

国外计算机科学教材系列

面向对象 程序设计导论

An Introduction to Object-Oriented
Programming with Java

[美] C. Thomas Wu 著

侯国峰 李 湘 张 迪 译

Mc
Graw
Hill



电子工业出版社

Publishing House of Electronics Industry
URL: <http://www.phei.com.cn>

国外计算机科学教材系列

面向对象程序设计导论

An Introduction to Object-Oriented
Programming with Java

[美] C.Thomas Wu 著

侯国峰 李 湘 张 迪 译

电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书通过大量与现实世界程序设计有关的实例全面而详尽地讲解了面向对象的程序设计思想和设计方法。Java 作为一种面向对象的程序设计语言,在本书中仅仅是作为讲授的工具。

本书作为面向对象的程序设计的入门教科书,用于一个学期的课程。已经具有其他传统面向过程程序设计语言经验的开发人员也可以使用本书作为面向对象的程序设计、图形用户界面以及事件驱动程序设计的入门。

本书可供大专院校计算机专业的本科生使用。

Efrem G. Mallach: An Introduction to Object-Oriented Programming with Java,2/e
ISBN 0-07-239684-9

Copyright © 2001 by the McGraw-Hill Companies, Inc.

Authorized translation from the English language edition published by McGraw-Hill, Inc.

All rights reserved. For sale in the People's Republic of China only.

本书中文简体字版由电子工业出版社和美国麦格劳-希尔国际公司合作出版。未经出版者许可,不得以任何方式复制或抄袭本书的任何部分。

版权所有,翻印必究

图书在版编目(CIP)数据

面向对象程序设计导论 / (美) 吴 (Wu, C.T) 著; 侯国峰等译. -北京: 电子工业出版社, 2001.8

(国外计算机科学教材系列)

书名原文: An Introduction to Object-Oriented Programming with Java

ISBN 7-5053-6930-X

I. 面... II. ①吴...②侯... III. 面向对象语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2001) 第 057465 号

丛 书 名: 国外计算机科学教材系列

书 名: 面向对象程序设计导论

原 书 名: An Introduction to Object-Oriented Programming with Java

著 者: [美] C. Thomas Wu

译 者: 侯国峰 李 湘 张 迪 等

责任编辑: 寇国华

印 刷 者: 北京市天竺颖华印刷厂

出版发行: 电子工业出版社 URL: <http://www.phei.com.cn>

北京市海淀区万寿路 173 信箱 邮编: 100036 电话: 68279077

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 42.5 字数: 1023 千字

版 次: 2001 年 8 月第 1 版 2001 年 8 月第 1 次印刷

书 号: ISBN 7-5053-6930-X
TP · 3948

定 价: 68.00 元

版权贸易合同登记号 图字: 01-2001-2495

凡购买电子工业出版社的图书,如有缺页、倒页、脱页、所附磁盘或光盘有问题者,请向购买书店调换。

若书店售缺,请与本社发行部联系调换。

出版说明

随着21世纪的到来,计算机技术的发展更加迅猛,在各行各业的应用更加广泛,越来越多的高等院校增设了有关计算机科学的课程内容,或对现有计算机课程设置进行了适当调整,以紧跟前沿技术。在这个教学体系和学科结构变革的大环境下,对适合不同院系、不同专业、不同层次的教材的需求量与日俱增。此时,如果能够借鉴、学习国外一流大学的先进教学体系,引进具有先进性、实用性和权威性的国外一流大学计算机教材,汲取其精华,必能更好地促进中国高等院校教学的全面改革。

美国 Prentice Hall 出版公司是享誉世界的高校教材出版商,自1913年成立以来,一直致力于教材的出版,所出版的计算机教材为美国众多大学采用,其中有不少是专业领域中的经典名著,已翻译成多种文字在世界各地的大学中使用,成为全人类的共同财富。许多蜚声世界的教授、学者都是该公司的资深作者,如道格拉斯·科默(Douglas E. Comer)、威廉·斯大林(William Stallings)等。早在1997年,电子工业出版社就从 Prentice Hall 引进了一套计算机英文版专业教材,并将其翻译出版,同时定名为《国外计算机科学教材系列》(下称:第一轮教材)。截至2000年12月,该系列教材已出版23种,深受读者欢迎,被许多大学选为高年级学生和研究生教材或参考书。

4年过去了,已出版的教材中多数已经有了后续版本。因此,我们开始设计新一轮教材(第二轮教材)的出版,成立了由我国计算机界著名专家和教授组成的“教材出版委员会”,并结合第一轮教材的使用情况和师生反馈意见,组织了第二轮《国外计算机科学教材系列》出版工作。

第二轮教材的出版原则为:

