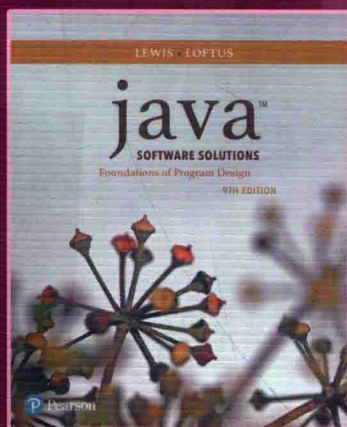


Java程序设计教程 (第九版)

Java Software Solutions
Foundations of Program Design, Ninth Edition



[美] John Lewis 著
William Loftus

洛基山 张君施 等译

国外计算机科学教材系列

Java 程序设计教程

(第九版)

Java Software Solutions

Foundations of Program Design

Ninth Edition

[美] John Lewis 著
William Loftus

洛基山 张君施 等译



电子工业出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书对面向对象的思想和机制进行了准确而透彻的剖析,为读者深入学习 Java 语言程序设计提供了全面、详细的指导,并在前几版的基础上进行了全面的修订与更新。全书覆盖了面向对象设计的广泛内容,介绍了 Java 语言的基本数据类型、流程控制、类和对象等。在深入分析面向对象设计方法的基础上,介绍了封装机制、继承机制和多态性的实现与应用、异常的捕捉和处理、集合类的定义方法与泛型类的概念。书中提供了大量具有可实践性的程序实例、自测题及答案、练习题和编程项目、生动的“软件失误案例”,并且在每章中都补充了针对本章主题的图形用户界面(GUI)设计知识和实例,讲解了如何使用开发工具 JavaFX 来设计 GUI,这部分内容可形成完整的 Java GUI 设计知识体系。

本书适合作为高等院校软件专业与计算机应用专业相关课程的教材,同时对工程技术人员也有很高的参考价值。

Authorized translation from the English language edition, entitled Java Software Solutions: Foundations of Program Design, Ninth Edition, 9780134462028 by John Lewis and William Loftus, published by Pearson Education, Inc., Copyright © 2017 Pearson Education, Inc.

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording or by any information storage retrieval system, without permission from Pearson Education, Inc.

CHINESE SIMPLIFIED language edition published by PEARSON EDUCATION ASIA LTD., and PUBLISHING HOUSE OF ELECTRONICS INDUSTRY Copyright © 2018.

本书中文简体字版专有出版权由 Pearson Education(培生教育出版集团)授予电子工业出版社。未经出版者预先书面许可,不得以任何方式复制或抄袭本书的任何部分。

本书贴有 Pearson Education(培生教育出版集团)激光防伪标签,无标签者不得销售。

版权贸易合同登记号 图字:01-2017-4582

图书在版编目(CIP)数据

Java 程序设计教程:第九版/(美)约翰·刘易斯(John Lewis), (美)威廉·洛夫特斯(William Loftus)著;洛基山等译. —北京:电子工业出版社,2018.7

书名原文:Java Software Solutions: Foundations of Program Design, Ninth Edition

国外计算机科学教材系列

ISBN 978-7-121-34093-2

I. ①J… II. ①约… ②威… ③洛… III. ①JAVA 语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2018)第 078746 号

策划编辑:冯小贝

责任编辑:冯小贝

印 刷:三河市鑫金马印装有限公司

装 订:三河市鑫金马印装有限公司

出版发行:电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本:787×1092 1/16 印张:30.25 字数:874 千字

版 次:2005 年 5 月第 1 版(原著第 4 版)

2018 年 7 月第 6 版(原著第 9 版)

印 次:2018 年 7 月第 1 次印刷

定 价:89.00 元

凡所购买电子工业出版社图书有缺损问题,请向购买书店调换。若书店售缺,请与本社发行部联系,联系及邮购电话:(010)88254888, 88258888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

本书咨询联系方式: fengxiaobei@phei.com.cn。

译者序

诞生于 20 世纪 90 年代中期的 Java 编程语言，开创了一种全新的编程思维模式，并迅速风靡整个 IT 业界。从 Java 开始，涌现出了许多各具特色的面向对象编程工具。Java 语言不仅可用于开发大型软件，而且特别适合于开发 Web 应用，目前已成为网络时代最重要的程序设计语言之一，并且其开发工具和功能也正随着网络技术的不断发展而同步进行更新。Java 语言已经成为事实上适合初学者学习面向对象程序设计的首选语言。

本书是一本学习 Java 编程的入门性教材，也是读者初步了解面向对象思想及其实现方法的一本理想读物。全书深入、透彻地分析了 Java 语言的面向对象机制，指导读者如何在学会设计类和对象前先学会使用类和对象，以大量程序实例体现了面向对象的核心概念（如类、对象、重载、接口及多态性等）和封装、继承等机制，并基于解决问题的思路，灵活运用各种 Java 程序设计技术，详细示范了一些典型算法的实现方法。

