

MATLAB

开发实例系列图书



MATLAB

GUI设计学习手记

罗华飞 编著

 北京航空航天大学出版社

MATLAB 开发实例系列图书

MATLAB GUI 设计学习手记

罗华飞 编著

北京航空航天大学出版社

内 容 简 介

本书由浅入深、循序渐进地介绍了 MATLAB GUI 设计的基础知识,旨在使读者在较短时间内掌握 GUI 设计的精要所在。

本书首先介绍了 GUI 设计的预备知识,然后详细介绍了 GUI 对象的属性以及两种建立 GUI 的方法:采用函数建立和采用 GUIDE 建立。在介绍了 GUI 设计基础知识的基础上,深入讲解了 ActiveX 控件、定时器、串口及 mcc 编译的相关知识。书中穿插了大量的图表,直观、简要地列出了相关知识点。对于一些重要、难懂的知识点,配以大量的典型例题,方便读者边学边练,加强理解。最后,书中给出了 3 个典型的综合实例,供读者研究学习。

本书适合需要短时间内迅速掌握 MATLAB GUI 设计的初学者使用,也可作为相关专业师生、相关工程开发人员的参考手册。

图书在版编目(CIP)数据

MATLAB GUI 设计学习手记/罗华飞编著. —北京:北京
航空航天大学出版社,2009. 8

(MATLAB 开发实例系列图书)

ISBN 978-7-81124-880-7

I. M… II. 罗… III. 算法语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2009)第 143182 号

MATLAB GUI 设计学习手记

罗华飞 编著

责任编辑 陈守平

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(100191) 发行部电话:(010)82317024 传真:(010)82328026

<http://www.buaapress.com.cn> E-mail:bhpress@263.net

涿州市新华印刷有限公司印装 各地书店经销

*

开本:787 mm×1 092 mm 1/16 印张:23.75 字数:608 千字

2009 年 8 月第 1 版 2009 年 8 月第 1 次印刷 印数:5 000 册

ISBN 978-7-81124-880-7 定价:39.80 元

前 言

MATLAB 是当今科研领域最常用的应用软件之一,它具有强大的矩阵计算、符号运算和数据可视化功能,是一种简单易用、可扩展的系统开发环境和平台。

MATLAB GUI 设计使读者不必深入掌握面向对象的编程语言,也能设计出精美的人机界面。与 Visual C++ 相比, MATLAB GUI 具有学习起点低(只需要了解 MATLAB 的基本操作和 C 语言的基础知识)、易学易懂、开发周期短的优点,设计出来的界面,完全可与 Visual C++ 编写的界面媲美。

本书共分 11 章,每章(最后 3 章除外)由 3 节内容组成:知识点归纳、重点与难点分析和典型例题解析。知识点归纳详细介绍了各章的内容与知识点,容易理解错的知识点用【注意】标明,个别地方配以少量例题讲解;重点与难点分析简要概括了各章的重点和难点,便于读者重点掌握;典型例题解析列举了大量的经典例题,侧重各章重、难点的考察,建议读者先独立将例题完成,然后参考例题解析。这样,边学边练,以进一步掌握每章的重点和难点。

第 1 章 GUI 设计预备知识,首先介绍了 MATLAB 的基本程序元素,然后重点介绍了几种 GUI 设计中经常使用的数据类型和矩阵操作函数,最后介绍了程序设计的 5 种句型(for、while 循环结构,if、switch 条件分支结构和 try...catch 结构)。熟悉和掌握这些内容,有助于读者更快、更扎实地掌握本书的内容。

第 2 章文件 I/O,主要介绍了文件 I/O 操作的相关函数,分为高级文件 I/O 和低级文件 I/O 两部分。高级文件 I/O 介绍了读写 MAT 或 ASCII 文件、读写 TXT 文件、读写 Excel 文件和读写图像文件的方法及相关函数;低级文件 I/O 介绍了读写二进制文件和读写文本文件的方法及相关函数。数据的导入与导出是 GUI 设计面向工程应用的基础。因此,文件 I/O 操作必须重点掌握。

