

高  
等  
学  
校  
教  
材

# 信号与系统

# MATLAB 实践

● 孟桥 董志芳 王琼



高等教育出版社  
Higher Education Press

高等学校教材

# 信号与系统

## MATLAB 实践

● 孟桥 董志芳 王琼



高等教育出版社  
Higher Education Press

## 内容简介

本书以信号与系统、数字信号处理等课程相关内容作为应用背景,结合 MATLAB 工具,介绍如何应用计算机技术解决工程实践中遇到的问题。

全书共分八章。第一章为 MATLAB 简介,包括基本计算、作图语句及其系统帮助的使用。第二、三、四、五章则分别以信号分析、系统分析以及系统对信号响应分析等信号与系统课程相关内容为主线,介绍了如何用 MATLAB 解决相关的问题。第六章结合通信中的调制解调以及取样定理,应用计算机仿真工具对相关内容进行研究和验证。第七章为数字滤波器设计,介绍如何用 MATLAB 工具设计出各种类型的数字滤波器。第八章结合信号与系统中的框图、系统串并联等部分内容,简单介绍了 MATLAB 中的图形化仿真工具 SIMULINK。

本书强调知识的基础性、算法的共通性以及基础知识与实践的结合,内容上力求简单明了,易于掌握,同时也留下了自主探究的空间,强化对学生自主学习能力的培养。

本书可以作为电气信息类专业相关课程的实践指导教材,也可以作为学生在学习信号与系统课程之后,进一步理解相关理论知识,掌握计算机仿真基本方法及了解相关理论在实际工程中应用的自学性读物。

## 图书在版编目(CIP)数据

信号与系统 MATLAB 实践/孟桥,董志芳,王琼. —北京:  
高等教育出版社,2008. 12

ISBN 978 - 7 - 04 - 024889 - 0

I. 信… II. ①孟…②董…③王… III. 信号系统 - 计  
算机辅助计算 - 软件包, MATLAB IV. TN911. 6 - 39

中国版本图书馆 CIP 数据核字(2008)第 178350 号

策划编辑 杜 炜 责任编辑 曲文利 封面设计 于文燕 责任绘图 尹 莉  
版式设计 王艳红 责任校对 杨凤玲 责任印制 毛斯璐

出版发行 高等教育出版社  
社 址 北京市西城区德外大街 4 号  
邮政编码 100120  
总 机 010 - 58581000

经 销 蓝色畅想图书发行有限公司  
印 刷 北京机工印刷厂

开 本 787 × 960 1/16  
印 张 13  
字 数 240 000

购书热线 010 - 58581118  
免费咨询 800 - 810 - 0598  
网 址 <http://www.hep.edu.cn>  
<http://www.hep.com.cn>  
网上订购 <http://www.landaco.com>  
<http://www.landaco.com.cn>  
畅想教育 <http://www.widedu.com>

版 次 2008 年 12 月第 1 版  
印 次 2008 年 12 月第 1 次印刷  
定 价 16.70 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 24889 - 00

## 郑重声明

高等教育出版社依法对本书享有专有出版权。任何未经许可的复制、销售行为均违反《中华人民共和国著作权法》，其为人将承担相应的民事责任和行政责任，构成犯罪的，将被依法追究刑事责任。为了维护市场秩序，保护读者的合法权益，避免读者误用盗版书造成不良后果，我社将配合行政执法部门和司法机关对违法犯罪的单位和个人给予严厉打击。社会各界人士如发现上述侵权行为，希望及时举报，本社将奖励举报有功人员。

反盗版举报电话：(010)58581897/58581896/58581879

传 真：(010)82086060

E - mail: dd@ hep. com. cn

通信地址：北京市西城区德外大街4号

高等教育出版社打击盗版办公室

邮 编：100120

购书请拨打电话：(010)58581118

# 前 言

随着当代计算机技术的不断发展,计算机逐渐融入了人们生活、工作和学习的方方面面。计算机应用已经成为当代大学生必须掌握的基本技能之一,他们不仅需要掌握计算机编程语言,更需要掌握利用计算机解决实际工作中遇到的问题能力,而这其中一个重要的方面就是应用计算机解决工程中实际计算问题的能力。在当前高等教育对实践性教学环节的重视程度不断提高的背景下,很多院校都开设了这方面的实践教学课程,而且大家都将目光不约而同地投向了一款优秀的计算软件——MATLAB。