本书注重完整的程序设计实践，每一章都附有大量具有可实践性的程序实例、自测题、练习题、编程项目，以及富有趣味性的图形用户界面设计实例，以着重培养读者的软件设计能力和解决问题的能力。有些章末尾的“软件失误案例”虽然短小，但它们是来自于现实世界的惨痛教训，其目的在于培养读者（未来的程序员）“质量高于一切”的思想，强调良好的软件设计思想和严谨合理的设计规范的重要性。

各章末尾的“图形设计之路”部分，已经全部重新编写，为图形设计的教学提供了较大的灵活性，它们可作为相对独立的一部分根据需要在教学上进行取舍。

本书最大的更新，在于讲解如何用最新的开发工具 JavaFX 设计图形用户界面，它全面替换了以前使用的抽象窗口工具集（AWT）和 Swing，使设计 GUI 变得更容易，功能更强大。并且还对编程项目进行了调整，从而更加适应教学的要求。此外，对占本书相当篇幅的附录进行了全面更新，以反映 Java 语言本身的调整并与正文各章相呼应。本书还为学生和主讲教师提供了丰富的配套资源。

全书共分 13 章及 12 个附录，包括计算机系统概述、数据与表达式、类与对象、编写类、条件判断与循环（两章）、面向对象设计、数组、继承、多态性、异常处理、递归和集合，附录则主要提供术语表及练习题答案等。全书由洛基山主持翻译工作。前言、第 2~4 章由首都经济贸易大学刘丽丽翻译；第 7~9 章由西北农林科技大学闫锋欣翻译；第 10~12 章由北京工商大学张君施翻译；张莉翻译了第 13 章及附录 A 和附录 B；附录 C~F 由徐景辉翻译；陈艳羽翻译了附录 G~K；常征翻译了第 1 章、附录 L 以及各章的自测题、练习题、编程项目和自测题答案；陕西师范大学孟永宏翻译了第 5~6 章和部分附录；蔡志强翻译了“软件失误案例”和部分自测题；全书最后由洛基山负责修订和统稿。

由于译者水平有限，并且各种专业术语在业界尚无严格统一的译法，如果译稿中有不妥或错误之处，恳请广大读者批评指正。

前言

欢迎使用本书第九版。多年来，本书能够满足广大师生的需求并为教学服务，这使我们深感欣慰。这一版本做了进一步的内容组织和调整，完善了关键技术问题的重点分解和概念解释。

本版的改进

这一版最大的变化，是彻底重写了书中的“图形设计之路”部分，以充分利用 JavaFX API。Oracle 已经不再支持 Swing。现在，JavaFX 已经成为开发图形和图形用户界面（GUI）的首选方法，因此本书也将采用它。

书中的更新如下：

- 涵盖 JavaFX 图形形状。
- 讲解 JavaFX 控件，包括按钮、文本框、复选框、单选钮、选项框、颜色选择器、日期选择器、滚动条以及微调器等。
- 使用 Java 8 的方法引用和 lambda 表达式来定义事件处理器。
- 探究 JavaFX 的类层次。
- 解释 JavaFX 中的属性及属性绑定。
- 优化了各章末尾的练习题以及编程项目。
- 新增加的附录 G，讲解了 JavaFX 的布局面板。
- 新增加的附录 H，介绍了 JavaFX Scene Builder 软件。

关于 JavaFX 有两个令人激动的部分。首先，它提供了比 Swing 更清晰的 GUI 开发方法。用 JavaFX 开发的程序，会比用 Swing 开发的更短、更易理解。

其次，与 Swing 相比，JavaFX 更遵循面向对象的核心原则。例如，所有的图形，都是用包含基本数据元素的类体现的，比如包含半径值的 Circle 类。第 3 章讲解的有关各种形状的类，都是经过精心设计的基本类，只需学生具备对类和对象的基本理解即可。

通过 Java 8 的方法引用，为定义事件处理器提供了一种容易理解的途径。或者，学生也可以使用（底层的）lambda 表达式来定义事件处理器。

关于 JavaFX 的布局面板，在书中的一些示例中解释；有关它的完整描述，在附录 G 中给出。我们认为，与 Swing 布局管理器相比，JavaFX 布局面板更好，书中的一章里会专门解释。

书中提供的所有 GUI，都是用 Java 代码“手工”编码出来的。对于初学者而言，这样做很重要。附录 H 讲解的是 JavaFX Scene Builder 软件，它能进行鼠标拖放操作。但是，本书正文中并没有使用它。