第 3 章二维绘图简介,主要介绍了与 GUI 设计密切相关的线性二维绘图及其相关函数、绘图工具函数和绘图注释函数。二维绘图函数常用于 GUI 设计中的数据可视化模块。

第 4 章句柄图形系统,首先介绍了句柄图形对象的概念及其操作函数,然后详细介绍了各种句柄图形对象的创建方法、属性及其含义。通过本章的学习,读者可以了解每种 GUI 对象的属性及其创建方法。本章是 GUI 设计的重点内容,需要熟练掌握。

第 5 章预定义对话框,介绍了 MATLAB 环境下可调用的所有预定义对话框,包括公共对话框和 MATLAB 自定义的对话框。这些对话框使得 GUI 设计更加直观、灵活。

第 6 章采用 GUIDE 建立 GUI,首先介绍了采用 GUIDE 建立 GUI 的基本操作,然后着重介绍了 GUI 的 M 文件结构、回调函数的分类以及回调函数的编写方法,最后介绍了 GUIDE 环境下 GUI 组件的使用方法。通过本章的学习,读者可以设计出精美的 GUI 界面,实现复杂的功能。本章是 GUI 设计的重点内容,需要熟练掌握。

第 7 章 ActiveX 控件,详细介绍了 7 大类的 ActiveX 控件:LED 状态显示、七段 LED 数码显示、表盘显示、线性测量、滑动条、进度条和视频播放。熟练掌握这些控件,可以使 GUI 的界面更加美观。本章是 GUI 设计的精华之处,只有掌握了 ActiveX 控件的设计,才能设计出与

Visual C++相媲美的精美界面。

第8章定时器,详细介绍了GUI设计中定时器的使用方法。熟练掌握定时器,可以实现更复杂、实时性更高的GUI设计。

第9章串口编程,介绍了GUI设计中的串口使用方法。熟练掌握串口编程,可以实现GUI与外部设备之间的实时通信。

第10章mcc编译,简要介绍了GUI编译为独立可执行文件的方法、mcc编译的局限性和P文件的使用方法。通过本章的学习,读者可以轻松编译带有ActiveX控件的GUI为EXE格式文件。

第11章综合实例,通过详细讲解密码登录框、科学计算器和串口通信助手3个实例,使读者深入、熟练地掌握MATLAB GUI设计的精髓。每个实例都有详细的构思和源程序,源程序包含详细的注释说明。通过本章的练习,读者可以独立完成复杂的GUI设计工程项目,设计出精美、稳定可靠的GUI。

最后,附录部分列出了常用的GUI设计相关函数,供读者参考查询。

本书试图全面细致地讲解MATLAB GUI设计所涉及的所有内容,希望能使读者对GUI设计产生浓厚的兴趣,从而达到抛砖引玉的目的。笔者相信,只要熟读本书,掌握本书之精髓,并勤加练习,数周之内定可精通GUI设计。

本书在编写过程中,参考了大量的网络资料,在此感谢MATLAB中文论坛提供的珍贵资料和math给予的热情帮助。

另外,我要特别感谢我的女友刘琴,在本书的创作过程中,她在背后给了我无微不至的照顾和鼓励。

本书全部源程序均在MATLAB 7.1环境下运行通过,并在MATLAB R2008a中测试运行通过。程序源代码请到北京航空航天大学出版社(<http://www.buaapress.com.cn>)下载中心下载。

同时,北京航空航天大学出版社联合MATLAB中文论坛(<http://www.iLoveMatlab.cn>)为本书设立了在线交流版块,网址为<http://www.iLoveMatlab.cn/forum-155-1.html>,有问必答!作者会第一时间在MATLAB中文论坛勘误,也会根据读者要求陆续上传更多案例和相关知识链接,还会随着MATLAB版本的升级增添必要的内容以满足读者的需求。希望这本不断“成长”的书能最大限度地解决您在学习、研究、工作中遇到的MATLAB GUI相关问题。

