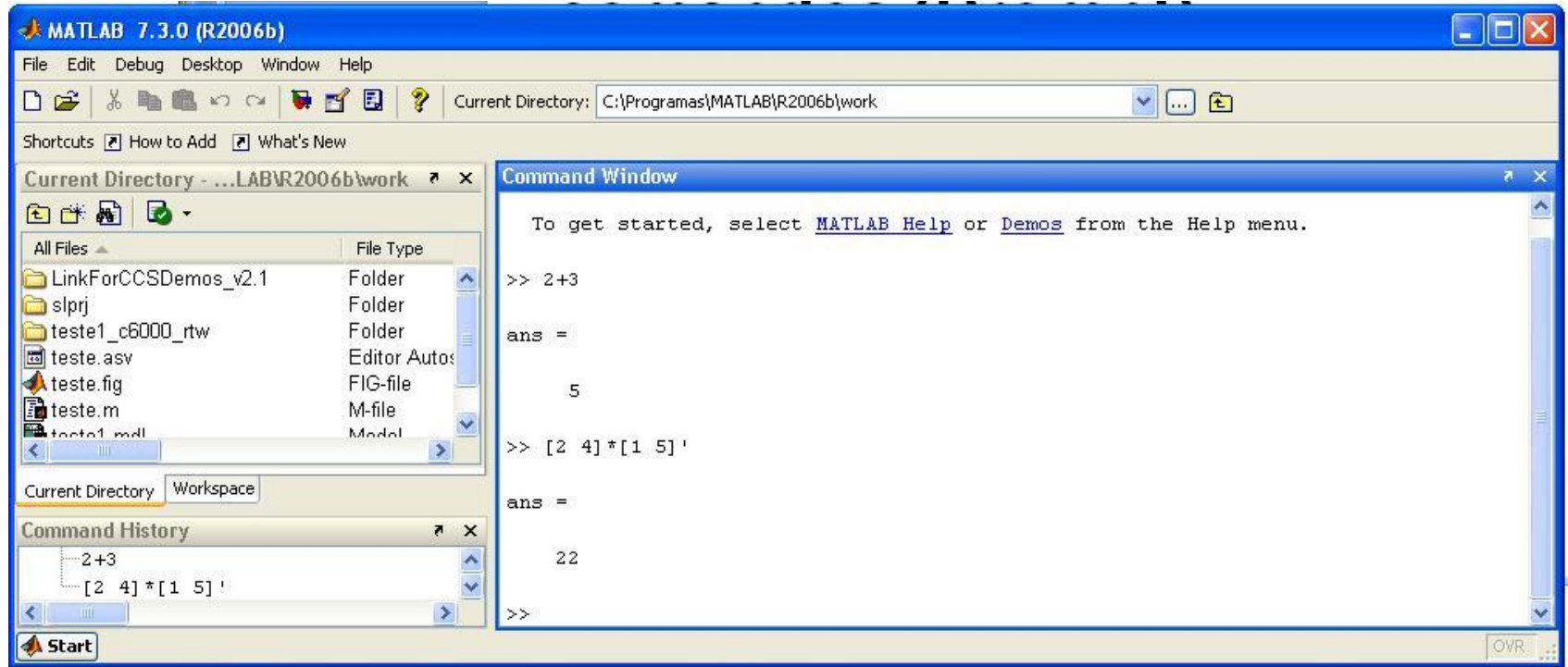
The text "Editor e Debugger" is overlaid on the right side of the Editor window.

2. Formas de Utilização

- Existem três formas distintas de executar código:
 - a) Escrita directa na janela de comandos (à semelhança da *command prompt*)
 - b) Scripts - conjunto de comandos em sequência, num ficheiro de texto
 - c) Funções - conjunto de comandos em sequência, num ficheiro de texto, com semântica de parâmetros de entrada, parâmetros de saída e *stack*



