

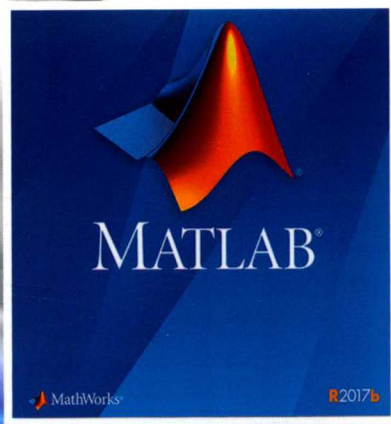


扫描下载本书
示例程序代码

MATLAB

面向对象和C/C++ 编程

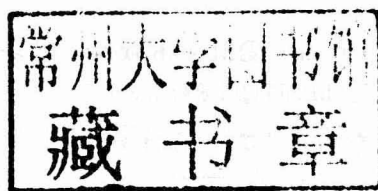
凌 云 张 志 涌 编 著



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

MATLAB 面向对象和 C/C++ 编程

凌 云 张志涌 编著



扫描下载本书
示例程序代码

北京航空航天大学出版社

内 容 简 介

本书正文由两篇组成:A篇“MATLAB面向对象编程”和B篇“MATLAB面向C/C++编程”。每篇各5章,各篇内容按照由浅入深的原则编排,篇中各章内容的设计,既有联系,又相对独立,以适应读者系统阅读 and 随时翻阅的不同需求。A篇的内容包括:MATLAB的类定义和基本操作、类对象数组、类方法的重载与覆盖、类的继承与组合,以及事件与响应。B篇的内容包括:MATLAB外部应用的数据接口、MATLAB对C/C++程序的调用、C/C++程序对MATLAB的MAT函数库的调用、C/C++程序对MATLAB引擎函数库的调用、MATLAB编译器和独立应用程序的开发。

全书包含70多个示例。所有示例都由作者围绕叙述内容精心设计而成。示例的背景知识适于理工类高校任何专业的师生,适于理工类软件开发的各类专业从业人员。所有示例表述清晰、完整,关键程序代码都附有注释,便于读者阅读和理解;示例中的运行结果,都可被读者重现,以帮助读者建立学习信心。此外,本书每个示例的运行代码或程序都保证准确、完整。读者亲自实践本书示例时所需的代码,既可以直接循书本代码键入而得,也可以扫描二维码或按出版社提供的下载地址下载。

本书既可以作为理工类高校各专业师生的学习、教学用书,也可以用作科研院所各类研发人员的自学用书和参考手册。

图书在版编目(CIP)数据

MATLAB面向对象和C/C++编程 / 凌云, 张志涌编著

. -- 北京:北京航空航天大学出版社,2018.3

ISBN 978-7-5124-2693-1

I. ①M… II. ①凌… ②张… III. ①Matlab 软件—程序设计②C语言—程序设计 IV. ①TP317②TP312.8

中国版本图书馆CIP数据核字(2018)第058764号

版权所有,侵权必究。

MATLAB面向对象和C/C++编程

凌 云 张 志 涌 编 著

责任编辑 蔡 喆

*

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(邮编100191) <http://www.buaapress.com.cn>

发行部电话:(010)82317024 传真:(010)82328026

读者信箱:goodtextbook@126.com 邮购电话:(010)82316936

北京宏伟双华印刷有限公司印装 各地书店经销

*

开本:787×1092 1/16 印张:27 字数:691千字

2018年3月第1版 2018年3月第1次印刷 印数:4000册

ISBN 978-7-5124-2693-1 定价:58.00元

前 言

1. 编写背景

经 30 多年的实践检验、市场筛选和时间凝炼, MATLAB 已经成为国际上广泛认可和普遍使用的数学分析和数值计算软件,也是教学、科研、工程界极可信赖的科学计算环境、标准仿真平台和十分可靠的数学资源库。