由于作者水平有限,加之时间仓促,书中的不足与疏忽之处,敬请读者批评指正。

编者

2009年6月

目 录

第 1 章 GUI 设计预备知识	1
1.1 知识点归纳	1
1.1.1 基本程序元素	1
1.1.2 数据类型	7
1.1.3 矩阵操作	31
1.1.4 程序设计	37
1.2 重点与难点分析	46
1.2.1 矩阵、向量、标量与数组	46
1.2.2 字符串与数值的相互转换	47
1.2.3 结构体	47
1.2.4 单元数组	48
1.3 典型例题解析	48
第 2 章 文件 I/O	52
2.1 知识点归纳	52
2.1.1 高级文件 I/O 操作	52
2.1.2 低级文件 I/O 操作	63
2.2 重点与难点分析	76
2.2.1 二进制文件与文本文件	76
2.2.2 sprintf 与 fprintf 函数	76
2.2.3 fscanf 与 textscan 函数	77
2.2.4 Excel 文件操作	77
2.2.5 图像数据的操作	78
2.2.6 低级文件 I/O 操作	78
2.3 典型例题解析	78
第 3 章 二维绘图简介	82
3.1 知识点归纳	82
3.1.1 常用的二维绘图函数	82
3.1.2 绘图工具	87
3.1.3 绘图注释	89
3.2 重点与难点分析	99
3.2.1 二维绘图的相关函数	99
3.2.2 Tex 字符	100

3.3	典型例题解析	100
第4章	句柄图形系统	104
4.1	知识点归纳	104
4.1.1	句柄图形对象	104
4.1.2	句柄图形对象的基本操作	106
4.1.3	句柄图形对象的属性	115
4.2	重点与难点分析	181
4.2.1	句柄式图形对象的常用函数总结	181
4.2.2	figure 对象的几个重要属性	182
4.2.3	axes 对象的几个重要属性	183
4.2.4	line 对象的几个重要属性	184
4.2.5	text 对象的几个重要属性	184
4.2.6	uicontrol 对象中的 text 控件与核心图形对象中的 text 对象的比较	185
4.2.7	对象的 Tag 值与句柄值的概念比较	185
4.2.8	uimenu 与 uicontextmenu 对象	185
4.3	典型例题解析	186
第5章	预定义对话框	195
5.1	知识点归纳	195
5.1.1	文件打开对话框(uigetfile)	196
5.1.2	文件保存对话框(uiputfile)	199
5.1.3	颜色设置对话框(uiSetColor)	200
5.1.4	字体设置对话框(uiSetFont)	201
5.1.5	页面设置对话框(pagesetupdlg)	202
5.1.6	打印预览对话框(printpreview)	202
5.1.7	打印设置对话框(printdlg)	202
5.1.8	进度条(waitbar)	202
5.1.9	菜单选择对话框(menu)	208
5.1.10	普通对话框(dialog)	208
5.1.11	错误对话框(errordlg)	209
5.1.12	警告对话框(warndlg)	211
5.1.13	帮助对话框(helpdlg)	212
5.1.14	信息对话框(msgbox)	213
5.1.15	提问对话框(questdlg)	214
5.1.16	输入对话框(inputdlg)	215
5.1.17	目录选择对话框(uiGetDir)	217
5.1.18	列表选择对话框(listdlg)	217
5.2	重点与难点分析	218
5.2.1	uiGetfile	218
5.2.2	uiPutfile	218