2. A Janela de Comandos



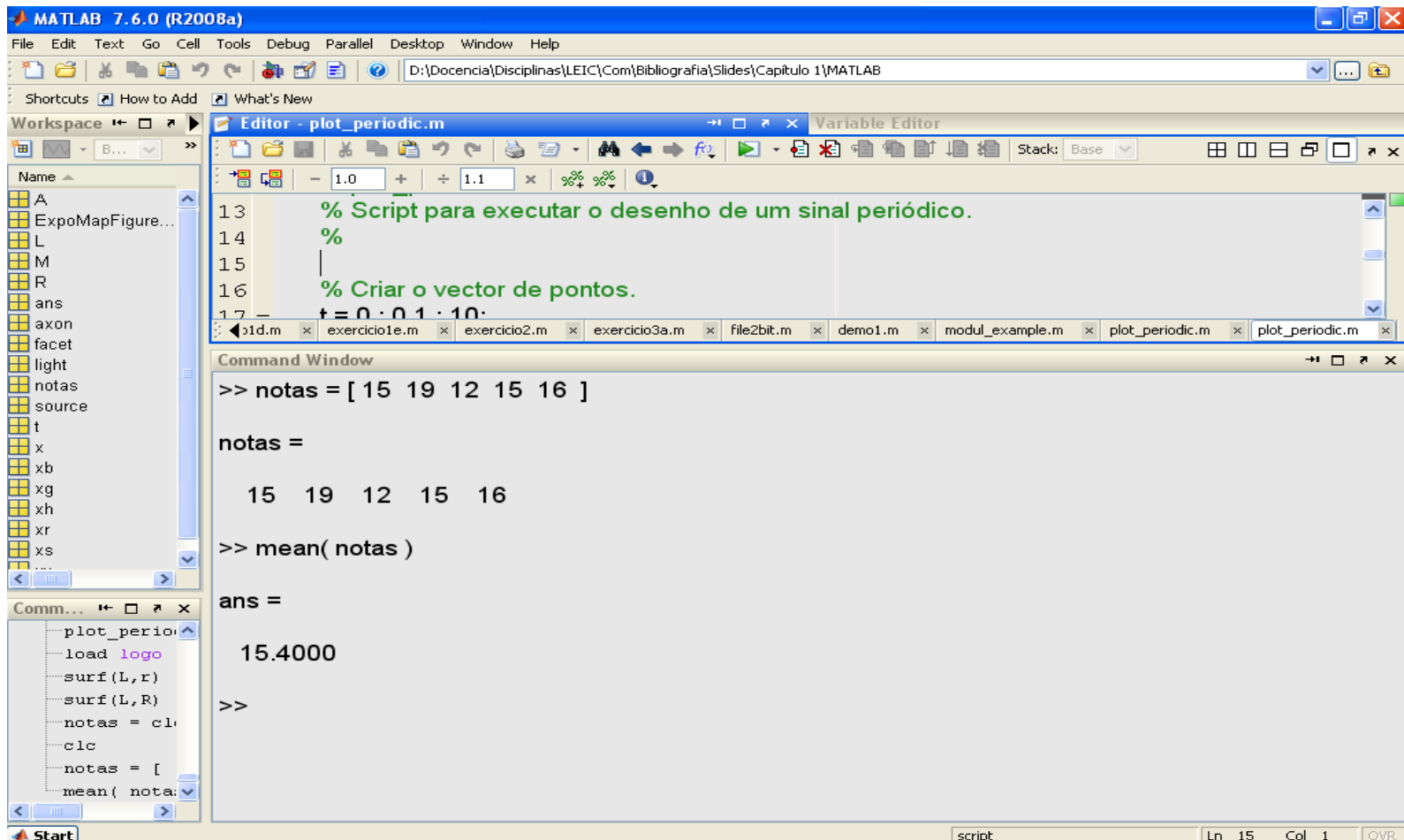
Exemplo de utilização da consola para executar operações aritméticas:

```
>> 2+3
```

```
>> plot(t,x,'r.');
```



2. A Janela de Comandos



The screenshot displays the MATLAB 7.6.0 (R2008a) environment. The Command Window shows the following commands and their outputs:

```
>> notas = [ 15 19 12 15 16 ]  
  
notas =  
  
    15    19    12    15    16  
  
>> mean( notas )  
  
ans =  
  
    15.4000  
  
>>
```

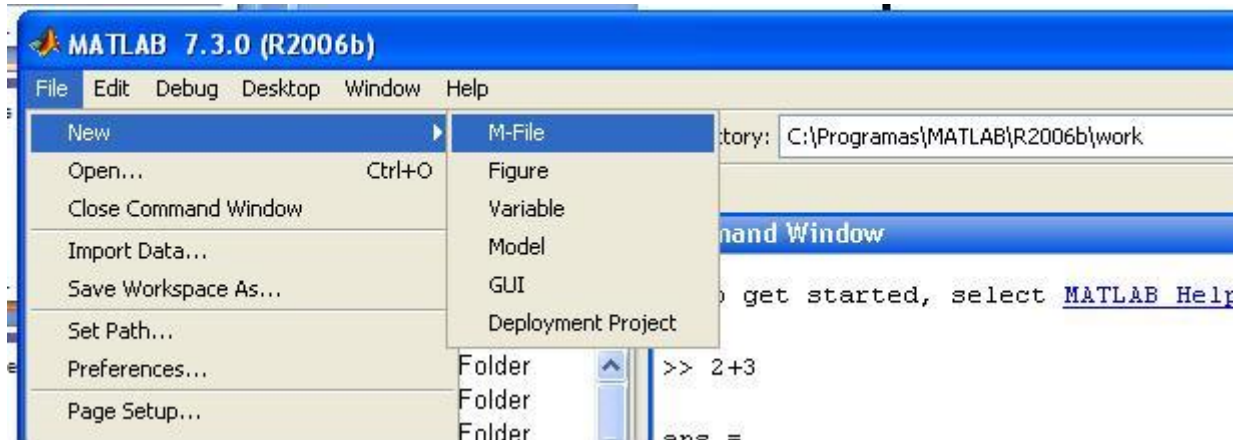
The Editor window shows the script `plot_periodic.m` with the following code:

```
13 % Script para executar o desenho de um sinal periódico.  
14 %  
15 |  
16 % Criar o vector de pontos.  
17 t = 0 : 0.1 : 10;
```

