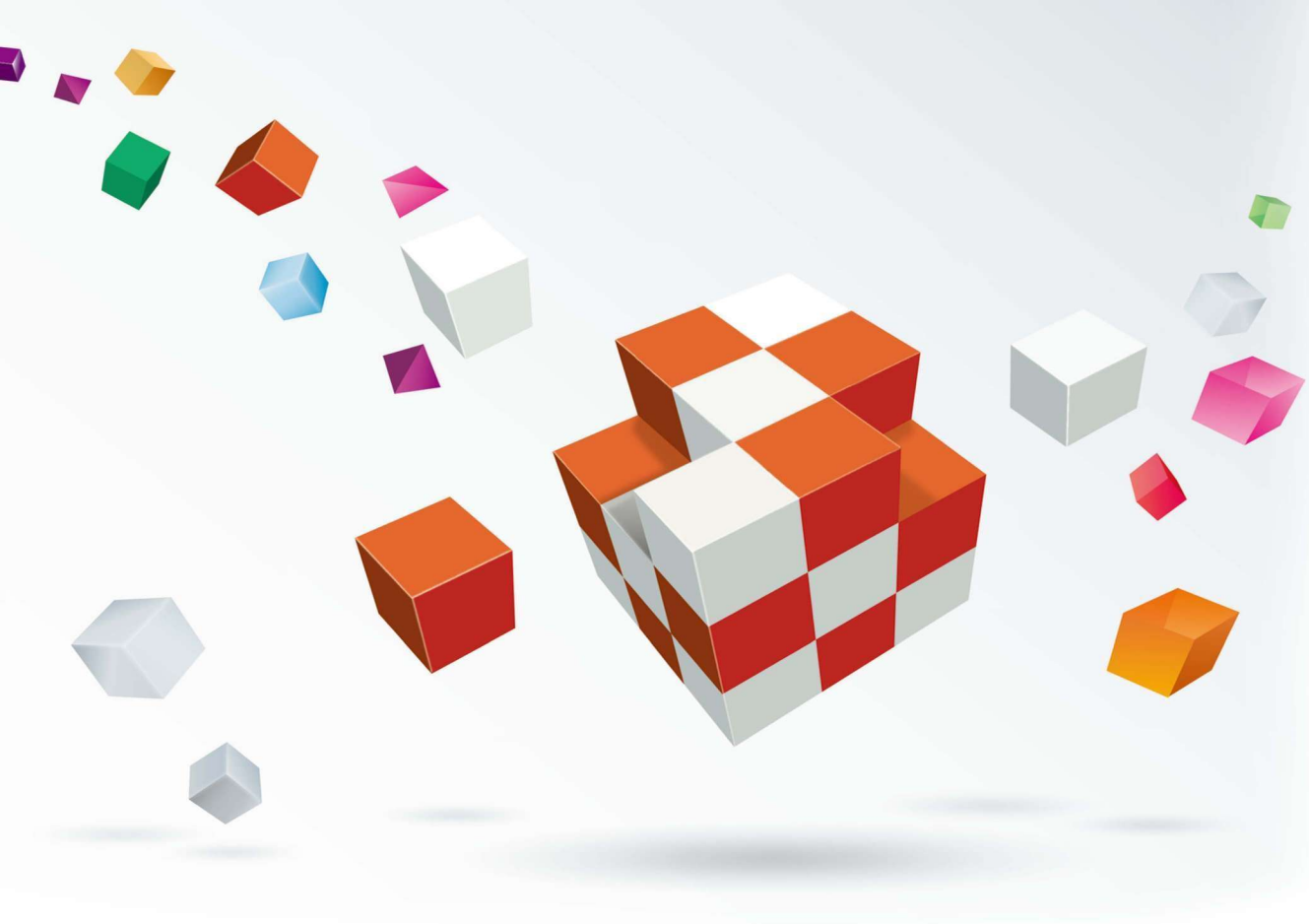




MATLAB

苗志宏◎编 著

面向对象程序设计 案例解析



中国商务出版社
CHINA COMMERCE AND TRADE PRESS

MATLAB 面向对象程序 设计案例解析

苗志宏 编著

中国商务出版社

图书在版编目 (CIP) 数据

MATLAB 面向对象程序设计案例解析/苗志宏编著

.—北京:中国商务出版社,2015.8

ISBN 978-7-5103-1366-0

I. ①M… II. ①苗… III. ①Matlab 软件—程序设计
案例解析—案例 IV. ①TP317

中国版本图书馆 CIP 数据核字 (2015) 第 192278 号

MATLAB 面向对象程序设计案例解析

MATLAB MIANXIANG DUIXIANG CHENGXU SHEJI ANLI JIEXI

苗志宏 编著

出 版:中国商务出版社

发 行:北京中商图出版物发行有限责任公司

社 址:北京市东城区安定门外大街东后巷 28 号

邮 编:100710

电 话:010—64269744 64218072 (编辑一室)