若您对此书内容有任何疑问，可以凭在线交流卡登录MATLAB中文论坛与作者交流。

5.2.3	waitbar	218
5.2.4	msgbox	219
5.2.5	questdlg	219
5.2.6	inputdlg	219
5.2.7	listdlg	219
5.3	典型例题解析	219
第 6 章	采用 GUIDE 建立 GUI	224
6.1	知识点归纳	224
6.1.1	GUIDE 界面基本操作	225
6.1.2	GUI 的 M 文件	231
6.1.3	回调函数	241
6.1.4	GUI 跨平台的兼容性设计	243
6.1.5	采用 GUIDE 创建 GUI 的步骤	244
6.1.6	GUI 组件详解	244
6.2	重点与难点分析	263
6.2.1	回调函数中的数据传递	263
6.2.2	KeyPressFcn 与 CurrentCharacter	264
6.2.3	WindowButtonDownFcn、Callback 与 SelectionType	265
6.3	典型例题解析	265
第 7 章	ActiveX 控件	271
7.1	知识点归纳	271
7.1.1	LED 状态显示(LED ActiveX Control)	272
7.1.2	七段 LED 数码显示(Numeric LED ActiveX Control)	278
7.1.3	表盘显示(Angular Gauge ActiveX Control)	280
7.1.4	线性测量(Linear Gauge ActiveX Control)	285
7.1.5	滑动条(Slider Activex Control)	289
7.1.6	进度条(Percent ActiveX Control)	292
7.1.7	视频播放(Windows MediaPlayer)	295
7.2	重点与难点分析	296
7.2.1	LED ActiveX Control	296
7.2.2	Numeric LED ActiveX Control	296
7.2.3	Angular Gauge ActiveX Control	297
7.2.4	Slider Activex Control	297
7.3	典型例题解析	297
第 8 章	定时器	303
8.1	知识点归纳	303
8.1.1	定时器对象及其属性	303
8.1.2	定时器的执行模式	305
8.1.3	定时器的回调函数	306

若您对此书内容有任何疑问，可以凭在线交流卡登录 MATLAB 中文论坛与作者交流。

8.1.4	定时器的操作函数	307
8.1.5	定时器的操作步骤	308
8.2	重点与难点分析	309
8.2.1	TimerFcn 函数	309
8.2.2	常用的定时器操作函数	309
8.3	典型例题分析	310
第 9 章	串口编程	317
9.1	知识点归纳	317
9.1.1	串口概述	317
9.1.2	串口对象的属性	319
9.1.3	串口的基本操作	322
9.1.4	串口 I/O 函数汇总	325
9.2	重点与难点分析	326
9.2.1	串口对象的创建	326
9.2.2	重要的串口操作函数	327
第 10 章	mcc 编译	328
10.1	mcc 编译介绍	328
10.2	mcc 编译的局限性	329
10.3	MATLAB 保护文件(P 文件)	330
第 11 章	综合实例	331
附录	MATLAB GUI 设计常用函数	367

若您对此书内容有任何疑问，可以凭在线交流卡登录 MATLAB 中文论坛与作者交流。

1.1 知识点归纳

本章内容：

- ◆ 基本程序元素
 - ◇ 变量
 - ◇ 特殊值
 - ◇ 关键字
 - ◇ 运算符
 - ◇ 特殊表达式
- ◆ 数据类型
 - ◇ 数值型
 - ◇ 逻辑型
 - ◇ 字符数组
 - ◇ 结构数组
 - ◇ 单元数组
 - ◇ 函数句柄
 - ◇ 日期和时间
- ◆ 矩阵操作
 - ◇ 创建矩阵
 - ◇ 连接矩阵
 - ◇ 重塑矩阵形状
 - ◇ 矩阵元素移位和排序
- ◆ 程序设计
 - ◇ 函数参数
 - ◇ for、while 循环结构
 - ◇ if、switch 条件分支结构
 - ◇ try...catch 结构
 - ◇ continue、break 和 return
 - ◇ 其他常用函数