- Declaração de vector
- Cálculo da média dos elementos do vector



2. Chamada a *scripts*

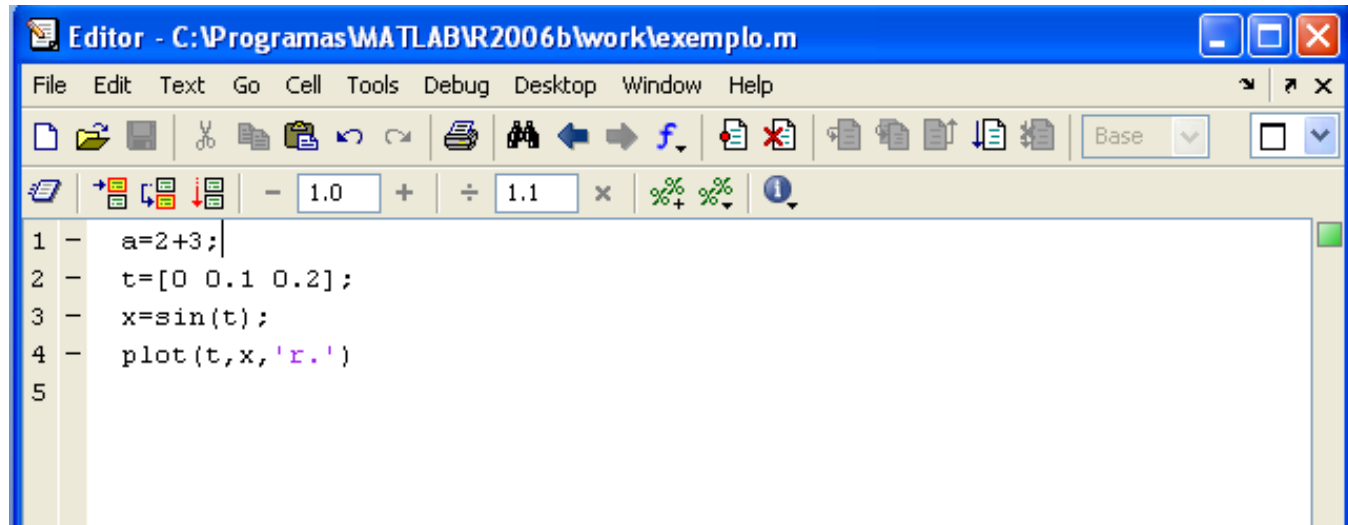


- Sempre que seja necessário executar vários comandos é conveniente usar um *script*
- Um *script* consiste num ficheiro texto de extensão .m, com comandos
- Criar um ficheiro *script* (<nomefich>.m).
- Exemplo:

```
>> edit exemplo.m
```



2. Chamada a *scripts*



```
Editor - C:\Programas\MATLAB\R2006b\work\exemplo.m
File Edit Text Go Cell Tools Debug Desktop Window Help
[Icons] Base [Dropdown]
- 1.0 + ÷ 1.1 x [Icons]
1 - a=2+3;
2 - t=[0 0.1 0.2];
3 - x=sin(t);
4 - plot(t,x,'r.')
5 -
```

- Ao executar o *script* as linhas de código serão interpretadas sucessivamente
- Equivalente a serem executadas em sequência na janela de comandos
- Deverá ser utilizado para pequenos testes
- Todas as variáveis permanecem em memória



2. Chamada a funções

The screenshot displays the MATLAB 7.6.0 (R2008a) environment. The main window is the Editor, showing a file named 'decrease_exponencial.m'. The code in the editor is as follows:

```
19 % y1, vector com as amostras da exponencial com factor de escala 'ScaleFactor'.
20 % y2, vector com as amostras da exponencial com factor de escala 2*'ScaleFactor'.
21 %
22
23 function [y1, y2] = decrease_exponencial ( NSamples, ScaleFactor )
24
25 % Construir o vector de amostras.
26 n = 0 : 0.1 : NSamples;
27
28 % Aplicar a exponencial decrescente.
29 y1 = exp( -n * ScaleFactor );
30 y2 = exp( -n * 2* ScaleFactor );
```

The Command Window at the bottom shows the function being called:

```
>> [y1, y2] = decrease_exponencial ( 100, 2 );
>>
```

The Workspace window on the left shows various variables, including 'A', 'ExpoMapFigure...', 'L', 'M', 'R', 'ans', 'axon', 'facet', 'light', 'notas', 'source', 't', 'x', 'xb', 'xg', 'xh', 'xr', and 'xs'. The Command Window also shows a list of variables: 'surf(L,r)', 'surf(L,R)', 'notas = cl...', 'clc', 'notas = [...]', 'mean(nota:', 'clc', and '[y1, y2] = ...'.

- O nome da função deve coincidir com o do ficheiro .m, no qual se insere



2. Chamada a funções

- Declaração e chamada da função stat.m

The screenshot displays the MATLAB 7.6.0 (R2008a) environment. The Editor window shows the definition of a function named `stat.m`. The function signature is `function [mean,stdev] = stat(x)`. The code includes a comment explaining that the existence of a file named `stat.m` defines a new function that calculates the mean and standard deviation of a vector. The function body consists of three lines: `n = length(x);`, `mean = sum(x)/n;`, and `stdev = sqrt(sum((x-mean).^2/n));`. The Command Window shows the execution of the function: `>> x = rand(1,1000) + 50; [m,s] = stat(x)`. The output shows the mean value `m = 50.4888` and the standard deviation value `s = 0.2831`. The Workspace window on the left lists various variables, and the Command Window history shows the sequence of commands entered.

```
function [mean,stdev] = stat(x)
%Example 1
%The existence of a file on disk called stat.m
%containing this code defines a new function
%called stat that calculates the mean and
%standard deviation of a vector:
n = length(x);
mean = sum(x)/n;
stdev = sqrt(sum((x-mean).^2/n));
```

```
>> x = rand(1,1000) + 50; [m,s] = stat(x)

m =

    50.4888

s =

    0.2831

>>
```



2. Chamada a funções

- Sintaxe de declaração

Syntax

function [out1, out2, ...] = funname(in1, in2, ...)