010—64266119 (发行部)

010—64263201 (零售、邮购)

网 址:<http://www.cctpress.com>

网 店:<http://cctpress.taobao.com>

邮 箱:cctp@cctpress.com bjys@cctpress.com

照 排:北京科事洁技术开发有限责任公司

印 刷:

开 本:787 毫米×980 毫米 1/16

印 张:13.25 字 数:295 千字

版 次:2015 年 8 月第 1 版 2015 年 8 月第 1 次印刷

书 号:ISBN 978-7-5103-1366-0

定 价:36.00 元

版权专有 侵权必究

盗版侵权举报电话:010-64245984

如所购图书发现有印、装质量问题,请及时与本社出版部联系。电话:010-64248236

前 言

MATLAB 是一款功能非常强大的科学计算软件。美国 MathWorks 公司自 20 世纪 80 年代中期把 MATLAB 正式推向市场以来,其功能不断扩展,目前已发展成为集数值处理、图形处理、数学建模、实时控制、动态仿真和信号处理等为一体的应用软件。与其他数学软件不同,MATLAB 软件提供了一种简洁直观的程序设计环境,其 MATLAB 语言与数学表达形式、工程计算中常用的习惯形式十分相似,符合人们的应用习惯。与其他编程语言相比较,MATLAB 语言的语法规则更加简单,编程特点更加贴近人们的思维方式。

MATLAB 早期的版本支持面向过程的结构化设计方法,很多数学计算和工程计算问题都可以在 MATLAB 得到解决。目前的版本(R2008a 版本以上),开始全面支持面向对象程序设计方法,这使得进行大型应用程序的开发、维护变得更加便捷,并显著增加代码的可重用性、可维护性和可扩展性。在 MATLAB 面向对象程序设计过程中还可以充分利用 MATLAB 所提供各种工具箱和其他数据可视化功能,以便更加方便、快捷地设计应用程序,从而解决不同领域中的实际问题。

目前国内有关 MATLAB 书籍琳琅满目,但大多数偏重于 MATLAB 语言编程介绍和各种工具箱的专题介绍,有关介绍 MATLAB 面向对象程序设计的书籍还很少。鉴于此,我们在 2014 年编写出版了《MATLAB 面向对象程序设计》一书,希望对 MATLAB 面向对象编程方法的普及起到一定的作用。在《MATLAB 面向对象程序设计》一书中,我们围绕面向对象程序设计的基本特征,比较全面地介绍了 MATLAB 面向对象程序设计的基本原理和方法。《MATLAB 面向对象程序设计》一书侧重于系统、全面地介绍面向对象程序设计的基本原理和方法。然而,面向对象程序设计的原理比较抽象,读者在短时间内很难掌握其精髓。只有通过具体的实践操作,才能深刻理解这些原理。在解决一些实际问题中,也逐步地养成用面向对象程序设计方法编写程序,才能真正体会到面向对象程序设计带来的那些优势。为此我们在原书基础上编写了这本《MATLAB 面向对象程序设计案例解析》。

本书采用案例解析的方式进行编排。围绕不同案例求解的需求,对所涉及的知识点了进行了针对性的讲解。全书知识点比较全面地涵盖了 MATLAB 面向对象程序设计的相关内容。在案例选取过程中,遵循了具有一定的专业性和实用性的原则,同时为增强一定的趣味性,还穿插编排了一些游戏类应用程序的设计案例,在每章中还适当安排一些具有综合性的案例。在章节组织上,按照面向对象程序设计基本特征进行分类编排,内容组织由浅入深、层层推进,绝大多数程序设计过程提供了完整的代码。

全书共分7章，编选了27个案例。第1章介绍了MATLAB的集成环境和基本操作；第2章主要讲解MATLAB程序设计问题，主要介绍了MATLAB中的程序设计的基本控制结构、函数概念及图形用户界面设计问题，这一章是为全面讲解面向对象程序设计原理做准备的；第3章介绍MATLAB中类的声明、对象的定义等内容，是面向对象程序设计原理的初级阶段；第4章围绕面向对象程序设计的主要特征“继承与派生”进行讲解；第5章针对“多态性”特征，介绍了函数重载、运算符重载和抽象类等概念；第6章主要介绍MATLAB句柄型类（即handle型类），这章内容也为事件模型的介绍做了准备；第7章介绍了MATLAB面向对象程序设计中的事件模型，通过几个案例比较全面地介绍了基于事件驱动的程序设计过程。