除了与 JavaFX 相关的改变，本书还更新了一些示例和讨论的内容，以满足当今的教学需要。

除了为学生提供本书，我们也很乐意听到读者对本书的评价和疑问。

本书的基石

本书是基于下面的基本思想编写的，相信这有助于使其成为一本优秀的教材。

- **真正的面向对象。**一本真正介绍全面面向对象方法的教材，必须使用对象语言，即所有问题都应该用面向对象的术语来讨论。但这并不意味着必须对第一个程序就讨论多个类和方法的编写方式，学生应当在学会编写对象之前先了解如何使用对象。本书采用一种自然的学习进程，使学生逐步达到具有设计实际的面向对象解决问题的能力。
- **良好的程序设计实践。**本书的主要目的不是教学生如何编程，而是讲解如何写出好的软件，二者是有差别的。编写软件并不是按照说明完成一套动作，一个好程序也远非仅仅是一些语句的集合。本书汇集了一些可作为良好程序设计技巧基础的实践经验，它们不仅应用于本书所有的实例，而且还在讨论中进一步强化。学生将学会如何解决问题以及如何实现设计方案。全书贯穿了软件工程基本技术的介绍，“软件失误案例”通过生动的实例，展示不遵守这些程序设计实践原则所带来的后果，进一步复习了软件工程基本技术课程。
- **示例程序。**学生将通过程序实例来学习。本书通过大量已完整实现的实例来讲解特定的概念。书中穿插列举了易理解的小例子和实用的大例子，二者相辅相成，并对图形和非图形实例的采用进行了合理的选择。
- **图形与图形用户界面（GUI）。**图形能够激发学生极大的学习兴趣，而且图形设计可以充当面向对象应用的绝佳实例。因此，本书每一章都安排了精心设计的几部分，称为“图形设计之路”。本书完整讲解了 JavaFX API，它提供对 Java 图形和 GUI 的全面支持，是首选的开发工具。随着图形设计主题讨论的开展，学生将很自然地学会以适当的方式创建 GUI。对于不需要涉及图形设计的学生，则可以忽略“图形设计之路”的内容。

各章概述

第 1 章（计算机系统概述）一般性地介绍计算机系统，包括计算机基本结构和硬件、网络、程序设计以及语言翻译。本章对 Java 也进行了介绍，并讨论了通用程序开发的基础和面向对象编程。这一章广泛涵盖了学生所要熟悉的开发环境介绍。

第 2 章（数据与表达式）探讨 Java 程序中使用的一些基本数据类型和计算表达式的用法，还讨论了数据类型间的转换，以及如何借助标准的 Scanner 类与用户进行交互式输入。

第 3 章（类与对象）讲解预定义类的使用及其对象的创建方法，这些类和对象用于处理字符串、产生随机数、执行复杂计算和格式化输出。此外还讨论了枚举数据类型。

第 4 章（编写类）分析了与类和方法的编写有关的问题，包括实例数据、可见性、引用范围、方法参数和返回类型。此外，还讨论了封装和构造方法。与编写类相关的其他问题在第 6 章讨论。

第 5 章（条件判断与循环）讨论用于判断的布尔表达式的用法，然后详细地讲解了 if 语句和 while 循环语句。一旦建立了循环，就可引入迭代的概念，并进一步讨论用 Scanner 类不断接收、解析输入数据以及读取文本文件的方法。最后，介绍可用于管理大量对象的 ArrayList 类。

第 6 章（其他条件判断与循环）讨论第 5 章未涉及的其他 Java 条件语句（switch）和循环语句（do, for），包括 for 循环语句的增强版本。for-each 循环语句也可用于处理迭代和 ArrayList 对象。

第 7 章（面向对象设计）进一步深入和扩展了关于类的设计问题的讨论，包括解决问题时需要确定类和对象的识别技术，以及类与对象间的关系。本章还探讨了静态类成员、接口以及枚举型类的

设计。这一章对方法设计和方法重载也进行了讨论。

第 8 章（数组）详细讲解了数组以及数组的处理。将作为最基本的程序设计数据结构的数组与高层次的对象管理方法进行了对比分析。此外，还讨论了命令行参数、可变长度参数表以及多维数组。

第 9 章（继承）讲解类的派生及其相关概念，例如类层次、重写和可见性。本章还强调了应合理使用继承，以及在软件设计中使用继承的原则。

第 10 章（多态性）探讨了绑定的概念以及绑定与多态性的关系。接着，讲解了如何使用继承或接口实现多态性，并利用排序示范了多态性。最后，讨论了有关多态性的程序设计问题。

第 11 章（异常处理）关注 Java 标准库中用于定义异常的类层次结构，以及定义用户异常对象的方法。同时还讨论了处理输入/输出异常时的方法，并分析了一个编写文本文件的程序示例。

第 12 章（递归）讲解递归的概念、递归的实现以及合理使用递归的方法，并利用几个不同应用领域的程序实例，演示了递归技术如何使得某些特定问题解决得更完美。

第 13 章（集合）涉及集合的概念及其基本数据结构，并进一步深入探讨了抽象概念和经典的数据结构。此外，还介绍了泛型。本章可作为 CS2（计算机科学课程第二阶段）的入门性内容。

补充资料^①

学生的在线资源

可从本书的配套网站 www.pearsonhighered.com/cs-resources 获取如下的学生在线资源：

- 书中所有程序的源代码
- Java 开发环境的链接

在线练习和评估^②

MyProgrammingLab（编程实验室）可帮助学生全面掌握程序设计的逻辑性、语义和句法。在 MyProgrammingLab 中，学生可以进行实际的练习并即时获得问题反馈信息，从而提高对一些常用的基本概念和高级程序设计语言范例感到困惑的初学者的编程能力。

