

# Noções de programação em Matlab

## Implementação de Métodos Recursivos

Jefferson Bertolai

24/out/2011

# Operações Básicas

Operação	Símbolo	Exemplos
$x + y$	+	$2 + 3$
$x - y$	-	$5 - 7$
$x \cdot y$	*	$7 * 8$
$x \div y$	/ ou \	$9/3$ ou $3 \setminus 9$
$x^y$	^	$2^3$

# Operadores Lógicos

Descrição	Símbolo	Exemplo	Valor lógico
maior do que	$>$	$2 > 0,5$	1
maior ou igual a	$\geq$	$2 \geq 2,5$	0
igual a	$==$	$3 == (4 - 1)$	1
diferente de	$\sim$	$2 * 3 \sim 6$	0
e	$\&$	$(3 > 1) \& (2 > 2)$	0
disjunção	$ $	$(3 > 1)   (1 < 2)$	1
disjunção exclusiva	<i>xor</i>	$\text{xor}((3 > 1), (1 < 2))$	0

# Definição e Manipulação de Objetos

```
var = 5    % cria uma variável com o nome 'var'  
% e valor inicial de '5'
```

```
v = [1 4 2]    % cria o vetor de três elementos a seguir
```

$$v = ( 1 \ 4 \ 2 )$$

```
x = [var v]    % concatena var e v, criando vetor  
% de 4 elementos
```

$$x = ( 5 \ 1 \ 4 \ 2 )$$

```
var = var + 2;    % soma 2 à variável var
```

## Definição e Manipulação de Objetos

`y = v'`;    % transpõe o vetor v, ou seja,

$$y = \begin{pmatrix} 1 \\ 4 \\ 2 \end{pmatrix}$$

`c = 1:2:7`;    % cria o vetor a seguir

$$c = ( 1 \ 3 \ 5 \ 7 )$$

`Z = zeros(2,2)`    % cria matriz 2x2 de zeros

$$Z = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$$

# Definição e Manipulação de Objetos

```
O = ones(3)    % cria matriz 3x3 a seguir
```

$$O = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

```
d = [-5,3;2 7];    % cria a matriz a seguir
```

$$d = \begin{pmatrix} -5 & 3 \\ 2 & 7 \end{pmatrix}$$

```
d(:,1)    % retorna primeira coluna de d
```

```
d(2,:)    % retorna segunda linha de d
```

```
d(1,2)    % retorna o elemento da linha 1 da coluna 2
```

# Definição e Manipulação de Objetos

```
dd = (-1).*d    % multiplica matriz d (elemento  
                % a elemento) por (-1)
```

$$dd = \begin{pmatrix} 5 & -3 \\ -2 & -7 \end{pmatrix}$$

```
dd.*d    % multiplica as matrizes dd e d,  
         % elemento a elemento
```

```
dd*d     % multiplicação matricial
```

```
z = dd.^d   % para cada (i,j) calcula a matriz com  
            % elemento geral dado por z(i,j)=dd(i,j)^d(i,j)
```

# Definição e Manipulação de Objetos

```
eye(n);      % cria matriz identidade de dimensão nxn

plot(x,y)    % constrói e exhibe o
              % gráfico da função f tal que
              % para todo i vale  $y(i) = f(x(i))$ 

kron(d,dd);  % produto Kronecker entre d e dd

size(dd);    % retorna as dimensões da matriz dd
```



# Funções e Constantes pré-definidas

Comando	Função / Constante	Exemplo
sqrt	raiz quadrada	sqrt(2)
pi	$\pi = 3,1416\dots$	-
cos	cosseno	cos(0)
tan	tangente	tan(pi/4)
exp	exponencial	exp(1)
log	logarítmica	log(exp(3))
inf	$\infty$	-
sum	somatória	sum([5,9,1])
prod	produtório	prod([1,3,7])
max	máximo	max([1,4,2])

# M-Files

**Script M-files:** define uma rotina a ser executada pela Matlab

```
% salvar script como routine.m  
X = [1,2;3,4];  
Y = [2,1;4,3];  
Z = X*Y
```

**Execução:** invocar comando `routine` na janela de comandos

**Function M-files:** função definida pelo usuário

```
% salvar m-file como func.m
function [out1 out2] = func(arg1,arg2)
out1 = arg1+arg2;
out2 = arg1*arg2
```

**Execução:** invocar comando `func(2,3)` na janela de comandos

# Loops

**for:** determina que uma rotina seja repetida um número determinado de vezes. A rotina abaixo resulta em  $f = 1 * 2 * 3 * 4 = 24$

```
f = 1;
for n=2:4
    f = f*n;
end
f
```

# Loops

**while**: determina que uma rotina seja repetida enquanto uma determinada condição for verdadeira

```
n = 1;    oldsum = -1;    newsum = 0;
while newsum > oldsum
    oldsum = newsum;
    newsum = newsum + n ^(-4);
    n = n + 1;
end
newsum
```

# Branching (if)

```
function y = absval(x)
if x >= 0
    y = x;
else
    y = -x;
end
```

```
function y = signum(x)
if x > 0
    y = 1;
elseif x==0
    y = 0;
else
    y = -1;
end
```

# Comandos Especiais

<b>Comando</b>	<b>Descrição</b>
Ctrl + c	interrompe procedimento
;	impede exibição do output do comando
%	insere comentário
Ctrl + r	transforma comando em comentário
Ctrl + t	transforma comentário em comando
whos	exibe objetos atualmente definidos
F5	salva e executa script
F9	executa seleção
eye(n)	cria matriz identidade com dimensões n×n
clear x	deleta objeto x
clear all	deleta todos os objetos