1.1.1 基本程序元素

1. 变 量

程序中,为了方便操作内存中的值,需要给内存中的值设定一个标签,这个标签称之为变

量。变量不需事先声明，MATLAB 在遇到新的变量名时，会自动建立变量并分配内存。给变量赋值时，如果变量不存在，会创建它；如果变量存在，会更新它的值；赋值时，右边的表达式必须有一个值（即使值为空也行）。

变量命名规则如下：

- 始于字母，由字母、数字或下划线组成。
- 区分大小写。
- 可任意长，但仅使用前 N 个字符。N 与硬件有关，由函数 `namelengthmax` 返回，一般 $N=63$ 。
- 不能使用关键字作为变量名。
- 避免使用函数名作为变量名。

如果变量采用函数名，该函数失效。

例如：

```
>> clear = 3;
>> clear
clear =
    3
```

`clear` 函数失效，不能清除基本工作空间里的变量。

又如：

```
>> i = 3;
>> 1 + 2 * i
ans =
    7
```

虚数单位 `i` 失效。

与变量有关的函数见表 1.1。

表 1.1 与变量有关的函数

函数名	函数说明
<code>clear</code>	移除工作空间里的数据项，释放内存
<code>isvarname</code>	检查输入的字符串是否为有效的变量名
<code>genvarname</code>	采用字符串构建有效的变量名
<code>ans</code>	当没指定输出变量时，临时存储最近的答案
<code>namelengthmax</code>	返回最大的标识符长度

MATLAB 存储变量在一块内存区域中，该区域称为基本工作空间。脚本文件或命令行创建的变量都存在基本工作空间中。函数不使用基本工作空间，每个函数都有自己的函数空间（连子函数也不例外）。

变量有 3 种基本类型：

① 局部变量。每个函数都有自己的局部变量，这些变量只能在定义它的函数内部使用。当函数运行时，局部变量保存在函数的工作空间里，一旦函数退出，这些局部变量将不复存在。

如果要观察函数的工作空间,可以在函数内部设置断点。

脚本(没有输入输出参数、由一系列 MATLAB 命令组成的 M 文件)没有单独的工作空间,只能共享脚本调用者的工作空间。当从命令行调用,脚本变量存在基本工作空间内;当从函数调用,脚本变量存在函数空间内。

② 全局变量。在函数或基本工作空间内,用 global 声明的变量为全局变量。例如,声明变量 a 为全局变量:

```
global a
```

声明了全局变量的函数或基本工作空间,共享该全局变量,都可以给它赋值。

如果函数的子函数也要使用全局变量,也必须用 global 声明。

为增强程序的逻辑性和可读性,应谨慎使用全局变量。

③ 永久变量。永久变量用 persistent 声明,只能在 M 文件函数中定义和使用,只允许声明它的函数存取。当声明它的函数退出时, MATLAB 不会从内存中清除它。例如,声明变量 a 为永久变量:

```
persistent a
```

最好在函数开始处声明永久变量。声明后,缺省初始值为空矩阵。

2. 特殊值

一些函数返回重要的特殊值,这些值可以在 M 文件中使用,见表 1.2。

表 1.2 特殊值

函 数	函数说明
eps	浮点数相对精度;MATLAB 计算时的容许误差
intmax	本计算机能表示的 8 位、16 位、32 位、64 位的最大整数
intmin	本计算机能表示的 8 位、16 位、32 位、64 位的最小整数
realmax	本计算机能表示的最大浮点数
realmin	本计算机能表示的最小浮点数
pi	圆周率,3.1415926535897...
i, j	虚数单位
inf	无穷大。当 $n > 0$ 时, $n/0$ 的结果是 inf;当 $n < 0$ 时, $n/0$ 的结果是 -inf
NaN	非数,无效数值。比如 $0/0$ 或 inf/inf ,结果为 NaN
computer	计算机类型。比如当返回字符串 PCWIN 时,操作系统为 Microsoft Windows
version	MATLAB 版本字符串如 7.0.0.19920 (R14)