本书编写过程中有关MATLAB面向对象程序设计的基本原理内容参考了MATLAB软件帮助中的相关文档，对其中一些例子也进行了适当引用和修改。所涉及的MATLAB的基本原理部分参考了文献[2—8]中的部分内容。有关面向对象程序设计基本原理还参照了文献[9—13]的部分内容，并其中的一些案例进行了移植和修改。案例中有关涉及数学问题的求解，参考了文献[14]的部分内容。在此向这些作者们表示谢意。作者在编写书稿过程中，为适应MATLAB面向对象程序设计方法对一些常见的案例进行了引申和再创作，书中的大多数案例的程序设计具有一定的原创性。

本书全部章节内容由苗志宏编写。本书写作过程中得到了马金强老师的协助和支持，还得到了中国商务出版社张永生编辑的鼓励和支持，在此深表谢意。由于作者水平有限，书中难免存在错误和不足之处，恳请批评指正。

苗志宏

2015年5月26日

目 录

第 1 章 MATLAB 快速入门	(1)
1.1 MATLAB 集成环境介绍	(1)
1.1.1 MATLAB 启动和退出	(1)
1.1.2 MATLAB 命令窗口	(1)
1.1.3 MATLAB 工作空间管理窗口	(3)
1.1.4 MATLAB 当前目录窗口	(3)
1.1.5 MATLAB 命令历史窗口	(4)
1.2 MATLAB 入门操作	(5)
1.2.1 MATLAB 命令窗口基本操作	(5)
1.2.2 M 文件的操作	(7)
第 2 章 MATLAB 程序设计基础	(12)
2.1 案例 1: 线性方程组的求解问题——矩阵基本运算	(12)
2.2 案例 2: 百钱买百鸡问题——算法的基本控制结构	(19)
2.3 案例 3: 一个日历程序的设计——函数调用	(23)
2.4 案例 4: 汉诺塔问题——函数递归调用	(28)
2.5 案例 5: 一个猜数游戏——图形用户界面设计	(32)
2.6 案例 6: 一个图形演示系统的设计——图形用户界面设计工具的应用 ...	(36)
第 3 章 类与对象	(42)
3.1 案例 1: 银行账户的信息管理——从面向过程结构化设计到面向 对象程序设计	(42)
3.2 案例 2: 世界各地时钟——类的声明, 对象的创建	(49)
3.3 案例 3: 统计一个储户在多家银行的存款总和——类的优先级	(59)
3.4 案例 4: 学生信息管理系统之一——构造函数的实现	(66)
3.5 案例 5: 基于面向对象程序设计的前馈神经网络设计——综合设计	(74)

第 4 章 继承与派生	(85)
4.1 案例 1: 学生信息管理系统之二——派生类的声明	(85)
4.2 案例 2: 一元线性拟合问题的程序设计——内置类型的派生类	(91)
4.3 案例 3: 高校人员信息管理——多重继承下的构造函数	(98)
4.4 案例 4: 用高斯消元法求线性方程组的解——综合设计	(107)
4.5 案例 5: 两人对弈五子棋游戏的实现——综合设计	(112)
第 5 章 多态性	(122)
5.1 案例 1: 正交多项式曲线拟合的实现——内置函数的重载	(122)
5.2 案例 2: 模糊集合运算的实现——运算符重载	(129)
5.3 案例 3: 模糊综合决策数学模型的实现——运算符重载	(134)
5.4 案例 4: 定积分的几种数值解法实现——抽象类	(140)
第 6 章 handle 型类	(146)
6.1 案例 1: 银行账户的信息管理——handle 型类的设计	(146)
6.2 案例 2: 函数曲线的自定义绘制——标准 set/get 接口实现	(153)
6.3 案例 3: 文件读写操作函数的封装——handle 型类的析构函数	(159)
第 7 章 事件模型	(167)
7.1 案例 1: 一个简单事件模型的实现——事件模型	(167)
7.2 案例 2: 用二分法求方程根的程序设计——属性成员事件	(173)
7.3 案例 3: 日历程序的设计——属性成员事件	(178)
7.4 案例 4: 多级倒立摆控制仿真系统设计——综合设计	(184)
参考文献	(203)

第 1 章 MATLAB 快速入门

MATLAB 是一个面向科学计算和工程计算的数学软件，目前在各个领域中的应用越来越广泛。MATLAB 的主要功能包括进行数值计算、数据可视化、符号计算等，其中还包含了很多直接涉及相关研究领域的工具箱。