近年来,随着深度学习、大数据分析、人工智能的突破和迅速市场化,学术界、工程界、科研部门和开发机构所面临的软件建模和程序开发问题越来越复杂。它们或输入来源多种,或输出任务多样,或输入输出间关系错综复杂,或被处理数据规模庞大,或离散事件随机多发,或数学计算和多界面操控高度交叉。面对这类综合性问题,既不再能单凭一连串数学公式的“MATLAB 面向过程”编程所能解决,也不再可单凭“C/C++ 面向对象”编程所能应对。于是,“MATLAB 面向对象”和“MATLAB 面向 C/C++”编程的应用需求强烈凸显。这也促使高校本科和研究生的 MATLAB 教学必须向高层次方向发展。

另一方面,从书籍出版角度看,迄今可见的千余种 MATLAB 书籍,绝大多数只涉及数学计算、建模及“面向过程”的编程,论及“面向对象”或“面向 C/C++”的书籍则屈指可数。

科技发展的需求和书籍出版的现实,驱使我们决心编写一本以理工共通知识为素材、以系统阐述及具体示例为经纬、面向对象和面向 C/C++ 的 MATLAB 用书。我们希望此书能帮助已经学习和使用 MATLAB 的高校学生、科研人员较快地理解和掌握“面向对象”编程和“面向 C/C++”编程。同时,我们也希望,具有 C/C++ 编程基础的高校学生及科研开发人员能借助本书理解和掌握“C/C++ 与 MATLAB”的联合编程。

2. 编写宗旨与特点

本书两位作者基于自身的长期教学和程序开发经验,对本书的内容组织和程序设计遵循如下宗旨:

- 在保持内容系统完整的前提下,充分体现 MATLAB 的特征。
 - ◇ 凡使用 MATLAB 进行过数值和科学计算的读者,都定会被 MATLAB 特有的高效便捷灵活的魅力所吸引。MATLAB 面向对象编程的概念、语法、结构和函数也同样展现其一贯的简洁友善的特性。这是因为 MATLAB 语言本身是由 C/C++、JAVA 等原生型面向对象编程语言构建的。本书的内容组织将让读者充分体会 MATLAB 的这些优点,以较短的时间掌握面向对象编程的精髓。
 - ◇ 基于 MATLAB 数学计算以数组形式实施的本质,本书在面向对象编程的内容组织中,特别强调了类属性取值数组时类方法编写的注意事项,以及类对象数组的创建和应用。

◇ 基于 MATLAB 各类数据都采用同一种阵列形式组织的基础性特征,本书在面向 C/C++ 编程的内容组织中,对专司 MATLAB 和 C/C++ 间数据交换的 `mxArray` 和 `mwArray` 阵列结构给予特别阐述。

● 在保证示例典范性的前提下,尽量使示例内容贴近大部分 MATLAB 用户的专业知识。

◇ 本书示例都是作者围绕演示目的精心设计的。所涉知识都由 MATLAB 类专业用户通识的复数、极坐标、向量、加减法等构成。

◇ 读者学习和实践本书示例时,一方面可免受专业隔阂的心理影响,另一方面可从通识要素的应用中得到向其专业拓展的启示。

● 在保证内容系统性的前提下,尽量保证示例的独立性。

◇ 全书章节内容按其自身逻辑和认知过程编排,保证内容的连贯性、系统性,以利读者系统阅读学习。

◇ 各章节示例设计遵循三个原则:一,示例内容和代码围绕所在章节内容设计;二,对可能生疏的非本章节命令或函数,给予简明注释;三,保证示例程序代码的可运行性、可验证性。在非系统学习的情况,读者可根据需要挑选示例在电脑上进行实际的演练体验。读者还可根据本书示例结果是否被重现,而修整自己可能的失误,建立起理解和掌握本书内容的自信。

