Amos Gilat
Department of Mechanical Engineering
The Ohio State University

SEGUNDA EDIÇÃO

MATLAB®

COM APLICAÇÕES EM ENGENHARIA

Tradução:
Glayson Eduardo de Figueiredo
Bacharel em Física pela UFMG
Mestrando em Engenharia Elétrica (UFMG)

Consultoria, supervisão e revisão técnica desta edição:
Antonio Pertence Júnior
Engenheiro Eletrônico e de Telecomunicações
Especialista em Processamento de Sinais (Ryerson University - Canadá)
Professor de Telecomunicações da FUMEC (MG)
Professor Titular da Faculdade de Sabará/MG

Reimpressão 2008

bookman®

2006

CAPÍTULO 1

Iniciação ao Ambiente do MATLAB

Este capítulo descreve as principais características e propósitos das diferentes janelas do MATLAB. Primeiramente, discutiremos em detalhes a janela Command Window e a exploraremos até o final do capítulo. O Capítulo 1 mostra também como usar o MATLAB para realizar operações aritméticas com números de um modo bastante parecido com as operações realizadas em uma calculadora simples (isso inclui algumas funções matemáticas elementares). O capítulo termina ensinando a declarar as variáveis escalares (por meio do operador de atribuição) e a utilizá-las em cálculos numéricos.

1.1 INICIALIZANDO O MATLAB: JANELAS DO MATLAB

Considerando-se que o *software* encontra-se instalado e inicializado no computador do usuário, as janelas abertas são semelhantes àquelas mostradas na Figura 1-1. A figura contém três pequenas janelas: Command Window, Current Directory Window e Command History Window. Esse é o modo de abertura padrão (*default*) do MATLAB. Ao todo, o MATLAB pode apresentar oito janelas diferentes. A lista completa das janelas do MATLAB com os respectivos propósitos está resumida na Tabela 1-1. O botão **Start**, no canto esquerdo inferior da tela, é uma novidade das versões mais recentes. Pode-se acessar todas as ferramentas e propriedades do MATLAB por esse botão.

Quatro dessas janelas – Command Window, Figure, Editor e Help – serão usadas extensivamente ao longo deste livro e terão uma breve descrição nas páginas seguintes. Descrições mais detalhadas estão incluídas nos capítulos em que se faz uso específico de cada uma delas.

Command Window: A principal janela do MATLAB é a Command Window. É ativada sempre que o MATLAB é inicializado. Muitas vezes é conveniente mantê-la aberta sem as demais. Isso pode ser feito fechando-se todas as outras (clique no **x** no canto direito

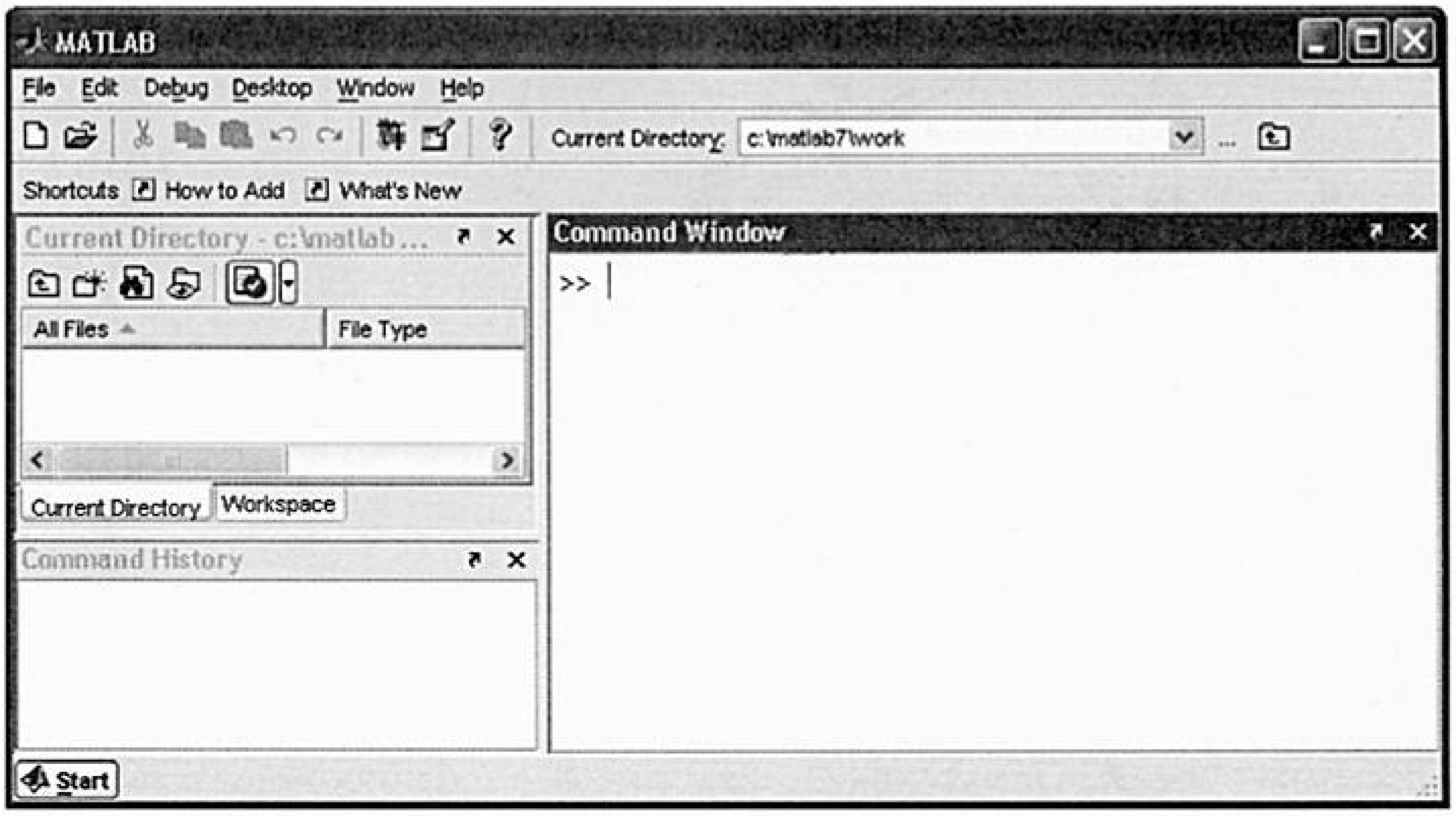


FIGURA 1-1 Desktop padrão do MATLAB.

TABELA 1-1 Janelas do MATLAB

Janela	Propósito
Command Window	Janela principal, inicializa variáveis e executa programas.
Figure	Apresenta o(s) resultado(s) dos comandos gráficos.
Editor	Permite a edição e a depuração de rotinas (*script files*) e funções.
Help	Ajuda na utilização do programa
Launch Pad	Fornece acesso às ferramentas, demos e documentação.
Command History	Apresenta o histórico dos comandos mais recentes digitados na janela Command Window.
Workspace	Disponibiliza informação sobre as variáveis que estão em uso.
Current Directory	Exibe os arquivos presentes no diretório ou na pasta atual.

superior da janela que você deseja fechar) ou, então, escolhendo a opção **Desktop Layout** no menu **Desktop** da barra de ferramentas e no submenu a opção **Command Window Only**. A Seção 1.2 descreve como trabalhar com a janela Command Window.

Janela Figure: A janela Figure é aberta toda vez que um comando gráfico é executado. Exibe o(s) gráfico(s) criado(s) por esse(s) comando(s). A Figura 1-2 mostra um exemplo gráfico na janela Figure. Uma descrição mais detalhada é apresentada no Capítulo 5.