MATLAB 不仅是一个数学软件，而且也是一个很强大的编程语言。相对于其他编程语言，它更加接近于书写计算公式的一般方式，用 MATLAB 编写程序犹如在演算纸上书写公式和进行求解，它是一种解析执行的语言。MATLAB 编程语言既支持结构化程序设计方法，在 MATLAB R2008a 版本后还支持面向对象程序设计方法。

本章主要介绍 MATLAB 的集成环境和简单的入门操作。

1.1 MATLAB 集成环境介绍

1.1.1 MATLAB 启动和退出

MATLAB 与一般的 Windows 应用程序一样，有三种常见启动方法。

- (1) 双击 Windows 窗口桌面上 MATLAB 图标即可启动。
- (2) 选择 Windows 任务栏上“开始”按钮，在菜单项中选择“程序”菜单项下的“MATLAB”，就可启动 MATLAB 系统。
- (3) 选择 Windows 任务栏上“开始”按钮，在菜单项中单击“运行”按钮，输入 matlab 并单击“确定”按钮或按 Enter 键，也可启动 MATLAB 系统。

启动 MATLAB 后，就可进入 MATLAB 集成环境，如图 1-1 所示。

集成环境中包括 MATLAB 主窗口、命令窗口 (Command Window)、命令历史窗口 (Command History)、工作空间管理窗口 (Workspace) 和当前目录窗口 (Current Directory)。

要退出 MATLAB，有如下三种方法。

- (1) 单击 MATLAB 主窗口的“关闭”按钮。
- (2) 在菜单栏中选择“File”，再选择其中的“Exit MATLAB”选项。
- (3) 在命令窗口中输入“Exit”或“Quit”命令。

1.1.2 MATLAB 命令窗口

MATLAB 命令窗口是主要的人机交互窗口，用于输入命令并显示除图形以外所执

行的结果，如图 1-2 所示。

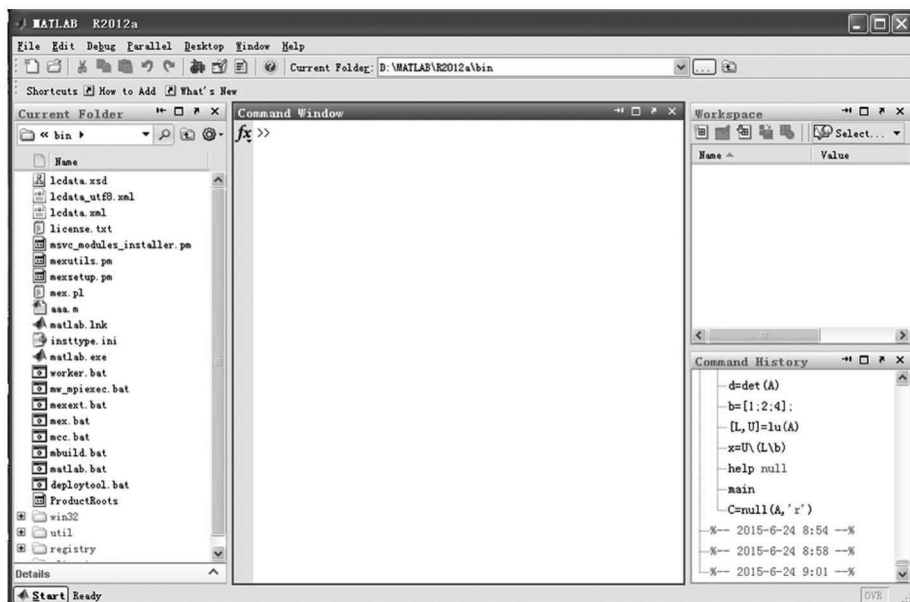


图 1-1 MATLAB 集成环境初始界面



图 1-2 MATLAB 的命令窗口

命令窗口中的“>>”为命令提示符，表示 MATLAB 处于输入等待状态。在命令提示符后输入一个命令后，按 Enter 键，MATLAB 就会解释执行所输入的命令，并在其后给出计算结果。通常，一个命令行输入一条命令，以按 Enter 键结束，但也可以在一行中输入多个命令，命令之间以逗号或分号隔开，如：

```
>> x=12; y=45;
```

```
>> a=2, b=6
```

如果结果不希望被显示，则只要在语句之后加上一个分号即可。此时尽管结果没有显示，但它依然被赋值，并驻留在 Matlab 的工作空间中。

如果一个命令行中要输入很长的命令，需要分为多个物理行，这时可以在各物理行之后加上三个连续的点“...”（续行符）并按 Enter 键续写下一行。

1.1.3 MATLAB 工作空间管理窗口

工作空间是 Matlab 用于暂时存储各种变量和结果的内存空间。而在工作空间管理窗口中可以显示工作空间中所有变量的名称、大小、字节数和变量类型说明，可对变量进行观察、编辑、保存和删除。