1. 引进 Prentice Hall 出版公司2000年和2001年推出的新版教材,作为替换版本。
2. 在著名高校教授的建议下,除了从 Prentice Hall 新选了一些教材之外,还从 McGraw-Hill 和 Addison Wesley Longman 等著名专业教材出版社、麻省理工学院出版社和剑桥大学出版社等著名大学出版社引进了一些经典教材,作为增补版本。
3. 对于第一轮中无新版本的优秀教材,我们将其作为沿用版本,直接进入第二轮使用。
4. 对于第一轮中翻译质量较好且无新版本的教材,我们将其进行了修订,也作为沿用版本,进入第二轮使用。

这次推出的教材覆盖学科范围广、领域宽、层次多,既有本科专业课程教材,也有研究生课程教材,以适应不同院系、不同专业、不同层次的师生对教材的需求。广大师生可自由选择和自由组合使用。

按照计划,本轮教材规划出版37种,其中替换版本8种,新增版本14种,沿用版本15种。教材内容涉及的学科方向包括网络与通信、操作系统、计算机组织与结构、算法与数据结构、数据库与信息处理、编程语言、图形图像与多媒体、软件工程等。本轮教材计划于2001年7月前全部出版。教材的使用年限平均为3年。我们还将陆续推出一些教材的参考课件,希望能为授课老师提供帮助。

为了保证本轮教材的选题质量和翻译质量,我们约请了清华大学、北京大学、北京航空航天大学、复旦大学、上海交通大学、南京大学、浙江大学、哈尔滨工业大学、华中科技大学、西安交通

大学、国防科学技术大学、解放军理工大学等著名高校的教授和骨干教师参与了本轮教材的选题、翻译和审校工作。他们中既有讲授同类教材的骨干教师和博士,也有积累了几十年教学经验的教授和博士生导师。

在本轮教材的选题、翻译和编辑加工过程中,为提高教材质量,我们做了大量细致的工作,包括:

1. 对于新选题和新版本进行了全面论证。
2. 对于沿用版本,认真审查了前一版本教材,修改了其中的印刷错误。
3. 对于译者和编辑的选择,达到了专业对口。
4. 对于从英文原书中发现的错误,我们通过与作者联络、从网上下载勘误表等方式,一一做了修改。
5. 对于翻译、审校、编辑、排版、印刷质量进行了严格的审查把关。

通过这些工作,保证了本轮教材的质量较前一轮有明显的提高。相信读者一定能够从字里行间体会到我们的这些努力。

今后,我们将继续加强与各高校教师的密切联系,为广大师生引进更多的国外优秀教材和参考书,为我国计算机科学教学体系与国际教学体系的接轨做出努力。

由于我们对国际计算机科学、我国高校计算机教育的发展存在认识上的不足,在选题、翻译、出版等方面的工作中还有许多有待提高之处,恳请广大师生和读者提出批评和建议。

电子工业出版社
2001 年春

教材出版委员会

主 任	杨芙清	北京大学教授 中国科学院院士 北京大学信息与工程学部主任 北京大学软件工程研究所所长
委 员	王 珊	中国人民大学信息学院院长、教授
	胡道元	清华大学计算机科学与技术系教授 国际信息处理联合会通信系统中国代表
	钟玉琢	清华大学计算机科学与技术系教授 中国计算机学会多媒体专业委员会主任
	谢希仁	中国人民解放军理工大学教授 全军网络技术研究中心主任、博士生导师
	尤晋元	上海交通大学计算机科学与工程系教授 上海分布计算技术中心主任
	施伯乐	上海国际数据库研究中心主任、复旦大学教授 中国计算机学会常务理事、上海市计算机学会理事长
	邹 鹏	国防科学技术大学计算机学院教授、博士生导师 教育部计算机基础教学课程指导委员会副主任委员
	张昆藏	青岛大学信息工程学院教授

译者序

在现实世界中，有很多事情可以用程序表示。在计算机的世界中，没有程序的计算机只是一堆废物而已。在计算机发展的早期时代，把现实世界中的程序编写成计算机可以理解并正确执行的程序是一项非常复杂和枯燥的工作。面向对象的程序设计出现之后，这项工作渐渐变得简单而有趣。

作为早期的程序员，他们可能还在留恋往日随心所欲编写程序的时光。虽然他们自觉或不自觉地运用面向对象的程序设计的概念，然而他们毕竟不得不依靠经验、精心规划，才能得到一个与面向对象的程序类似的程序。开发另外一个新程序时，他们必须改写自己心爱的程序。

作为程序设计初学者，他们可能还在惊奇为什么计算机可以实现如此之多的功能且计算机如何实现这些功能。到了课堂上，大量程序设计的新概念夹杂着计算机原理铺天盖地而来，使他们感到手足无措。实际上，对他们来说，无需掌握计算机科学每一个领域的细节，而需掌握一种既简单易懂，又便于运用的计算机程序设计方法。