MyProgrammingLab 是一个辅助学生自主学习和完成课外作业的工具，其中包括配合本书教学内容和组织结构的数百个小型练习题。对于学生，该工具可对学生提交的代码自动检查逻辑和语法错误，并提供明确的错误提示信息，使学生能够分析出错在何处、为什么出错。对于教师，该工具提供了一个综合性的记分册，可跟踪记录学生的正确答案和错误答案，并可保存学生提交的代码供教师审阅。

MyProgrammingLab 提供给本书用户使用，实现该工具的合作方是 TC（Turing's Craft），即 CodeLab 交互式程序设计训练系统的开发者。通过查看来自教师和学生的反馈信息，或在学习本书内容时使用该工具（访问网站 www.myprogramminglab.com），可以全面了解 MyProgrammingLab 的功能和用法。

教师资源^③

下面的补充材料仅供教师使用。访问 Pearson 教师资源中心（www.pearsonhighered.com/irc），可以得到如何获取下述资源的信息：

① 相关的在线资源也可登录华信教育资源网（www.hxedu.com.cn）免费注册下载。

② MyProgrammingLab 需要登录相关网站单独付费购买。

③ 教师资源申请方式请参见前言后的“教学支持说明”。

- 幻灯片文件——使用 PowerPoint 制作的幻灯片
- 各章末尾练习题答案
- 各章末尾编程项目答案

本书特点

重要概念 全书用“重要概念”文本框显示基础概念和重点内容，并在各章末尾对这些概念进行了总结。

程序示例 所有的程序示例都进行了编号，并在程序后面给出了程序的输出结果、运行实例或者屏幕截图。程序中的代码用不同的灰度进行区分，以突出注释和保留字。

语法流程图 书中以特殊的流程图方式讨论了 Java 语言的语法元素，语法流程图清晰地描述了一条语句或语句构造的有效格式。Java 语言的全部语法流程图在附录 K 中给出。

图形设计之路 所有关于图形与 GUI 处理的讨论，都安排在每章的最后几节中，这部分内容称为“图形设计之路”。如果忽略这部分内容，也不会影响本书的完整性。当然，也可以根据需要根据专门关注“图形设计之路”的内容。每个“图形设计之路”小节的内容，都与其所在章的讨论主题相关。

重要概念小结 每章末尾都给出了该章重要概念的小结。

自测题和答案 每章给出的自测题是对该章建立的基本概念和术语的复习，并且可以使学生评估自己对该章学习内容的掌握程度。自测题的答案在附录 L 中给出。

练习题 这些阶段性的练习题涉及计算、分析或编写代码段，它们要求对本章内容有全面的理解。尽管练习题可能涉及代码编写，但通常并不需要上机实践。

编程项目 编程项目要求设计和实现 Java 程序，它们的难度各不相同。

MyProgrammingLab 学生可以进行实际的练习并即时获得个性化的问题反馈信息。对于那些对基本概念和高级程序设计语言范例感到困惑的初学者而言，MyProgrammingLab 可提高他们的编程能力。

软件失误案例 一些穿插在各章之间的短文，讨论了现实世界中软件设计的种种缺陷，激励学生从编程学习的开始就注意采用良好的软件设计原则。

致谢

非常感激世界各地的教师和学生们对本书以前各个版本提出的宝贵意见与建议。我们很高兴地看到教师对学生的深切关心和学生对知识的渴求，并欢迎你们继续对本书给出评价和质疑。

特别感谢 Brigham Young 大学的 Robert Burton 为本书提供的热心帮助、深刻见解及密切关注。多年来，Robert 一直不断地提出有助于完善和改进本书的有价值的反馈意见。

在将本书内容更新至 JavaFX 的过程中，瑞士西北部 Applied Sciences 大学的 Bradley Richards 提供了有用的建议和资源。Simon Fraser 大学的 Brian Fraser 也提出了一些非常好的反馈意见，有助于澄清一些问题。我们深切感受到，与计算机教育者之间的这种交流，的确非常有益。

感谢 Villanova 大学的 Dan Joyce，是他完成了最初的自测题的编写工作，并确保有足够的自测题覆盖本书中每一个相关的主题。

我们不断地惊叹于 Pearson 出版社为本书付出的才智和努力。本书编辑 Matt Goldstein 具有惊人的洞察力和责任心；他的助手 Kristy Alaura 一直给我们提供帮助和支持；市场部经理 Demetrius

Hall 使我们确信，教师们能够理解和认同本书在教学方法方面的优势；天才 Joyce Wells 为本书设计了封面；Scott Disanno 和 Carole Snyder 负责本书的生产；Lakeside Editorial Services 公司的 Louise Capulli 是本书的项目经理，她每天都在为我们提供帮助。我们感谢所有确保这本书达到高质量出版标准的人们。