3. 关键字

MATLAB 为程序语言保留的一些字称为关键字。变量名不能为关键字,否则会出错。查看 MATLAB 所有的关键字,用 iskeyword 函数:

```
>> iskeyword
ans =
```

```

'break'
'case'
'catch'
'continue'
'else'
'elseif'
'end'
'for'
'function'
'global'
'if'
'otherwise'
'persistent'
'return'
'switch'
'try'
'while'
    
```

4. 运算符

运算符主要分为算术运算符、关系运算符和逻辑运算符 3 大类,还包括一些特殊运算符。下面详细介绍。

(1) 算术运算

算术运算分为两类:矩阵运算和数组运算。矩阵运算是按线性代数的规则进行运算,而数组运算是数组对应元素间的运算。算术运算符及相关运算方式、说明见表 1.3。

表 1.3 算术运算符

运算符	运算方式	说明	运算符	运算方式	说明
+、-	矩阵运算、数组运算	加、减	+、-	矩阵运算、数组运算	单目的加、减
、/	矩阵运算	乘、除	.	数组运算	数组乘
\	矩阵运算	左除,左边为除数	./	数组运算	数组左除
^	矩阵运算	乘方	.\	数组运算	数组右除
'	矩阵运算	转置	.^	数组运算	数组乘方
:	矩阵运算、数组运算	索引,用于增量操作	.'	数组运算	数组转置

MATLAB 数组的算术运算,是两个同维数组对应元素之间的运算。一个标量与数组的运算,是标量与数组每个元素之间的运算,这种特性称之为**标量扩展**。

(2) 关系运算

关系运算用于比较两个同维数组或同维向量的对应元素,结果为一个同维的逻辑数组。如果运算对象一个为标量,另一个为数组或向量,那么先进行标量扩展,然后再比较。关系运算符及其说明见表 1.4。

(3) 逻辑运算

MATLAB 提供了两种类型的逻辑运算:一般逻辑运算和捷径运算,见表 1.5。

表 1.4 关系运算符

关系运算符	说明
<	小于
<=	小于等于
>	大于
>=	大于等于
==	等于
~=	不等于

表 1.5 逻辑运算符与函数

运算类型	运算符与函数	说明
元素运算	& (and)	逻辑与
	(or)	逻辑或
	~ (not)	逻辑非
	xor	逻辑异或
捷径运算	&&	对标量值的捷径与
		对标量值的捷径或

捷径运算符只能对标量值执行逻辑与和逻辑或运算。捷径运算首先判断第一个运算对象,如果可以知道结果,直接返回,而不继续判断第二个运算对象。捷径运算提高了程序的运行效率,可以避免一些不必要的错误。例如:

```
>> x = b&&(a/b > 10) % 相当于 x = (b&&(a/b > 10))
```

如果 b 为 0,捷径运算符就不会计算(a/b > 10)的值了,也就避免了被 0 除的错误。

(4) 位运算

位运算相关函数见表 1.6。

表 1.6 位运算相关函数

位运算函数	说明	位运算函数	说明
bitand	位与	bitget	返回指定位的数值,值为 0 或 1, double 型
bitor	位或	bitset	设定指定位的值为 0 或 1, 返回运算结果
bitcmp	位比较,反码	bitshift	移位运算,返回运算结果
bitxor	位异或	swapbytes	翻转字节的位顺序,返回运算结果

【注意】 位运算函数的输入必须同为无符号整数、无符号整数数组或标量浮点数,且输出与输入的数值类型一致。若输入为标量浮点数, MATLAB 会先将其转换为无符号整数,再进行位运算。