本书既满足了以往习惯于运用面向过程的程序设计思想编写程序的老一代程序员学习面向对象的程序设计思想的需要，也满足了教师从一开始就向程序设计初学者灌输现实世界程序设计思想的需要。本书讲授的是面向对象的程序设计思想，而不是 Java 程序设计语言。Java 作为一种面向对象的程序设计语言，在本书中仅仅是作为讲授面向对象的程序设计思想的工具。

你想掌握面向对象的程序设计思想和方法吗？这是一本非常好的简单易学的教科书。

本书第 0 章至第 2 章由侯震峰翻译，第 3 章至第 9 章、序言、附录 B 及附录 C 由侯国峰翻译，第 10 章和第 11 章由江海峰翻译，第 12 章至第 15 章由李湘翻译，第 16 章和附录 A 由张迪翻译，参加本书翻译工作的还有程洁、宋黎松、裘岚、韩珊、官章全、舒志勇、郑城荣以及李鑫等同志。在翻译过程中，除对原文个别错误作了更正外，我们力求忠实于原文。但由于译者的知识水平和实际工作经验有限，因此不当之处在所难免，恳请读者批评指正。

译者

前 言

这是本书的第 2 版，在第 2 版中我们做出了许多改进，但主要目标不变。希望本书作为面向对象的程序设计的入门教科书，用于一个学期的 CS1 课程，而且假设学生以前没有程序设计经验。那些已经具有诸如 C、BASIC 以及其他传统的面向过程的程序设计语言经验的人也可以使用本书作为面向对象的程序设计、图形用户界面以及事件驱动的程序设计的入门。本书的两个主要目标是讲述：

- 面向对象的程序设计。
- 现实世界程序设计的基础。

面向对象已经成为所有计算机科学领域中的一种重要模式，从第 1 个程序设计课程就开始讲授面向对象的程序设计是非常重要的，讲授面向对象的程序设计多于讲授面向对象的程序设计语言的语法和语义。掌握了面向对象的程序设计，意味着精通面向对象的概念并能够将其有效地、系统地应用于程序开发过程中。

本书的第 2 个目标是使学生习惯于现实世界的程序设计，知道面向对象的概念是不够的，学生们必须能够运用这些知识开发现实世界的程序。大多数入门教科书中的示例程序过于简单，学生们在其他教科书中很少遇见定义超过 3 个类的示例程序。但是在现实世界的应用项目中，程序员必须使用库中的许多类，还必须定义其自己的多个类。在本书中，我们教学生如何使用类库中的类和如何定义其自己的类。例如，第 15 章的示例程序定义了 10 个类并使用了现有类库中大量的类。

第 2 版的新特点

我们愿意借此机会感谢第 1 版的使用者，我们特别感谢使用者和他们的学生提出的大量建议和鼓励。至于第 2 版，我们集中改进了第 1 版中的不足并尽可能多地采纳建议。因为无法把全部建议都编写到本书中，因此我们设法通过将资料放到我们的站点上以满足各种各样的需求。请访问站点有关辅助资料方面更详细的内容。

进入本书正题之前，我们首先简要强调第 2 版带来了哪些新特点：

1. **使用 Javadoc 注释。**除了前面的几章以外，所有示例代码和程序都以标准化的 javadoc 的风格文档化。修订的 javabook 类也使用 javadoc 注释完全文档化，由 javadoc 注释生成的 javabook 类的 HTML 文档文件可以从我们的站点访问到。

2. **双色页面。**对于第 1 版中的插图，我们得到了许多称赞。我们进一步改善了这些插图，把用丰富的资料和视觉上吸引人的图解与插图说明问题的风格描述为 *形象化教学*，我们相信形象化教学是讲授初步程序设计最合适的方法。

3. **新的和改进的 javabook 类。**Javabook 包增加两个新类：Clock 和 SimpleInput。Clock 类提供基本的时钟函数，诸如读取当前时间、今天的日期和提供秒表函数。利用秒表函数，程序员可以很容易地记录程序的运行时间。例如，这些秒表函数可以很方便地比较不同排序

算法的运行时间。第 2 个新类, 即 SimpleInput, 提供不基于 GUI 的输入例行程序。许多使用者要求用于非 GUI 环境的 InputBox 的功能性。我们增加这个类以满足他们的要求。除了这两个新类以外, 我们还对现有的类做了许多较小的改进。

4. 基于 Swing 的 javabook 类。随着 Java 2 平台中的 Swing 类的出现, javabook 包的实现基于 Swing 版本。使用 Swing 类的直接好处包括简化若干 javabook 类和诸如将一个图标放到 MessageBox 对象上的新功能的实现, 有关基于 Swing 的 javabook 类的信息可以在我们的站点上找到。使用最初的 javabook 类, 还是使用基于 Swing 的 javabook 类取决于教师在课程中涉及的 Swing 类的范围, 即使课程中不包括 Swing 类。如果教师不打算讲授 javabook 包的内部处理, 也可以使用基于 Swing 的 javabook 类。