特别要感谢下面这些人，他们通过兴趣小组、访谈、评论等不同方式为本书提供了有价值的建议。他们以及其他许多的教师和朋友，都为本书提供过很好的反馈。这些人的名单如下：

Elizabeth Adams	James Madison University
Hossein Assadipour	Rutgers University
David Atkins	University of Oregon
Lewis Barnett	University of Richmond
Thomas W. Bennet	Mississippi College
Gian Mario Besana	DePaul University
Hans-Peter Bischof	Rochester Institute of Technology
Don Braffitt	Radford University
Robert Burton	Brigham Young University
John Chandler	Oklahoma State University
Robert Cohen	University of Massachusetts, Boston
Dodi Coreson	Linn Benton Community College
James H. Cross II	Auburn University
Eman El-Sheikh	University of West Florida
Sherif Elfayoumy	University of North Florida
Christopher Eliot	University of Massachusetts, Amherst
Wanda M. Eanes	Macon State College
Stephanie Elzer	Millersville University
Matt Evett	Eastern Michigan University
Marj Feroe	Delaware County Community College, Pennsylvania
John Gauch	University of Kansas
Chris Haynes	Indiana University
James Heliotis	Rochester Institute of Technology
Laurie Hendren	McGill University
Mike Higgs	Austin College
Stephen Hughes	Roanoke College
Daniel Joyce	Villanova University
Saroja Kanchi	Kettering University
Gregory Kapfhammer	Allegheny College
Karen Kluge	Dartmouth College
Jason Levy	University of Hawaii
Peter MacKenzie	McGill University
Jerry Marsh	Oakland University
Blayne Mayfield	Oklahoma State University
Gheorghe Muresan	Rutgers University

Laurie Murphy Pacific	Lutheran University
Dave Musicant	Carleton College
Faye Navabi-Tadayon	Arizona State University
Lawrence Osborne	Lamar University
Barry Pollack	City College of San Francisco
B. Ravikumar	University of Rhode Island
David Riley	University of Wisconsin (La Crosse)
Bob Roos	Allegheny College
Carolyn Rosiene	University of Hartford
Jerry Ross Lane	Community College
Patricia Roth	Southeastern Polytechnic State University
Carolyn Schauble	Colorado State University
Arjit Sengupta	Georgia State University
Bennet Setzer	Kennesaw State University
Vijay Srinivasan	JavaSoft, Sun Microsystems, Inc.
Stuart Steiner	Eastern Washington University
Katherine St. John	Lehman College, CUNY
Alexander Stoytchev	Iowa State University
Ed Timmerman	University of Maryland, University College
Shengru Tu	University of New Orleans
Paul Tymann	Rochester Institute of Technology
John J. Wegis	JavaSoft, Sun Microsystems, Inc.
Ken Williams	North Carolina Agricultural and Technical University
Linda Wilson	Dartmouth College
David Wittenberg	Brandeis University
Wang-Chan Wong	California State University (Dominguez Hills)

还要感谢我的朋友和以前在 Villanova 大学工作的同事，他们提供了很多建设性意见，这些同事包括 Bob Beck、Cathy Helwig、Anany Levitin、Najib Nadi、Beth Taddei 以及 Barbara Zimmerman。还要感谢 Pete DePasquale，他以前在 New Jersey 学院工作，现供职于 SailThru 公司。

还有许多人以各种方式提供了帮助，包括 Ken Arnold、Mike Czepiel、John Loftus、Sebastian Niezgoda 和 Saverio Perugini。在此还向所有遗漏了名字的朋友们致以深深的歉意。

美国计算机学会（ACM）的计算机科学教育特别兴趣组（SIGCSE）是一个巨大的教育信息交流机构，他们的会议给来自不同层次与不同类型的学校的教师提供了交流教育新观点、新思路和资源的机会。对于从事某一计算机领域的教育工作者来说，如果还未加入 SIGCSE，就会错过参与这种教育信息交流的机会。



Pearson

尊敬的老师:

您好!

为了确保您及时有效地申请培生整体教学资源,请您务必完整填写如下表格,加盖学院的公章后传真给我们,我们将会 在 2-3 个工作日内为您处理。

请填写所需教辅的开课信息:

采用教材			☐ 中文版 ☐ 英文版 ☐ 双语版
作 者		出版社	
版 次		ISBN	
课程时间	始于 年 月 日	学生人数	
	止于 年 月 日	学生年级	☐ 专 科 ☐ 本科 1/2 年级 ☐ 研究生 ☐ 本科 3/4 年级

请填写您的个人信息:

学 校			
院系/专业			
姓 名		职 称	☐ 助教 ☐ 讲师 ☐ 副教授 ☐ 教授
通信地址/邮编			
手 机		电 话	
传 真			
official email(必填) (eg:XXX@ruc.edu.cn)		email (eg:XXX@163.com)	
是否愿意接受我们定期的新书讯息通知: ☐ 是 ☐ 否			

系 / 院主任: _____ (签字)

(系 / 院办公室章)

年 月 日

资源介绍:

—教材、常规教辅(PPT、教师手册、题库等)资源: 请访问www.pearsonhighered.com/educator; (免费)

—MyLabs/Mastering 系列在线平台: 适合老师和学生共同使用; 访问需要 Access Code; (付费)

100013 北京市东城区北三环东路 36 号环球贸易中心 D 座 1208 室
电话: (8610) 57355003 传真: (8610) 58257961

Please send this form to:

目 录

第 1 章 计算机系统概述	1
1.1 计算机处理过程	1
1.1.1 软件分类	2
1.1.2 数字计算机	4
1.1.3 二进制数	5
1.2 硬件组件	7
1.2.1 计算机结构	7
1.2.2 输入/输出设备	8
1.2.3 主存储器 and 辅助存储器	9
1.2.4 中央处理器	11
1.3 网络	13
1.3.1 网络连接	13
1.3.2 局域网和广域网	14
1.3.3 因特网	15
1.3.4 万维网	15
1.3.5 统一资源定位器	16
1.4 Java 编程语言	17
1.4.1 Java 程序	18
1.4.2 注释	19
1.4.3 标识符和保留字	20
1.4.4 空白符	22
1.5 程序开发	24
1.5.1 编程语言的级别	24
1.5.2 编辑器、编译器和解释器	25
1.5.3 开发环境	26
1.5.4 语法和语义	27
1.5.5 错误	27
1.6 面向对象编程	28
1.6.1 问题求解	29
1.6.2 面向对象的软件原理	30
重要概念小结	32
练习题	33
编程项目	34

第 2 章 数据与表达式	36
2.1 字符串	36
2.1.1 print 和 println 方法	36
2.1.2 字符串的连接	37
2.1.3 转义序列	39
2.2 变量与赋值	41
2.2.1 变量	41
2.2.2 赋值语句	42
2.2.3 常量	43
2.3 基本数据类型	44
2.3.1 整型与浮点型	44
2.3.2 字符型	46
2.3.3 布尔型	47
2.4 表达式	47
2.4.1 算术运算符	47
2.4.2 运算符优先级	48
2.4.3 自增和自减运算符	50
2.4.4 赋值运算符	51
2.5 数据类型转换	52
2.5.1 数据类型转换技术	53
2.6 交互式程序	55
2.6.1 Scanner 类	55
重要概念小结	58
练习题	58
编程项目	60
第 3 章 类与对象	62
3.1 创建对象	62
3.1.1 别名	64
3.2 String 类	65
3.3 包	67
3.3.1 import 声明	69
3.4 Random 类	70
3.5 Math 类	72

3.6 格式化输出	74	5.1.2 逻辑运算符	122
3.6.1 NumberFormat 类	74	5.2 if 语句	124
3.6.2 DecimalFormat 类	76	5.2.1 if-else 语句	126
3.6.3 printf 方法	77	5.2.2 使用语句块	129
3.7 枚举类型	78	5.2.3 嵌套 if 语句	131
3.8 包装器类	80	5.3 数据比较	133
3.8.1 自动装箱	81	5.3.1 浮点数比较	133
3.9 JavaFX 简介	81	5.3.2 字符比较	133
3.10 基本形状	84	5.3.3 比较对象	134
3.11 颜色呈现	88	5.4 while 语句	135
重要概念小结	89	5.4.1 无限循环	139
练习题	90	5.4.2 嵌套循环	139
编程项目	91	5.4.3 break 语句与 continue 语句	142
第 4 章 编写类	93	5.5 迭代器	143
4.1 类与对象的核心概念	93	5.5.1 读取文本文件	143
4.2 类的分析	94	5.6 ArrayList 类	145
4.2.1 实例数据	98	5.7 确定事件源	148
4.2.2 UML 类图	98	5.8 管理字体	150
4.3 封装	99	5.9 复选框	151
4.3.1 可见性修饰符	100	5.10 单选钮	154
4.3.2 访问器和修改器	100	重要概念小结	157
4.4 方法的分析	101	练习题	157
4.4.1 return 语句	102	编程项目	159
4.4.2 参数	103	第 6 章 其他条件判断与循环	163
4.4.3 局部数据	103	6.1 switch 语句	163
4.4.4 银行账户示例	104	6.2 条件运算符	166
4.5 构造方法回顾	107	6.3 do 语句	167
4.6 弧	107	6.4 for 语句	169
4.7 图形	110	6.4.1 for-each 循环	172
4.7.1 视口	111	6.4.2 各种循环的比较	172
4.8 图形用户界面	112	6.5 对图形使用循环和条件	173
4.8.1 指定事件处理器的其他 方法	114	6.6 图形转换	176
4.9 单行文本框	114	6.6.1 平移	176
重要概念小结	117	6.6.2 缩放	177
练习题	117	6.6.3 旋转	177
编程项目	118	6.6.4 错切	178
第 5 章 条件判断与循环	121	6.6.5 按组进行转换	178
5.1 布尔表达式	121	重要概念小结	181
5.1.1 相等性运算符与关系运算符	122	练习题	181