(5) 特殊运算符

除了以上运算符, MATLAB 还经常使用一些特殊的运算符,见表 1.7。

表 1.7 特殊运算符

特殊运算符	说明	特殊运算符	说明
[]	生成向量和矩阵	...	续行符
{ }	给单元数组赋值,或创建一个空单元数组	,	分隔矩阵下标和函数参数
()	在算术运算中优先计算;封装函数参数;封装向量或矩阵的下标	;	在括号内结束行;禁止表达式显示结果;隔开声明
=	用于赋值语句	:	创建矢量、数组下标;循环迭代
'	在矩阵或向量之后表示复共轭转置;两个'之间的字符为字符串	%	注释;格式转换定义符中的初始字符
.	域访问	@	函数句柄,类似于 C 语言中的取址运算符

(6) 运算优先级

在包含前面介绍的运算符的表达式中,运算顺序按优先级进行。优先级高的先执行,同优先级的从左至右执行。运算符按优先级从高到低排列见表 1.8。

表 1.8 运算优先级

序号	运算符	备注	序号	运算符	备注
1	()	优先级最高	7	< <= > >= == ~=	
2	.' ^ ' ^		8	&	
3	+ - ~	单目运算	9		
4	. * ./ \ * / \		10	&&	
5	+ -	双目运算	11		优先级最低
6	:				

5. 特殊表达式

MATLAB 提供了一种非常重要的特殊表达式:字符串计算表达式。字符串计算有两个函数:eval 计算包含表达式的字符串;feval 以字符串或函数句柄为函数名进行计算。

eval 的调用方法为:eval(expression)。expression 为包含 MATLAB 表达式的字符串。

feval 的调用方法为:[y1, y2, ...] = feval(fun_name, x1, ..., xn)。fun_name 为所执行函数的函数名,可以为字符串或函数句柄;x1, ..., xn 为函数 fun_name 的输入参数;y1, y2, ... 为函数 fun_name 的输出参数。

eval 调用函数时可写成[a1, a2, a3, ...] = eval('function(b1, b2, b3, ...)')。但这种形式避开了 MATLAB 分析程序的严格检查,可能产生不可捕捉的错误和不希望看到的结果,不提倡使用。

在命令行分别输入下面 3 个程序。

程序一:

```
>> t = 0:.1:2 * pi;
>> y = eval('sin(t)');
>> plot(y)
```

程序二:

```
>> t = 0:.1:2 * pi;
>> y = feval('sin', t);
>> plot(y)
```

程序三:

```
>> t = 0:.1:2 * pi;
>> y = feval(@sin, t);
>> plot(y)
```

若您对此书内容有任何疑问,可以凭在线交流卡登录 MATLAB 中文论坛与作者交流。

3个程序的运行结果相同,如图1.1所示。

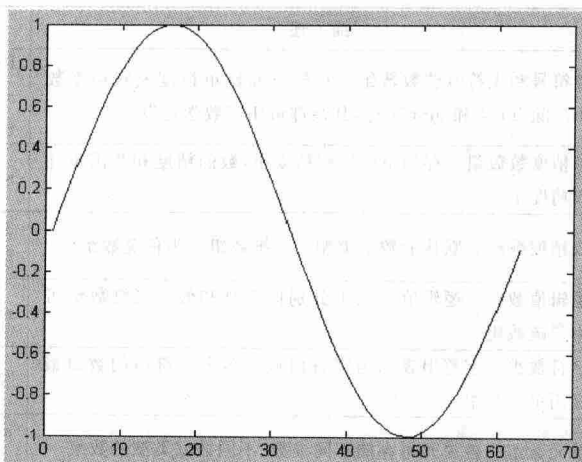


图 1.1 eval 与 feval 的运行结果

eval 函数的使用非常灵活。例如:

```
>> eval('a1 = 3')
a1 =
3
```

eval 函数可以从字符串中创建一个变量。

在这里,eval 的功能是将字符串转化为数字,相当于 str2num 或 str2double,但 eval 的转换速度要快于 str2num 和 str2double。str2num 和 str2double 的区别在后面讲解。