工作空间管理窗口如图 1-3 所示。

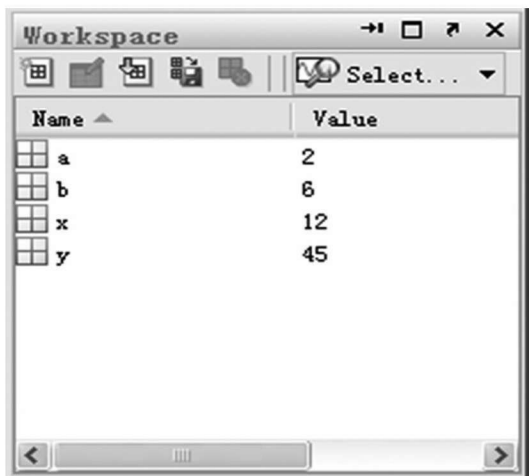


图 1-3 MATLAB 的工作空间管理窗口

1.1.4 MATLAB 当前目录窗口

MATLAB 中当前目录指定了 MATLAB 运行文件时的目录，称为工作目录。工作目录的选择可以通过 MATLAB 主窗口工具栏中的“Browse for folder”按钮来选择，并在“Current Folder”编辑框中显示当前工作目录。

只有在当前工作目录或搜索路径下的文件、函数才可以被运行或调用。在当前目录窗口中可以显示或改变当前目录，还可以显示当前目录下的文件并提供搜索功能，MATLAB 中当前目录窗口如图 1-4 所示。