5. 补充的课题。尽管我们认为详细的集合类的范围属于 CS2 卷, 但是我们还是收到使用者要求包括 Vector 方面的讨论。我们同意他们的观点, 为 CS1 学生介绍使用方便和功能强大的 Vector 类是可取的, Vector 类在第 9 章中描述。第 2 版中包括的另外一个新课题是堆排序, 将排序算法从原来的第 15 章移动到新的第 10 章以后, 我们增加了堆排序, 用一个非递归的 $Mlog_2N$ 算法加强这一章的内容, 堆排序作为巧妙使用数组排序堆节点的一个主要例子。

6. 改进的辅助资料。我们改进了现有的辅助资料并增加了许多新的辅助资料, 从我们的站点可以得到有关的详细信息。

主要特点

本书有许多教学法方面的特点, 在面向对象的程序设计方面的入门教科书之中, 本书是独特的。下面我们说明本书的主要特点:

特点 1 java

我们为本书选择了 Java, 与 C++ 不同, Java 是一个纯粹的面向对象的语言, 是讲授面向对象的程序设计的理想语言。因为 Java 是合乎逻辑的并且易于编程的, 所以使其成为最易于编程的面向对象的语言, Java 没有任何可能成为初学者学习面向对象概念的障碍。尽管我们使用了 Java, 然而必须强调本书不是关于 Java 程序设计的。由于本书是有关面向对象的程序设计的, 因此没有涉及 Java 的各个方面。然而我们确实论及了足够的 Java 语言特点, 以使學生可以胜任 Java 程序员。

特点 2 javabook 包

我们提供了一个称为“javabook”的类库(Java 术语中的“包”), 包括本书从头到尾使用的许多类。1993 年我们在《*Journal of Object-Oriented Programming*》中(第 6 卷第 1 部分, 第 6 卷第 4 部分和第 6 卷第 5 部分)撰写了一系列关于如何讲授面向对象的程序设计的文章。在这个系列中, 我们描述的教学法概念的核心是在成为一名对象设计者之前, 必须首先成为一名对象用户。换句话说, 在能够有效地设计某人自己的类之前, 一个程序员必须首先学会如何使用预先定义的类, javabook 的使用正是基于这个原则。

使用 javabook 包有许多优点:

1. 向学生说明现实世界的程序是如何开发的。我们不是从零开始开发实际的程序, 相反, 只要可能我们就使用预先定义的类。面向对象的程序设计的好处之一是通过重用现有的类提

高程序员的生产力。学生们通过使用 javabook 包中的类，将得到代码重用的实际经验。

2. 使程序设计语言语法和语义的影响最小。Javabook 类的使用让学生集中精力学习概念，而不是 Java 语言的特点。我们已经见过许多这样的事例，没有经验的程序员由一个设计优秀的程序开始，却以一个结构非常差劲的程序结束。通常，因为他们没有完全理解程序设计语言，所以他们的设计没有被编译为语法和语义正确的程序。在他们开发程序的过程中遇到错误时，不是修改程序代码，而是改变程序的设计。使用预先定义的类，使得程序设计语言的影响最小化，因为这些预先定义的类隐藏了底层程序设计语言的复杂性。学生们更容易将其程序设计为一个使用 javabook 类的工作程序代码。

3. 使学生从一开始就编写有用的程序，有助于保持学生最初的学习兴趣和动机。如果不使用预先定义的类，那么在学生们可以开始编写有趣的和实际的程序之前，他们必须学习程序设计语言多得不可忍受的细节。但是在他们达到这种程度之前，许多人可能已经失去了程序设计的兴趣，而淹没在语言语法和语义的令人厌烦的细节之中。然而一开始就使用诸如 java.awt 标准 Java 类库中预先定义的类是不实际的，因为这需要初学的学生具备他们尚未掌握的程序设计经验，诸如 javabook 这样易用且直观的预先定义的类更适合于程序设计初学者。

4. 在学生可以开始设计其自己的类之前提供必要的基础。学习面向对象的程序设计的最终目标是掌握设计有效类所必须的能力，但是在有能力设计这样的类之前，学生必须首先学会如何使用现有的类。此外一开始就向新的程序员讲授如何使用标准的 Java 类，从教学方法上说是不合理的，因为 java.awt、java.io 和其他类库的多数类对于程序设计初学者来说不是很容易掌握。我们从程序设计初学者的实际出发设计了 javabook 类。

5. 可以自定义 javabook 包满足自己的需要。例如，这个包中有一个称为“MainWindow”的类，用作一个程序的顶层窗口。你可以很容易地扩充这个类，当顶层窗口出现在屏幕上时显示自己学校的徽标。或者可以增加一个帮助菜单，列表显示 T.A. 办公室和电话号码，也可以扩充其他的 javabook 类。Javabook 包还可以是毕业生或高年级未毕业学生的习题，通过设计由几百个新学生使用的用于 javabook 包的类，他们可以直接学到需要采用什么技术才能保证类的可靠和真正可重用。