编程项目

编程项目

编程项目

编程项目

编程项目

编程项目

第7章 面向对象设计	185	8.6 二维数组	245
7.1 软件开发活动	185	8.6.1 多维数组	248
7.2 明确类和对象	186	8.7 多边形和折线	249
7.2.1 类职责的分派	187	8.8 Color 对象数组	250
7.3 静态类成员	188	8.9 选项框	253
7.3.1 静态变量	188	重要概念小结	255
7.3.2 静态方法	188	练习题	256
7.4 类间关系	190	编程项目	257
7.4.1 依赖关系	190	第9章 继承	260
7.4.2 同一个类的对象之间的 依赖性	191	9.1 创建子类	260
7.4.3 聚合	195	9.1.1 protected 修饰符	264
7.4.4 this 引用	198	9.1.2 super 引用	264
7.5 接口	199	9.1.3 多继承	267
7.5.1 Comparable 接口	203	9.2 重写方法	267
7.5.2 Iterator 接口	203	9.2.1 影子变量	269
7.6 枚举类型	204	9.3 类层次结构	269
7.7 方法设计	205	9.3.1 Object 类	270
7.7.1 方法分解	206	9.3.2 抽象类	271
7.7.2 方法参数的传递方式	209	9.3.3 接口层次	272
7.8 方法重载	213	9.4 可见性	273
7.9 测试	214	9.5 继承关系的设计	275
7.9.1 评审	215	9.5.1 继承的限制	276
7.9.2 缺陷测试	215	9.6 JavaFX 中的继承	276
7.10 GUI 设计	217	9.7 颜色和日期选择器	278
7.11 鼠标事件	217	9.8 对话框	280
7.12 按键事件	221	9.8.1 文件选择器	282
重要概念小结	223	重要概念小结	284
练习题	223	练习题	285
编程项目	224	编程项目	285
第8章 数组	227	第10章 多态性	288
8.1 数组元素	227	10.1 后绑定	288
8.2 声明和使用数组	228	10.2 利用继承实现多态性	289
8.2.1 边界检查	230	10.3 利用接口实现多态性	297
8.2.2 数组声明方式	233	10.4 排序	299
8.2.3 数组初始值表	233	10.4.1 选择法排序	299
8.2.4 数组作为参数	234	10.4.2 插入法排序	303
8.3 对象数组	235	10.4.3 排序算法的比较	304
8.4 命令行实参	242	10.5 搜索	305
8.5 可变长度参数表	243	10.5.1 线性搜索	305
		10.5.2 二分法搜索	308

10.5.3 搜索算法的比较	309	练习题	360
10.6 多态性设计	309	编程项目	361
10.7 属性	310	第 13 章 集合	363
10.7.1 变化监听器	312	13.1 集合与数据结构	363
10.8 滑动条	314	13.1.1 分离接口与实现	363
10.9 微调器	315	13.2 数据结构的动态表示	364
重要概念小结	317	13.2.1 动态结构	364
练习题	318	13.2.2 动态链表	365
编程项目	318	13.2.3 其他的动态列表	368
第 11 章 异常处理	319	13.3 线性集合	369
11.1 异常	319	13.3.1 队列	369
11.2 未捕获的异常	320	13.3.2 栈	370
11.3 try-catch 语句	321	13.4 非线性数据结构	372
11.3.1 finally 子句	323	13.4.1 树	372
11.4 异常的传递	324	13.4.2 图	373
11.5 异常类层次结构	326	13.5 Java 集合类 API	374
11.5.1 检查型与非检查型异常	328	13.5.1 泛型	374
11.6 I/O 异常	329	重要概念小结	375
11.7 工具提示与禁用控件	332	练习题	375
11.8 滚动面板	334	编程项目	376
11.9 分隔面板和列表视图	336	附录 A 术语表	378
重要概念小结	339	附录 B 数制系统	395
练习题	339	附录 C Unicode 字符集	399
编程项目	340	附录 D Java 运算符	401
第 12 章 递归	341	附录 E Java 修饰符	405
12.1 递归思想	341	附录 F Java 编码指南	407
12.1.1 无穷递归	342	附录 G JavaFX 布局面板	411
12.1.2 数学中的递归运算	342	附录 H JavaFX Scene Builder	418
12.2 递归编程	343	附录 I 正则表达式	425
12.2.1 递归与迭代的比较	344	附录 J javadoc 文档生成器	426
12.2.2 直接递归与间接递归	345	附录 K Java 语法	429
12.3 递归的应用	346	附录 L 自测题答案	438
12.3.1 迷宫旅行	346		
12.3.2 汉诺塔	349		
12.4 平铺图形	353		
12.5 分形	355		
重要概念小结	360		

第 1 章 计算机系统概述

本章目标

- 描述计算机软、硬件之间的关系
- 定义软件的各种类型及用法
- 定义并解释计算机的核心硬件组件及其作用
- 解释计算机硬件组件如何协同执行程序和管理数据
- 描述计算机如何与网络连接以便共享信息
- 介绍 Java 编程语言
- 描述程序编译和执行的步骤
- 概述面向对象技术的基本原理

本书主要讨论如何编写设计精良的软件。为了理解什么是软件，首先必须对软件在计算机系统中的地位 and 作用有基本的理解。计算机系统软件 and 硬件相互配合，共同完成各种复杂的任务。理解各种硬件组件的作用及这些组件连接到网络中的方式，是学习软件开发的先决条件。本章首先讨论基本的计算机处理过程，然后通过介绍 Java 编程语言及面向对象的编程原理来探讨软件开发。