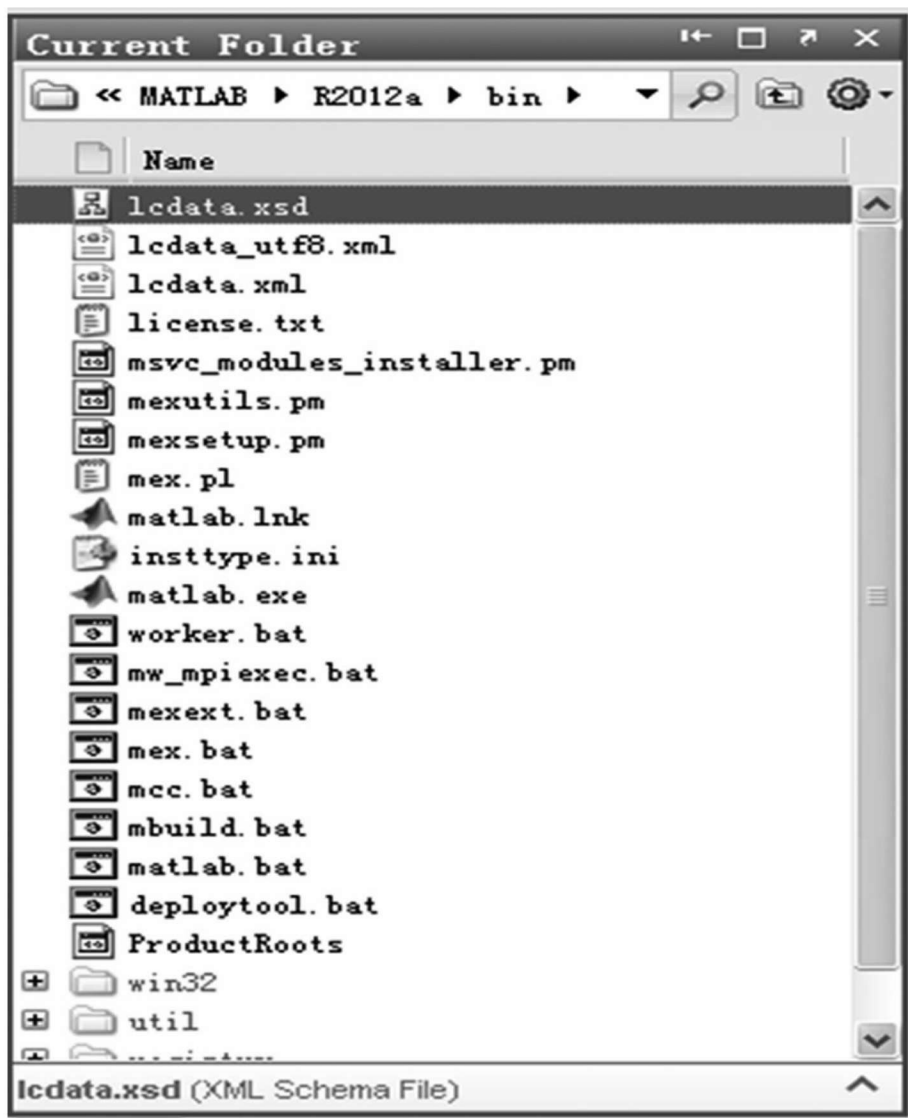


图 1-4 MATLAB 的当前目录窗口

除通过“Browse for folder”按钮来选择当前工作目录外，还可以在 MATLAB 命令窗口中使用 `cd` 命令来改变当前目录。例如，要将用户目录 `d:\mywork` 设置为当前目录，可在命令窗口输入命令：

```
>> cd d:\mywork
```

1.1.5 MATLAB 命令历史窗口

在默认设置下，命令历史窗口中会自动保留自 MATLAB 安装起用过的所有命令，并标明使用时间，从而方便用户查询。通过双击命令可进行历史命令的再运行。如果要清除这些历史命令，可以选择 Edit 菜单中的“Clear Command History”命令。图

1—5 是一个命令历史窗口。



图 1-5 MATLAB 的命令历史窗口

1.2 MATLAB 入门操作

1.2.1 MATLAB 命令窗口基本操作

在这一小节，通过一些简单的计算问题，介绍 MATLAB 的基本操作。

〔例 1〕 试计算两个矩阵的和。

在命令窗口下输入如下命令：

```
>> A=[1 4 7; 2 5 8; 3 6 10];  
>> B=[1 2 3; 4 5 6; 7 8 9];  
>> C=A+B
```

C=

2	6	10
6	10	14
10	14	19

以上每行命令输入完后，按 Enter 键，命令就被执行。最后一个命令后没有写分号，执行后就显示计算结果。

〔例 2〕 绘制函数 $y=e^{-x/2}\sin(2\pi x)$ 的曲线。

在命令窗口下输入如下命令：

```
>> x=0:0.01:2*pi;  
>> y=exp(-x/2). * sin(2*pi*x);
```

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

这里 plot 是绘制二维曲线图形的函数，运行结果如图 1-6 所示。

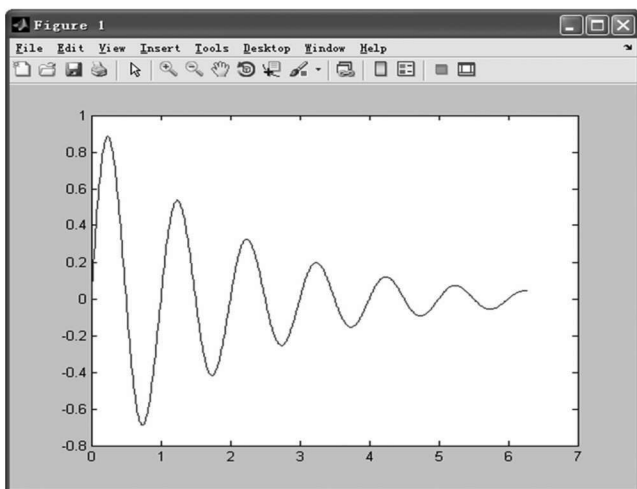


图 1-6 绘制函数曲线

〔例 3〕 绘制函数 $z = \frac{\sin(x^2 + y^2)}{x^2 + y^2}$ 的曲面图。

在命令窗口下输入如下命令：

```
>> [X,Y] = meshgrid(-10:0.5:10);
```

```
>> W=sqrt(X.^2+Y.^2)+eps;
```

```
>> Z=sin(W)./W;
```

```
>> surf(X,Y,Z);
```