关于 javabook 的使用出现的一个担心是学生们是否能够不使用 javabook 包就编写程序，答案当然是可以。Javabook 不是目的，而是学生学习标准包的一种手段。它是达到目标的手段，是通往标准包的培训方向。除了 javabook 类之外，我们还论及许多来自诸如 java.awt 和 java.io 标准 Java 包的类。

Javabook 类的全部源代码已经提供，鼓励学生在实践可重用类的例子时研究。学习第 13 章之后，学生们能够理解几乎全部 javabook 类，我们说“几乎”是因为 javabook 中的某些类是用本书中未解释的标准类实现的。如果学生们花时间查阅参考手册中的这些标准类，他们应该能够百分之百地理解 javabook 类。

特点 3 完全沉浸式方法

我们采用一种完全沉浸式方法，学生以这种方法从第 1 个程序就开始学习如何使用对象，从一开始就保证强调面向对象的程序设计的核心概念是非常重要的。第 1 章的第 1 个示例程序如下：

```
/*
    Program FunTime

    The program will allow you to draw a picture by
    dragging a mouse (move the mouse while holding the left mouse
    button down; hold the button on Mac). To erase the picture and
    start over, click the right mouse button (command-click on Mac).
*/

import javabook.*;

class FunTime
{
    public static void main(String[ ] args)
    {
        SketchPad  doodleBoard;
        doodleBoard = new SketchPad();
        doodleBoard.setVisible( true );
    }
}
```

这个程序集中了面向对象的程序设计的大多数基本概念，即一个面向对象的程序使用对象。正如听起来的那样明显，许多入门书并没有真正强调这个事实。在该程序中，我们使用一个称为“doodleBoard”的 SketchPad 对象，允许用户绘制一幅图。几乎所有其他的入门教科书都以如下所示的示例程序开始：

```
/*
    Hello World Program
*/

class HelloWorld
{
    public static void main(String args[])
    {
        System.out.println("Hello World");
    }
}
```

或：

```
/*
    Hello World Applet
*/
import java.applet.*;
import java.awt.*;

public class HelloWorld extends Applet
{
    public void paint( Graphics g)
    {
        g.drawString("Hello World", 50, 50);
    }
}
```

这两个程序都有问题，它们都没有阐明面向对象的程序使用对象这个关键的概念。第 1

个程序确实使用了一个 System.out 对象，但是 System.out 对象的使用并没有说明对象的声明和创建。初学者通常不能分辨类和对象，因此在开始使用对象之前强调必须声明并创建来自类的对象。我们的第 1 个示例程序做到了这一点。

另外一个由 System.out 程序带来的问题是没有真正的基于窗口的程序用其输出。有些教科书不但在第 1 个程序中使用 System.out，而且几乎专门依靠 System.out 用于程序输出，这不是现实世界的程序设计。本书中，我们仅在程序开发过程中为了检查的目的才使用 System.out。

第 2 个 HelloWorld 程序是一个 applet，正如其名字所暗示的，它是一个具有非常特殊用法的小程序。尽管 applet 很有趣，然而专门讲授 applet 却是个问题，因为学生们将只能学到非常有限的程序设计观点。在序言中稍后我们将讨论更多有关应用程序与 applet 的内容。

由这两个程序带来的另外一个主要问题是它们不适应现实世界的情况，相反，我们的第 1 个示例程序通过使用另外一个类，假设是 WordProcessor，替换 SketchPad，即可作为一个商业性应用程序的主程序。事实上，来自第 2 章的第 2 个示例程序如下(说明：实际上这是我们逐行解释的第 1 个程序)：

```
/*
   Program MyFirstApplication
   The first sample Java application.
*/
import javabook.*;

class MyFirstApplication
{
    public static void main(String args[])
    {
        MainWindow mainWindow;
        mainWindow = new MainWindow();//create and
        mainWindow.setVisible( true );//display a window
    }
}
```

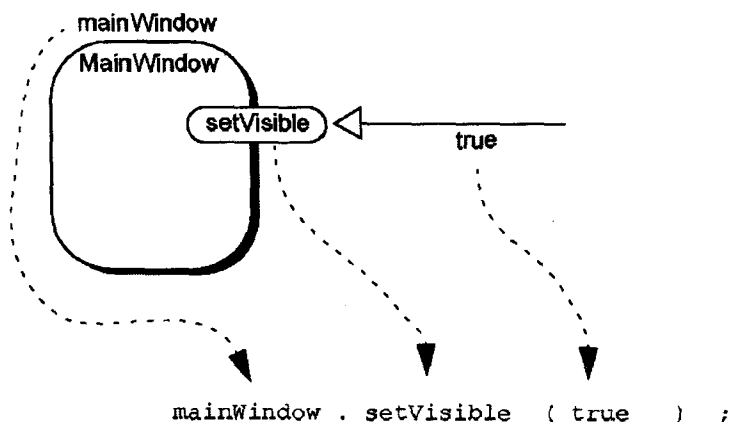
这个程序的结构与第 1 个示例程序的结构完全相同，第 2 个示例程序补充了我们通过使用对象和通过改变对象而创建一个新的程序的编程概念。

