

高等教育质量工程信息技术系列示范教材

新概念

Java程序设计 大学教程

(第3版)

张基温 编著

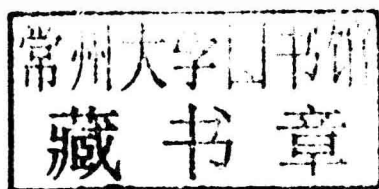
清华大学出版社



高等教育质量工程信息技术系列示范教材

新概念 Java程序设计大学教程 (第3版)

张基温 编著



清华大学出版社
北京

内 容 简 介

本书结构新颖、概念清晰、面向应用,体现了作者提出的“程序设计=计算思维+语言艺术+工程方法”的教学思想。全书共分为4篇:第1篇为面向对象启步,用4个例子引导读者逐步建立面向对象的思维方式和培养基本的设计能力,将Java基本语法贯穿其中;第2篇为面向类的程序设计,在介绍了抽象类和接口这两个基本机制后,用一个故事引入了面向对象程序设计原则,接着用设计模式举例加深对面向对象结构优化必要性的认识,为进一步学习设计模式打下基础,最后介绍了反射技术;第3篇为基于API的开发,包括网络编程、JDBC、JavaBean、程序文档化、程序配置和打包与发布;第4篇为Java高级技术,包括泛型编程、多线程技术、数据结构和接口。通过这4篇可以达到夯实基础、面向应用、领略全貌的教学效果,并适应不同层次的教学需求。

本书采用问题体系,具有零起点、快启动、立意新、重内涵的特点,可作为高等学校有关专业的程序设计课程的教材,也适合培训和自学。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

新概念Java程序设计大学教程/张基温编著. —3版. —北京:清华大学出版社,2018

(高等教育质量工程信息技术系列示范教材)

ISBN 978-7-302-49057-9

I. ①新… II. ①张… 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(2017)第296435号

责任编辑:白立军 王冰飞

封面设计:常雪影

责任校对:时翠兰

责任印制:刘海龙

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦A座

邮 编:100084

社总机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质量反馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课件下载: <http://www.tup.com.cn>, 010-62795954

印 装 者:三河市君旺印务有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:26

字 数:607千字

版 次:2013年8月第1版 2018年3月第3版

印 次:2018年3月第1次印刷

印 数:1~1500

定 价:59.00元

产品编号:077035-01

第3版前言

(一)

程序设计是一个很古老的概念,它从算筹和算盘开始就存在了。程序设计也是一个不断更新的概念,它随着计算工具、程序设计语言、程序设计方法的不断创新而不断更新。并且随着现代科学技术的进步,其基础性、重要性日益彰显。在各级各类学校中,关于如何教好程序设计课程的问题,也越来越受到广泛的关注。

但是,从各个方面得到的信息说明,大家对于程序设计的教学效果还是没有肯定。

我应邀到过许多学校进行交流,那里的院长、系主任们抱怨最多的是,学生学了好几门程序设计语言课,但还是遇到编程就头疼。

2004年12月,在苏州大学举办的全国计算机教学研讨会上,我根据在许多学校调研的情况,提出了计算机专业的课程中学得最不好的课程就是程序设计,真正的过关率也就30%。2008年,在北京举办的一次院长、系主任论坛上,时任教育部高教司司长的张尧学院士提出,计算机专业中,80%左右的学生程序设计没有过关。这足以说明问题的严重性。

这一局面正是我不断进行程序设计教材探索的动力。

(二)

本书是《新概念Java程序设计大学教程》的第3版。它是我在教学和交流中不断探索的一个新的成果。在写作中,我注意解决了如下几个问题。

1. 先入为主地让学生一开始就进入面向对象的世界

人人都说,由于对象反映的是真实世界的对象,面向对象程序设计比较直接,容易理解。但是,实际情况是,学生学习了面向对象程序设计后,写出来的程序却是面向过程的。原因就在于我们许多教材是从面向过程开始介绍面向对象的。这种先入为主的面向过程,是不能很好地培养面向对象的思维和方法的。只有先入为主地从面向对象开始,才能让学习者根深蒂固地掌握面向对象的思维和方法。