● 全书 A、B 两篇的章节内容,按各自内在关系由浅入深编排。

◇ 本书每篇最前的一两个示例,都较简单而不失综合性。初学读者,通过这些示例就可感受或触摸到该篇内容的轮廓特点。

◇ 几乎每章的最后,都安排了较为综合的示例,以向读者提供进一步的联想和启示。

3. 全书结构梗概

全书由正文、附录和参考文献组成。

● A 篇 MATLAB 面向对象编程

“第 1 章 类的定义和基本操作”的内容是全篇的基础。它包括:类和对象概念;类定义的基本框架;属性及其秉质;方法及其秉质;构造函数编写准则及典型结构等。通过本章的学习,读者就可初步具备以“面向对象”的思维解题编程的能力。

“第 2 章 对象数组”的内容包括:对象数组的创建;对象数组的编址及寻访;特殊对象数组生成;对象数组的常用操作。由于 MATLAB 自身的各种运算是建立在数组基础上的,所以,相较于 C/C++ 等而言,本章内容在 MATLAB 面向对象编程中具有特殊意义。

“第 3 章 重载与覆盖”的主要内容有:重载与覆盖在类设计中的意义;算符与对应的后台控制函数;改变对象显示的重载和覆盖技术。

“第 4 章 类的继承与组合”系统介绍:类继承定义和子父类关系;继承的多态性;抽象类概念及应用;属性包含型类组合。此外,本章最后,以示例形式展现很常用的“界面类+算法类”程序设计模式。

“第 5 章 事件与响应”的内容有:事件/响应机制;响应函数的型式;事件发布方/响应函数之间的数据传递与侦听件。该章最后采用综合示例形式介绍事件和响应在客户端/服务器

(C/S)架构软件设计与开发中的应用。

● B 篇 MATLAB 面向 C/C++ 编程

“第 6 章 数据接口”集中叙述 MATLAB 与 C/C++ 等交换信息所依赖的 mxArray、mwArray 等两种阵列结构的创建、读取、赋值和删除等函数的具体调用格式,并着重阐述指针、引用、堆内存管理在相关函数设计和具体调用中的作用。本章内容具有较强的查阅功能。

“第 7 章 MATLAB 调用 C/C++ 程序”围绕在 MATLAB 中调用由 C/C++ 源文件编译而来的 MEX 文件展开。具体内容有:MEX 文件的定义;源文件的构成和格式;MEX 文件的生成;MEX 文件执行流程;MEX 函数库。该章最后叙述在 Visual Studio 环境中编译、调试 MEX 文件。

“第 8 章 C/C++ 程序调用 MAT 函数库”的内容为:MATLAB 的 MAT 函数库,以及在 MATLAB 环境和 Visual Studio 环境内编译调用 MAT 函数库的 C/C++ 程序。

“第 9 章 C/C++ 程序调用 MATLAB 引擎”系统阐述:MATLAB 引擎概念和功用;引擎函数库;在 MATLAB 环境和 Visual Studio 环境内编译调用引擎函数库的 C/C++ 程序。该章最后以综合示例形式介绍“C/C++ 程序+MATLAB 引擎”联合开发模式的在线调试方法。

“第 10 章 MATLAB 编译器”专述:如何将 MATLAB 函数编译生成可独立运行的程序或动态链接库;MATLAB 编译器的外部 C/C++ 编译器配置;生成外用文件的 mcc 命令和应用编译器。该章最后以综合示例展示:MFC 应用程序框架及其综合应用。

● 附 录

“附录 A 基础准备与入门”包含两节,“A. 1: MATLAB 的配置及入门”和“A. 2: Visual Studio 集成开发环境入门”。A. 1 节为不熟悉 MATLAB 的读者而设,用于简介:MATLAB 的安装启动;在 Windows 中对 MATLAB 的设置;MATLAB 桌面和编辑器的使用入门。A. 2 节为不熟悉 C/C++ 的读者而设。简介 Visual Studio 2010 的安装启动;简介 Visual Studio 集成开发环境的使用入门。