我校在 20 世纪 90 年代末开始在电类专业中开设了 MATLAB 实践课程,该课程在信号与系统课程之后开设,通过信号与系统中的一些实例,介绍 MATLAB 软件的应用,在十多年的长期教学实践中取得了良好的效果。本教材就是在这门课程的讲义基础上进一步整理深化的成果。本教材的编写思路,就是以信号与系统、数字信号处理等课程相关内容作为应用背景,介绍如何应用计算机技术解决工程实践中遇到的问题。

首先,信号与系统等课程与工程实践有着直接联系。目前,绝大多数国内的工科电类本科专业都开设了这门课程,它涉及很多数学方面的知识,同时各个知识点都有很强的应用背景,而这些应用背景自然成为了计算机在这方面应用的极好素材。在这些背景下,学生通过实验能够理解各种算法的前因后果,加深对相关技术的认识,起到举一反三的作用;同时,还可以加深学生对相关理论课程中知识点的理解,便于为他们学习后续课程,如自动控制、通信原理和图像处理等,使用计算机工具验证、加深对相关理论和方法的理解打下良好的基础。书中很多理论和应用涉及的知识点以管致中教授主编的《信号与线性系统》(第四版)(书中简称为教材)中的内容为参考,同时许多信号与系统教材中都可以找到。

其次,本书主要目标力求介绍计算机仿真计算中涉及的一些基本方法。计算机工具软件并不限于 MATLAB 一种,只不过这种语言对于信号与系统、数字信号处理等的仿真计算方面更容易发挥其特长。虽然 MATLAB 提供了非常丰富的工具和函数,但是它也不是万能的,很多问题也需要用户自己编程实现;同时,在数字信号处理器等实际应用场合中也无法直接使用 MATLAB 编写的 m 程序(虽然 MATLAB 提供了编译、形成 C 语言程序等手段试图解决这个问题),更

不用说是调用其函数库了。所以掌握计算机仿真计算的基本方法才是本课程的重要的基本目标之一。例如,虽然这里所有的例题都是以 m 程序的形式提供的,但是在编写的时候应尽量避免使用 MATLAB 所独有的语句(例如冒号赋值计算语句),很多例题使用常规的 for、while 等循环语句来实现;还有些算法虽然在 MATLAB 中已经有现成的函数,但是这里也给出了具体实现的程序,以便学生掌握算法的实质,如第五章中介绍的状态方程的龙格-库塔数值计算法。在这些 m 程序的基础上,按照相似的软件流程,相关的程序在其他的计算机语言中也可以实现<sup>①</sup>。

最后,本书并不是一本全面系统介绍 MATLAB 软件的书籍,而是基于相关知识的需要,循序渐进地对其进行介绍。MATLAB 提供了很多丰富的界面,针对各种专门应用设计了很多工具箱;同时,随着软件版本的不断升级, MATLAB 目前不仅具有科学计算功能,而且也提供了界面设计、可执行文件编译以及实时数字信号等方面功能的支持,体系非常庞大。对这些内容的详细介绍不可能在本书中完成,而且这些详细的内容在 MATLAB 提供的帮助中有非常详细的解释,没有必要通过教材的形式再次介绍。所以,本书只是将其作为学习的工具,而不是学习的目的,对其知识的掌握不求全面。例如,对 MATLAB 操作的介绍集中于其命令窗口,通过窗口指令可以得到所有需要的信息,而对 MATLAB 其他的界面、按钮、快捷键和子窗口等内容不进行详细介绍(SIMULINK 方面的内容除外);对丰富的 MATLAB 函数仅仅择其中重要的介绍,而且对很多这些函数也只介绍其最基本的功能,对其扩展的功能由学生自己去探究。本书除了第一章是对 MATLAB 的入门性介绍以外,其他章节都是按照信号与系统以及数字信号处理等课程的内容组织和安排的,在对相关的实际问题数值计算介绍中逐步深入介绍 MATLAB,其内容融入了各章的基本内容之中。

