

# Octave Quick Reference Octave Version 1.1.1

## Iniciando o Octave

<b>octave</b>	inicia seção interativa do Octave
<b>octave <i>arquivo</i></b>	roda Octave com os comandos em <i>arquivo</i>
<b>octave --help</b>	descreve opções de linha de comando

## Parando o Octave

<b>quit</b> ou <b>exit</b>	sai do Octave
<b>INTERRUPT</b>	( <i>e.g.</i> <b>C-c</b> ) termina o comando corrente e retorna para o prompt do nível pai

## Conseguindo Ajuda

<b>help</b>	lista todos os comando e variáveis internas
<b>help <i>comando</i></b>	descreve brevemente o <i>comando</i>
<b>help -i</b>	manual Info do Octave
<b>help -i <i>comando</i></b>	busca por <i>comando</i> no manual do Octave

## Movendo-se no info

<b>SPC</b> ou <b>C-v</b>	rola uma página abaixo
<b>DEL</b> ou <b>M-v</b>	rola uma página acima
<b>C-l</b>	redesenha a tela

## Seleção do Nó no Info

<b>n</b>	seleciona o nó seguinte
<b>p</b>	seleciona o nó anterior
<b>u</b>	seleciona o nó acima ('up')
<b>t</b>	seleciona o nó pai ('top')
<b>d</b>	seleciona o nó <i>directory</i>
<b>&lt;</b>	seleciona o primeiro nó do arquivo corrente
<b>&gt;</b>	seleciona o último nó do arquivo corrente
<b>g</b>	lê o nome de um nó e o seleciona
<b>C-x k</b>	mata o nó corrente

## Procurando no Info

<b>s</b>	procura por uma <i>string</i>
<b>C-s</b>	procura a frente incrementalmente
<b>C-r</b>	procura para traz incrementalmente
<b>i</b>	procura no índice e vai para o
<b>,</b>	próxima ocorrência do último comando i

## Movimentação do Cursor

<b>C-b</b>	retorna um caracter
<b>C-f</b>	avança um caracter
<b>C-a</b>	move para o começo da linha
<b>C-e</b>	move para o final da linha
<b>M-f</b>	move para a próxima palavra
<b>M-b</b>	move para a palavra anterior
<b>C-l</b>	limpa a tela, reescrevendo a linha atual

## Inserindo ou Alterando Texto

<b>M-TAB</b>	insere um caracter de tab
<b>DEL</b>	apaga caracter a direita do cursor
<b>C-d</b>	apaga caracter sob o cursor
<b>C-v</b>	insere próximo caracter textualmente
<b>C-t</b>	transpõe dois caracteres
<b>M-t</b>	transpõe duas palavras

[ ] delimita argumentos opcionais ... exibe um ou mais argumentos

## Cortando e Colando

<b>C-k</b>	corta até o final da linha
<b>C-y</b>	cola o texto cortado mais recentemente
<b>M-d</b>	corta até o final da palavra corrente
<b>M-DEL</b>	corta a palavra anterior ao cursor
<b>M-y</b>	roda a lista de cortes e cola o topo

## Completamento de Comandos e Histórico

<b>TAB</b>	completa o comando ou o nome da variável
<b>M-?</b> ou <b>TAB TAB</b>	lista os possíveis completamentos
<b>RET</b>	entra a linha corrente
<b>C-p</b>	move para 'cima' no histórico
<b>C-n</b>	move para 'baixo' no histórico
<b>M-&lt;</b>	move para a primeira linha do histórico
<b>M-&gt;</b>	move para a última linha do histórico
<b>C-r</b>	busca para trs no histórico
<b>C-s</b>	busca para frente no histórico

**history [-q] [N]** lista *N* linhas anteriores do histórico, omitindo a numeração se **-q**

**history -w [arq]** salva histórico em *arq* (~/.octave\_hist se *arq* não fornecido)

**history -r [arq]** lê histórico de *arq* (~/.octave\_hist se *arq* não fornecido)

**edit\_history *linha*** edita e roda comando do histórico

**run\_history *linha* [ini] [fim]** roda comando do histórico

Especifica o primeiro e último comando do histórico para editar e rodar. Se *ini* for maior que *fim*, reverte a lista de comandos antes de edita-la. Se *fim* for omitido, seleciona comandos a partir de *ini* até o final do histórico. Se ambos forem omitidos, edita o ítem anterior do histórico.