“附录 B 索引”汇集了本书所涉及的所有 MATLAB 和 C/C++ 函数及命令。所有函数及命令按英文字母的次序排列,并列出与之对应的章节序号。该索引可为读者提供另一条据英文关键词检索本书内容的途径。

● “参考文献”

列出了本书成文的文献基础。读者如有疑问或想寻根问源,这部分资料十分有用。

4. 读者对象和使用建议

本书的编写动机和本书的内容组织,决定了本书的读者对象为:理工类高校的本科生、研究生;科研机构中从事数学仿真或应用开发的科技人员。

为帮助读者阅读、使用,本书作者提出如下建议。

● A 篇的内容组织和阅读建议:

- ◇ A 篇适于循章节次序,由浅入深,循序渐进,但也不必拘泥于连续通读、一气到底。
- ◇ 实际上,在 A 篇第 1 章仔细阅读并实践示例后,读者就不难理解“面向对象”编程的思维模式,而顺利跨入“面向对象”编程的大门。

◇ 其余章节,读者完全可以根据自己需要挑选阅读。本书的内容组织及示例命令后的注释说明,都能有力地帮助读者克服理解困难。

● B 篇的内容组织和阅读建议:

◇ 第 6 章数据接口的内容主要用于各种函数的调用格式介绍。该章内容是为读者查阅使用方便而编写的,因此不必系统阅读。读者只要浏览几节,能理解 `mxArray` 和 `mwArray` 阵列结构在 MATLAB 和 C/C++ 之间交换数据中的作用便可。

◇ 其余各章内容之间没有前后关联,相互独立。因此,读者可以根据需要,分章学习。本书作者建议读者,认真阅读各示例的相关说明和命令后的注解,并静下心来在机器上对示例进行操作实践。这样,定能事半功倍地掌握 MATLAB 与 C/C++ 的联合编程。

5. 致 谢

本书成稿历经 3 年。在这期间,周围朋友、同事及相关部门领导的期盼、鼓励,帮助作者克服了困惑、犹豫和疲惫。在这不短的成稿期间,作者之所以能有独立的空间、专心静思的时间、无扰的环境从事本书的写作和试验,那都因为背后有着家人的理解、支持、帮助、鼓励和默默的付出。值此本书出版之际,作者向他们一并表示诚挚的感谢。

最后,还要感谢北京航空航天大学出版社对我们完稿时间一再后延的宽容和理解,并感谢他们为本书出版所做的一切。

本书虽历时 3 年,几经易稿,示例几经修改和验证,并最终在 MATLAB R2017b 和 Visual Studio 2010 环境下定稿,但仍难免缺陷,或发生因版本变更引起的不适性错误。对此,我们恳请各相关方面专家和广大读者不吝指正,以便再次印刷时加以修正。联系的电子邮箱为: `ly00519@126.com`; `zyzh@njupt.edu.cn`。

凌 云 张志涌
2018 年 1 月于南京

版权声明

© MATLAB[®] 和 Simulink[®] 是 MathWorks 公司注册商标,为叙述简洁,本书用 MATLAB 代替 MATLAB[®],用 Simulink 代替 Simulink[®],特此说明。

© 2018,北京航空航天大学出版社有限公司,版权所有,侵权必究。

未经本书出版者书面许可,任何单位和个人不得以任何形式或手段复制或传播本书及其所附电子文档的内容。

资料下载说明

本书为读者免费提供书中示例程序源代码,请扫描二维码下载。读者也可以通过以下网址从百度云盘下载全部资料:

<https://pan.baidu.com/s/1S-7HTXpxZIOgvLxWmIMS6Q>

如您阅读本书时遇到任何问题,或者有好的意见和建议,请发送电子邮件至 goodtextook@126.com,或致电 010-82317036 咨询处理。

二维码使用提示:使用手机微信或浏览器扫描二维码,可将资料保存到用户自己的“百度网盘”中,未安装“百度网盘”的用户可以直接下载使用。



目 录

A 篇 MATLAB 面向对象编程

第 1 章 类定义和基本操作	3
1.1 引 导	3
1.1.1 面向过程的程序设计方法概述	3
1.1.2 面向对象的程序设计方法概述	4
1.1.3 面向过程和面向对象程序设计的区别	4
1.2 类和对象	9
1.3 类定义的基本结构	9
1.4 类的属性	12
1.4.1 属性块和属性	12
1.4.2 属性块的乘质	13
1.4.3 对象属性名的获取和属性值的查询	14
1.4.4 属性赋值的三个不同阶段	19
1.4.5 属性赋值和 set 函数	19
1.4.6 从属属性和 get/set 函数	23
1.5 类的方法	28
1.5.1 方法块和方法的构成	28
1.5.2 方法块的乘质	32
1.5.3 类对象方法名的查询和获取	33
1.5.4 类方法的调用与程序设计	36
1.6 类对象构造函数	41
1.6.1 理解和应用默认构造函数	42
1.6.2 自编构造函数的编写准则	45
1.6.3 自编构造函数的典型结构	46
1.6.4 自编构造函数的编写	47
1.7 全值类和句柄类的差别	54
1.7.1 两类的类定义和对象性状差别	54
1.7.2 两类对象变量在函数内外传递中的差别	64
1.7.3 全值类和句柄类适用性差别	73
1.8 类方法中的数组运算和矩阵运算	74

第 2 章 对象数组	78
2.1 对象数组的构成和创建	78
2.1.1 对象数组基础	78
2.1.2 小规模对象数组的方括号创建法	78
2.1.3 对象数组的外循环创建法	82
2.1.4 对象数组的 deal 属性赋值创建法	85
2.1.5 对象数组的内循环创建法	86
2.1.6 对象数组的编址赋值创建法	88
2.2 对象数组的编址和寻访	91
2.2.1 对象数组的编址	91
2.2.2 对象数组的元素对象寻访	91
2.2.3 对象数组属性值的获取和组织	94
2.3 对象数组操作和特殊对象数组	98
2.3.1 对象数组的常用操作命令	98
2.3.2 空对象数组	98
2.3.3 特殊对象数组的生成	101
第 3 章 重载和覆盖	111
3.1 引 导	111
3.1.1 重载和覆盖的基本含义	111
3.1.2 什么情况下需要重载和覆盖	111
3.1.3 MATLAB 语言格式及控制函数的默认行为表现	111
3.2 类别转换和算符重载	112
3.2.1 对象类别的转换方法函数	112
3.2.2 MATLAB 算符及后台控制函数	115
3.2.3 代数运算符的重载	116
3.3 改变对象显示的重载和覆盖技术	121
3.3.1 显示函数、对象形态及显示内容	121
3.3.2 改变对象显示的覆盖技术	127
3.3.3 改变对象显示的重载技术	137
第 4 章 类的继承与组合	142
4.1 类继承定义	142
4.1.1 单一父类继承	142
4.1.2 多个父类继承	143
4.2 子类如何构造父类	147
4.3 类继承中的覆盖与多态性	151
4.4 抽象类及其应用	157

4.4.1	抽象类的定义	157
4.4.2	抽象类的应用	163
4.5	类的组合应用	170
4.5.1	属性包含型组合	170
4.5.2	界面和算法两类的包含型组合	174
第 5 章	事件与响应	182
5.1	概 述	182
5.1.1	事件概念及其代码特征	182
5.1.2	侦听和响应概念	183
5.2	事件定义和消息发布函数	183
5.2.1	事件块及乘质	184
5.2.2	消息发布函数 notify	185
5.3	侦听件的创建和响应函数的型式	185
5.3.1	侦听件创建函数的一般格式	185
5.3.2	不同型式响应函数的注册格式	186
5.4	事件/响应的简单应用示例	188
5.5	事件在复杂软件开发中的综合应用	196
5.5.1	MATLAB 和 C++ 中事件应用的区别	196
5.5.2	事件在 MATLAB 和 C++ 联合开发中的应用	197