Janela Editor: A janela Editor é usada para escrever e editar programas. É possível abri-la a partir do menu **File** na janela Command Window. Um exemplo de função na janela Editor é mostrada na Figura 1-3. Mais detalhes sobre essa janela serão apresentados no Capítulo 4 (criando rotinas e *scripts*) e no Capítulo 6 (criando funções).

Janela Help: Contém um descritivo detalhado do Help. É possível abri-la a partir do menu **Help** na barra de ferramentas de qualquer janela do MATLAB. Ela é interativa e pode ser utilizada para obtenção de informação sobre qualquer característica do MATLAB. A Figura 1-4 mostra uma janela típica do Help.

A primeira inicialização do MATLAB exibirá telas semelhantes à da Figura 1-1. Para a maioria dos iniciantes, pode ser conveniente fechar todas as janelas, exceto a Command Window. Não se preocupe, as janelas fechadas podem ser reabertas selecionando-as novamente no menu **Desktop**. Por fim, as janelas mostradas na Figura 1-1 po-

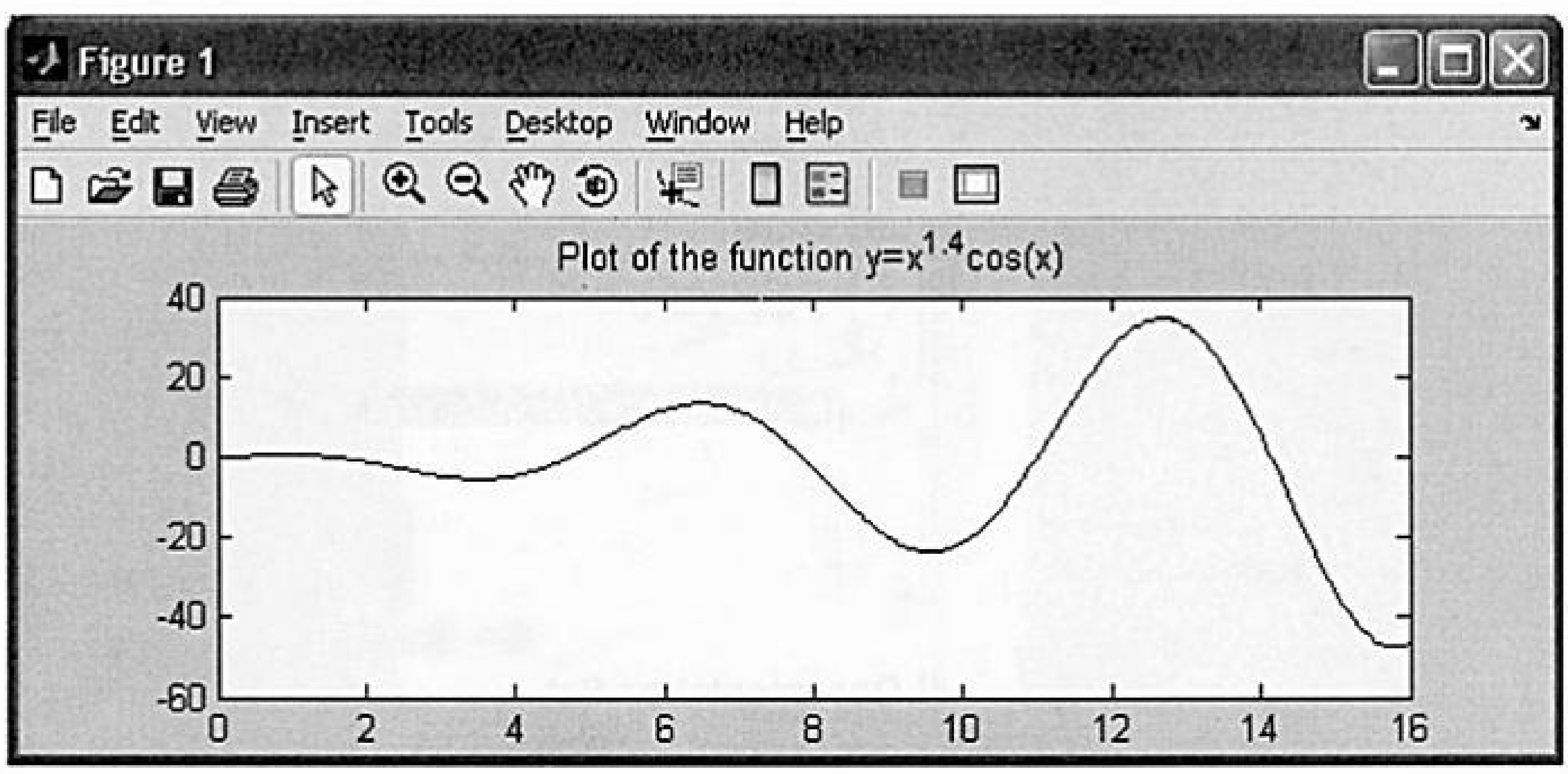


FIGURA 1-2 Exemplo de uma função na janela Figure.

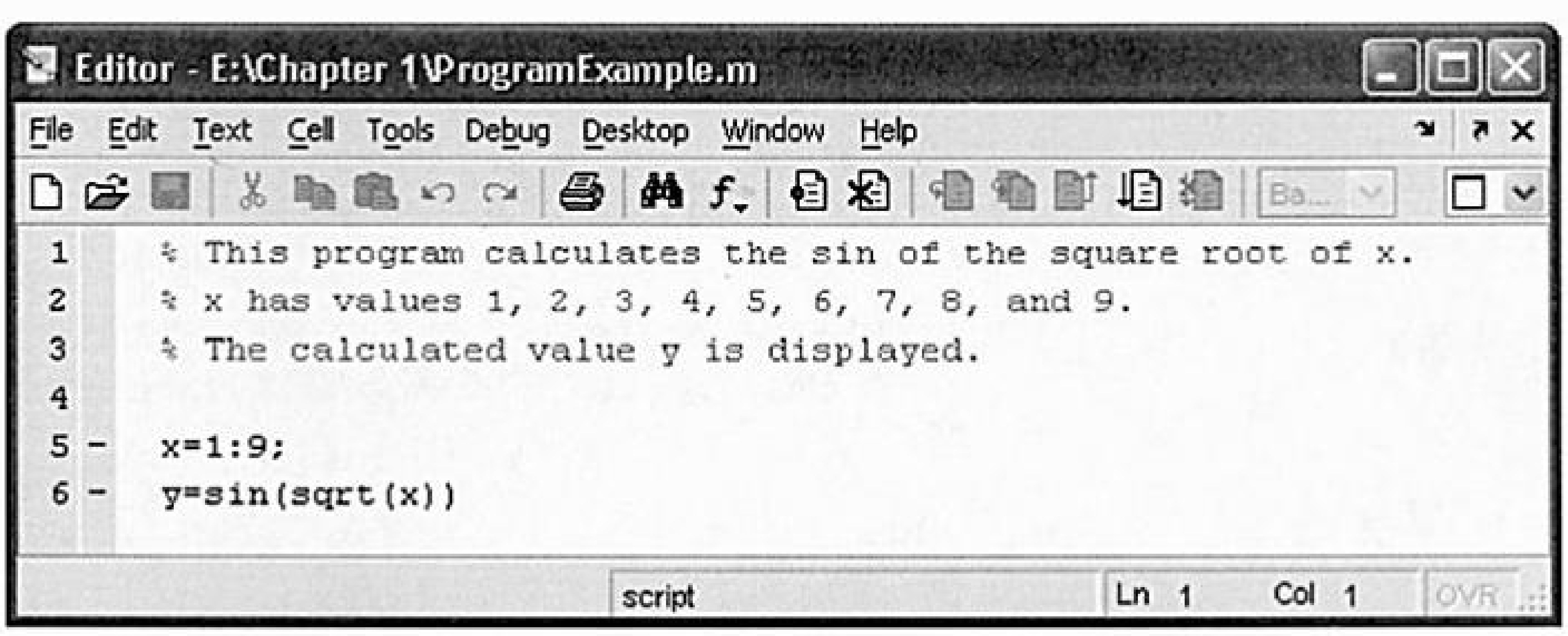


FIGURA 1-3 Exemplo de aplicação na janela Editor.