Description

function [out1, out2, ...] = funname(in1, in2, ...) defines function funname that accepts inputs in1, in2, etc. and returns outputs out1, out2, etc.



3. Sintaxe, Variáveis, Matrizes e Vectores

- Não existe necessidade de declarar o tipo das variáveis. Por omissão: double (64 bits)
- Declaração e afectação feita simultaneamente, através do operador =, exemplo:

```
>> 2+3  
ans =  
    5
```

```
>> a = 2 + 3  
a =  
    5
```

```
>> b=pi  
b =  
    3.1416
```



3. Sintaxe, Variáveis, Matrizes e Vectors

■ Definição de um vector:

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

```
A =
```

```
    1    2    3
```

■ Definição de uma matriz:

```
>> A=[1 2 3; 4 5 6]
```

```
A =
```

```
    1    2    3
```

```
    4    5    6
```

Nota:

Operador ; permite definir várias linhas

■ Determinação do tamanho de dada variável:

```
>> size(A)
```

```
ans =
```

```
    2    3
```

```
>> length(A)
```

```
ans =
```

```
    3
```



3. Sintaxe, Variáveis, Matrizes e Vectores

■ Operador :, sintaxe:

- valorInicial : incremento : valorFinal
- Por omissão o incremento é 1
- Exemplos

```
>> d=1:4
```

```
d =
```

```
1      2      3      4
```

```
>> d=1:2:4
```

```
d =
```

```
1      3
```



3. Sintaxe, Variáveis, Matrizes e Vectors

■ Operador ' , sintaxe:

□ Exemplos

```
>> d=1:4
```

```
d =
```

```
    1    2    3    4
```

```
>> e=d'
```

```
e =
```

```
    1  
    2  
    3  
    4
```

```
>> size(d)
```

```
ans =
```

```
    1    4
```

```
>> size(e)
```

```
ans =
```

```
    4    1
```



3. Sintaxe, Variáveis, Matrizes e Vectors

- A indexação dos vectores e matrizes começa em 1
- É realizada recorrendo aos parênteses curvos

```
>> d=1:4
```

```
d =
```

```
1      2      3      4
```

```
>> d(1)
```

```
ans =
```

```
1
```

```
>> A=[1 2 3; 4 5 6]
```

```
A =
```

```
1      2      3
```

```
4      5      6
```

```
>> A(1,3)
```

```
ans =
```

```
3
```



3. Sintaxe, Variáveis, Matrizes e Vectores

- Pode ser usada para selecção ou afectação de mais que um valor simultaneamente, usando o operador :

```
>> A=[1 2 3; 4 5 6]
```

```
A =
```

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

```
>> A(:,1)
```

```
ans =
```

```
    1
    4
```

```
>> A(2,:)
```

```
ans =
```

```
    4    5    6
```

```
>> A(2,1:2)
```

```
ans =
```

```
    4    5
```

```
>> A(2,:)=0
```

```
A =
```

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



3. Sintaxe, Variáveis, Matrizes e Vectors

■ Inserção (concatenação) e remoção de linhas/colunas

```
>> A(3,:)=[7 8 9]      >> A(:,3)=[]      >> X=[1 2]
A =                    A =                    X =
     1     2     3           1     2           1     2
     4     5     6           4     5           >> A=[A; X]
     7     8     9           7     8           A =
                                     1     2
                                     4     5
                                     1     2
```

■ Indexação Linear (*column-wise*)

```
>> A=[1 2 3; 4 5 6]
A =
     1     2     3
     4     5     6
     >> A(1)      >> A(4)
ans =            ans =
         1         5
```



3. Sintaxe, Variáveis, Matrizes e Vectors

■ Funções zeros e ones

```
>> A=zeros(3)
```

```
A =
```

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

```
>> ones(1,3)
```

```
ans =
```

```
    1    1    1
```