## Commandos da Shell

<b>cd <i>dir</i></b>	troca o diretório de trabalho para <i>dir</i>
<b>pwd</b>	exibe o diretório corrente
<b>ls [opts]</b>	lista o conteúdo do diretório
<b>getenv (<i>var</i>)</b>	retorna o valor da variável de ambiente prescrita
<b>system (<i>cmd</i>)</b>	executa o comando da shell prescrito

## Matrizes

Colchetes delimitam matrizes literais. Vírgulas separam elementos na mesma linha. Ponto-e-vírgula separam linhas. Vírgulas podem ser substituídas por espaços, e ponto-e-vírgulas pode ser substituídos por quebras de linha. Elementos de uma matriz pode ser expressões arbitrárias, desde que as dimensões coincidam.

[ <i>x, y, ...</i> ]	vetor linha
[ <i>x; y; ...</i> ]	vetor coluna
[ <i>w, x; y, z</i> ]	matriz 2×2

## Faixas

*base* : *limite*  
*base* : *incr* : *limite*

Especifica um intervalo de valores começando em *base*, com elementos menores ou iguais a *limite*. Se omitido, o valor padrão para *incr* é 1. Incrementos negativos são permitidos.

## Strings e Caracteres Especiais

Uma *string constante* consiste de uma seqüência de caracteres entre aspas simples ou duplas.

\\	uma barra invertida literal
\"	um caracter de aspas duplas literal
\'	um caracter de aspas simples literal
\n	quebra de linha, código ASCII 10
\t	tabulação horizontal, código ASCII 9

## Indexando Expressões

<b>var (<i>idx</i>)</b>	seleciona elementos de um vetor
<b>var (<i>idx1, idx2</i>)</b>	sleciona elementos de uma matriz
<b><i>escalar</i></b>	seleciona linha (coluna) correspondente ao <i>escalar</i>
<b><i>vetor</i></b>	seleciona linhas (colunas) correspondentes aos elementos do <i>vetor</i>
<b><i>faixa</i></b>	seleciona linhas (colunas) correspondentes aos elementos da <i>faixa</i>
<b>:</b>	seleciona todas as linhas (colunas)

## Variáveis Globais

**global *var1 ...*** declara uma variável global

Variáveis globais podem ser acessadas dentro do corpo de funções sem terem sido passadas na lista de parâmetros da função, desde que também sejam declaradas como globais dentro da função.

## Variáveis Internas Selecionadas

<b>EDITOR</b>	editor utilizado em <b>edit_history</b>
<b>Inf, NaN</b>	Infinito no padrão IEEE, NaN
<b>LOADPATH</b>	caminho de busca por arquivos de funções
<b>PAGER</b>	programa utilizado para paginar a tela
<b>ans</b>	último resultado não explicitamente atribuído
<b>eps</b>	precisão da máquina
<b>pi</b>	$\pi$
<b>realmax</b>	maior valor absoluto representável
<b>realmin</b>	menor valor absoluto representável

<b>automatic_replot</b>	redesenha automaticamente gráficos
<b>do_fortran_indexing</b>	estilo Fortran de indexamento
<b>implicit_str_to_num_ok</b>	permite conversão string-número
<b>output_max_field_width</b>	comprimento máximo para campo numérico
<b>output_precision</b>	min dígitos significativos exibidos
<b>page_screen_output</b>	controla se a saída é paginada
<b>prefer_column_vectors</b>	cria vetores coluna por padrão
<b>resize_on_range_error</b>	redimensionamento automático de matrizes
<b>save_precision</b>	dígitos gravados pelo comando <b>save</b>
<b>silent_functions</b>	elimina saída de funções
<b>warn_divide_by_zero</b>	elimina erros de divisão por zero

**commas\_in\_literal\_matrix** controla com lidar com espaços em matrizes

**ignore\_function\_time\_stamp** ignora alterações em arquivos de funções durante sessão

**ok\_to\_lose\_imaginary\_part** habilita conversão de complexo para real

**prefer\_zero\_one\_indexing** na dúvida, prefere estilo de indexação 0-1

## Operadores Aritméticos e Incrementais

<code>x + y</code>	adição
<code>x - y</code>	subtração
<code>x * y</code>	multiplicação matricial
<code>x .* y</code>	multiplicação elemento por elemento
<code>x / y</code>	divisão à direita, conceitualmente equivalente a <b>(inverse (y') * x')</b>
<code>x ./ y</code>	divisão à direita elemento por elemento
<code>x \ y</code>	divisão à esquerda, conceitualmente equivalente a <b>inverse (x) * y</b>
<code>x .\ y</code>	divisão à esquerda elemento por elemento
<code>x ^ y</code>	potenciação
<code>x .^ y</code>	potenciação elemento por elemento
<code>- x</code>	troca de sinal
<code>+ x</code>	soma unária (sem efeito)
<code>x ' </code>	transposto conjugado complexo
<code>x .' </code>	transposto
<code>++ x ( -- x)</code>	incrementa (decrementa) <i>x</i> , retorna o <i>novo</i>
<code>x ++ ( x --)</code>	incrementa (decrementa) <i>x</i> , retorna <i>velho</i>