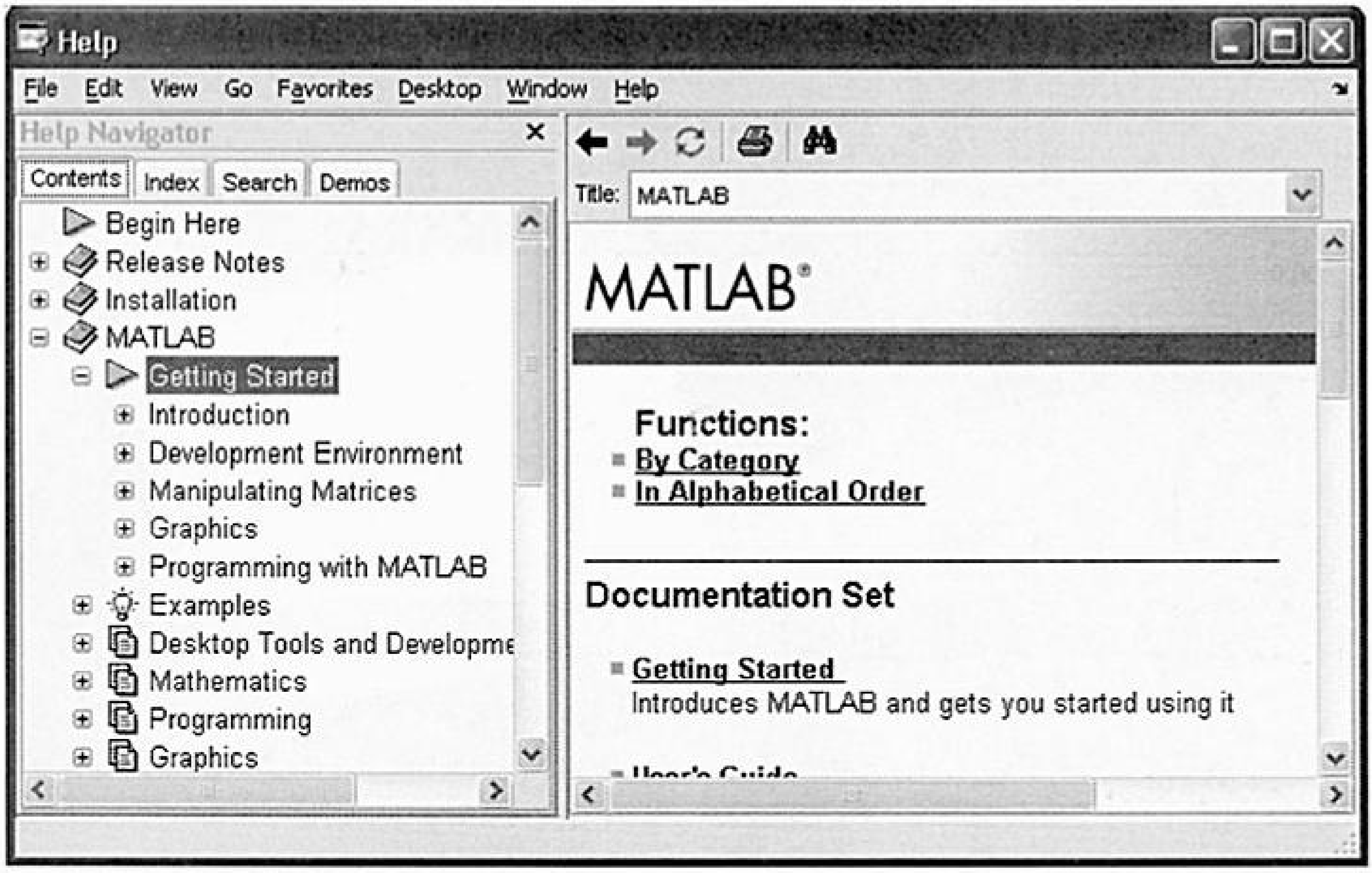


FIGURA 1-4 O Help do MATLAB.

dem ser exibidas selecionando-as primeiramente no **Desktop Layout** no menu **Desktop** e, a seguir, na opção **Default** do submenu.

1.2 TRABALHANDO COM A JANELA COMMAND WINDOW

Conforme mencionado, a principal janela do MATLAB é a Command Window. Ela pode ser usada para executar comandos, abrir outras janelas, rodar programas escritos pelo usuário e/ou gerenciar o uso do MATLAB. Um exemplo simples de uso da janela Command Window é mostrado na Figura 1-5.

Observações quanto ao uso da janela Command Window:

- Para digitar um comando, o cursor deve ser posicionado junto ao *prompt* de comando (>>).
- Uma vez que o comando foi digitado e a tecla **Enter** foi pressionada, o comando é executado. Contudo, somente o último comando é executado. Qualquer comando digitado anteriormente torna-se inacessível (a menos que seja reescrito ou chamado novamente).
- Muitos comandos podem ser digitados na mesma linha. Isso pode ser feito separando-os por vírgulas. Quando a tecla **Enter** é pressionada, os comandos são executados na ordem em que foram digitados, sucessivamente da esquerda para a direita.
- Não é possível retornar à última linha exibida na janela Command Window para fazer uma correção (edição) e, então, executar o comando novamente, produzindo um resultado nessa mesma linha.