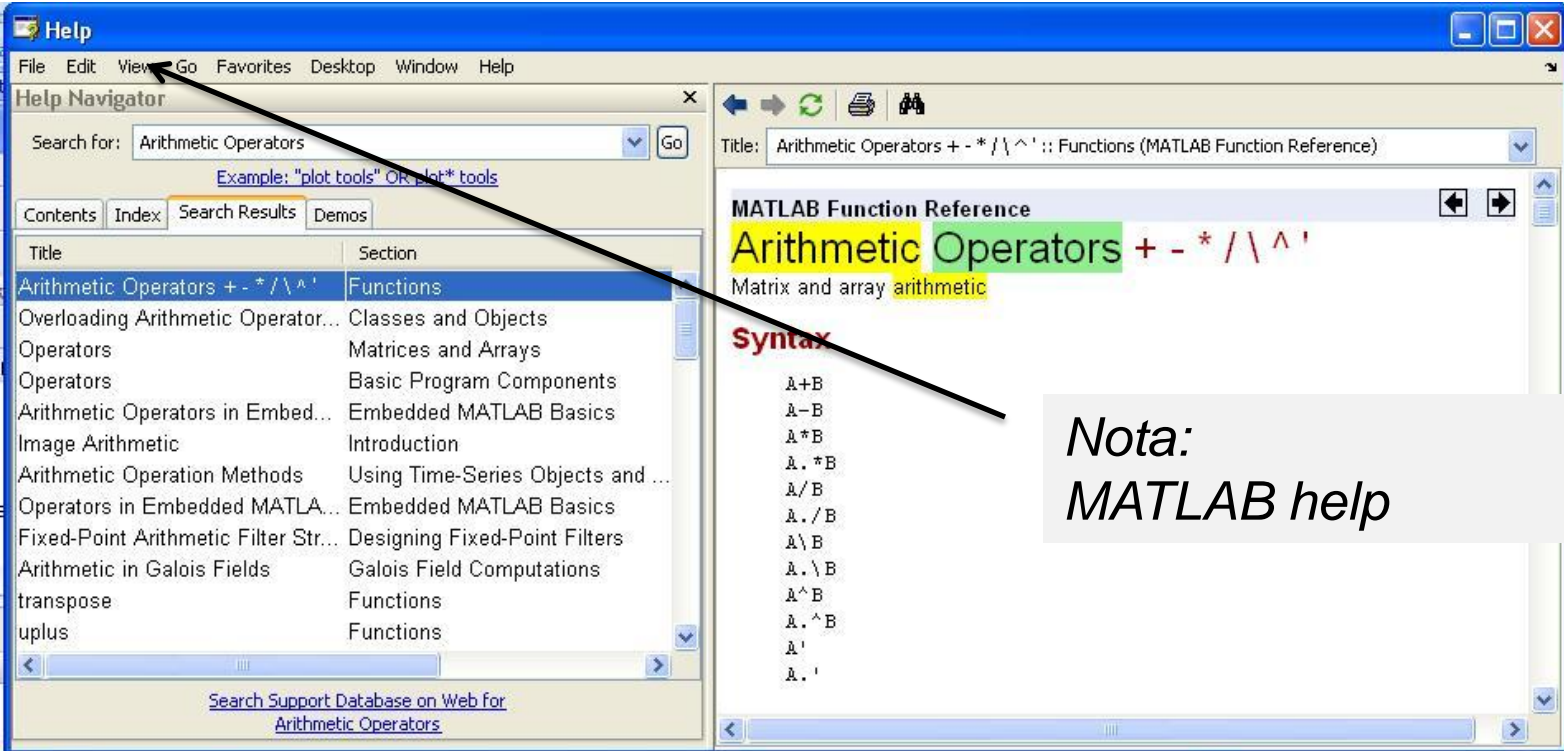
■ Sintaxe:

□ zeros(nlinhas, ncolunas) ou zeros(num)



3. Sintaxe, Variáveis, Matrizes e Vectors

■ Operações algébricas e ponto a ponto



The screenshot shows the MATLAB Help Navigator window. The search bar contains 'Arithmetic Operators'. The 'Search Results' tab is active, displaying a list of results. The first result, 'Arithmetic Operators + - * / \ ^ \'', is selected and highlighted in blue. The main content area shows the 'MATLAB Function Reference' for 'Arithmetic Operators', with the title and the operators themselves highlighted in yellow and green. Below the title, the text 'Matrix and array arithmetic' is visible. A 'Syntax' section lists various expressions: $A+B$, $A-B$, $A*B$, $A.*B$, $A./B$, $A./B$, $A\B$, $A.\B$, $A.^B$, $A.^B$, A' , and $A.'$. A black arrow points from the search bar to the 'Syntax' section. A grey callout box on the right contains the text 'Nota: MATLAB help'.

Title	Section
Arithmetic Operators + - * / \ ^ \'	Functions
Overloading Arithmetic Operator...	Classes and Objects
Operators	Matrices and Arrays
Operators	Basic Program Components
Arithmetic Operators in Embed...	Embedded MATLAB Basics
Image Arithmetic	Introduction
Arithmetic Operation Methods	Using Time-Series Objects and ...
Operators in Embedded MATLA...	Embedded MATLAB Basics
Fixed-Point Arithmetic Filter Str...	Designing Fixed-Point Filters
Arithmetic in Galois Fields	Galois Field Computations
transpose	Functions
uplus	Functions

Nota:
MATLAB help



3. Sintaxe, Variáveis, Matrizes e Vectores

■ Operações Algébricas ou matriciais (+ - * / ^)

```
>> v=1:3
```

```
v =
```

```
    1    2    3
```

```
>> v*v'
```

```
ans =
```

```
    14
```

```
>> v'*v
```

```
ans =
```

```
    1    2    3  
    2    4    6  
    3    6    9
```