B 篇 MATLAB 面向 C/C++ 编程

第 6 章	数据接口	233
6.1	MATLAB 数据	233
6.1.1	MATLAB 阵列	233
6.1.2	数据存储方式	234
6.1.3	C/C++ 数据接口	236
6.2	mxArray 阵列结构	237
6.2.1	创建和删除 mxArray	237
6.2.2	读取和赋值 mxArray 中的数据	243
6.2.3	获取 mxArray 属性	249
6.2.4	mxArray 的 API 函数综合应用示例	251
6.3	mwArray 阵列结构	259
6.3.1	创建 mxArray	260
6.3.2	读取和赋值 mxArray 中的数据	262
6.3.3	获取 mxArray 属性	264

6.3.4	mwArray 类方法综合应用示例	265
第 7 章 MATLAB 中调用 C/C++ 程序 269		
7.1	关于 MEX 文件的一般性说明	269
7.2	C_MEX 源文件的构成	270
7.3	编译生成 C 语言 MEX 文件	272
7.3.1	编译器的配置	272
7.3.2	借助编译器生成 MEX 文件	273
7.3.3	编译指令 mex 简介	274
7.4	MEX 文件的执行流程	275
7.5	MEX 函数库介绍	276
7.6	C 语言 MEX 文件应用示例	277
7.6.1	MATLAB 环境下编译 C_MEX 源文件	277
7.6.2	Visual Studio 环境下编译 C_MEX 源文件	282
7.6.3	Visual Studio 环境下调试 MEX 文件	287
第 8 章 C/C++ 程序对 MAT 函数库的调用 291		
8.1	MATLAB 中数据的输入输出方法	291
8.2	MAT 函数库介绍	291
8.3	调用 MAT 函数库的 C/C++ 源码应用程序编写	293
8.4	利用 Visual Studio 环境编写调用 MAT 函数库程序	298
第 9 章 C/C++ 程序调用 MATLAB 引擎 309		
9.1	MATLAB 引擎概念和功用	309
9.2	MATLAB 引擎函数库介绍	309
9.3	MATLAB 环境下编译调用引擎函数库的 C/C++ 源文件	310
9.4	Visual Studio 环境编译调用引擎函数库的 C/C++ 源文件	316
9.5	C/C++ 应用程序与 M 函数联合调试技术	319
9.5.1	联合调试流程	319
9.5.2	联合调试应用示例	320
第 10 章 MATLAB 编译器 327		
10.1	MATLAB 编译器概述	327
10.1.1	编译器的功能	328
10.1.2	编译器的局限性	328
10.1.3	把脚本文件改写为函数文件	328
10.2	MATLAB 编译器配置和入门	330
10.2.1	为正确使用 MATLAB 编译器进行配置	330
10.2.2	MATLAB 编译器使用入门	331

10.2.3 编译指令 mcc 简介	333
10.2.4 编译器工作流程介绍	338
10.3 应用程序编译器	344
10.3.1 应用程序编译器概述	344
10.3.2 应用程序编译器使用入门	344
10.4 编译器生成可独立执行应用程序示例	351
10.5 编译器生成共享动态链接库示例	357
10.5.1 M 函数文件生成 C 语言动态链接库及调用	357
10.5.2 M 函数文件生成 C++ 语言动态链接库及调用	361
10.5.3 MFC 应用程序框架及其综合应用	365

附 录

附录 A 基础准备与入门	395
A.1 MATLAB 的配置及入门	395
A.1.1 MATLAB 的安装与启动	395
A.1.2 为 MATLAB 配置 Windows 搜索路径	397
A.1.3 MATLAB 基本使用方法	399
A.2 Visual Studio 集成开发环境入门	403
A.2.1 Visual Studio 2010 的安装与启动	403
A.2.2 Visual Studio 环境的基本使用方法	405
附录 B 索 引	413
B.1 MATLAB 的英文关键词	413
B.1.1 面向对象编程的函数和命令	413
B.1.2 面向 C/C++ 的函数和命令	415
B.2 C/C++ 的英文关键词	418
参考文献	419