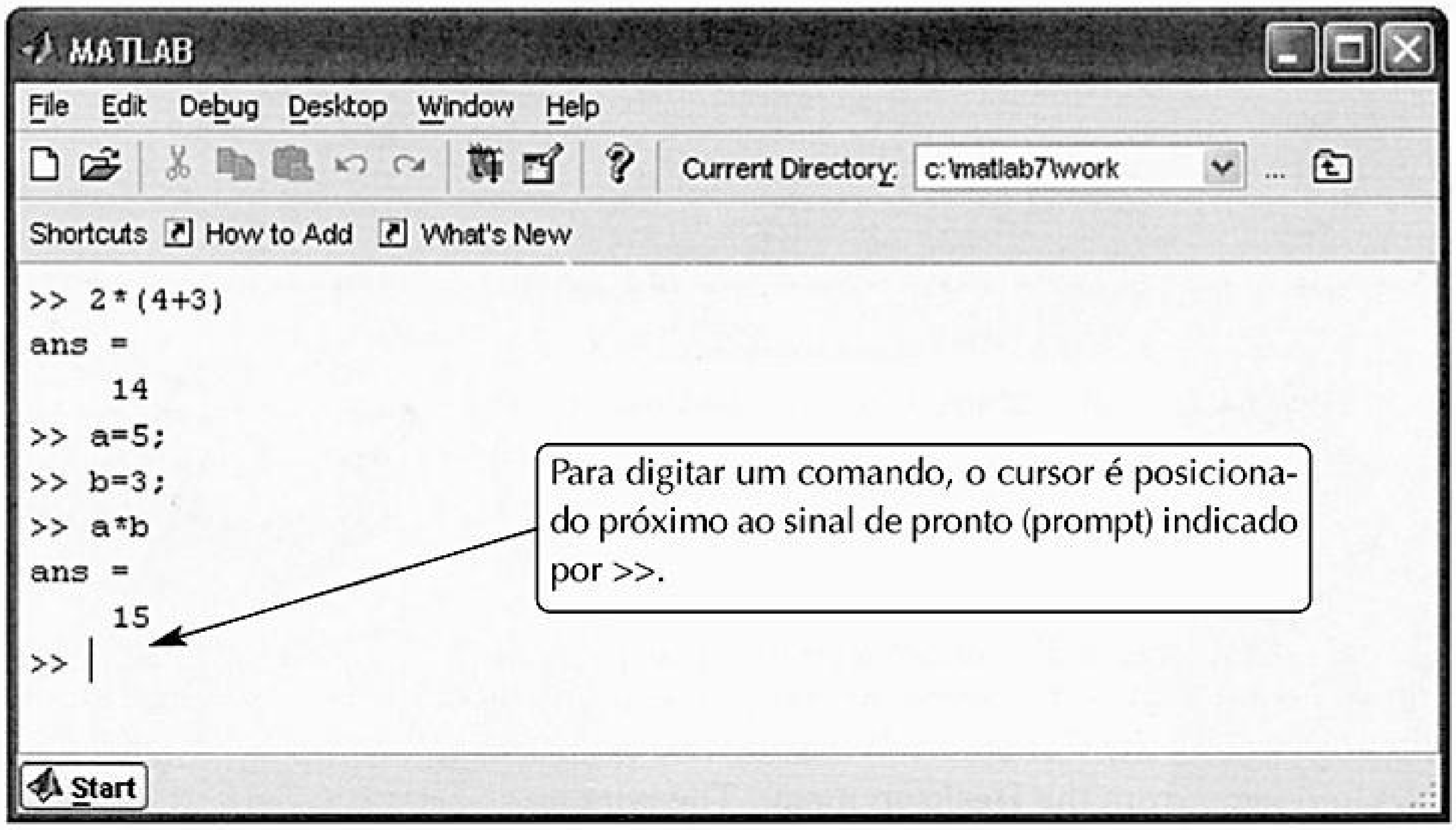


FIGURA 1-5 A janela Command Window.

- Um comando anteriormente digitado pode ser chamado outra vez através das teclas de navegação ($\uparrow$) e ($\downarrow$). Assim que o comando desejado for exibido no *prompt* de comando, é possível modificá-lo (se necessário) e executá-lo.
- Se um comando for extenso demais para caber em uma linha, basta digitar reticências (...) e pressionar a tecla **Enter** para continuar na próxima linha. A continuação de um comando no MATLAB pode ocupar 4.096 caracteres após a linha inicial.

O ponto-e-vírgula (;):

Um comando é executado quando é digitado na janela Command Window e a tecla **Enter** é pressionada. Qualquer resultado que o comando produza é mostrado no Command Window. O resultado de saída será ocultado se um ponto-e-vírgula (;) for digitado no final do comando. Digitar um ponto-e-vírgula é interessante quando o resultado da operação é óbvio ou conhecido, ou então quando a saída é grande demais.

Se múltiplos comandos forem digitados na mesma linha, havendo ponto-e-vírgula entre eles (em vez de uma vírgula), nenhum resultado será exibido.

Digitando %:

Quando o símbolo de percentagem (%) é digitado no início de uma linha de comando, toda a linha é designada como um comentário. Significa que, quando a tecla **Enter** for pressionada, a linha não será executada. O caractere %, seguido de um comentário, também pode ser digitado após um comando (na mesma linha). Isso não produzirá efeitos na execução desse comando.

A janela de Command Window não possui comentário padrão. Porém, comentários são freqüentemente usados em programas para adicionar descrições ou explicar algum ponto particular sobre programa (veja Capítulos 4 e 6).

O comando* `clc`*:

O comando `clc` limpa os últimos resultados exibidos na janela Command Window. Muitas vezes, a janela Command Window mostra resultados que já não são úteis ou que podem trazer algum tipo de dispersão na análise. Assim que o comando `clc` é digitado e confirmado, todos os resultados são apagados da saída. Entretanto, o comando não altera o que foi executado anteriormente. Por exemplo, se foram declaradas variáveis antes do comando `clc` (veja Seção 1.6), elas continuam existindo após esse comando e podem ser utilizadas. Outra vez, as teclas de navegação (↑ ou ↓) podem ser acionadas para chamar comandos digitados anteriormente.

1.3 OPERAÇÕES ARITMÉTICAS COM ESCALARES

Neste capítulo, discutiremos apenas operações aritméticas com escalares (números). Conforme será explicado adiante, números podem ser usados diretamente em cálculos aritméticos (como em uma calculadora) ou então podem ser atribuídos às variáveis a serem empregadas em cálculos futuros. Os símbolos para operações aritméticas são:

Operação	Símbolo	Exemplo
Adição	+	5 + 3
Subtração	–	5 – 3
Multiplicação	*	5 * 3
Divisão à direita	/	5 / 3
Divisão à esquerda	\	5 \ 3 = 3 / 5
Exponenciação	^	5 ^ 3 (ou seja, $5^3 = 125$)

Observe que todos os símbolos, exceto o da divisão à esquerda, são os mesmos encontrados na maioria das calculadoras. Em se tratando de escalares, a divisão à esquerda é a operação inversa da divisão à direita. Entretanto, a divisão à esquerda encontra maior utilidade em operações com matrizes (que examinaremos no Capítulo 3).

1.3.1 Ordem de precedência

O MATLAB executa os cálculos de acordo com a ordem de precedência mostrada a seguir. Essa ordem é a mesma utilizada na maioria das calculadoras.

Precedência	Operação matemática
Mais alta	Parênteses. Quando ocorrem parênteses aninhados, os parênteses mais internos são executados em primeiro lugar.
Segunda	Exponenciação
Terceira	Multiplicação, divisão (mesma precedência)
Mais baixa	Adição e subtração

Nas expressões que possuírem muitas operações, as operações com maior ordem de precedência serão executadas em primeiro lugar. Se duas ou mais operações tiverem a mesma ordem de precedência, a expressão mais à esquerda será executada primeiro. A próxima seção ilustra como os parênteses podem ser utilizados para mudar a ordem de precedência nos cálculos.

1.3.2 Utilizando o MATLAB como uma calculadora

O modo mais simples de utilizar o MATLAB é como uma calculadora. Isso pode ser feito digitando-se uma expressão matemática no *prompt* da janela Command Window e pressionando a tecla **Enter**. O MATLAB calcula a expressão, emite uma resposta do tipo `ans =` e exibe o resultado numérico da expressão nas linhas seguintes. Esses procedimentos são demonstrados passo a passo no Tutorial 1-1:

TUTORIAL 1-1 Utilizando o MATLAB como uma calculadora

```
>> 7 + 8/2
ans =
   11
>> (7 + 8)/2
ans =
   7.5000
>> 4 + 5/3 + 2
ans =
   7.6667
>> 5^3/2
ans =
   62.5000
>> 27^(1/3) + 32^0.2
ans =
   5
>> 27^1/3 + 32^0.2
ans =
   11
>> 0.7854 – (0.7854)^3/(1*2*3) + 0.785^5/(1*2*3*4*5)...
– (0.785)^7/(1*2*3*4*5*6*7)
ans =
   0.7071
>>
```

Digite e pressione **Enter**.

8/2 é executado primeiro.

Digite e pressione **Enter**.

7 + 8 é executado primeiro.

5^3 é executado primeiro.

5^3 é executado primeiro, /2 é executado em seguida.

1/3 é executado primeiro, 27^(1/3) e 32^0.2 são executados em seguida e "+" é executado por último.

27^1 e 32^0.2 são executados inicialmente, /3 é executado em seguida e "+" é executado por último.

Digite reticências... (e pressione **Enter**) para continuar a expressão na linha seguinte.

A última expressão contém os quatro primeiros termos da série de Taylor para sin(π/4).

1.4 FORMATANDO DADOS NUMÉRICOS

O usuário pode controlar o formato segundo o qual o MATLAB exibe os dados de saída na tela. No Tutorial 1-1, o formato de saída é o ponto fixo com quatro dígitos decimais (chamado `short`), que é o formato padrão para valores numéricos. O formato de saída pode ser modificado através do comando `format`. Uma vez digitado um comando `format`, todos os resultados exibidos na saída seguem esse formato específico. Muitos formatos disponíveis no MATLAB estão listados e descritos na Tabela 1-2.

O MATLAB possui diversos outros formatos para dados numéricos. Mais informações sobre esses formatos podem ser obtidas digitando-se `help format` na linha do *prompt* do Command Window. O formato escolhido para os dados numéricos não afeta a maneira como o MATLAB calcula e salva os números.

TABELA 1-2 Formatos padronizados dos números

Comando	Descrição	Exemplo
`format short`	Notação ponto fixo com 4 dígitos decimais para: 0.001 ≤ *número* ≤ 1000 Caso contrário, formato do número `short e`.	>> 290/7 ans = 41.4286
`format long`	Notação ponto fixo com 14 dígitos decimais para: 0.001 ≤ *número* ≤ 1000 Caso contrário, formato do número `long e`.	>> 290/7 ans = 41.42857142857143
`format short e`	Notação científica com 4 dígitos decimais.	>> 290/7 ans = 4.1429e+001
`format long e`	Notação científica com 15 dígitos decimais.	>> 290/7 ans = 4.142857142857143e+001
`format short g`	Melhor em 5 dígitos entre a notação de ponto fixo ou ponto flutuante.	> 290/7 ans = 41.429
`format long g`	Melhor em 15 dígitos entre a notação de ponto fixo ou ponto flutuante.	>> 290/7 ans = 41.4285714285714

TABELA 1-2 Formatos padronizados dos números *(continuação)*

Comando	Descrição	Exemplo
`format bank`	Dois dígitos decimais.	>> 290/7 ans = 41.43
`format compact`	Elimina espaços para permitir que mais linhas de informação sejam mostradas na tela.	
`format loose`	Adiciona espaços entre linhas (oposto ao formato `compact`).	

1.5 FUNÇÕES ELEMENTARES NATIVAS DO MATLAB

Expressões no MATLAB podem incluir funções, além das operações aritméticas básicas. O MATLAB possui uma biblioteca extensa de funções nativas. Uma função é caracterizada por um nome e um argumento entre parênteses. Por exemplo, a função que calcula a raiz quadrada de um número é `sqrt(x)`. O nome da função é `sqrt` e o argumento é `x`. O argumento de uma função pode ser um número, uma variável (veja Seção 1.6) ou uma expressão composta de números e/ou variáveis. Funções também podem ser incluídas no argumento de outras funções, assim como em expressões. O Tutorial 1-2 ensina como usar a função `sqrt(x)` operando com escalares.

Listas de funções elementares nativas do MATLAB são mostradas nas Tabelas 1-3, 1-4 e 1-5. Uma lista completa de funções, organizadas pelo nome da categoria a que pertencem, pode ser encontrada em Help Window.

TUTORIAL 1-2 Usando a função nativa `sqrt`

```
>> sqrt(64)                          O argumento é um número.
ans =
   8
>> sqrt(50 + 14*3)                   O argumento é uma expressão.
ans =
   9.5917
>> sqrt(54 + 9*sqrt(100))            O argumento inclui uma função.
ans =
   12
>> (15 + 600/4)/sqrt(121)            Uma função é incluída em uma expressão.
ans =
   15
>>
```

TABELA 1-3 Funções matemáticas elementares

Função	Descrição	Exemplo
`sqrt(x)`	Raiz quadrada	>> sqrt(81) ans = 9
`exp(x)`	Exponencial (e^x)	>> exp(5) ans = 148.4132
`abs(x)`	Valor absoluto (módulo)	>> abs(-24) ans = 24
`log(x)`	Logaritmo neperiano (ln)	>> log(1000) ans = 6.9078
`log10(x)`	Logaritmo base 10	>> log10(1000) ans = 3
`factorial(x)`	Fatorial de *x* (*x*!) (*x* deve ser um inteiro positivo)	>> factorial(5) ans = 120

TABELA 1-4 Funções trigonométricas

Função	Descrição	Exemplo
`sin(x)`	Seno do argumento *x* (*x* em radianos)	>> sin(pi/6) ans = 0.5000
`cos(x)`	Co-seno do argumento *x* (*x* em radianos)	>> cos(pi/6) ans = 0.8660
`tan(x)`	Tangente do argumento *x* (*x* em radianos)	>> tan(pi/6) ans = 0.5774
`cot(x)`	Cotangente do argumento *x* (*x* em radianos)	>> cot(pi/6) ans = 1.7321

As funções trigonométricas inversas são: `asin(x)`, `acos(x)`, `atan(x)` e `acot(x)`. Já as funções trigonométricas hiperbólicas são: `sinh(x)`, `cosh(x)`, `tanh(x)` e `coth(x)`. A tabela acima usa `pi` representando π* (veja a Seção 1.6.3).

* N. de T.: Essa representação é apenas simbólica. Numericamente, π é um número irracional. Nenhum programa de computador é capaz de representá-lo em sua totalidade.

TABELA 1-5 Funções de arredondamento

Função	Descrição	Exemplo
`round(x)`	Arredonda para o inteiro mais próximo.	>> round(17/5) ans = 3
`fix(x)`	Arredonda para o inteiro positivo imediatamente menor.	>> fix(13/5) ans = 2
`ceil(x)`	Arredonda para o inteiro positivo imediatamente maior.	>> ceil(11/5) ans = 3
`floor(x)`	Arredonda para o inteiro negativo imediatamente menor.	>> floor (-9/4) ans = –3
`rem(x,y)`	Retorna o resto da divisão x por y.	>> rem(13,5) ans = 3
`sign(x)`	Função sinal. Retorna: 1 (se $x > 0$); –1 (se $x < 0$) e 0 (se $x = 0$).	>> sign(5) ans = 1

1.6 DECLARANDO VARIÁVEIS ESCALARES

Uma variável escalar no MATLAB é um nome formado por uma letra ou uma cadeia de letras (e números) ao qual é atribuído um valor. Uma vez atribuído um valor numérico à variável, podemos usá-la em expressões matemáticas, em funções, em sentenças e comandos do MATLAB. Tecnicamente, uma variável é um espaço de memória reservado para armazenar um certo tipo de dado, tendo um nome para referenciar o seu conteúdo. Quando uma variável é declarada, o MATLAB aloca um espaço de memória onde o conteúdo da variável é armazenado. Ao se utilizar a variável, automaticamente o conteúdo (dado armazenado) é passado ao comando ou à sentença que fará uso dela. Se for atribuído um novo valor à variável, o conteúdo da posição de memória será substituído. No Capítulo 1, atribuiremos somente valores numéricos (escalares) às variáveis. A atribuição e o endereçamento de arranjos – matrizes e vetores – serão discutidos no Capítulo 2.

1.6.1 O operador de atribuição

No MATLAB, o sinal de igualdade (=) é denominado operador de atribuição. O operador de atribuição inicializa ou modifica o valor de uma variável.

Nome_variável = Um valor ou uma expressão numérica

- O lado esquerdo do operador de atribuição pode conter apenas uma variável. O lado direito pode ser um número ou uma expressão numérica, possuindo números e/ou variáveis previamente inicializadas. Teclando-se **Enter**, o resultado ou o número à direita do operador é atribuído à variável no lado esquerdo e o MATLAB exibe a variável com o seu respectivo valor nas duas próximas linhas.

Os exemplos a seguir mostram como o operador de atribuição funciona:

```
>> x = 15
x =
   15

>> x = 3*x - 12
x =
   33
>>
```

O número 15 é atribuído à variável x.

O MATLAB exibe a variável e o valor atribuído.

Um novo valor é atribuído à variável x. O novo valor é 3 vezes o valor anterior de x menos 12.

A última sentença ($x = 3x - 12$) ilustra a diferença entre o operador de atribuição e o sinal de igualdade. Se, nessa sentença, o sinal (=) significasse igualdade, o valor de x deveria ser 6 (resolvendo-se a equação para x).

A utilização de variáveis previamente declaradas para definir uma nova variável é demonstrada a seguir:

```
>> a = 12
a =
   12
>> B = 4
B =
   4
>> C = (a - B) + 40 - a/B*10
C =
   18
>>
```

Atribui 12 à variável a.

Atribui 4 à variável B.

Atribui o valor da expressão no lado direito do sinal (=) à variável C.

- Se um ponto-e-vírgula é digitado no fim de um comando, quando a tecla **Enter** for pressionada o MATLAB não exibirá o valor atribuído à variável (a variável é inicializada e encontra-se armazenada na memória).
- Qualquer variável declarada pode ter seu valor exibido a qualquer momento simplesmente digitando-se o nome da variável e pressionando-se **Enter**.

Por exemplo, se a última demonstração é repetida usando-se ponto-e-vírgula:

```
>> a = 12;
>> B = 4;
>> C = (a – B) + 40 – a/B*10;
>> C
C =
   18
>>
```

As variáveis a, B e C foram declaradas e inicializadas, mas não tiveram o conteúdo exibido em virtude do ponto-e-vírgula em cada linha.

O valor da variável C é exibido se o nome da variável é digitado e a tecla ENTER é pressionada.

- Muitas atribuições podem ser feitas numa mesma linha. Cada atribuição deve ser separada com uma vírgula ou ponto-e-vírgula, conforme conveniência (espaços também podem ser adicionados após uma vírgula ou ponto-e-vírgula). Pressionando a tecla **Enter**, as atribuições são efetuadas da esquerda para a direita e as variáveis com os respectivos valores são exibidas na tela (exceto aquelas terminadas em ponto-e-vírgula). Por exemplo, as atribuições das variáveis a, B e C acima podem ser feitas na mesma linha.

```
>> a = 12, B = 4; C = (a – B) + 40 – a/B*10
a =
   12
C =
   18
```

A variável B e o conteúdo dela não são exibidos porque um ponto-e-vírgula foi digitado após a inicialização de B.

- Uma variável previamente declarada pode ter seu conteúdo modificado. Por exemplo:

```
>> ABB = 72;
>> ABB = 9;
>> ABB
ABB =
   9
>>
```

O número 72 é atribuído à variável ABB.

Um novo número (9) é atribuído à variável ABB.

O valor real da variável ABB é exibido se o nome da variável é digitado na linha do *prompt* e a tecla **Enter** é pressionada.

- Uma vez declarada a variável, ela pode ser utilizada no argumento de funções. Por exemplo:

```
>> x = 0.75;
>> E = sin(x)^2 + cos(x)^2
E =
   1
>>
```

1.6.2 Regras quanto ao uso de nomes de variáveis

Os nomes das variáveis:

- Podem conter até 63 caracteres no MATLAB 7.0 (31 caracteres no MATLAB 6.0).
- Podem conter letras, números e o caractere sublinhar (*underscore*).
- Devem iniciar com uma letra.
- O MATLAB é *case sensitive*, ou seja, faz distinção entre nomes de variáveis com letras maiúsculas e minúsculas. Por exemplo: `AA`, `Aa`, `aA` e `aa` podem ser nomes de quatro variáveis diferentes.
- Evite usar nomes de funções nativas do MATLAB para nomear variáveis (por exemplo: `cos`, `sin`, `exp`, `sqrt`, etc). Uma vez que o nome de uma função é utilizado para definir uma variável, essa função não pode mais ser usada.

1.6.3 Variáveis predefinidas

Ao inicializar o uso do MATLAB, uma certa quantidade de variáveis é carregada automaticamente na memória do programa. Por exemplo:

`ans` Variável que assume o valor da última expressão não atribuída a uma variável especificada (veja Tutorial 1-1). Se o usuário não atribui o valor de uma expressão a uma variável, o MATLAB armazena, automaticamente, o resultado em `ans`.

`pi` O número π (aproximação).

`eps` A menor diferença entre dois números. Equivale a 2^(−52) ou, aproximadamente, 2,2204e−16.

`inf` Infinito.

`i` Definido como $\sqrt{-1}$, isto é: 0 + 1.0000i.

`j` O mesmo que i.

`NaN` Abreviação de Not-a-Number. Usado quando o MATLAB não pode determinar um valor numérico válido. Por exemplo, 0/0.

As variáveis predefinidas podem ser redefinidas a qualquer momento. É recomendado que as variáveis `pi`, `eps` e `inf` não sejam redefinidas, porque muitas aplicações fazem uso delas. Outras variáveis, como `i` e `j`, às vezes são redefinidas quando os números complexos não estão envolvidos numa aplicação particular (como acontece na linguagem de programação C/C++, dentre outras, `i` e `j` freqüentemente são associados aos índices de contagem em laços – *loops* – de programas).

1.7 COMANDOS ÚTEIS NO MANUSEIO DE VARIÁVEIS

A seguir são mostrados alguns comandos usados para eliminar variáveis ou obter informação a respeito das variáveis declaradas dentro de um sessão do MATLAB. Quando esses comandos são digitados na janela Command Window e a tecla **Enter** é pressionada, eles fornecem a informação desejada ou realizam uma tarefa, conforme discriminado a seguir:

Comando	Resultado
`clear`	Apaga da memória todas as variáveis.
`clear x y z`	Apaga da memória somente as variáveis `x`, `y` e `z`.
`who`	Exibe uma lista de variáveis declaradas/ativas na memória.
`whos`	Exibe uma lista de variáveis declaradas na memória, com o respectivo tamanho em *bytes* e a classe de armazenamento.