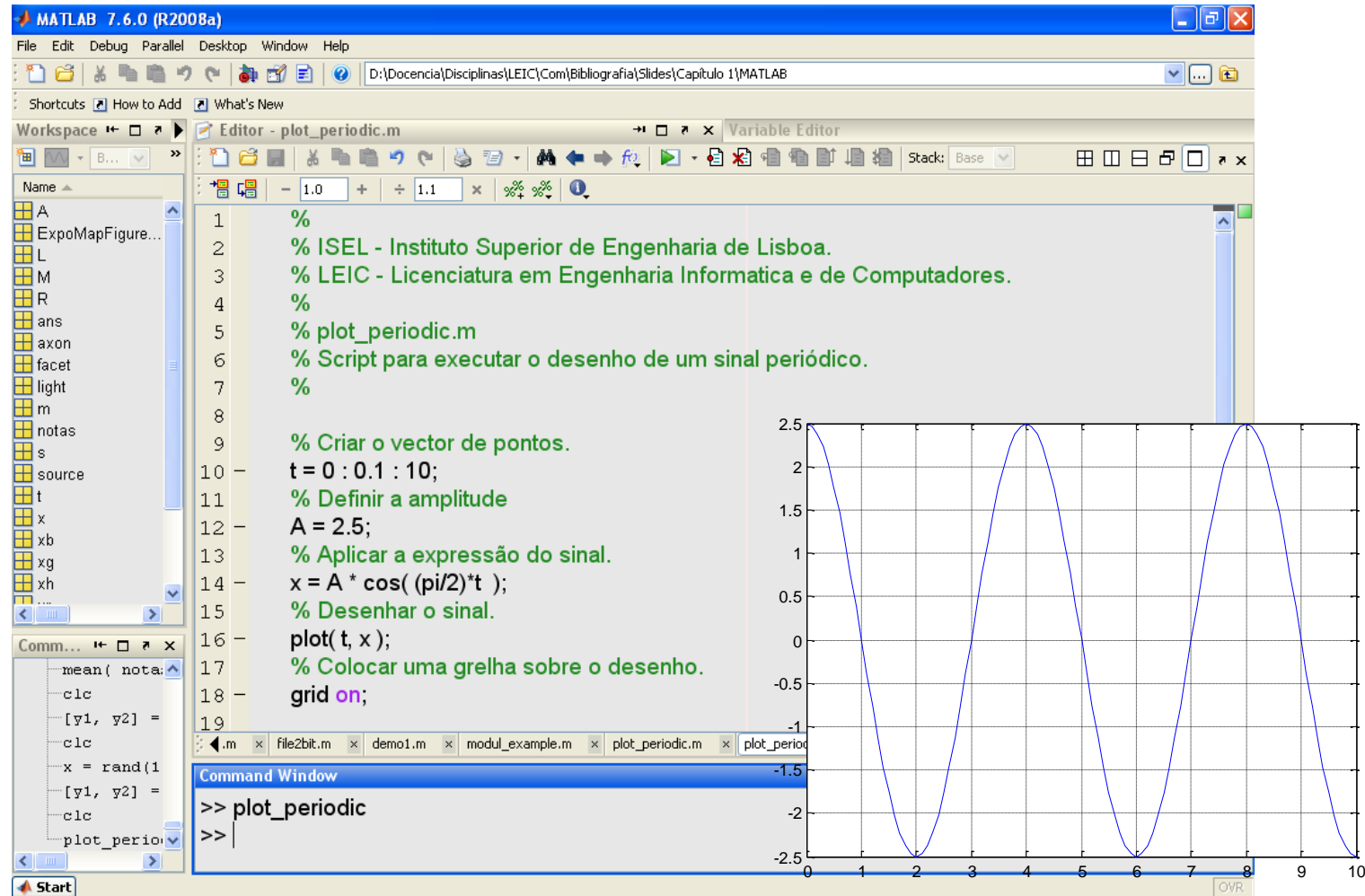
■ Operações elemento a elemento (ponto a ponto)

- .* produto ponto a ponto entre vectores/matrizes da mesma dimensão
- ./ divisão ponto a ponto entre vectores/matrizes da mesma dimensão



4. Operações comuns

■ Desenho de uma sinusóide



4. Operações comuns

- Cálculo de energia do sinal $x[n]$ (vector linha)

```
>> sum( x.^2 )
```

```
>> sum( x .* x )
```

```
>> x * x'
```

- Cálculo do valor médio do sinal $x[n]$ (vector linha)

```
>> mean( x )
```

```
>> sum( x ) / length(x)
```



5. Controlo de fluxo: if, for, while

■ Decisão binária If

```
if expression,  
statements,  
end
```

```
if expression1  
    statements1  
elseif expression2  
    statements2  
else  
    statements3  
end
```



5. Controlo de fluxo: if, for, while

■ Ciclo for

```
for x=initval:endval, statements, end
```

```
for x=initval:stepval:endval, statements, end
```

```
for variable = initval:endval  
    statement  
    ...  
    statement  
end
```