## Expressões de Atribuição

<code>var = expr</code>	atribui expressão à variável
<code>var (idx) = expr</code>	atribui expressão à variável indexada

## Comparações e Operadores Booleanos

Estes operadores trabalham elemento por elemento. Ambos os argumentos são sempre avaliados.

<code>x &lt; y</code>	verdadeiro se <i>x</i> for menor que <i>y</i>
<code>x &lt;= y</code>	verdadeiro se <i>x</i> for menor que ou igual a <i>y</i>
<code>x == y</code>	verdadeiro se <i>x</i> for igual a <i>y</i>
<code>x &gt;= y</code>	verdadeiro se <i>x</i> for maior que ou igual a <i>y</i>
<code>x &gt; y</code>	verdadeiro se <i>x</i> for maior que <i>y</i>
<code>x != y</code>	verdadeiro se <i>x</i> for diferente de <i>y</i>
<code>x &amp; y</code>	verdadeiro se <i>x</i> e <i>y</i> são ambos verdadeiro
<code>x   y</code>	verdadeiro se <i>x</i> ou <i>y</i> for verdadeiro
<code>! bool</code>	verdadeiro se <i>bool</i> for falso

## Operadores Booleanos Curto-circuíto

Operadores avaliam da esquerda para a direita, esperando escalares. Operandos são avaliados apenas se necessário, parando quando já for possível determinar o resultado da operação. Operandos são convertidos para escalar pela aplicação da função **all**.

<code>x &amp;&amp; y</code>	verdadeiro se <i>x</i> e <i>y</i> são ambos verdadeiros
<code>x    y</code>	verdadeiro se <i>x</i> ou <i>y</i> for verdadeiro

## Precedência de Operadores

Tabela de operadores em ordem de precedência.

<code>;</code> ,	separadores
<code>=</code>	atribuição
<code>   &amp;&amp;</code>	“ou” e “e” lógicos
<code>  &amp;</code>	“ou” e “e” lógicos por elementos
<code>&lt; &lt;= == &gt;= &gt; !=</code>	operadores relacionais
<code>:</code>	dois pontos
<code>+ -</code>	adição e subtração
<code>* / \ .* ./ .\ ' .'</code>	mutiplicação e divisão
<code>' .'</code>	transposto
<code>+ - ++ -- !</code>	menos unário, incremento, negação
<code>^ .^</code>	exponenciação

## Statements

**for** *identifíer* = *expr lista-cmds* **endfor**  
Executa *lista-cmds* uma vez para cada coluna de *expr*. A variável *identifíer* guarda o valor da coluna corrente durante cada itereção.

**while** (*condição*) *lista-cmds* **endwhile**  
Executa *lista-cmds* enquanto *condição* for verdadeira.

**break** abandona o loop mais interno  
**continue** vai para o começo do loop mais interno  
**return** retorna para a função chamadora

**if** (*condição*) *corpo-do-if* [**else** *corpo-do-else*] **endif**

Executa *corpo-do-if* se *condição* for verdadeira, caso contrário executa *corpo-do-else*.

**if** (*cond1*) *lista-cmd1* [**elseif** (*cond*) *lista-cmd*] **endif**

Executa *lista-cmd1* se *cond1* for verdadeira, caso contrário executa *lista-cmd* correspondendo à primeira condição **elseif** que seja verdadeira, caso contrário executa o corpo de comandos do bloco **else**.

Num **if** pode haver quantidade arbitrária de blocos **elseif**.

**unwind\_protect** *cmds1* **unwind\_protect\_cleanup** *cmds2* **end**

Executa *cmds1*. Executa *cmds2* independentemente de como se tenha saído do bloco *cmds1*.

## Definindo Funções

**function** [*lista-retorno*] *function-name* [(*lista-de-args*)]  
*corpo-da-função*  
**endfunction**

*lista-de-retorno* pode ser um único identificador ou uma lista de identificadores, separada por vírgulas e delimitada por colchetes.