与本书相关的课程是一门实践性课程,在教学中课程讲授只占总学时很少的部分,而其他部分则通过自主学习的方式完成,需要学生自己去实践和体会。所以本书在编写上也考虑到了这方面的因素,力求其能够成为一个“自学教材”。各章内容有相关理论、应用背景和 MATLAB 相关内容的简要介绍,也提供了丰富的例题供学生参考和体会。后面的实验题目则按照内容的需要,分为操作式(指定操作流程,让学生观察结果)、目标函数式(要求学生设计一个指定的 MATLAB 函数)和习题式三种。操作式的实验可以直接加深学生对相关知识的理解;目标函数式的实验可以提高学生的编程能力,而习题式的实验则可以培养学生综合分析、解决问题的能力。本教材还重点突出了对 MATLAB 的帮助(help 命令)的介绍,“授人以渔”,给学生一把深入探究 MATLAB 的利器,借以强

<sup>①</sup> 可能一些程序在使用其他编程语言实现的时候,需要使用一些相关的科学计算软件包。

化学生主动学习的能力。

本书共分八章。

第一章简单介绍了 MATLAB 入门知识,包括基本计算、作图语句以及其系统帮助的使用。

第二、三、四、五章则以信号分析、系统分析以及系统对信号响应分析等信号与系统课程相关内容为主线,介绍了如何用 MATLAB 解决相关的问题。其中,第三章为拓展内容,简单介绍了图像信号处理的一些内容,将信号与信号处理的研究范畴扩展到了二维,展示了信号与系统课程与后续的图像信号处理等课程的联系。

第六章结合通信中的调制解调以及取样定理,介绍了如何应用计算机仿真工具对相关内容进行研究和验证。

第七章的内容是数字滤波器设计。这一章与其他各章不同之处在于其教学重点在设计,介绍如何用 MATLAB 工具完成各种类型的数字滤波器设计。数字滤波器的概念在信号与系统课程中有所介绍,而其设计则是数字信号处理中的一个重要内容。考虑到目前数字滤波器在实际应用中使用越来越广泛,教会学生如何用 MATLAB 设计出滤波器比相关的滤波器设计理论公式更有实用价值。

SIMULINK 是 MATLAB 推出的一个图形化的仿真计算工具,通过它可以形象生动地完成系统的构造和仿真,在实际教学中的效果远远优于枯燥的公式推导以及流程化的仿真程序。在第八章中简要介绍了 SIMULINK 工具的使用,通过这部分内容学生不仅能够进一步加深对信号与系统中的框图、系统串并联等部分内容的理解,也可以为后续的自动控制、数字信号处理、通信系统等课程的学习中相关模型的建立和仿真计算打下基础。

为了便于读者自学,教材除了正常的叙述性文字,还列举了大量仿真实例,并用灰色背景突出显示。读者可以在 MATLAB 中进行验证。此外,实验中的提高题目加“\*”表示。

本教材由孟桥主编,其中第一、二章由孟桥编写,第三章至第六章以及附录由董志芳编写,第七、八章由王琼编写,孟桥完成了整个书稿的统稿、修改和润色。东南大学信息科学与工程学院副院长、国家级教学名师吴镇扬教授审阅了全稿并提出了很多非常宝贵的修改意见,谨致以衷心感谢。该教材的出版得到了高等教育出版社和东南大学的大力支持,在此也致以诚挚的谢意。

限于编者的水平,本书中错误和不足之处难免,恳请广大读者指正。

编者

2008年8月于东南大学

# 目 录

<b>第一章 MATLAB 简介</b> .....	1
§ 1-1 科学计算、计算机语言与 MATLAB .....	1
§ 1-2 MATLAB 入门 .....	3
§ 1-3 MATLAB 中的简单作图语句 .....	17
§ 1-4 MATLAB 的系统帮助 .....	22
[实验一] 熟悉 MATLAB 基本操作 .....	24
<b>第二章 信号分析</b> .....	26
§ 2-1 信号的计算 .....	26
[实验二] 信号的运算 .....	36
§ 2-2 信号的谱分析 .....	38
[实验三] 信号的谱分析 .....	47
§ 2-3 信号的正交变换 .....	49
[实验四] 信号的正交变换 .....	54
<b>第三章 二维信号分析</b> .....	56
§ 3-1 图像信号的格式与表示 .....	56
§ 3-2 图像的频域分析与处理 .....	60
§ 3-3 图像的空间域处理 .....	70
[实验五] 二维信号的分析与处理 .....	76
<b>第四章 系统分析</b> .....	79
§ 4-1 系统在 MATLAB 中的表示 .....	79
§ 4-2 系统的频率响应与波特图 .....	85
[实验六] 系统在 MATLAB 中的表示和频率特性 .....	92
§ 4-3 系统的极零点及稳定性分析 .....	94
§ 4-4 线性反馈系统稳定性分析 .....	101
[实验七] 系统极零点及其稳定性 .....	107
<b>第五章 系统对信号响应分析</b> .....	110
§ 5-1 系统对激励的响应 .....	110
[实验八] 系统的响应 .....	120
§ 5-2 通过状态方程求系统响应的数值解 .....	122
§ 5-3 非线性系统的数值解 .....	129
[实验九] 状态方程的数值解法 .....	134