5. Controlo de fluxo: if, for, while

■ Ciclo while

```
while expression, statements, end
```

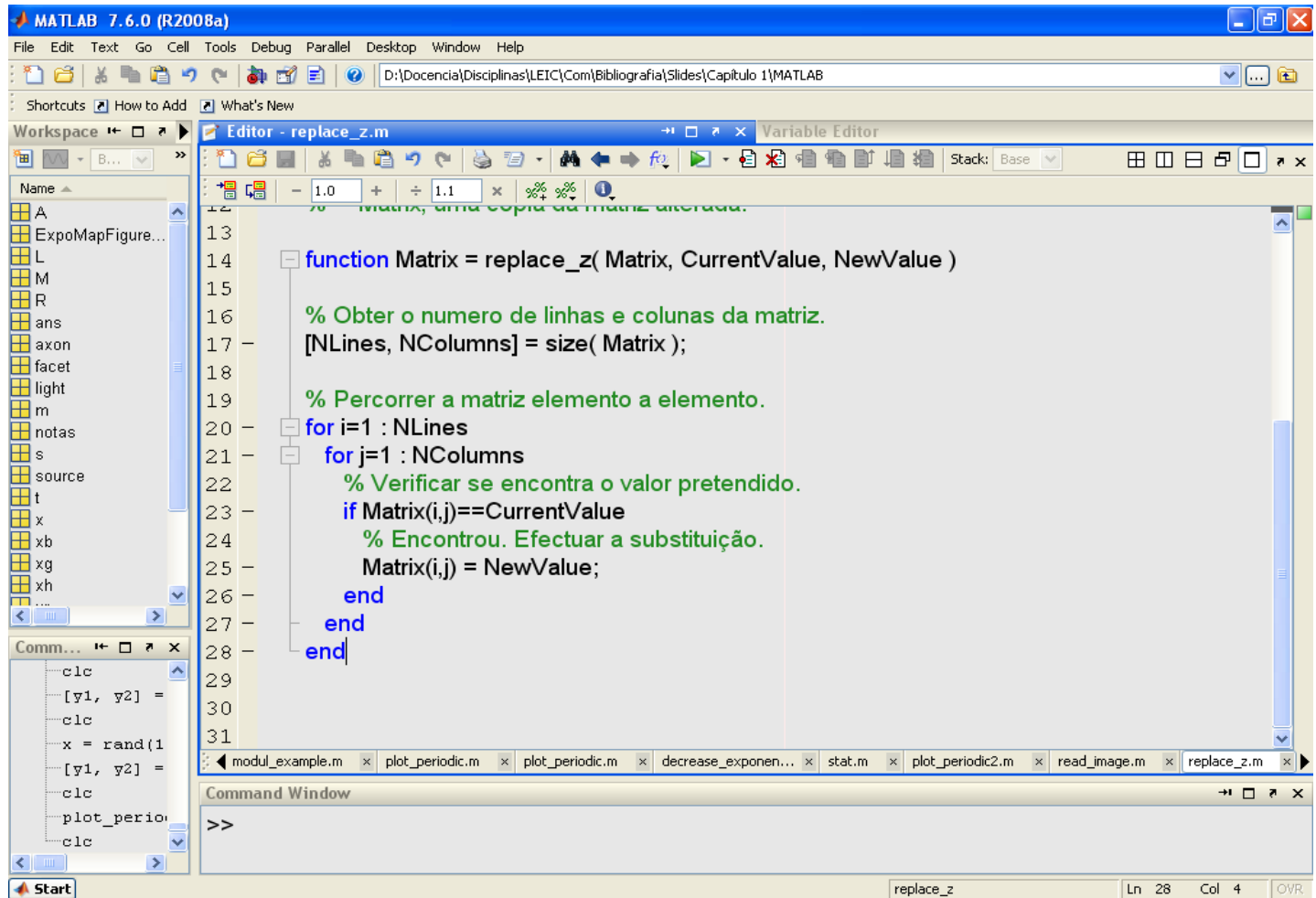
■ Exemplo

```
eps = 1;  
while (1+eps) > 1  
    eps = eps/2;  
end  
eps = eps*2
```



5. Controlo de fluxo: if, for, while

- Ciclo for – actualização de matriz



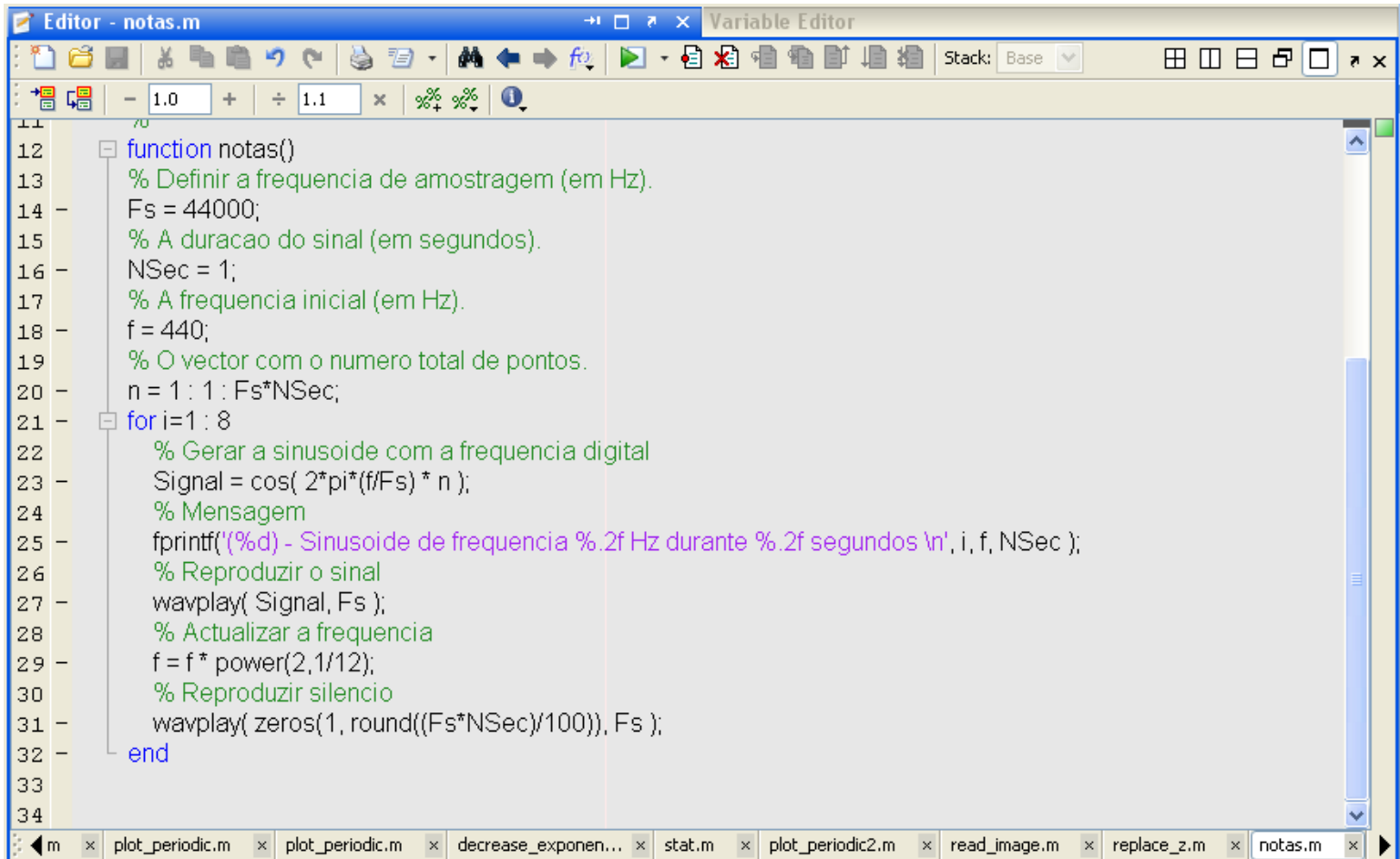
```
function Matrix = replace_z( Matrix, CurrentValue, NewValue )  
  
% Obter o numero de linhas e colunas da matriz.  
[NLines, NColumns] = size( Matrix );  
  
% Percorrer a matriz elemento a elemento.  
for i=1 : NLines  
    for j=1 : NColumns  
        % Verificar se encontra o valor pretendido.  
        if Matrix(i,j)==CurrentValue  
            % Encontrou. Efectuar a substituição.  
            Matrix(i,j) = NewValue;  
        end  
    end  
end
```