特点 4 图解

我们相信一幅图胜千言，用易懂的插图可以很清楚地解释深奥的概念。我们使用 *对象框图(object diagram)* 展示对象和类之间的关系。框图是设计程序和文档化程序一种重要的工具，程序员开发现实世界的软件应用程序都要使用某种形式的框图绘制工具。我们使用简单且非正式的框图，但是我们在本书中使用的框图效仿业界标准的对象框图。熟练掌握本书中使用的对象框图之后，学生们就具备了研究更为正规的面向对象设计方法学的很好基础。对于愿意介绍正式对象框图的使用者，我们有用于示例程序和 javabook 类的 UML 框图，可以在我们的站点查看或下载。

作为教学方法的工具，本书包括数量众多的插图，解释诸如继承、私有方法与公共方法之间的差别等这样的核心概念。对象框图中使用的表示法在形形色色的插图中使用一致。图

1 是第 2 章中的一个例子，本书中有 230 多个这样的插图和框图。其他有代表性的插图可以在有关章节中找到。



除了对象框图以外，我们还使用方法调用顺序框图(method call sequence diagrams)表示诸如图 15.1 所示的方法调用的顺序。方法调用顺序框图在说明消息的流程中是非常有用的，方法调用顺序框图广泛使用于第 15 章一个高级示例程序的文档化。

特点 5 增量开发

我们在本书中讲授软件工程原理。程序开发原理和技术与其他课题是交织在一起的，而不是每章一个课题。从第 2 章到第 14 章，每章至少包括一个示例程序用来举例说明该章论及的课题，而且我们始终使用同一种设计方法开发程序。第 15 章是案例研究的一章，我们开发了一个适合于 CS1 标准的相当大的程序。

目前市场上许多其他介绍性的程序设计书籍带来的一个主要问题是讲授一种有 20 多年历史的结构化的程序设计，这对面向对象的程序简直是行不通的。本书真正讲授一种面向对象的程序设计的软件设计方法学。本书中的所有示例程序都是采用我们描述为增量开发的技术开发的，增量开发技术基于现代的迭代方法(有些人称其为“螺旋方法”)，这是专业的面向对象的程序员首选的方法。

程序设计初学者往往混淆高层次的设计与低层次的代码细节，他们的思维过程完全混乱。完成最终的程序是不够的，如果我们要教会学生如何开发程序，则必须说明开发过程。仅仅通过观察已经完成的产品，无论是家具还是房屋，一名学徒是不可能成为师傅的，软件构造却很容易。其他教科书常常有这种情况，单独的一章专门讲述软件开发，这是不够的，我们必须多次说明开发过程。本书中，我们增量开发的每一个示例程序都是为了向学生说明如何以一种合乎逻辑，而且以有条理的方式开发程序。

每个开发阶段的所有示例程序的源代码都可以从我们的站点访问，然而我们不鼓励学生只是简单遵循书中介绍的开发，也不鼓励学生阅读源代码。我们鼓励学生遵循书中介绍的开发步骤实际建立示例程序，这是学生真正掌握软件开发最稳妥、快捷的途径。

特点 6 设计原则、有帮助的提示和单元测试

遍及全书都包括设计原则和有帮助的提示，几乎各章的每一节都以单元测试结束，以保证学生掌握该节的要点。

设计一个实现单独的、意义明确的任务的类，不要使类因为多个任务而负担过重。

密切注意由于一个错误中断(OBOE)。

第 2 种风格是 Caffeine 博士的长篇大论：

有时，程序设计使人非常沮丧，因为在你这部分程序中即使付出了再多的努力也可能使程序运行不正确。不仅你如此，专业的程序员常常有同感，包括在下。但是如果花时间思考并保持冷静得出问题的结论，你就能够找到解决问题的方法。如果不这样做，那么它仅仅是一个程序。你的健康比运行一个程序和好分数重要的多。

第 3 种风格是 Caffeine 博士与其引以为荣的学生 Latte 女士或 Espresso 先生的一段对话，Latte 女士和 Espresso 先生在各章中轮流出现。