1.1.2 数据类型

MATLAB 有 17 种基本的数据类型,每种类型的数据都以矩阵或数组形式存在。矩阵或数组的最小尺寸是 0×0 ,它能够扩展为任意大小的 n 维数组。所有的基本数据类型用小写字符显示在图 1.2 中。

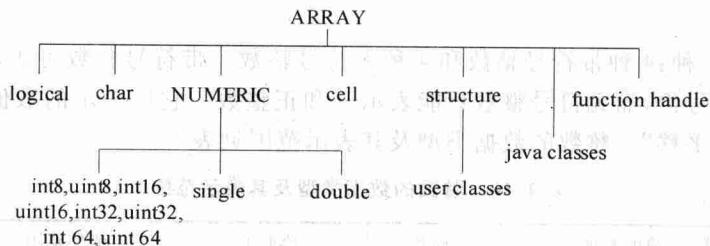


图 1.2 基本数据类型

表 1.9 详细描述了这些数据类型。

表 1.9 数据类型

数据类型	描述	举例
int8, uint8, int16, uint16, int32, uint32, int64, uint64	带符号和无符号整数数组。存储空间比单精度或双精度数小。除 int64 和 uint64 外,其余都可用于数学运算	flag=uint16(0); a=uint8(3)+uint8(10); b=int8(1:10)
single	单精度数数组。存储空间比双精度小,数的精度和范围也比双精度小	single(5*10^38)
double	双精度数组。默认的数字类型。二维数组可为稀疏数组	3*10^300; 5+6i
logical	逻辑值数组。逻辑值 0 或 1 分别代表真和假。二维数组可为稀疏数组	magic(4) > 7
char	字符数组。字符串表示为字符向量。多个字符串的数组最好用单元数组	'MATLAB'
cell array	单元数组。各单元可存储不同维数、不同数据类型的数组	a{1,1}='Red'; a{1,2}=magic(4)
structure	结构数组。类似于 C 语言中的结构体。每个域可保存不同维数和不同类型的数组	a.day=12; a.color='Red'
function handle	函数句柄,指向一个函数。能传递给其他函数	@sin
user class	从用户定义的类构造的对象	polynom([0 -2 -5])
Java class	从一个 Java 类构造的对象	java.awt.Frame

1. 数值型

数值型数据包括有符号和带符号整数、单精度和双精度浮点数。MATLAB 默认将所有数字值存为双精度浮点数(double 型),但整数和单精度数组更节省内存空间。

所有的数值型都支持基本的数组操作,如下标操作和尺寸重塑。除 int64 和 uint64 外,都可用于数学运算。

以下介绍的内容为:

- 整数;
- 浮点数;
- 复数;
- 其他常用函数。

(1) 整数

整数类型有 8 种:4 种带符号整数和 4 种无符号整数。带符号整数可表示负整数、0 和正整数,最高位为符号位,而无符号整数只能表示 0 和正整数。它们表示的数值范围一样大,只是对范围进行了“平移”。整数的数据类型及其表示范围见表 1.10。

表 1.10 整数的数据类型及其表示范围

数据类型	值的范围	转换函数	数据类型	值的范围	转换函数
单精度 8 位整数	$-2^7 \sim 2^7 - 1$	int8	无符号 8 位整数	$0 \sim 2^8 - 1$	uint8
单精度 16 位整数	$-2^{15} \sim 2^{15} - 1$	int16	无符号 16 位整数	$0 \sim 2^{16} - 1$	uint16
单精度 32 位整数	$-2^{31} \sim 2^{31} - 1$	int32	无符号 32 位整数	$0 \sim 2^{32} - 1$	uint32
单精度 64 位整数	$-2^{63} \sim 2^{63} - 1$	int64	无符号 64 位整数	$0 \sim 2^{64} - 1$	uint64

若您对此书内容有任何疑问,可以凭在线交流卡登录 MATLAB 中文论坛与作者交流。