The screenshot shows the MATLAB 7.6.0 (R2008a) interface. The main window is the Editor, displaying a function named 'replace_z.m'. The function uses nested for loops to iterate through the rows and columns of a matrix 'Matrix'. Inside the inner loop, an if statement checks if the current element 'Matrix(i,j)' is equal to 'CurrentValue'. If true, it replaces it with 'NewValue'. The Command Window at the bottom shows the prompt '>>'.



5. Controlo de fluxo: if, for, while

■ Ciclo for – reprodução de notas musicais



```
11
12 function notas()
13 % Definir a frequencia de amostragem (em Hz).
14 Fs = 44000;
15 % A duracao do sinal (em segundos).
16 NSec = 1;
17 % A frequencia inicial (em Hz).
18 f = 440;
19 % O vector com o numero total de pontos.
20 n = 1 : 1 : Fs*NSec;
21 for i=1 : 8
22 % Gerar a senoide com a frequencia digital
23 Signal = cos( 2*pi*(f/Fs) * n);
24 % Mensagem
25 fprintf('%d) - Senoide de frequencia %.2f Hz durante %.2f segundos \n', i, f, NSec);
26 % Reproduzir o sinal
27 wavplay( Signal, Fs );
28 % Actualizar a frequencia
29 f = f * power(2,1/12);
30 % Reproduzir silencio
31 wavplay( zeros(1, round((Fs*NSec)/100)), Fs );
32 end
33
34
```



6. Funções Úteis

conv	Calcula a convolução linear entre dois sinais
fft e fftshift	Calcula o espectro de um sinal.
figure	Cria uma nova figura.
filter	Realiza a filtragem de um sinal por um determinado sistema.
hist	Desenha o histograma dos dados presentes num vector.
length	Calcula a dimensão de um vector.
max	Calcula o valor máximo de um vector.
mean	Calcula o valor médio de um vector.
min	Calcula o valor mínimo de um vector.
plot	Desenha vectores unindo os pontos, tornando o sinal aparentemente contínuo.
rand, randn e randint	Gera matriz com valores aleatórios
ones	Cria uma matriz com todos os elementos iniciados a 1.
size	Obtém as dimensões de uma matriz.
sound ou soundsc	Reproduzem para a card audio, o conteúdo de um vector.
stem	Desenha vectores, mostrando as amostras individualmente.
subplot	Define vários gráficos dentro da mesma figura.
sum	Soma todos os elementos de um vector.
wavplay	Reproduz para a card audio, o conteúdo de um vector.
wavread	Lê o conteúdo de um ficheiro wave para um vector.
wavrecord	Adquire um sinal a partir da card audio.
wavwrite	Escreve para ficheiro wave, o conteúdo de um vector.
zeros	Cria uma matriz com todos os elementos iniciados a zero.



6. Comandos Úteis

xlabel	Coloca a label eixo xx, num gráfico
Ylabel	Coloca a label eixo yy, num gráfico
title	Coloca título no gráfico
legend	Define a legenda para o gráfico
clc	Apaga o conteúdo da janela de comandos.
clear	Remove todas as variáveis em memória.
clear x	Remove a variável x de memória.
close all	Fecha todas as janelas de figuras.
grid on	Coloca uma grelha sobre o eixo gráfico.



7. Links

- Mathworks, empresa que desenvolve e comercializa o MATLAB
 - <http://www.mathworks.com/>
- MatlabCentral, troca de informação, código e resolução de problemas em MATLAB
 - <http://www.mathworks.com/matlabcentral/>