<b>第六章 调制解调与取样定理</b> .....	137
§ 6-1 AM 调制与解调 .....	137
§ 6-2 脉冲幅度调制 .....	143
[实验十] 信号调制与解调 .....	149
§ 6-3 取样定理 .....	150
[实验十一] 取样定理 .....	154
<b>第七章 数字滤波器设计</b> .....	156
§ 7-1 数字滤波器简介 .....	156
§ 7-2 IIR 滤波器设计 .....	157
§ 7-3 FIR 滤波器设计 .....	161
[实验十二] 滤波器设计 .....	164
<b>第八章 系统仿真</b> .....	166
§ 8-1 系统仿真工具 SIMULINK 简介 .....	166
[实验十三] SIMULINK 模拟仿真 .....	179
§ 8-2 SIMULINK 进阶 .....	180
[实验十四] 系统仿真综合 .....	190
<b>附录:MATLAB 基本运算符表</b> .....	192
<b>索引</b> .....	194
<b>参考文献</b> .....	198

# 第一章 MATLAB 简介

## §1-1 科学计算、计算机语言与 MATLAB

在航空航天、地质勘探、天气预报和建筑工程等领域的科学研究和工程实践中,常常需要进行大量的科学计算、仿真以及作图工作,这些工作在工程的设计阶段非常重要,一个成功、精确的科学计算可以大大节省后续开发的费用,为工程项目的成果打下坚实的基础。如何更快、更精确地进行计算自然就成为人们关心的问题,为此人们研制了各种方便计算的工具,从古代的算盘,到近代的计算尺、手摇计算器等,都是为了方便计算而发明的工具。在 20 世纪 60 年代,中国的科学家们利用这些早期的计算器,自力更生,在两弹一星的研制中创造了一个个奇迹。电子计算机的出现大大地方便了科研中的科学计算和仿真工作,使得人们摆脱了以往的计算尺、手摇计算器和稿纸等原始的计算工具,工作效率和计算精度有了极大地提高。随着计算机的速度、内存、硬盘等资源的不断扩大,计算机仿真的速度和计算规模不断地扩大。现在人们可以利用计算机求解复杂系统的特性,求解高维代数方程,证明一些数学难题,甚至可以在计算机上模拟核爆炸这样复杂的实验,节省了大量的研究开支。如今计算机已经在各个领域起到了非常重要的作用。

计算机的运行离不开软件,而为了方便使用计算机进行科学计算,相应的计算机语言的研制自然就成为人们关心的重点。早期的计算机编程主要是由程序员直接设计机器码。当时,程序员把生成一个有效程序看做一项复杂而又富有创造力的艺术活动。他们把精力主要都花在克服当时计算机引发的各种障碍上,如没有索引寄存器,没有嵌入式浮点操作等。那时程序员是一个技术高度复杂的职业,在 1954 年前后,花在程序员上的投资已接近计算机本身的价值。而且计算机的使用时间有  $1/4 \sim 1/2$  是花在程序的排错上。这样,编程和排错占了一台计算机运行投资的  $3/4$ 。随着计算机的价格越便宜,这种状况越加严重。特别是对于一些复杂的科学计算,花费在编程上的时间远远超出了算法设计和实际的计算时间,成为科学研究的瓶颈。所以研究一种简单、与计算机无关、适合科学计算使用的计算机语言就成了当务之急。

历史上第一个用于科学计算的高级语言当属 20 世纪 50 年代初期美国国际


商用机器 (IBM) 公司的约翰·巴克斯领衔开发的 FORTRAN 语言。从表面看来,它就像是英语速记和代数的结合,同工程师及科学家日常使用的代数公式有很多相似之处。经过一段时间培训,他们就可以十分轻松地将科学和工程问题转化为计算机能够“理解”的语言。实际上其名称 FORTRAN 就是“公式转换器 (Formula Translator)”的简称,这个名称形象地描述了这种计算机语言的特色,成为所有计算机高级语言的鼻祖。而被称为“FORTRAN 语言之父”约翰·巴克斯也在 1977 年获得了计算机界的最高荣誉之一——图灵奖。其后,随着 C 语言、BASIC 语言、PASCAL 语言等其他各种类型的高级语言的出现,FORTRAN 的应用已经不太常见了,但是由于历史的原因在很多科学计算上还是可以见到它的身影,它在计算机应用和计算机语言发展历史上起着不可替代的里程碑作用。

虽然 FORTRAN 等高级语言提供了如日常代数公式一样的计算机语言,大大方便了编程,但是科学计算往往还是需要很多复杂的计算,其中涉及的编程工作量还是很大的。计算机语言只能完成标量的加减乘除,如果需要复杂的计算就需要编制大量的程序。例如一个二维矩阵的运算往往需要至少 2 层 for 循环,矩阵的求逆计算的过程就更加复杂。由此有人将科学计算中常用的算法编成一系列的子函数,构成“科学计算库”,方便其他人使用。但是这种“库”的方式在实际使用中还是需要大量的调整,以便与实际工程计算程序和计算平台相融合。所以人们不断地在寻求更加简化的科学计算程序的设计方式,在此期间也推出了不少各具特色的实用软件,MATLAB 程序就是其中的佼佼者之一。

MATLAB 语言是 20 世纪 70 年代后期,美国墨西哥大学计算机系系主任 Cleve Moler 教授在讲授线性代数课程时构思并开发的,其名称 MATLAB 是 Matrix Laboratory 的缩写,即矩阵实验室。顾名思义,这个语言是为了解决科学计算中大量使用的矩阵计算而设计的,它利用了当时数值线性代数领域最高水平的 EISPACK 和 LINPACK 两大软件包中可靠的子程序,是用 FORTRAN 语言编写的集命令翻译、科学计算于一身的一套交互式软件系统。人们给出一条矩阵计算命令,立即就可以得出该命令的结果,并且可以方便地将计算结果用图形的方式显示出来。刚开始,MATLAB 是为了解决矩阵计算的问题,后来随着其功能的不断推进,其计算上也同时向统计分析、多元方程求解、系统综合等各种科学计算领域发展,同时在功能上也从单一的科学计算向界面设计、外部接口,甚至实时信号处理等多方面发展,而且可以运行于 Windows 9X/NT、OS/2、Macintosh、Sun、UNIX、Linux 等多种操作系统平台,逐渐演变成为一种流行的科学与工程计算的软件工具,现在有人称它为“第四代”计算机语言,在国内外高校和研究部门正扮演着重要的角色。

## §1 - 2 MATLAB 入门

### 1. MATLAB 界面与启动

虽然 MATLAB 可以在很多操作系统上运行,在本书中还是以 Windows 为例介绍 MATLAB 的使用。在安装了 MATLAB 程序以后,在桌面上就可以看到图标 (这里以 MATLAB7.0 为例,其他不同版本号的图标类似),点击该图标,就可以进入到 MATLAB 的操作桌面,如图 1-1 所示。下面是默认操作桌面。这个界面与 Word 之类的程序界面很相似,由菜单、快捷工具栏以及三个主要的窗口部分组成。这种结构与常见的 Word 等 Windows 下的应用程序非常相似,其中最重要的是其右侧的命令窗口(Command Window)。从 DOS 界面下的 MATLAB 开始,无论在什么操作平台下,命令窗口都是 MATLAB 最核心的部分,在这个窗口中可以完成所有的 MATLAB 的操作。其他部分只是起辅助作用,帮助用户进行快速操作,完成对 MATLAB 环境的观察。我们对 MATLAB 的介绍就着重对命令窗口进行。对于其他部分的功能可以参见一些 MATLAB 的参考书籍,或者自己在使用中摸索,很快就可以掌握。

命令窗口中的操作是通过命令行来实现的。这种方式只要显示器终端可以显示字符就可以实现,比图形化的界面显示和控制要简单得多,在 DOS、UNIX、Linux 等操作系统中被广泛地应用。用鼠标在命令窗口中点击一下,就可以进入命令行编辑状态,这时候可以看见命令窗口中的“>>”号后面有一个小的光标在闪动,表示用户可以在这里输入命令。这里“>>”被称为“命令行提示符”,它标志着一个命令行的开始。用户可以在提示符后面输入相应的命令,并按 Enter 键,输入的命令就可以被执行。

**【例 1-1】** 简单计算:

(1) 求  $3 + 4 - 5$  的算术运算结果。

(2) 求  $\tan(1)$  的数值。

解:(1) 用键盘在 MATLAB 命令窗口中输入以下内容

```
>> 3 + 4 - 5
```

在上述表达式输入完成后,按 Enter 键,该指令就被执行, MATLAB 给出计算结果

```
ans =
```

```
2
```

```
>>
```

其中的 ans 是 answer 的缩写,表示当前命令执行的结果。如果输入的是一个直接的数学计算表达式,这结果就是计算的结果。在本例中计算的结果等于 2。

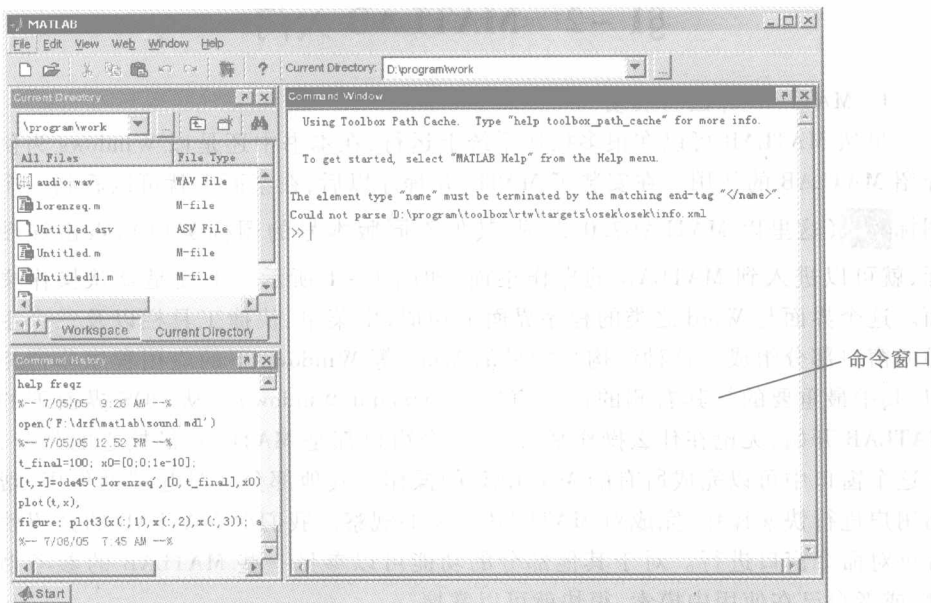


图 1-1 MATLAB 启动界面

当前命令结束以后,在下一行又出现了提示符“>>”,表示一个新的命令行的开始,用户可以输入新的命令。

(2) 在命令行下输入

```
>> tan(1)
```

运行结果

```
ans =
```

```
1.5574
```

当出现一些异常的结果时, MATLAB 将给出一定的警告。例如下面的运算可以得到这样的结果

```
>> 1/0
```

```
Warning: Divide by zero.
```

```
ans =
```

```
Inf
```

```
>>
```

```
>> 0/0
```

```
Warning: Divide by zero.
```

```
ans =
```

NaN

```
>>
```

其中 Inf 表示 $\infty$  (Infinity), NaN 表示  $0/0$ 、 $0 \times \infty$  或  $\infty/\infty$  等无法确定的运算结果 (Not a Number)。在 MATLAB 运行过程中要时刻注意系统给出的警告, 因为这种警告的出现往往意味着程序设计中存在一些错误或者考虑不周到的地方, 必须加以改进。

## 2. MATLAB 中的变量与赋值

与其他计算机语言一样, MATLAB 也可以将数值赋给一个变量, 然后用这个变量进行计算。例如, 要给变量 a 赋值 1.12, 可以在命令行中输入

```
>> a = 1.12
```

可以得到赋值结果

```
a =
```

```
1.1200
```

```
>>
```

MATLAB 中的变量不需要预定义类型, 名称可以任意指定, 只要满足下面几个条件:

(1) MATLAB 的变量名是区分大小写的, 例如“Ab”和“ab”在 MATLAB 中表示不同的变量。

(2) 变量的第一个字符必须为英文字母, 而且不能超过 63 个字符<sup>①</sup>。

(3) 变量名可以包含下连字符、数字, 但不能为空格符、标点。例如“a\_b”、“a123b”都是合法的变量名, 而“a,b”、“a:b”等不能作为变量名。

(4) 变量名不包含任何中文。

从上面的例子也可以看出 MATLAB 与其他计算机语言不同之处, 就是变量可以在赋值的时候自动加以定义, 而不用像 FORTRAN、C 语言那样事先定义数据类型 (这一点与 BASIC 语言很相近)。这样做的好处是使用比较方便, 但是缺点在于这种隐性的变量定义在实际使用中会引起一些程序设计上的失误。在使用中必须加以重视。

MATLAB 环境中定义了一些常用的变量, 例如上面介绍的 NaN 和 Inf, 还有表示圆周率  $\pi$  的变量 pi。用户定义的变量名不能与之重复。

MATLAB 中数一般都用双精度 (Double), 而且不仅可以表示实数, 同时也可以表示复数。例如要将 a 赋一个复数  $1 + j3$ , 可以输入

```
>> a = 1 + 3j
```

---

<sup>①</sup> 最大变量名长度可能随 MATLAB 的版本和运行平台的不同而有差异。

可以得到

```
a =
```

```
1.0000 + 3.0000i
```

```
>>
```

这里请注意两点：

(1) 在一般文献中,有的使用  $i$  表示复数,有的使用  $j$ 。MATLAB 统一使用  $i$  表示虚数单位。但是在赋值的时候,无论用  $i$  或者  $j$ ,MATLAB 都可以识别。

(2) 在一般的教科书上表示虚数的时候  $i$  或者  $j$  都出现在虚部的开头,例如  $j2$ ,但是在 MATLAB 中则是出现在虚部的尾部,例如上例中的  $3.0000i$ 。这是为了避免与变量名相混淆。在 MATLAB 中, $i2$  表示一个变量名,而  $2i$  表示一个虚数。

MATLAB 中的变量不仅可以表示整数、浮点、复数,更加重要的是可以用一个变量表示一个矩阵。几乎所有 MATLAB 中的变量都是矩阵,正如其名称所说明的,矩阵(Matrix)是构成 MATLAB 的最基本的单元。例如,如果要将一个矩阵

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

赋给变量  $a$ ,只要在命令行中输入

```
>> a = [1,2,3;4,5,6;7,8,9]
```

就可以在命令窗口得到赋值结果

```
a =
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

由于这里是对一个变量赋值,所以结果直接就是变量名,而不是前面简单计算中的  $ans$ 。

这里请读者注意一下赋值语句中,矩阵各个元素的位置:

- (1) 矩阵赋值内容含括在一对中括号“ $[ ]$ ”之中。
- (2) 赋值元素按照先行后列的顺序输入。
- (3) 每行的各个元素之间用空格或者逗号隔开;行与行之间用分号隔开,分号表示一个新的矩阵行的开始。

(4) 每一行含有的元素个数(列数)必须相等,否则系统会给出出错提示。

MATLAB 中的矩阵变量可以通过其所在的行和列的位置进行单独访问。例如要读出上面定义的  $a$  变量的第 1 行第 2 列元素的内容,可以在命令行中输入

```
>> a(1,2)
```

可以得到

```
ans =  
2
```

也可以通过这种方式单独地改变矩阵中某个元素的数值。例如:将原来变量 a 矩阵中第 1 行第 2 列的元素改成 10,可以输入

```
>> a(1,2) = 10
```

可以得到

```
a =  
1 10 3  
4 5 6  
7 8 9
```

可见对应的元素的数值发生了变化。从用法上看,这里的矩阵与 C、FORTRAN 等语言中的数组很相似,但是 MATLAB 中矩阵行列号是从 1 开始的,这一点与 C 等计算机语言中的数组的下标有所不同。

与 C、FORTRAN 等其他计算机语言相比, MATLAB 中的矩阵相当于其他高级语言中的数组。但是, MATLAB 直接提供了矩阵之间的计算功能,这是很多其他高级语言没有的。此外, MATLAB 中的矩阵的元素可以是复数,这给很多需要用到复数的科学计算程序的设计带来了很大的方便。

在实际的计算程序中上,矩阵也可以作为一般的数组或者数列使用。例如,在计算中可以用  $1 \times 10$  的矩阵表示一个 10 个元素构成的数列,可以用一个  $640 \times 480$  的矩阵表示一个二维数组来存储一个像素尺寸为  $640 \times 480$  的二维图形或者图像。MATLAB 中提供了一些专用的数列赋值的语句。例如,可以用冒号赋值方式定义一个等差数列,具体的语句为

```
a = s1: s2: s3
```

其中 a 是被赋值的变量, s1 为等差数列的起始值, s2 为步长,而 s3 为终止值。MATLAB 自动按照给定的参数产生等差数列(或者向量)。例如,在 MATLAB 命令窗口下输入

```
>> a1 = 1: 10: 100
```

可以得到数列(或者矩阵)

```
a1 =  
1 11 21 31 41 51 61 71 81 91
```

这种赋值语句中的 s2 可以省略,这时等差数列的差固定为 1。例如,下列赋值的结果为

```
>> a2 = 1: 10
```

```
a2 =
```



1 2 3 4 5 6 7 8 9 10

这种赋值方法将给后面的计算带来很多方便。

在 MATLAB 中除了可以提取出矩阵的元素以外,也可以提取一个大矩阵中的部分内容并赋值给一个小矩阵。例如:

(1) 命令“ $a = A(3,1:3)$ ”表示将矩阵 A 的第 3 行的第 1~3 列的内容赋值给变量 a (这里的结果 a 是一个 1 行 3 列的矩阵)。

(2) 命令“ $b = B(5:7,2)$ ”表示将矩阵 B 的第 2 列第 5 行到第 7 行的内容赋值给变量 b (这里的结果 b 是一个 3 行 1 列的矩阵)。

(3) 命令“ $c = C(5:7,2:5)$ ”表示将矩阵 C 的第 5 行到第 7 行中的第 2 列到第 5 列的内容赋值给变量 c (这里的结果 c 是一个 3 行 4 列的矩阵)。

(4) 命令“ $d = D(3,:)$ ”表示将矩阵 D 的第 3 行所有列的内容赋值给变量 d。

(5) 命令“ $e = E(:,2)$ ”表示将矩阵 E 的第 2 列所有行的内容赋值给变量 e。

除了上面介绍的赋值语句以外, MATLAB 也提供了其他很多快捷的矩阵赋值方式用于产生一些特殊的矩阵,例如:可以通过 eye 命令产生单位矩阵 I,通过 zeros 命令产生全零矩阵、通过 ones 命令产生全 1 矩阵。具体相关内容可以参考 MATLAB 的帮助文件(详见 § 1-4 对 MATLAB 帮助系统的介绍)。

从上面介绍的 MATLAB 的相关内容中,可以看到,对每一个用户输入的赋值或者计算指令, MATLAB 立刻显示出结果。这样做的优点是可以让用户立刻看到结果,但是在实际的应用中,一个计算往往需要几个步骤,通过好几个指令才能够完成,如果显示过多的指令执行结果过多则会造成显示屏幕显示的信息非常杂乱。为此 MATLAB 提供了一种控制指令结果输出的机制:只要在命令的结尾加上分号“;”,该指令的结果就不会被显示。还以上面的等差数列赋值为例,如果在原来的赋值语句后面加上分号,其结果将为

```
>> a2 = 1:10;
```

```
>>
```

可见在这时候将跳过对赋值语句结果的显示,直接显示下一个命令的提示符。通过适当的控制命令结果的显示,将使 MATLAB 命令窗口的显示内容变得简洁。此外,从这里也可以看出分号“;”在 MATLAB 有着非常重要的作用:在命令行结尾出现的分号可以屏蔽命令的结果的显示,在矩阵赋值命令的矩阵元素之间的分号则标志着矩阵现有一行的结束和新的的一行的开始。

前面我们提到 MATLAB 没有专门的变量定义语句,变量的定义是在变量赋值时一同进行的。MATLAB 提供了两个命令供使用者检查当前 MATLAB 环境中定义的变量信息,一个是“who”,另一个是“whos”。例如在进行了一定的变量赋值以及计算之后,在 MATLAB 命令窗口下输入“who”,可以得到