1.8 EXEMPLOS DE APLICAÇÕES DE MATLAB

Problema-exemplo 1-1: Identidade trigonométrica

Uma identidade trigonométrica é dada por:

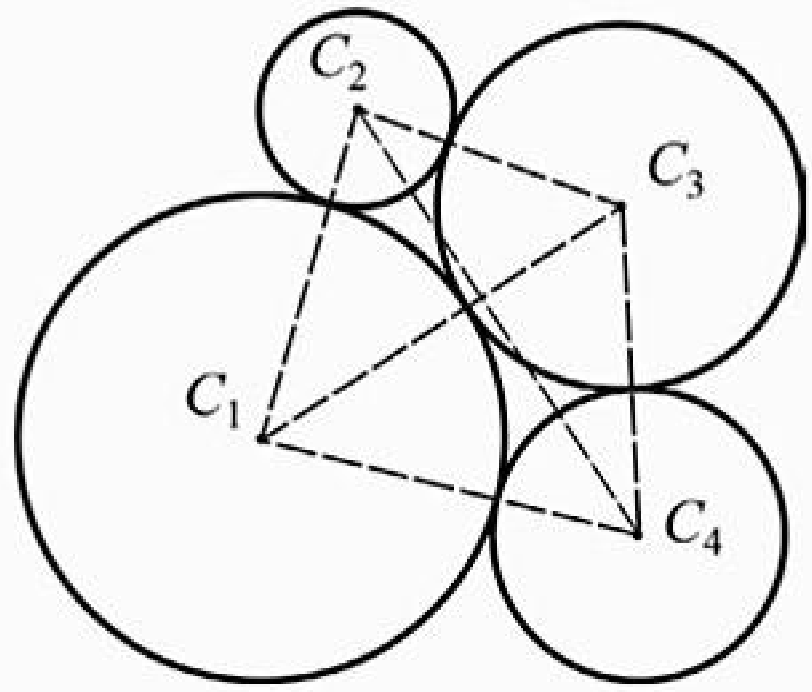
$$\cos^2\frac{x}{2} = \frac{\tan x + \sin x}{2\tan x}$$

Substituindo $x = \frac{\pi}{5}$, verifique a identidade calculando cada lado da equação.

Solução

```
>> x = pi/5;                                   Declara e inicializa x.
>> LE = cos(x/2)^2                             Calcula o lado esquerdo (LE) da identidade.
LE =
   0.9045
>> LD = (tan(x) + sin(x))/(2*tan(x))           Calcula o lado direito (LD) da identidade.
LD =
   0.9045
>>
```

Problema-exemplo 1-2: Geometria e trigonometria

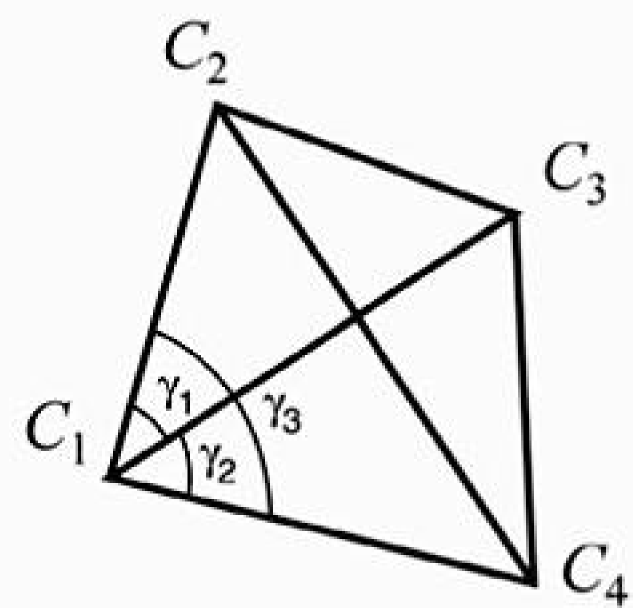


Quatro círculos estão dispostos como mostra a figura. Os círculos tangenciam-se dois a dois num determinado ponto. A partir daí, determine a distância entre os centros C_2 e C_4.

Os raios dos círculos são: $R_1 = 16$mm, $R_2 = 6.5$mm, $R_3 = 12$mm e $R_4 = 9.5$mm.

Solução

As retas que ligam os centros dos círculos geram quatro triângulos. Em dois desses triângulos, $\Delta C_1C_2C_3$ e $\Delta C_1C_3C_4$, os comprimentos de todos os lados são conhecidos. Essa informação é usada para calcular os ângulos γ_1 e γ_2, por meio da lei dos co-senos. Por exemplo, γ_1 é calculado a partir de:



$$(C_2C_3)^2 = (C_1C_2)^2 + (C_1C_3)^2 - 2(C_1C_2)(C_1C_3)\cos\gamma_1$$

Em seguida, o comprimento do lado C_2C_4 é calculado tomando-se como base o triângulo $\Delta C_1C_2C_4$. Para tanto, basta usar novamente a lei dos co-senos (os comprimentos dos lados C_1C_2 e C_1C_4 são conhecidos e o ângulo γ_3 é a soma dos ângulos γ_1 e γ_2).

```
>> R1 = 16; R2 = 6.5; R3 = 12; R4 = 9.5;
>> C1C2 = R1 + R2; C1C3= R1 + R3;
>> C1C4 = R1 + R4; C2C3 = R2 + R3;
>> C3C4 = R3 + R4;
>> gama1 = acos((C1C2^2 + C1C3^2 – C2C3^2)/(2*C1C2*C1C3));
>> gama2 = acos((C1C3^2 + C1C4^2 – C3C4^2)/(2*C1C3*C1C4));
>> gama3 = gama1 + gama2;

>> C2C4 = sqrt (C1C2^2 + C1C4^2 – 2*C1C2*C1C4*cos(gama3))
C2C4 =
   33.5051
```

Declara e inicializa os *R*s.

Determina os comprimentos dos lados.

Determina γ_1, γ_2 e γ_3.

Determina o comprimento do lado C_2C_4.

Problema-exemplo 1-3: Transferência de calor

Um objeto que está a uma temperatura inicial T_0 é colocado em $t = 0$ dentro de uma câmara à temperatura constante (T_s). A mudança na temperatura do objeto segue a lei:

$$T = T_s + (T_0 - T_s)\, e^{-k\,t}$$

onde T é a temperatura do objeto em um tempo t qualquer e k é uma constante. De posse dessas informações, considere uma lata de refrigerante exposta a uma temperatura 48.9°C. Em seguida, é colocada dentro de um refrigerador onde a temperatura é 3.3°C. Determine a temperatura da lata em graus Celsius, em valores inteiros, três horas após a lata ser colocada no refrigerador. Considere $k = 0.45$. Declare inicialmente todas as variáveis e então calcule a temperatura usando apenas um comando do MATLAB.

Solução

```
>> Ts = 3.3; T0=48.9; k=0.45; t=3;
>> T= round (Ts + (T0-Ts)*exp(-k*t))
T =
   15
```

Arredonda para um valor inteiro de temperatura.

Problema-exemplo 1-4: Juros compostos

O saldo B de uma conta de poupança após t anos é dado por:

$$B = P\left(1 + \frac{r}{n}\right)^{nt} \qquad (1)$$

Investidas inicialmente P unidades monetárias, sendo a taxa de juros anual r e os juros compostos sendo calculados n vezes ao ano. Sendo os juros compostos calculados anualmente, o saldo é dado por:

$$B = P(1 + r)^t \qquad (2)$$

Em uma conta de poupança são depositados \$ 5,000.00, investidos durante 17 anos em um banco, com juros compostos anualmente. Em outra conta são depositados \$ 5,000.00, com juros compostos mensalmente. Em ambas as contas, a taxa de juros anual é 8.5%. Use o MATLAB para determinar em quanto tempo (em anos e meses) o saldo da segunda conta será idêntico ao saldo da primeira conta, decorridos os 17 anos.

Solução

Passos:

(*a*) Determine B, usando a Equação (2), para os \$5,000.00 investidos na conta onde os juros são compostos anualmente, durante 17 anos.

(*b*) Determine *t* para o saldo *B*, calculado na letra (*a*), a partir dos juros compostos mensalmente, Equação (1).

(*c*) Determine o número de anos e meses que correspondem ao intervalo de tempo *t*.

```
>> P = 5000; r = 0.085; ta = 17; n = 12;
>> B = P*(1 + r)^ta                          Passo (a): determina B a partir da equação (2).
B =
   2.0011e+004
>> t = log(B/P)/(n*log(1 + r/n))             Passo (b): resolve a equação (1) para t e determina o valor de t.
t =
   16.3737
>> anos = fix(t)                             Passo (c): determina o número de anos.
anos =
   16
>> meses = ceil((t – anos)*12)               Determina o número de meses.
meses =
   5
```

1.9 PROBLEMAS

Resolva os problemas a seguir utilizando a janela Command Window.