2. 把程序测试的基本方法融入程序设计

在计算机专业的教学中,程序测试是软件工程中的内容。而在此之前就开设了程序设计课程。这样,就把程序设计与程序测试割裂开来。在学习程序设计时,一般都是以“试通”方式进行。这种“劣习”到了学习软件工程时,已经积重难返。而且,在软件工程学习时,也没有更多时间和机会进行程序测试训练了。因此,在学习程序设计的同时,让学生掌握一些基本的程序测试方法非常重要。

3. 设计模式与面向对象程序设计准则

在面向对象程序设计的实践中,人们发掘出了一些模式。这些模式对于设计者具有标杆性的启示作用。但是,就 23 种模式来说,也不是一门程序设计课程所能容纳的。所幸,人们又从这些模式中总结出了开闭原则、面向抽象、接口分离、单一职责、迪米特等法则。这些法则精炼,指导意义更大。但是,它们又非常抽象。经过反复琢磨,本书采取了故事引导的方法来介绍这些原则,而把几个常用的典型模式作为实例的方法。

4. 程序文档化

本书还注意了程序文档化的训练,介绍了标注(annotation)和 Javadoc 等。

(三)

本书没有沿袭 Java 教材从数据类型到控制结构的思路,而是在第 1 篇直接用 6 个实例按照“定义类—定义引用—创建对象—操作对象”的过程进行面向对象思维的训练,形成面向对象的思维主线,把数据类型、控制结构等语法嵌于其中。同时介绍基于组件的测试方法。

不了解设计模式,就没有掌握面向对象的真谛。本书第 2 篇以面向抽象为题,首先介绍抽象类和接口,然后引出面向对象的几个基本原则,接着通过 3 个例子引出 GoF 设计模式,最后介绍反射、配置文件和程序打包发布。这一篇不仅让学生了解了面向对象的真正意义,也更贴近了应用实际。

第 3 篇通过多线程、图形用户界面、网络编程、JavaBean 和持久化,加深对于 API 意义和应用的理解决,为开发应用程序奠定基础。

第 4 篇介绍 Java Web 开发和软件架构。

这 4 篇有详有略。详者为夯实 Java 开发的坚实基础,并学到实用的本领;略者为读者了解 Java 技术的全貌,以便将来确定在何处突破。这 4 篇也形成了 4 个学习层次,便于相关教学单位根据教学对象和目的进行取舍。

(四)

我国著名教育家陶行知说道:“行动是老子,知识是儿子,创造是孙子”,并倡导“知行合一”。世界上第一所完全为发展现代设计教育而建立的学院的创始人——包豪斯(Bauhaus)的名言“干中学(learning from doing)”已经成为现代教育的重要思想。所有这些都表明了实践在学习中的重要性。程序设计更是这样,仅仅学习了一些程序设计语言的语法,仅仅了解了一些程序设计的方法,仅仅有了“知”,而不一定“会”。要想会,就要实践。

由 J. Piaget、O. Kernberg、R. J. Sternberg、D. Katz、Vogotsky 等人创建的建构主义(constructivism)学习理论认为,知识不是通过教师传授得到,而是学习者在一定的情境即社会文化背景下借助其他人(包括教师和学习伙伴)的帮助,利用必要的学习资料,通过意义建构的方式而获得的。在信息时代,人们获得知识的途径发生了根本性的变化,教师不再是单一的“传道、授业、解惑”者,帮助学习者构建一个良好的学习环境也成为其一项重要职责。

当然,这也是现代教材的责任。本书充分考虑了这些问题。

在这本书中,在每章后面都安排了概念辨析、代码分析、开发实践和探索深究 4 种自测和训练实践环节,为学习者搭建起一个立体化的实践环境。

1. 概念辨析

概念辨析主要提供选择和判断两类自测题目,帮助学习者理解本章中学习过的有关概念,把当前学习内容所反映的事物尽量和自己已经知道的事物相联系,并认真思考这种联系,通过“自我协商”与“相互协商”,形成新知识的同化与顺应。对于这种类型的习题,读者应当按照如下顺序完成:

- (1) 先给出自己的判断。
- (2) 设计一个小的程序,验证自己的判断。
- (3) 结合自己的判断对验证结果进行分析,说明原因。

2. 代码分析

代码阅读是程序设计者的基本能力之一。代码分析部分的主要题型有 4 种:

- (1) 要求给出执行结果。
- (2) 要求找出错误。
- (3) 选择一个答案。
- (4) 填写一个空白。

3. 开发实践

提高程序开发能力是本书的主要目标。本书在每章后面都给出了相应的作业题目。但是,完成这些题目并非就是简单地写出代码,而且要将它看作是一个思维+语法+方法的工程训练。因此,要求在上机之前先写出准备文档。准备文档包括以下内容:

- (1) 问题分析与建模。
- (2) 源代码设计。
- (3) 测试用例设计。

这些文档内容的准备应当作为是否可以上机作业的条件。没有做这些准备的学生,上机就是盲目行为,收获不会太大。在学校中,学生进入机房之前,教师应当先检查学生是否已经准备了这些内容,并将准备情况作为该次上机作业成绩的一部分或不允许上机。

经过上机作业,学生还应当提交作业报告。作业报告包括如下内容:

- (1) 上机作业时发现的问题。
- (2) 对于发现的问题采取的调试方法。
- (3) 对自己准备文档中给出的测试用例的评价。
- (4) 程序运行结果分析。
- (5) 编程心得。

4. 探索思考

建构主义提倡,学习者要用探索法和发现法去建构知识的意义,学习者要在意义建构的过程中主动地搜集和分析有关的信息资料,对碰到的问题提出各种假设并努力加以验证。按照这一理论,本书还提供了一个探索思考栏目,以培养学习者获取知识的能力和不断探索的兴趣,同时可以从更深层次上探究 Java 语法。

(五)

写书难,写教材更难。一本专著,仅用于表达自己的见地;而一本好的教材,不仅是科学技术知识和方法的精华,还应当是先进教育理念的结晶。离这些,我还差得很远。好在有 20 余年教学的经历和不断进行教学改革探索的积累,以及许多热心者的支持和帮助。

这一版终于要问世了。在此,我要衷心感谢在这本书的出版过程中付出了劳动的下列人员:姚威、古辉、陶利民、赵忠孝、张展为、张秋菊、史林娟、张友明、李磊、张展赫、戴璐、文明瑶、陈觉、吴灼伟(插图)。

本书的出版仅仅是我程序设计教学改革中的一个新台阶,前面的路还要继续走下去。为此,衷心希望能得到有关专家和读者的批评和建议,也希望结交一些志同道合者,把这项改革推向更新的境界。

张基温
2018 年 1 月

第 2 版前言

本书是《新概念 Java 程序设计大学教程》的第 2 版。这一版修订有如下考虑：

(1) 第 1 版把 JSP 作为 Java 技术的一部分介绍,这样可以形成一个完整的 Java 技术体系,在教学中一气呵成,可以提高教学效率。但是目前多数学校将 Java 基础与 Java Web 分开,作为两门课程进行讲授。第 2 版就是为了满足这种需求而进行的修订——将 Java Web 程序开发有关的部分删除,另行成册。

现在,Java 有两个应用方面:桌面系统应用和 Web 系统应用。做这两个题目时,把 GUI 加进来,就是一个完整的桌面系统开发实践。再加上 Servlet 和 JSP 的学习,完成一个 Web 系统的开发,就可以全面领略 Java 技术了。

(2) 在开发实践和教学中,越来越体会到设计模式的重要性。但其内容相当抽象,并且设计模式也不仅仅限于 GoF 的 23 种模式,它是一个宽泛又不断发展的概念。为此,在这次修订中,除了继续按照第 1 版的方法,用一个故事趣味性地引出面向对象程序设计原则、列举了几种设计模式,并介绍了 JavaBean 之外,还在 JDBC 最后引出了 DAO 模式,并根据 DAO 模式的需要对前面的设计模式举例进行了改写,形成一个贯穿全书的设计模式思想。

(3) 现代程序设计强调代码的可读性和安全性,因此,第 2 版增加了关于 Annotation 和 Javadoc 的介绍。

(4) 程序设计语言在不断发展,在教学中还应当适时介绍一些在基本内容上扩展的 Java 机制,如:

- 正则表达式。
- JVM 运行时数据区。
- 泛型编程、Java 数据接口,将它们与多线程一起作为 Java 高级技术介绍。

这样,就形成了第 2 版的 4 篇内容:

第 1 篇为面向对象启步,用 5 个例子引导读者逐步建立面向对象的思维方式和培养基本的设计能力,将 Java 基本语法贯穿其中;第 2 篇为面向抽象的编程,主要介绍抽象类、接口、面向对象程序设计原则、设计模式举例和反射技术;第 3 篇为基于 API 的开发,包括图形用户界面、网络编程、JDBC、JavaBean、程序文档化、程序配置和打包与发布;第 4 篇为 Java 高级技术,包括泛型编程、多线程技术、数据结构和接口。

在本书的前面一些单元中,除了基本内容外,还开辟了“知识链接”栏目,作为基本内容的扩展和延伸。这些内容可以作为选学或自学内容。

程序设计是一门实践性极强的课程。其实践包含两个方面的训练:思维能力训练和语言运用能力训练。本书为此提供了丰富的习题。此外,在学时安排上还建议增加相应课程设计。经过多年的教学实践,本人推荐 3 种课程设计题目:

- 多人聊天。
- 一个信息系统的 DAO。

- 关于数据结构的应用。

在做这些题目时可以分别提出如下要求：

- 使用图形用户界面。
- 使用 JavaBean。
- 使用反射。
- 使用有关模式以及 MVC。
- 使用多线程。
- 使用 Javadoc 和 Annotation。
-

在第 2 版即将出版之际,要特别感谢传智播客的高美云先生仔细审读了本书的有关章节,提出了非常中肯的修改意见;也要感谢参加过一些编写和校阅工作的赵忠孝、陶利民、姚威、古辉、张展为、李磊、董兆军、张秋菊、史林娟、张友明、张展赫、陈觉、吴灼伟(插图)等人。

本书没有采用传统的语法体系,而是采用了自己倡导的“问题 + 计算思维”体系编写。这个体系已经经过了十几年的摸索,在新一版即将面世之际,衷心地希望读过、用过本书的有关专家和学习者能不吝批评指正,提出宝贵意见,把程序设计教学改革向前再推进一步。

张基温
2016 年 6 月

第1版前言

这是我第3次写Java教材了。第1次是2001年,应清华大学出版社之约,写了《Java程序开发教程》,并配有一本习题解答。第2次是2010年,应中国电力出版社之约,写了《新概念Java教程》。这一次又应清华大学出版社之约,写了《新概念Java程序开发大学教程》。

近几年,我所写的程序设计教材都冠以“新概念”。所谓“新概念”,并非我能在一种程序设计语言中添加什么概念,而是企图建立一种新的程序设计教学的模式来改变程序设计课程教学效率不高,甚至不成功的现状。

(一)

在多年的程序设计课程教学实践以及广泛地与国内同行的交流中,自己感到或听到的,都是认为程序设计课程不太成功的说法。

有的学校的系主任(院长)抱怨,学生学习了几门程序设计课程,可是到了课程设计、毕业设计,遇到问题还是下不了手。

多数程序设计课程老师都认为,教了C++、Java,可是学生遇到的问题,写出来的代码还是面向过程的。

许多学生告诉我,不知道如何测试程序,甚至学习了软件工程以后,设计了程序也还是简单地试通一下,根本没有规范化测试的习惯。

企业界的朋友告诉我,企业界已经在关注设计模式和软件架构,而新入职的软件专业的大学生对此还完全没有概念……

所有这些问题都落实到一点上:程序设计课程应当教什么?应当如何教?

(二)

教材是教学的剧本。程序设计课程的改革首先应当从教材改革开始。创新是发展的动力,教材的改革要求教材有所创新。

计算机程序设计教材的创新,首先要改变程序设计教材基于语法体系的结构。说到底,语法体系的程序设计教材都是程序设计语言手册的翻版。这种语法体系造就了重语法教学、轻思维训练的教学模式,是学习了程序设计课程却不会编写程序的祸根。

新概念系列力图在这个方面创出一条新路。本书没有沿袭Java教材从数据类型、控制结构开始的思路,而是在第1篇直接用6个实例按照“定义类—定义引用—创建对象—操作对象”的过程进行面向对象思维的训练,形成面向对象的思维主线,把数据类型、控制结构等语法嵌于其中。同时介绍基于组件的测试方法。