这里 meshgrid 是网格化函数，surf 是绘制曲面的函数。运行结果如图 1-7 所示。

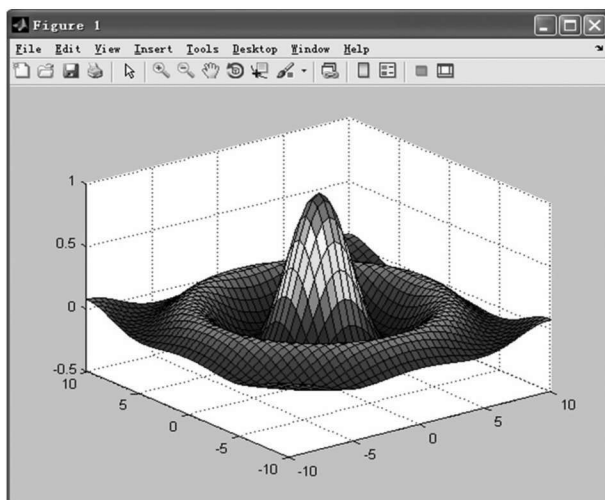


图 1-7 绘制函数曲面

〔例 4〕 用帮助命令 help 查询函数 surf。

在命令窗口下输入如下命令：

```
>> help surf
```

运行结果为：

```
surf 3-D colored surface.  
surf (X, Y, Z, C) plots the colored parametric surface defined by  
four matrix arguments. The view point is specified by VIEW.  
The axis labels are determined by the range of X, Y and Z,  
or by the current setting of AXIS. The color scaling is determined  
by the range of C, or by the current setting of CAXIS. The scaled  
color values are used as indices into the current COLORMAP.  
The shading model is set by SHADING.  
...
```

这里 help 是 MATLAB 的帮助命令，可以通过 help 加函数名来显示该函数的帮助说明。

〔例 5〕 用命令 format 改变命令窗口下的数据显示精度。

在命令窗口下输入如下命令：

```
>> pi
```

```
ans=  
3.1416
```

```
>> format long
```

```
>> pi
```

```
ans=  
3.141592653589793
```

默认情况下，MATLAB 命令窗口下的显示精度是 short 型，如果要查看更多位的数字，可以用命令 format 加一个参数来改变，这里参数 long 是长格式，表示保留小数点后 15 位。

1.2.2 M 文件的操作

MATLAB 提供了用于编写程序的一个文本编辑器，即 M 文件编辑器，其界面和 Windows 下的文本编辑软件类似，如图 1-8 所示。

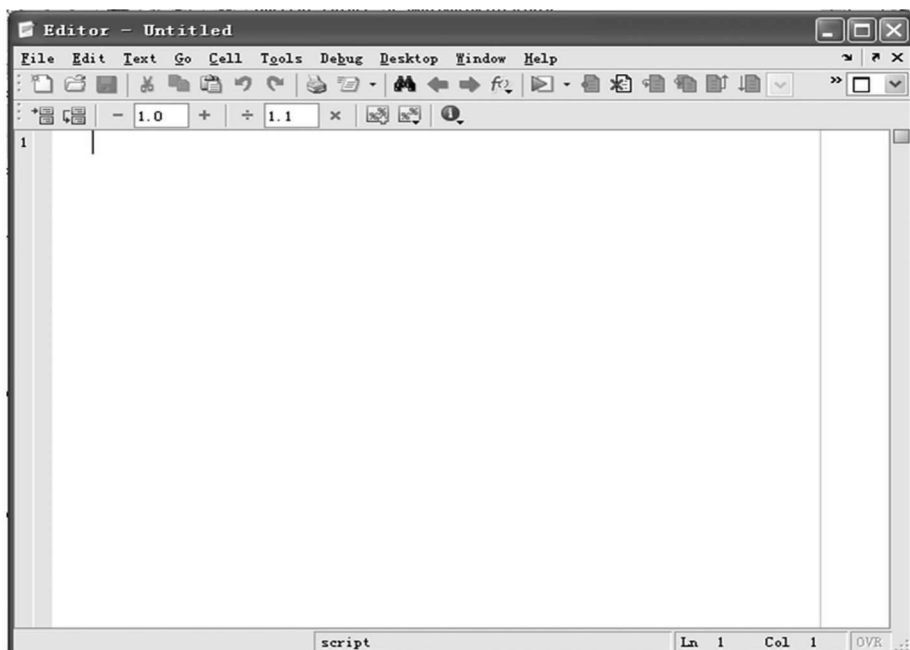


图 1—8 M 文件编辑器

要启动 M 文件编辑器建立一个 M 文件有以下几种方式。

- (1) 在菜单栏中：选择 File 菜单→New 菜单项→Script 项。
- (2) 在工具栏中：选择 New Script 按钮。
- (3) 在命令窗口：输入命令 edit。

启动 M 文件编辑器后，就可以在空白区域中编写自己的程序。编写好程序并保存后，选择工具栏中的 Run 按钮或按功能键 F5 可直接执行程序。

目前，MATLAB 中的 M 文件有三种类型：命令文件（Script File）、函数文件（Function File）和类定义文件（Classdef File）。命令文件实际上是一串指令的集合，与在命令窗口逐行执行文件中的所有指令，其结果是一样的。没有输入输出参数。函数文件的第一个有效行的开头是关键字 function，一般需要有输入变量，和输出变量。类定义文件中第一个有效行的开头是关键字 classdef，它是作为一个类类型的声明文件。以上三种类型的文件其扩展名都是 m。