Latte 女士：我出现在奇数章，提出重要的问题。

Caffeine 博士：很好，你的问题很有见解，对其他同学很有帮助。

Espresso 先生：我出现在偶数章，也提出问题。

Caffeine 博士：是的，我也喜欢你的问题。

解释 Swing 类的文档可以访问我们的站点：www.drcaffeine.com/additionaltopics

单元测试出现在各节的结尾。

特点 7 图形化用户界面和事件驱动的程序设计

由于现代现实世界的程序是基于 GUI 和事件驱动的，因此如果我们要讲授现实世界程序设计的基础，就不能回避它们。有些人可能认为基于 GUI 和事件驱动的程序对于初学者来说难于掌握，虽然它们不是很浅显，但是我们相信通过一部分一部分地介绍，初学的学生是可以逐渐掌握的。在第 13 章中，我们深入讨论基于 GUI 和事件驱动的程序设计。然而学生们将专门从第 2 章开始利用 `javabook` 类编写基于 GUI 的程序，在第 5 章中学习事件处理的初步内容。到第 13 章时，他们就做好了理解该内容的准备(为此，最好介绍第 5 章，但是正如我们稍后的解释，如果不希望涉及 `applet`，则可跳过该章)。

特点 8 应用程序与 applet

我们既讲授 Java 应用程序，也讲授 applet，但我们强调的是应用程序开发。有些入门书过分依赖或完全依赖 applet，但是 applet 的用途是有限的。在现实世界中，我们不能专门编写 applet，而且程序设计入门书籍也不应该只讲授 applet。同时，我们不应该回避讲授 applet，我们应该都讲授。虽然 applet 的用途是有限的，但是对学生来说编写 applet 通常很有趣，而且我们可以将 applet 作为一种有效的教学工具。在本书中，我们在第 5 章利用 applet 介绍事件处理和基于 GUI 的程序设计的基础。

虽然我们涉及 applet 并建议教师讲授，但是我们有机地组织本书，使教师可以跳过 applet 方面的内容而不影响连贯性。

补充材料

所有的补充材料都可以从我们的 www.mhhe.com/wu2 和 www.drcaffeine.com 站点得到。www.drcaffeine.com 在作者的控制之下，因此其更新比 McGraw-Hill 维护的站点更频繁。可以从我们的站点访问的补充材料包括：

1. 所有示例程序和 javabook 类的源代码。对于所有示例程序，可以得到每个开发步骤的源代码。既有 javabook 类的二进制代码，也有源代码，也有由 javadoc 注释生成的 HTML 格式的 javabook 包的文档。

2. 附加包。希望使用新的 Java GUI 对象的读者可以访问基于 Swing 的 javabook 类。对于希望将更多“趣味性”放到程序设计习题的读者，我们提供了实现 Logo 龟的 Turtle 类，Turtle 类和支撑类在 galapagos 包中。

3. 附加课题。为了满足教师各种各样的需要，我们在站点准备了附加的课题。可以访问的一些课题包括：

- Swing 类。
- 链表和动态分配。
- 不同的示例程序。

我们将尽最大努力定期提出新课题。

4. 如何使用 Java 编译器。有关如何利用朴素的 JDK 编译器和诸如 Symantec Visual Café、Borland JBuilder、Metroworks CodeWarrior 以及其他商业编译器编辑、编译和运行程序，我们有一个循序渐进的说明。

5. 单元测验和经挑选章节的习题答案。每个人都可以得到单元测验习题解答，每章的习题答案仅限于教师。教师可以下载这些解答，如果他们愿意，还可以让学生得到这些答案。

6. 幻灯片教师。可以得到动画版本和非动画版本的 PowerPoint 幻灯片，这两个版本的幻灯片可以通过浏览器联机查看，本书的使用者还可以下载并根据课程需要编辑这些幻灯片。

7. 每章的教学说明和建议。采用本教科书的教师可以得到此材料。

McGraw-Hill 联机学习中心

有关本书的资料位于 www.mhhe.com/wu2 的联机学习中心(Online Learning Center)，是一盘“数字盒式磁带”，包含本书的教学和补充。学生阅读本书时，可以联机分阶段自我测验、

访问源代码，并回顾相应章节的 PowerPoint 幻灯片。联机学习中心还有教师部分，包括 PowerPoint 幻灯片、源代码以及按章组织的有口令保护的解答，便于查阅。

本书组织

本书共 17 章。足够一个学期学习的内容了，各章基本上应该顺序讨论。如果希望直接转入程序设计，则可以跳过第 0 章，或者指定为课外读物。如果不希望讨论 applet，则可跳过整个第 5 章。3.8 节的数值表示法和 7.10 节的递归方法是选修的，可以跳过。

下面，我们简要描述每一章：

- 第 0 章是选修章，提供计算机和程序设计语言方面的背景信息，如果希望从面向对象的程序设计概念开始，可以跳过本章或指定为课外读物。

- 第 1 章提供面向对象的程序设计的概念基础，我们描述面向对象的程序设计的关键部分，并利用图解表示法举例说明每个概念。本章中我们介绍的第 1 个示例程序是一个有趣的画图程序。

- 第 2 章讨论 Java 程序设计的基础和编辑、编译以及运行一个程序的过程，本章介绍应用程序和 applet，还介绍两个来自 javabook 包的类——MainWindow 和 MessageBox。如果不讨论 applet，可以跳过本章讨论 applet 的最后一节。