1. Calcule:

a) $\dfrac{35.7 \cdot 64 - 7^3}{45 + 5^2}$

b) $\dfrac{5}{4} \cdot 7 \cdot 6^2 + \dfrac{3^7}{(9^3 - 652)}$

2. Calcule:

a) $(2+7)^3 + \dfrac{273^{2/3}}{2} + \dfrac{55^2}{3}$

b) $2^3 + 7^3 + \dfrac{273^3}{2} + 55^{3/2}$

3. Calcule:

a) $\dfrac{3^7 \log(76)}{7^3 + 546} + \sqrt[3]{910}$

b) $43 \cdot \dfrac{(\sqrt[4]{250}+23)^2}{e^{(45-3^3)}}$

4. Calcule:

a) $\cos^2\left(\dfrac{5\pi}{6}\right)\sin\left(\dfrac{7\pi}{8}\right)^2 + \dfrac{\tan\left(\dfrac{\pi}{6}\ln 8\right)}{\sqrt{7}}$

b) $\cos\left(\dfrac{5\pi}{6}\right)^2 \sin^2\left(\dfrac{7\pi}{8}\right) + \dfrac{\tan\left(\dfrac{\pi \ln 8}{6}\right)}{7 \cdot \dfrac{5}{2}}$

5. Declare e inicialize a variável x como $x = 13.5$. Em seguida, determine:

a) $x^3 + 5x^2 - 26.7x - 52$

b) $\dfrac{\sqrt{14x^3}}{e^{3x}}$

c) $\log|x^2 - x^3|$

6. Declare e inicialize as variáveis x e z como $x = 9.6$ e $z = 8.1$. Em seguida, determine:

a) $xz^2 - \left(\dfrac{2z}{3x}\right)^{\frac{3}{5}}$

b) $\dfrac{443z}{2x^3} + \dfrac{e^{-xz}}{(x+z)}$

7. Declare e inicialize as variáveis a, b, c e d como:
$a = 15.62$, $b = -7.08$, $c = 62.5$ e $d = 0.5(ab - c)$.

Em seguida, calcule:

a) $a + \dfrac{ab}{c}\dfrac{(a+d)^2}{\sqrt{|ab|}}$

b) $de^{\left(\frac{d}{2}\right)} + \dfrac{\dfrac{ad+cd}{\dfrac{20}{a}+\dfrac{30}{b}}}{(a+b+c+d)}$

8. Calcule (usando apenas um comando) o raio r de uma esfera de volume 5.74e-3m^3. Uma vez conhecido o valor de r, use-o para determinar a área da superfície da esfera.

9. Duas identidades trigonométricas são definidas como:

a) $\sin 2x = 2\sin x \cos x$

b) $\cos\frac{x}{2} = \sqrt{\frac{1+\cos x}{2}}$

Considerando $x = \frac{5}{24}\pi$, verifique cada uma das identidades calculando os dois lados da equação.

10. Duas identidades trigonométricas são definidas como:

a) $\tan 2x = \frac{2\tan x}{1-\tan^2 x}$

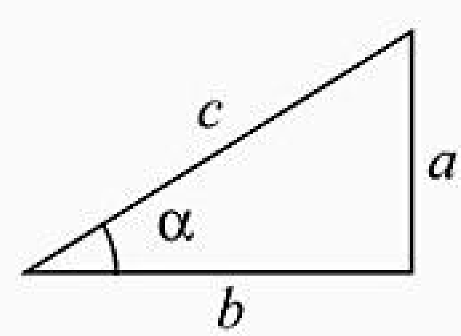
b) $\tan\frac{x}{2} = \sqrt{\frac{1-\cos x}{1+\cos x}}$

Considerando $x = \frac{3}{17}\pi$, verifique cada uma das identidades calculando os dois lados da equação.

11. Declare duas variáveis: *alfa* = 5π/9 e *beta* = π/7. Usando essas variáveis, mostre (calculando os dois lados da igualdade) que a identidade trigonométrica a seguir está correta.

$$\cos\alpha - \cos\beta = 2\sin\frac{1}{2}(\alpha+\beta)\sin\frac{1}{2}(\beta-\alpha)$$

12. No triângulo retângulo mostrado na figura ao lado, a = 11cm e c = 21cm. Declare as variáveis a e c e então:

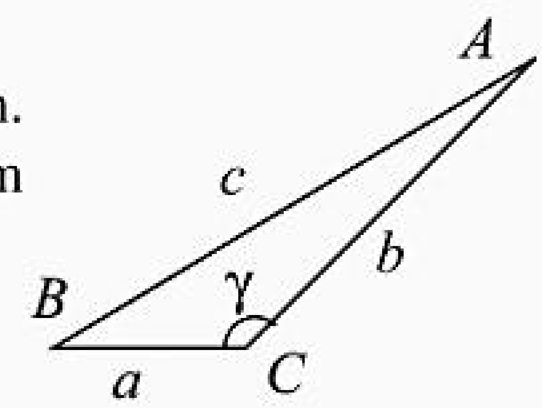


a) Usando o Teorema de Pitágoras, calcule b digitando apenas uma linha de comando.

b) Usando o valor de b encontrado em (*a*) e a função `acos(x)`, determine o ângulo α em graus, utilizando apenas uma linha de comando.

13. No triângulo mostrado, a = 18cm, b = 35cm e c = 50cm. Declare as variáveis a, b e c para calcular o ângulo γ (em graus) substituindo as variáveis na lei dos co-senos.

(Lei dos Co-senos: $c^2 = a^2 + b^2 - 2ab\cos\gamma$)



14. A distância d de um ponto (x_0, y_0) à reta $Ax + By + C = 0$ é dada por:

$$d = \frac{|Ax_0 + By_0 + C|}{\sqrt{A^2 + B^2}}$$

Determine a distância do ponto (2, −3) à reta $3x + 5y - 6 = 0$. Primeiramente, declare as variáveis A, B, C, x_0 e y_0, então calcule d. (Use as funções `abs` e `sqrt`.)

15. Flores são embaladas em caixas com capacidade para acomodar uma dúzia delas. Usando a função `ceil`, determine quantas caixas serão necessárias para embalar 751 flores.

16. Declare as seguintes variáveis:

preco_mesa = $ 256.95
preco_cadeira = $ 89.99

Mude o formato da janela Command Window para *bank* e:

a) Determine o preço total de duas mesas e oito cadeiras.
b) Repita o item (*a*), mas adicione 5.5% de juros.
c) Repita o item (*b*), mas arredonde o preço total em reais.

17. Um modo de adicionar frações é utilizando um denominador comum (no caso, o menor). Por exemplo, o menor denominador comum entre 1/4 e 1/10 é 20. Use a janela Help Window do MATLAB para determinar a função que determina o menor múltiplo comum de dois números. Em seguida, use essa função para mostrar que o menor múltiplo comum entre:

a) 4 e 10 é 20.
b) 6 e 38 é 114.

18. A intensidade M de um terremoto na escala Richter é dada por: $M = \frac{2}{3}\log\frac{E}{E_0}$, onde E é a energia liberada pelo tremor e $E_0 = 10^{4.4}$ Joules é uma constante (energia liberada em um tremor de terra de referência). Determine quantas vezes a energia liberada em um terremoto de 7.2 na escala Richter é maior do que um terremoto 5.3 nessa mesma escala.