命令文件和函数文件的主要区别是：命令文件中定义的变量在文件执行完后仍保存在工作空间中，而函数文件内定义的变量仅在函数文件内部起作用，执行完函数文件后，这些局部变量将被清除。

以下通过一些简单的编程问题，介绍 M 文件的操作。

〔例 1〕 已知三角形的三条边，编写程序计算其面积。

解：在 M 文件编辑器中编写如下命令文件：

```
%Triangle.m, 计算三角形的面积  
a=input('请输入三角形的边 a:'); %输入 a
```

```

b=input('请输入三角型的边 b:');%输入 b
c=input('请输入三角型的边 c:');%输入 c
%计算
s=(a+ b+ c)/2;
Area=sqrt(s*(s-a)*(s-b)*(s-c));%利用海伦公式计算面积
Area                                %输出结果

```

其中“%”是程序的注释符，MATLAB 不会去执行该行后续的字符，仅作为对程序的文字说明。程序中的 input 是带提示信息的数据输入函数。

将以上程序保存为一个 M 文件，这里文件名是 Triangle.m。在命令窗口下直接输入文件名 Triangle 执行程序：

```
>> Triangle
```

运行结果为：

```

请输入三角型的边 a: 3
请输入三角型的边 b: 4
请输入三角型的边 c: 5
Area=
    6

```

也可直接单击编辑器工具栏中的 Run 按钮运行程序。

〔例 2〕设计一个函数文件，用反正切函数 arctan 的泰勒展开式求其近似值，其中

$$\arctan(x) = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$$

解：新建一个 M 文件，在其中编写如下代码：

```

%myarctan.m, 利用泰勒展开式求 atan 的值
function y=myarctan(x)% myarctan 是函数名
    item=x;
    sign=1;
    y=x;
    for i=1: 10%截取前 11 项
        item=item*x*x;
        sign=-sign;
        y=y+sign*item/(2*i+1);
    end
end

```

保存文件时，默认情况下 MATLAB 会以函数数名作为文件名，也可以更改为其他名字，但在函数调用时以文件名为准。这里将上面程序保存为 myarctan.m 文件。

以下是在命令窗口中调用该函数：

```
>> myarctan(pi/10)
```


运行结果为：

```
ans=  
    0.3044
```

〔例 3〕 用 M 文件定义一个复数类 Complex。

编写一个类定义文件，可以直接新建一个 M 文件，然后编写相关的类声明代码。也可以通过 MATLAB 主窗口菜单栏中选择 File 项，再选 New 中的 Class 项直接生成一个简单的类定义结构框架，如图 1—9 所示。

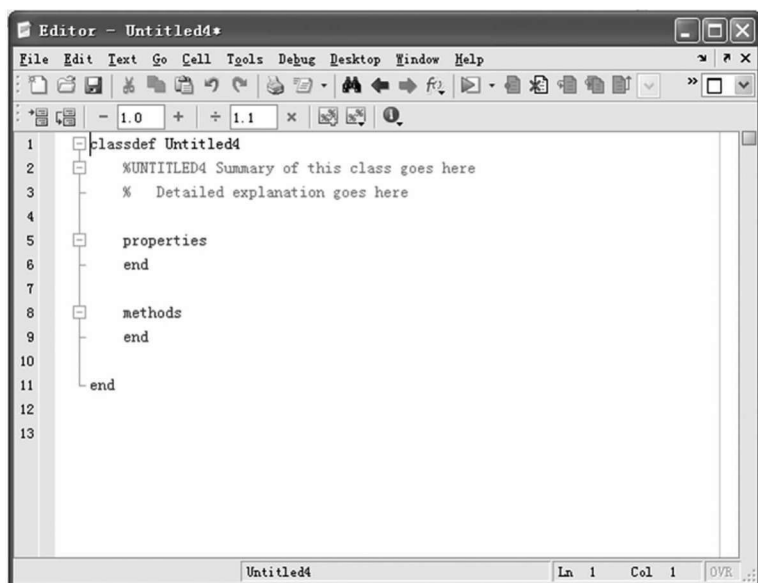


图 1—9 类定义文件编辑界面

修改和添加类名和相关属性后，得到复数类 Complex 的如下声明过程：

```
classdef Complex
```

```
%复数类的声明
```

```
properties %属性成员
```

```
    a; %实部
```

```
    b; %虚部
```

```
end
```

```
methods %方法
```

```
function obj=Complex(x, y)%构造函数
```

```
    obj.a=x;
```

```
    obj.b=y;
```

```
end
```

```
function Disp(obj)%显示复数
```