*lista-de-args* é uma lista de identificadores, separada por vírgulas, e pode ser vazia.

## Manipulações Matriciais Básicas

<b>rows</b> ( <i>a</i> )	retorna o número de linhas de <i>a</i>
<b>columns</b> ( <i>a</i> )	retorna o número de colunas de <i>a</i>
<b>all</b> ( <i>a</i> )	1 se todos elementos de <i>a</i> são não-nulos
<b>any</b> ( <i>a</i> )	1 se algum elemento de <i>a</i> é não-nulo
<b>find</b> ( <i>a</i> )	retorna os índices dos elementos não-nulos
<b>sort</b> ( <i>a</i> )	ordena os elementos em cada coluna de <i>a</i>
<b>sum</b> ( <i>a</i> )	soma os elementos nas colunas de <i>a</i>
<b>prod</b> ( <i>a</i> )	produto dos elementos nas colunas de <i>a</i>
<b>min</b> ( <i>args</i> )	encontra o valor mínimo
<b>max</b> ( <i>args</i> )	encontra o valor máximo
<b>rem</b> ( <i>x</i> , <i>y</i> )	encontra o resto de <i>x/y</i>
<b>reshape</b> ( <i>a</i> , <i>m</i> , <i>n</i> )	reformata <i>a</i> como $m \times n$
<b>diag</b> ( <i>v</i> , <i>k</i> )	cria matriz diagonal
<b>linspace</b> ( <i>b</i> , <i>l</i> , <i>n</i> )	cria vetor com elementos espaçados linearmente
<b>logspace</b> ( <i>b</i> , <i>l</i> , <i>n</i> )	cria vetor com elementos espaçados logaritmicamente
<b>eye</b> ( <i>n</i> , <i>m</i> )	cria matriz identidade $n \times m$
<b>ones</b> ( <i>n</i> , <i>m</i> )	cria matriz de uns $n \times m$
<b>zeros</b> ( <i>n</i> , <i>m</i> )	cria matriz de zeros $n \times m$
<b>rand</b> ( <i>n</i> , <i>m</i> )	cria matriz de valores aleatórios $n \times m$

## Álgebra Linear

<b>chol</b> ( <i>a</i> )	fatoração de Cholesky
<b>det</b> ( <i>a</i> )	determinante da matriz
<b>eig</b> ( <i>a</i> )	autovalores e autovetores
<b>expm</b> ( <i>a</i> )	exponencial da matriz
<b>hess</b> ( <i>a</i> )	decomposição de Hessianberg
<b>inverse</b> ( <i>a</i> )	inversa de matriz quadrada
<b>norm</b> ( <i>a</i> , <i>p</i> )	norma <i>p</i> da matriz <i>a</i>
<b>pinv</b> ( <i>a</i> )	pseudoinversa de <i>a</i>
<b>qr</b> ( <i>a</i> )	fatoração QR
<b>rank</b> ( <i>a</i> )	posto da matriz
<b>schur</b> ( <i>a</i> )	decomposição de Schur
<b>svd</b> ( <i>a</i> )	decomposição em valores singulares
<b>syl</b> ( <i>a</i> , <i>b</i> , <i>c</i> )	resolve a equação de Sylvester

## Equações, EDOs, EDAs, Quadratura

<b>*fsolve</b>	resolve um sistema algébrico de equações
<b>*lsode</b>	integra EDOs não-lineares
<b>*dassl</b>	integra EDAs não-lineares
<b>*quad</b>	integra funções não-lineares

**perror** (*nm*, *code*) para funções que retornam códigos numéricos, imprime a mensagem de erro para a função e o código de erro dado

\* Confira a documentação on-line ou o manual impresso para a lista completa de argumentos para essas funções.