不了解设计模式,就没有掌握面向对象的真谛。本书第2篇以面向抽象为题,首先介绍抽象类和接口,然后引出面向对象的几个基本原则,接着通过3个例子引出GoF设计模式,

最后介绍反射、配置文件和程序的打包与发布。这一篇不仅让学生了解了面向对象的真正意义,也更贴近了实际应用。

第3篇通过多线程、图形用户界面、网络编程、JavaBean和持久化加深对于API意义和应用的理 解,为开发应用程序奠定基础。

第4篇介绍Java Web开发和软件架构。

这4篇有详有略。详者为夯实Java开发的坚实基础,并学到实用的本领;略者为读者了解Java技术的全貌,以便将来确定在何处突破。这4篇也形成4个学习层次,便于有关教学单位根据教学对象和目的进行取舍。

(三)

我国著名教育家陶行知说过,“行动是老子,知识是儿子,创造是孙子”,并倡导“知行合一”。世界上第一所完全为发展现代设计教育而建立的学院——包豪斯(Bauhaus)的名言“干中学(learning from doing)”已经成为现代教育的重要思想。所有这些都表明了实践在学习中的重要性。程序设计更是这样,仅仅学习了一些程序设计语言的语法,仅仅了解了一些程序设计的方法,仅仅有了“知”,而不一定“会”。要想会,就要实践。

由J. Piaget、O. Kernberg、R. J. Sternberg、D. Katz、Vogotsky等创建的建构主义(constructivism)学习理论认为,知识不是通过教师传授得到的,而是学习者在一定的情境(即社会文化背景)下借助其他人(包括教师和学习伙伴)的帮助,利用必要的学习资料,通过意义建构的方式而获得的。在信息时代,人们获得知识的途径发生了根本性的变化,教师不再是单一的“传道、授业、解惑”者,帮助学习者构建一个良好的学习环境也成为其一项重要职责。当然,这也是现代教材的责任。本书充分考虑了这些问题。

本书的每一单元后面都安排了概念辨析、代码分析、开发实践和思考探索4种自测和训练实践环节,为学习者搭建起一个立体化的实践环境。

1. 概念辨析

概念辨析主要提供选择和判断两类自测题目,帮助学习者理解本节学习过的有关概念,把当前学习内容所反映的事物尽量和自己已经知道的事物相联系,并认真思考这种联系,通过“自我协商”与“相互协商”形成新知识的同化与顺应。对于这种类型的习题,读者应当按照如下顺序完成:

- (1) 先给出自己的判断。
- (2) 设计一个小的程序,验证自己的判断。
- (3) 结合自己的判断对验证结果进行分析,说明原因。

2. 代码分析

代码阅读是程序设计者的基本能力之一。代码分析部分的主要题型有4种:

- (1) 要求给出执行结果。
- (2) 要求找出错误。
- (3) 选择一个答案。

(4) 填写一个空白。

3. 开发实践

提高程序开发能力是本书的主要目标。本书在绝大多数单元后面都给出了相应的作业题目。但是,完成这些题目并非就是简单地写出其代码,而要将它看作是一个思维+语法+方法的工程训练。因此,要求在上机之前先写出准备文档。准备文档的内容包括:

- (1) 问题分析与建模。
- (2) 源代码设计。
- (3) 测试用例设计。

这些文档内容的准备应当作为是否可以上机作业的条件。没有这些准备,上机就比较盲目,收获不会太大。在学校中,学生进入机房之前,教师应当先检查学生是否已经准备了这些内容,并将准备情况作为该次上机作业成绩的一部分或不允许上机。

经过上机作业,学生还应当提交作业报告。作业报告包括如下内容:

- (1) 上机作业时发现的问题。
- (2) 对于发现的问题采取的调试方法。
- (3) 对自己所准备文档中给出的测试用例的评价。
- (4) 程序运行结果分析。
- (5) 编程心得。

4. 思考探索

建构主义提倡,学习者要用探索法和发现法去建构知识的意义,学习者要在意义建构的过程中主动地搜集和分析有关的信息资料,对碰到的问题提出各种假设并努力加以验证。按照这一理论,本书还提供了一个探索思考栏目,以培养学习者获取知识的能力和不断探索的兴趣,同时可以从更深层次上探究 Java 语法。

(四)

写书难,写教材更难。一本专著,仅用于表达自己的见地;而一本好的教材,不仅是科学技术知识和方法的精华,还应当是先进教育理念的结晶。离这些,我还差得很远,好在有二十余年教学的经历和不断进行教学改革探索的积累,以及许多热心者的支持和帮助。

在此,我要衷心感谢在这本书的编写中参加了一定工作的下列人员:姚威、陶利民、张展为、郎贵义、李磊、董兆军、张秋菊、史林娟、文明瑶、黄姝敏、陈觉、张友明、宋文炳、许剑生、贺竞峰等。此外,还要感谢清华大学出版社为本书出版所做的大量平凡而细致的工作。

本书就要出版了。它的出版是我在这项教学改革工作中跨上的一个新的台阶。我衷心希望能得到有关专家和读者的批评和建议,也希望结交一些志同道合者,把这项改革推向更新的境界。

张基温

2013年5月

目 录

第 1 篇 面向对象程序设计启步

第 1 单元 职员类	3
1.1 从现实世界中的对象到类模型	3
1.1.1 程序=模型+表现	3
1.1.2 现实世界中的对象分析	4
1.1.3 职员类的 UML 描述	5
1.1.4 职员类的 Java 语言描述	5
1.1.5 职员类的 Java 代码说明	6
1.2 类的应用与测试	8
1.2.1 对象引用及其创建	8
1.2.2 构造器与 this()	9
1.2.3 对象成员的访问与 this	11
1.2.4 主方法与主类	12
1.2.5 类文件与包	14
1.3 Java 程序开发	16
1.3.1 Java 编译器与 Java 虚拟机	16
1.3.2 JDK	16
1.3.3 Eclipse 开发环境	18
1.4 知识链接	24
1.4.1 Java 语言及其特点	24
1.4.2 Java 数据类型	26
1.4.3 字面值	27
1.4.4 基本类型的转换	29
1.4.5 Java 关键词与标识符	32
1.4.6 流与标准 I/O 流对象	33
1.4.7 Java 注释	34
习题 1	35
第 2 单元 计算器类	38
2.1 计算器类的定义	38
2.1.1 计算器建模	38
2.1.2 Calculator 类的 Java 描述	39

2.2	Calculator 类的测试	39
2.2.1	测试数据设计	39
2.2.2	规避整除风险——Calculator 类改进之一	40
2.3	异常处理——Calculator 类改进之二	42
2.3.1	Java 异常处理概述	42
2.3.2	Java 异常处理的基本形式	42
2.3.3	用 throws 向上层抛出异常	44
2.3.4	用 throw 直接抛出异常	45
2.3.5	Java 提供的主要异常类	47
2.4	用选择结构确定计算类型——Calculator 类改进之三	47
2.4.1	用 if...else 实现 calculate() 方法	47
2.4.2	关系操作符	49
2.4.3	用 switch 结构实现 calculate() 方法	50
2.5	用静态成员变量存储中间结果——Calculator 类改进之四	52
2.5.1	静态成员变量的性质	52
2.5.2	带有静态成员变量的 Calculator 类定义	52
2.6	知识链接	54
2.6.1	Java 表达式	54
2.6.2	静态方法——类方法	55
2.6.3	初始化块与静态初始化块	57
2.6.4	String 类	60
2.6.5	正则表达式	61
2.6.6	Scanner 类	64
习题 2	64
第 3 单元	素数序列产生器	72
3.1	问题描述与对象建模	72
3.1.1	素数序列产生器建模	72
3.1.2	getPrimeSequence() 方法的基本思路	72
3.2	使用 isPrime() 判定素数的 PrimeGenerator 类的实现	73
3.2.1	采用 while 结构的 getPrimeSequence() 方法	73
3.2.2	采用 do...while 结构的 getPrimeSequence() 方法	75
3.2.3	采用 for 结构的 getPrimeSequence() 方法	75
3.2.4	重复结构中的 continue 语句	76
3.2.5	采用 for 结构的 isPrime() 方法	76
3.2.6	将 isPrime() 定义为静态方法	77
3.2.7	不用 isPrime() 判定素数的 PrimeGenerator 类的实现	78
3.3	知识链接	79

3.3.1	变量的访问属性	79
3.3.2	变量的作用域	79
3.3.3	Java 数据实体的生命期	80
3.3.4	基本类型的包装	81
习题 3	83
第 4 单元	扑克游戏	90
4.1	数组与扑克牌的表示和存储	90
4.1.1	数组的概念	90
4.1.2	数组的声明与内存分配	90
4.1.3	数组的初始化	92
4.1.4	匿名数组	93
4.2	数组元素的访问	93
4.2.1	用普通循环结构访问数组元素	94
4.2.2	用增强 for 遍历数组元素	94
4.3	洗牌	95
4.3.1	随机数与 Random 类	95
4.3.2	洗牌方法设计	96
4.3.3	含有洗牌方法的扑克游戏类设计	98
4.4	扑克的发牌与二维数组	99
4.4.1	基本的发牌算法	99
4.4.2	用二维数组表示玩家手中的牌	100
4.4.3	使用二维数组的发牌方法	102
4.4.4	含有洗牌、发牌方法的扑克游戏类设计	102
4.5	知识链接	104
4.5.1	数组实用类 Arrays	104
4.5.2	java.util.Vector 类	105
4.5.3	命令行参数	107
4.5.4	Math 类	108
习题 4	109

第 2 篇 面向类的程序设计

第 5 单元	类的继承	115
5.1	学生类-研究生类层次结构	115
5.1.1	由 Student 类派生 GradStudent 类	115
5.1.2	super 关键字	117
5.1.3	final 关键字	119
5.2	Java 的访问权限控制	119

5.2.1	类成员的访问权限控制	119
5.2.2	类的访问权限控制	120
5.2.3	private 构造器	120
5.3	类层次中的类型转换	121
5.3.1	类层次中的赋值兼容规则	121
5.3.2	里氏代换原则	122
5.3.3	类型转换与类型测试	122
5.4	方法覆盖与隐藏	123
5.4.1	派生类实例方法覆盖基类中签名相同的实例方法	123
5.4.2	用@Override 标注覆盖	124
5.4.3	派生类静态方法隐藏基类中签名相同的静态方法	126
5.4.4	JVM 的绑定机制	127
5.5	知识链接	128
5.5.1	Object 类	128
5.5.2	@Deprecated 与@SuppressWarnings	131
5.5.3	Java 异常类和错误类体系	135
习题 5		137
第 6 单元	抽象类与接口	145
6.1	圆、三角形和矩形	145
6.1.1	3 个独立的类: Circle、Rectangle 和 Triangle	145
6.1.2	枚举	147
6.2	抽象类	148
6.2.1	由具体类抽象出抽象类	148
6.2.2	由抽象类派生出实例类	150
6.2.3	抽象类小结	153
6.3	接口	153
6.3.1	接口及其特点	153
6.3.2	接口的实现类	154
6.3.3	关于接口的进一步讨论	156
6.4	知识链接	157
6.4.1	Java 构件修饰符小结	157
6.4.2	对象克隆	158
习题 6		160
第 7 单元	面向对象程序架构优化原则	164
7.0	引言	164
7.1	从可重用说起: 合成/聚合优先原则	165

7.1.1	继承重用的特点	166
7.1.2	合成/聚合重用及其特点	166
7.1.3	合成/聚合优先原则	167
7.2	从可维护性说起：开-闭原则	168
7.2.1	软件的可维护性和可扩展性	168
7.2.2	开-闭原则	169
7.3	面向抽象的原则	170
7.3.1	具体与抽象	170
7.3.2	依赖倒转原则	170
7.3.3	面向接口原则	170
7.3.4	面向接口编程举例	171
7.4	单一职责原则	175
7.4.1	对象的职责	175
7.4.2	单一职责原则的概念	176
7.4.3	接口分离原则	176
7.5	不要和陌生人说话	181
7.5.1	狭义迪米特法则	181
7.5.2	广义迪米特法则	182
	习题 7	183

第 8 单元	设计模式	186
8.1	设计模式概述	186
8.2	设计模式举例——诉讼代理问题	186
8.2.1	无律师的涉讼程序设计	186
8.2.2	请律师代理的涉讼程序设计	187
8.2.3	关于代理模式	189
8.3	设计模式举例——商场营销问题	190
8.3.1	不用策略模式的商场营销解决方案	190
8.3.2	策略模式的定义	192
8.3.3	采用策略模式的商场营销解决方案	193
8.4	设计模式举例——图形对象的创建问题	197
8.4.1	简单工厂模式	197
8.4.2	工厂方法模式	202
8.4.3	策略模式与简单工厂模式结合	207
8.5	知识链接	209
8.5.1	类文件与类加载	209
8.5.2	Class 对象	209
8.5.3	反射 API	213