- 第 3 章介绍变量、常数和操作数值数据的表达式，我们说明来自 java.lang 的标准的 Math 类，并介绍和使用来自 javabook 包的两个另外的类——InputBox 和 OutputBox，这两个类处理应用程序的输入和输出。选修的一节解释数值数据在存储器空间中如何表示。

- 第 4 章讲授如何定义可实例化的类，本章涉及的关键课题是构造函数、可视性修饰符(Public 和 Private)、局部变量、参数传递以及返回值的方法。到第 4 章结束时，学生将基本理解如何使用来自包中的类和如何定义其自己(简单)的类。

- 第 5 章讨论 applet。讲授如何用 applet 处理输入，介绍 java.awt 包的 GUI 对象 Label、TextField 和 Button。本章第 1 次出现事件处理，为第 12 章中更全面地讨论 GUI 对象和事件驱动的程序设计打下一个良好的基础。我们建议教师介绍本章，但是我们理解有些教师宁愿不讨论 applet。如果是这种情况，则可以跳过本章而不影响连贯性。

- 第 6 章解释判断语句 if 和 switch。涉及逻辑表达式和嵌套的 if 语句，为了举例说明判断语句的用法，介绍 javabook 包的 ListBox 类并用于示例程序。

- 第 7 章解释循环语句 while、do-while 和 for，我们引入 javabook 包的另外两个类——ResponseBox 和 Format，并在示例程序中使用这两个类。本章最后的选修一节介绍递归，作为用于循环的另外一种技术。

- 第 8 章讨论非数值数据类型：字符和字符串，本章解释 String 和 StringBuffer 类。利用这两个类，我们说明简单的数据类型和引用的数据类型之间的不同，还说明对象如何作为参数传递给方法以及它们如何从方法中返回。

- 第 9 章讲授数组，我们讨论简单数据类型的数组和对象数组，数组在 Java 中是对象，在本章中我们利用数组作为例子重申对象如何传递给方法。本章引入自引用指针 this，我们描述如何处理二维数组并解释在 Java 中一个二维数组确实是数组的数组。本章最后引入并说明 Java.util 包的 Vector 类。

- 第 10 章介绍搜索和排序算法，涉及 N^2 和 $N\log_2 N$ 排序算法。根据学生的背景，可以

跳过搜索和排序算法的数学分析。

- **第 11 章**说明文件 I/O, 说明来自 java.awt 和 java.io 的标准对象, 诸如 File 和 FileDialog。我们讨论文件 I/O 的各种类型, 从低级字节 I/O 到高级对象 I/O。涉及对象 I/O, 这在入门教科书之中是独一无二的。作为文件处理的一部分, 本章引入了异常处理。

- **第 12 章**研究可重用的类并描述包的组织, 在本章中, 我们说明如何将对象分为 4 类, 并说明辅助类的设计。作为如何将一个类设计为可重用的举例说明, 我们将转换前面的实例类为可重用的类。

- **第 13 章**讨论事件驱动的程序设计和 GUI 对象, 本章中说明的 GUI 对象有 Button、Menu、Dialog、Frame 以及 Applet, 本章中还使用了第 5 章介绍的 GUI 对象。我们还将说明鼠标事件的处理。

- **第 14 章**讨论继承和多态以及在程序设计中如何有效地使用它们, 说明成员可访问性和构造程序继承的结果, 还说明抽象类和抽象方法的用途。

- **第 15 章**是案例研究的一章, 我们将利用本书中学过的技术开发一个大型的程序。除了我们为该程序设计的 10 个类以外, 该程序还使用来自 javabook 包和标准 Java 包的对象。没有哪本入门教科书像本书这样有包括了如此之多类的示例程序。

- **第 16 章**讨论递归, 因为我们想说明使用了递归的例子确实出色, 所以没有包括其实应该非递归编写的任何递归算法(除了那些用来说明的以外)。

致谢

首先, 我愿意感谢下列审阅者及其意见、建议与鼓励:

Lewis Barnett	University of Richmond
Peter Biggs	Brigham Young University
Allen Brookes	Linfield College
Edward Chow	University of Colorado, Colorado Springs
Stephanie Crouch	Texas A & M University
Ann Ford	University of Michigan
Homer C. Gerber	University of Central Florida
Michael Hoffhines	Maui Community College
Lily Hou	Carnegie Mellon University
Faisal Kaleem	Florida International University
Saroja Kanchi	Kettering University
Alan Kaplan	Clemson University
Chung Lee	California State Polytechnic University at Pomona
John Lowther	Michigan Technical University
Keith B. Olson	Montana Tech of the University of Montana
John Phillips	Dodge City Community College
Bina Ramamurthy	SUNY at Buffalo
Arturo Sanchez-Ruiz	University of Massachusetts, Dartmouth