## Processamento de Sinais

<b>fft</b> ( <i>a</i> )	Transformada Rápida de Fourier usando FFTPACK
<b>ifft</b> ( <i>a</i> )	FFT inversa usando FFTPACK
<b>freqz</b> ( <i>args</i> )	filtro FIR
<b>sinc</b> ( <i>x</i> )	retorna $\sin(\pi x)/(\pi x)$

## Processamento de Imagens

<b>colormap</b> ( <i>map</i> )	define o colormap corrente
<b>gray2ind</b> ( <i>i</i> , <i>n</i> )	converte escala de cinzas
<b>image</b> ( <i>img</i> , <i>zoom</i> )	exibe matriz como imagem
<b>imagesc</b> ( <i>img</i> , <i>zoom</i> )	exibe matriz escalada como imagem
<b>imshow</b> ( <i>img</i> , <i>map</i> )	exibe imagem do Octave
<b>imshow</b> ( <i>i</i> , <i>n</i> )	exibe imagem em tons de cinza
<b>imshow</b> ( <i>r</i> , <i>g</i> , <i>b</i> )	exibe imagem RGB
<b>ind2gray</b> ( <i>img</i> , <i>map</i> )	converte imagem para tons de cinza
<b>ind2rgb</b> ( <i>img</i> , <i>map</i> )	converte imagem para RGB
<b>loadimage</b> ( <i>arq</i> )	carrega um arquivo de imagem
<b>rgb2ind</b> ( <i>r</i> , <i>g</i> , <i>b</i> )	converte RGB para imagem
<b>saveimage</b> ( <i>arq</i> , <i>img</i> , <i>fmt</i> , <i>map</i> )	salva matriz para <i>arq</i>

## Conjuntos

<b>create_set</b> ( <i>a</i> , <i>b</i> )	cria vetor linha de valores únicos
<b>complement</b> ( <i>a</i> , <i>b</i> )	elementos de <i>b</i> que não estão em <i>a</i>
<b>intersection</b> ( <i>a</i> , <i>b</i> )	interseção dos conjuntos <i>a</i> e <i>b</i>
<b>union</b> ( <i>a</i> , <i>b</i> )	união dos conjuntos <i>a</i> e <i>b</i>

## Strings

<b>strcmp</b> ( <i>s</i> , <i>t</i> )	compara strings
<b>strcat</b> ( <i>s</i> , <i>t</i> , ...)	concatena strings

## Entrada e Saída no padrão C

<code>fopen</code> ( <i>nome, modo</i> )	abre o arquivo <i>nome</i>
<code>fclose</code> ( <i>arq</i> )	fecha <i>arq</i>
<code>printf</code> ( <i>fmt, ...</i> )	saída formatada para <code>stdout</code>
<code>fprintf</code> ( <i>arq, fmt, ...</i> )	saída formatada para <i>arq</i>
<code>sprintf</code> ( <i>fmt, ...</i> )	saída formatada para string
<code>scanf</code> ( <i>fmt</i> )	entrada formatada de <code>stdin</code>
<code>fscanf</code> ( <i>arq, fmt</i> )	entrada formatada de <i>arq</i>
<code>sscanf</code> ( <i>str, fmt</i> )	entrada formatada de <i>string</i>
<code>fgets</code> ( <i>arq, len</i> )	lê <i>len</i> caracteres de <i>arq</i>
<code>fflush</code> ( <i>arq</i> )	força saídas pendentes para <i>arq</i>
<code>ftell</code> ( <i>arq</i> )	ponteiro para a posição atual
<code>frewind</code> ( <i>arq</i> )	move ponteiro para início
<code>freport</code>	informações sobre arquivos abertos
<code>fread</code> ( <i>arq, tam, prec</i> )	lê arquivo binário
<code>fwrite</code> ( <i>arq, tam, prec</i> )	escreve arquivo binário
<code>feof</code> ( <i>arq</i> )	verifica se está no final do arquivo

Um arquivo pode ser referenciado tanto por seu nome quanto pelo número retornado por `fopen`. Três arquivos estão sempre preconnectados quando Octave inicia: `stdin`, `stdout`, and `stderr`.

## Outras Funções de Entrada e Saída

<code>save</code> <i>arq var ...</i>	salva variáveis em <i>arq</i>
<code>load</code> <i>arq</i>	lê variáveis de <i>arq</i>
<code>disp</code> ( <i>var</i> )	exibe valor de <i>var</i> na tela

## Funções Diversas

<code>eval</code> ( <i>str</i> )	avalia <i>str</i> como um comando
<code>feval</code> ( <i>str, ...</i> )	avalia a função chamada <i>str</i> , passando os argumentos restantes para a função
<code>error</code> ( <i>message</i> )	exibe a mensagem e retorna ao nível superior
<code>clear</code> <i>padrão</i>	apaga as variáveis que casam com o padrão
<code>exist</code> ( <i>str</i> )	verifica a existência da variável ou função
<code>who</code>	lista as variáveis correntes

## Polinômios

<code>compan</code> ( <i>p</i> )	matriz companheira
<code>conv</code> ( <i>a, b</i> )	convolução
<code>deconv</code> ( <i>a, b</i> )	deconvolve dois vetores
<code>poly</code> ( <i>a</i> )	cria polinômio a partir da matriz
<code>polyderiv</code> ( <i>p</i> )	derivada do polinômio
<code>polyreduce</code> ( <i>p</i> )	integral do polinômio
<code>polyval</code> ( <i>p, x</i> )	valor do polinômio em <i>x</i>
<code>polyvalm</code> ( <i>p, x</i> )	valor do polinômio em <i>x</i>
<code>roots</code> ( <i>p</i> )	raízes do polinômio
<code>residue</code> ( <i>a, b</i> )	expansão em frações parciais da razão <i>a/b</i>

## Estatística

<code>corrcoef</code> ( <i>x, y</i> )	coeficiente de correlação
<code>cov</code> ( <i>x, y</i> )	covariância
<code>mean</code> ( <i>a</i> )	valor esperado
<code>median</code> ( <i>a</i> )	mediana
<code>std</code> ( <i>a</i> )	desvio padrão
<code>var</code> ( <i>a</i> )	variância

## Gráficos Básicos

<code>gplot</code> [ <i>ranges</i> ] <i>expr</i> [ <i>using</i> ] [ <i>title</i> ] [ <i>style</i> ]	gráficos 2D
<code>gspplot</code> [ <i>ranges</i> ] <i>expr</i> [ <i>using</i> ] [ <i>title</i> ] [ <i>style</i> ]	gráficos 3D
<i>ranges</i>	especifica a faixa dos dados
<i>expr</i>	expressão a ser graficada
<i>using</i>	colunas a serem graficadas
<i>title</i>	especifica título para legendas
<i>style</i>	especifica o estilo da linha

Se forem fornecidos *ranges*, estes devem aparecer antes da expressão a ser graficada. As opções *using*, *title*, e *style* pode aparecer em qualquer ordem depois de *expr*. Várias expressões podem ser graficadas com um único comando, separando-as por vírgulas.

<code>set</code> <i>opções</i>	define opções para gráficos
<code>show</code> <i>opções</i>	exibe opções para gráficos
<code>replot</code>	redesenha o gráfico corrente
<code>closeplot</code>	encerra o fluxo para processamento pelo <code>gnuplot</code>
<code>purge_tmp_files</code>	apaga arquivos temporários
<code>automatic_replot</code>	variável interna

## Outras Funções Gráficas

<code>plot</code> ( <i>args</i> )	gráfico 2D com eixos lineares
<code>semilogx</code> ( <i>args</i> )	gráfico 2D com eixo x logarítmico
<code>semilogy</code> ( <i>args</i> )	gráfico 2D com eixo y logarítmico
<code>loglog</code> ( <i>args</i> )	gráfico 2D com eixos logarítmicos
<code>bar</code> ( <i>args</i> )	gráfico de barras
<code>stairs</code> ( <i>x, y</i> )	gráfico escada
<code>hist</code> ( <i>y, x</i> )	histograma

`title` (*string*) define título do gráfico

<code>axis</code> ( <i>limites</i> )	define a faixa dos eixos
<code>xlabel</code> ( <i>string</i> )	define o rótulo para o eixo x
<code>ylabel</code> ( <i>string</i> )	define o rótulo para o eixo y
<code>grid</code> [on off]	liga/desliga o grid
<code>hold</code> [on off]	liga/desliga sobreposição de gráficos
<code>ishold</code>	retorna status da sobreposição

<code>mesh</code> ( <i>x, y, z</i> )	gráfico 3D de superfícies
<code>meshdom</code> ( <i>x, y</i> )	cria matrizes de coordenadas para malhas

---

Edição 1.1 para Octave Versão 1.1.1. Copyright 1996, John W. Eaton (jwe@che.utexas.edu). O autor não se responsabiliza por qualquer erro neste cartão de referência.

Este cartão pode ser livremente distribuído sob os termos da Licença Pública Geral GNU.

Macros T<sub>E</sub>X para este cartão escritas por Roland Pesch (pesch@cygnus.com), originalmente para o cartão do GDB.

Octave é software livre; você pode redistribuí-lo e/ou modificá-lo sob os termos da Licença Pública Geral GNU, conforme publicada pela Free Software Foundation. Não há nenhuma garantia para Octave.

Tradução de Ricardo Biloti (biloti@ime.unicamp.br), 2006.