1.1 计算机处理过程

所有的计算机系统，不管是台式机、笔记本、智能手机、游戏机，还是诸如汽车导航系统的专用设备，都具有某些共同的特性。尽管形状和功能大不相同，但它们都以类似的方式处理数据。尽管本书的主要目的是讲解如何进行软件开发，但首先会回顾计算机的处理过程，以便于理解上下文。首先需定义一些基本术语，并描述计算机系统的核心组件是如何相互合作的。

计算机系统由软件和硬件组成。计算机硬件组件是实际存在的物理系统，支持着计算机的实际运算处理能力。这些组件包括芯片、机箱、电子线路、键盘、扬声器、磁盘、主存卡、USB 闪存（也称为 Jump 存储器）、电缆、插头、打印机、鼠标、显示器及路由器等。所有能直接触摸到的计算机系统组成部分，都是计算机硬件。

重要概念：计算机系统由硬件和软件组成，软、硬件协同工作帮助我们解决问题。

实际上，如果离开了指示计算机工作的相关指令，计算机硬件系统是没有任何用处的。程序就是一系列的指令，由硬件逐条执行。软件则包括程序及程序执行所需要的数据。软件是与硬件系统配合的无形部分。软件和硬件结合，便组成了我们最终解决问题的工具。

计算机系统的主要硬件组件如下：

- 中央处理单元（CPU）
- 输入/输出（I/O）设备
- 主存储器
- 辅助存储设备

这些重要的组件将在 1.2 节详细介绍，现在只需要简单了解它们的基本作用。中央处理单元（CPU）是执行每一条程序指令的设备。输入/输出（I/O）设备，比如键盘、鼠标、触控板和显示器等，提供了人机交互界面。

程序和数据则被保存在存储设备（称为存储器）中。存储器分为两种：主存储器和辅助存储器。主存储器用于保存正在被 CPU 执行的程序。辅助存储器则是以永久保存的方式存储软件。在一个典型的计算机系统中，最为重要的辅助存储器是安装在机箱中的硬盘。USB 闪存也是一种重要的辅助存储设备。通常而言，USB 闪存的容量不及硬盘。USB 闪存的优点是可移动性，它可随时卸载或者在计算机间来回使用。

图 1.1 演示了信息是如何在计算机硬件组件之间传递的。假设现在希望运行一个可执行程序，而该程序本身保存在某个辅助存储设备（例如硬盘）中。当计算机执行这个程序时，系统从辅助存储设备将该程序复制到主存储器中。CPU 则从主存储器中依次读取每一条程序指令并且执行，一次执行一条，直到程序结束。程序需要的数据（例如，加法中的两个加数），也同样保存在主存储器中。这些数据可以来自辅助存储设备，也可以是从输入设备（如键盘）得到的。执行过程中，程序可能会将一些信息显示到输出设备上，如显示器。

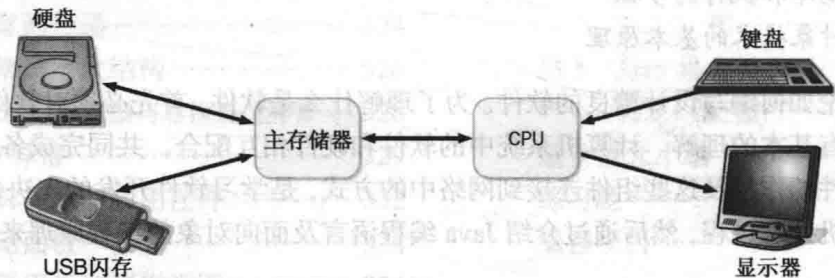


图 1.1 计算机系统的简单示意图

程序的执行过程是计算机的基本操作方式，所有的计算机系统本质上都以这种方式工作。

重要概念：CPU 从主存储器中读取程序的每一条指令，并且每次执行一条指令直到程序结束。

1.1.1 软件分类

依照不同的标准，可以将软件分为很多种类。这里只简单地将它们分为系统程序和应用程序。

操作系统是计算机的核心软件，它主要实现两种重要功能。首先，操作系统提供用户界面，使用户可以与计算机交互；其次，操作系统管理计算机资源，比如 CPU 和主存储器。操作系统决定什么时候执行程序、从哪里将程序载入存储器，以及硬件设备之间如何通信。操作系统的任务，是使得计算机更容易使用，并且保证计算机能有效地运行。

重要概念：操作系统提供用户界面并管理计算机资源。

现在有很多流行的操作系统在世界范围内使用。由 Microsoft 公司为个人计算机系统开发的 Windows 操作系统，占据操作系统市场的大份额。各种版本的 UNIX 操作系统也很流行，特别是在大型计算机系统中。UNIX 的一个版本称为 Linux，Linux 系统的开发是一个开源项目，这意味着很多人都可以参与该系统的开发，并且它的代码是公开的和共享的，因此，Linux 成为备受一些用户偏爱的系统。Mac OS 是用于苹果公司（Apple Computers）产品的操作系统，它由该公司开发。