A 篇

MATLAB 面向对象编程

过程和对象是客观世界中的真实存在。对象自身的运动、变化都以过程的形式体现；不同对象之间的交流也多以过程实施。所以，在为实际项目构建仿真模型时，既要有“面向对象”的思维框架，又要有“面向过程”的细致分析。现在，“面向对象”和“面向过程”的编程思想、实践和应用，几乎已相互交织地构成了所有成功的软件产品。MATLAB 自身就是以“面向对象”框架组织的，而其成千上万的使用者都便捷地使用 MATLAB 语言进行着各式各样的“面向过程”的编程。

随着模式识别、深度学习、大数据分析等日益广泛地应用于各种场合，其软件模型已经越来越难以仅凭“面向过程”的思维进行编程。然而，不论是正在接受 MATLAB 教学的高校学生，还是已经熟练运用

MATLAB 的高校教师 and 各类科研人员,他们中相当一部分人也只学习或仅使用传统而经典的“MATLAB 面向过程”编程。

本篇是作者在考虑现实需求后专门为学习和使用 MATLAB 软件的理工科高校师生及科研人员编写的,专门讲述 MATLAB 面向对象编程。作者编写本篇内容时,充分考虑了本书读者的知识背景,书中所用的术语、函数命令、叙述风格和所涉数学知识都是理工科专业最易理解和接受的。

全篇共 5 章,所有示例都在 MATLAB R2017b 下验证通过。

第 1 章是全篇的基础。它涉及面向对象编程的各种基本概念和基本要素。学过本章后,读者就可初步具备以“面向对象”的思维进行解题编程的能力。

第 2 章“对象数组”是 MATLAB 面向对象编程的特色所在。与 C/C++ 等语言不同,MATLAB 的各种运算是建立在数组基础上的。

第 3、4、5 章讲述“重载与覆盖”“类的继承与组合”“事件与响应”。它们充分体现了与“面向过程”编程的不同,鲜明地反映出“面向对象”编程的灵活和优势。

本篇最后一个示例采用 MATLAB 和 C/C++ 联合开发的客户端/服务器(C/S)架构软件。此例比较综合,可供读者进一步深入学习。

第 1 章

类定义和基本操作

本章专门阐述 MATLAB 面向对象编程的基础性内容。本章共分 8 节。第 1.1 节通过对比,着重阐述面向对象编程的思想。第 1.2 节从概念着手,介绍类和对象的关系。第 1.3 节给出了类定义的框架性结构,以帮助读者了解类定义的梗概。

从第 1.4 节起,用 3 节篇幅分节详述类属性、类方法及类对象构造函数。第 1.4 节不仅详细阐述了属性、属性乘质的概念、设置及使用含义,而且还着重表述了查询/赋值函数 `get/set` 的配用模式。第 1.5 节详述类方法的构建、乘质设置;此外,专辟节次叙述类方法的点调用格式、函数调用格式以及它们与类方法程序代码的适配。第 1.6 节专述类对象构造函数的特殊定义格式和重要性,详述构造函数的自编准则和典型结构。

第 1.7 节对比性地描述了 MATLAB 的两大基类——全值类和句柄类在定义、内涵、操作性状上的差异;归纳性地叙述全值类和句柄类的不同应用场合。

本章最后一节是为 MATLAB 的数组运算本质而专设的,该节专论当类属性值为数组时,如何设计类方法。

1.1 引 导

1.1.1 面向过程的程序设计方法概述

“面向过程的程序设计(Procedure Oriented Programming)”的编程思想以过程为中心。该方法从分析解决问题所需的步骤着手,然后用模块依次实现这些步骤,使用时再顺序调用那些模块。

面向过程是人们解决问题的一种直观思考方式。当程序规模不很大时,面向过程的设计方法很具优势:程序流程清晰,模块组织结构简明,特别适合于时序性的单任务问题的解决。以数据接收处理为例,整个过程如图 1.1-1 所示。

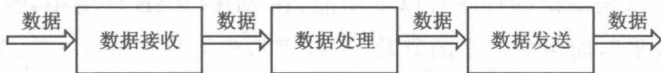


图 1.1-1 数据接收处理系统流程

处理过程涉及数据接收、数据处理、数据发送三个模块。这三个模块严格按照时序节拍运行;首先运行“数据接收”模块,随后“数据处理”模块工作,最后运行“数据发送”模块;各模块运行所需的数据被严格按照时间顺序传递。在这种以过程为核心的编程思想中,编程人员的主要任务就是:编写各种模块实现过程中的各个步骤。在这种设计方式中,模块的作用至关重要,而数据只起辅助的串接作用。

面向过程的程序设计方法,首先关心的是功能,强调以模块(过程)为中心,采用模块化、自

顶向下、逐步求精的设计过程。系统是实现模块功能的函数和过程的集合,结构清晰、设计简单、可读性好。然而,用户需求的更改和硬件技术的不断发展,会引发模块间流动数据的改变,轻则局部修改作为系统基本组成部分的功能模块,重则需完全重建系统,进而导致软件维护和升级的困难。因此,采用面向过程方法设计的系统,前期开发入手快、后期不得不频繁改动的案例,在实际应用中屡见不鲜。

1.1.2 面向对象的程序设计方法概述

“面向对象的程序设计(Object Oriented Programming)”的编程思想以对象为中心。该设计方法以类和继承为构造机制,通过类对象的构造、组合、互动来认识、理解和刻画客观世界。

现实生活中很多事物都可以理解为由单个或多个对象组合而成。例如几个人组合成一个家庭,多个家庭组合成社区,多个社区组合成城市等等。这里,每个人可以看成是一个对象,都有身高、体重之类的属性以及代表人行为能力的对象方法。


面向对象的程序设计方法,首先把对象的各种属性和行为进行抽象,进而形成类。类的设计以数据(类属性)为中心,并为这些数据的操作提供各种接口(类方法)。类的属性作为类的核心数据,一般不对外开放,外部只能通过类的接口来访问这些数据。这样的设计有三大优点。

- 类使用的方便:用户只需要了解类的对外公共接口函数,而不需要过多了解类内部的详细设计。
- 类属性的相对稳定:用户不能直接对类属性进行修改,有的属性甚至不能访问,从而避免了用户的误操作所引起的系统错误。
- 类及系统的维护、升级方便。

类、对象、继承、组合等概念的引入使用,显然令面向对象的设计方法具有一定的优势,能为生产可重用的软件构件和解决软件的复杂性问题提供一条有效的途径,特别是在具有界面的人机交互操作系统和软件出现之后,面向对象的程序设计方法得到了很快的发展。

1.1.3 面向过程和面向对象程序设计的区别

前面两小节介绍了面向过程和面向对象程序设计方法的基本概念,本小节将以复数表达和复数加法为例,向读者简略而完整地展示:在 MATLAB 环境中,面向过程的编程思维形式和程序格局;面向对象的编程思维形式和程序格局。

 **【1.1-1】** 要求编写出的程序有以下功能:在 MATLAB 环境中,数据的赋值、保存、运算保持在“双精度水平”,而由此产生的复数,不以直角坐标形式的 $x+jy$ 显示,而以极坐标形式的 $A\angle\theta$ 显示;极坐标表示法中的 θ 应以度数为单位,而不是弧度单位;加法运算应采用通常习惯的“+”符号进行,而不采用调用 M 函数的方式进行。

1) 复数的数学表述方式

为读者理解方便,先把理论数学教学和应用中复数的常见表达方式归纳如下:

● 实虚部形式:

◇ 直角坐标表达: $c = x + jy$;

◇ 三角函数表达: $c = A(\cos \theta + j\sin \theta)$ 。

● 模幅角形式: