

普通高校本科计算机专业 **特色** 教材精选

Java 语言程序设计

吕凤翥 马 皓 编著

<http://www.tup.com.cn>

清华大学出版社



普通高校本科计算机专业**特色**教材精选

Java 语言程序设计

吕凤翥 马 皓 编著

清华大学出版社
北 京

编审委员会

主任：蒋宗礼

副主任：李仲麟 何炎祥

委员：（排名不分先后）

王向东 宁 洪 朱庆生 吴功宜 吴 跃

张 虹 张 钢 张为群 余雪丽 陈志国

武 波 孟祥旭 孟小峰 胡金初 姚放吾

原福永 黄刘生 廖明宏 薛永生

秘书长：王听讲

出版说明

INTRODUCTION

在我国高等教育逐步实现大众化后,越来越多的高等学校将会面向国民经济发展的第一线,为行业、企业培养各级各类高级应用型专门人才。为此,教育部已经启动了“高等学校教学质量和教学改革工程”,强调要以信息技术为手段,深化教学改革和人才培养模式改革。如何根据社会的实际需要,根据各行各业的具体人才需求,培养具有特色显著的人才,是我们共同面临的重大问题。具体地,培养具有一定专业特色的和特定能力强的计算机专业应用型人才则是计算机教育要解决的问题。

为了适应 21 世纪人才培养的需要,培养具有特色的计算机人才,急需一批适合各种人才培养特点的计算机专业教材。目前,一些高校在计算机专业教学和教材改革方面已经做了大量工作,许多教师在计算机专业教学和科研方面已经积累了许多宝贵经验。将他们的教研成果转化为教材的形式,向全国其他学校推广,对于深化我国高等学校的教学改革是一件十分有意义的事。

清华大学出版社在经过大量调查研究的基础上,决定编写出版一套“普通高校本科计算机专业特色教材精选”。本套教材是针对当前高等教育改革的新形势,以社会对人才的需求为导向,主要以培养应用型计算机人才为目标,立足课程改革和教材创新,广泛吸纳全国各地的高等院校计算机优秀教师参与编写,从中精选出版确实反映计算机专业教学方向的特色教材,供普通高等院校计算机专业学生使用。

本套教材具有以下特点:

1. 编写目的明确

本套教材是深入研究各地各学校办学特色的基础上,面向普通高校的计算机专业学生编写的。学生通过本套教材,主要学习计算机科学与技术专业的基本理论和基本知识,接受利用计算机解决实际问题的基本训练,培养研究和开发计算机系统,特别是应用系统的基本能力。

2. 理论知识与实践训练相结合

根据计算学科的三个学科形态及其关系,本套教材力求突出学科的理论与实践紧密结合的特征,结合实例讲解理论,使理论来源于实践,又进一步指导实践得到自然地体现,学生通过实践深化对理论的理解,更重要的是使学生学会理论方法的实际运用。在编写教材时突出实用性,并做到通俗易懂,易教易学,使学生不仅知其然,知其所以然,还要会其如何然。

3. 注意培养学生的动手能力

每种教材都增加了能力训练部分的内容,学生通过学习和练习,能比较熟练地应用计算机知识解决实际问题。既注意培养学生分析问题的能力,也注重培养学生解决问题的能力,以适应新经济时代对人才的需要,满足就业要求。

4. 注重教材的立体化配套

大多数教材都将陆续配套教师用课件、习题及其解答提示,学生上机实验指导等辅助教学资源,有些教材还提供能用于网上下载的文件,以方便教学。

由于各地区各学校的培养目标、教学要求和办学特色均有所不同,所以对特色教学的理解也不尽一致,我们恳切希望大家在使用教材的过程中,及时地给我们提出批评和改进意见,以便我们做好教材的修订改版工作,使其日趋完善。

我们相信经过大家的共同努力,这套教材一定能成为特色鲜明、质量上乘的优秀教材,同时,我们也希望通过本套教材的编写出版,为“高等学校教学质量和教学改革工程”作出贡献。

清华大学出版社

前言

PREFACE

Java 语言是当前计算机应用较为广泛的面向对象的程序设计语言之一。该语言在网络编程方面有着广泛的应用。在当今的网络时代,为了更好地满足网络的发展和应用,越来越多的人需要学会和掌握网络编程语言。为此,我们总结多年来从事 Java 语言教学工作的经验编著了本书。

本书具有如下特点:作者多年从事 Java 语言的教学工作,熟悉学习对象的情况,力求做到突出重点、详析难点、解答疑点,使读者学习起来容易理解和掌握;本书语言通俗,由浅入深,简明实用,适于自学;书中通过大量的例题来解释相关的概念和方法,将明确概念和着重实用相结合,有助于读者对基本概念和方法的学习;本书每章备有大量的习题和上机题,可引导读者掌握本章内容并检查学习情况,有助于读者较快地掌握所学的知识。

本书共分 10 章。第 1、2 章介绍 Java 语言的特点和 Java 程序的实现,以及该语言的语法基础。第 3 章介绍 Java 语言面向对象的特征。第 4 章介绍 Java Applet 及其应用。第 5 章详细介绍了图形用户界面设计,这是 Java 语言的重要编程应用之一。第 6 章介绍了异常处理。第 7 章介绍了线程的创建、线程的状态,以及线程的重要特性。第 8 章介绍了有关 Java 语言的集合操作,这也是 Java 语言的主要应用。第 9 章介绍了 Java 语言的输入输出操作和文件操作。第 10 章介绍了 Java 语言的网络编程,这是该语言的重要应用。本书全面系统地介绍了 Java 语言的基本语法和重要应用,读者通过学习本书可以全面掌握 Java 语言和编程方法,并能使用该语言实现简单的网络编程。

学习任何一种计算机语言都离不开上机实践。读者应在边学习边上机的过程中,深入细致地掌握 Java 语言。本书备有上机练习题,读者应按照题目的要求,认真上机练习。通过上机调试程序,可以发现概念上的模糊和方法的错误,搞清概念和纠正错误对于深刻理解和熟练掌握 Java

语言十分重要。敬劝读者学习 Java 语言时，一定要做到边看书边上机，只有多做练习、多上机调试程序，才会较快地掌握这种语言。

由于作者水平有限，编写时间仓促，在该书中可能会出现一些错误，恳请读者提出指正和宝贵意见。谢谢看过本书的读者。

作 者

目 录

CONTENTS

第 1 章 Java 语言概述	1
1.1 Java 语言的历史	1
1.2 Java 语言的特点	2
1.3 Java 程序及其实现	4
1.3.1 Java Application 程序举例	5
1.3.2 Java Application 程序的结构特点	7
1.4 Java Application 程序的实现	8
1.4.1 编辑 Java 源程序	8
1.4.2 编译 Java 源程序	8
1.4.3 运行 Java 的字节码文件	9
1.5 Java Application 程序的不同界面输出	10
习题	13
问答题	13
选择题	14
判断题	14
分析程序的输出结果题	15
编程题	16
上机题	17
第 2 章 Java 语言语法基础	19
2.1 词法规则	19
2.1.1 标识符	19
2.1.2 关键字	20
2.1.3 分隔符	20
2.1.4 注释符	21
2.2 数据类型	21

2.2.1 基本数据类型	21
2.2.2 复合数据类型	22
2.2.3 类型转换	22
2.3 常量和变量	24
2.3.1 常量	24
2.3.2 变量	27
2.4 运算符和表达式	32
2.4.1 运算符	32
2.4.2 表达式	37
2.5 语句	42
2.5.1 表达式语句和空语句	42
2.5.2 块语句	42
2.5.3 选择语句	43
2.5.4 循环语句	46
2.5.5 转向语句	49
2.5.6 其他语句	53
2.6 数组和字符串	53
2.6.1 数组	53
2.6.2 字符数组和字符串	60
2.6.3 Java Application 命令行参数	62
习题	63
问答题	63
选择题	65
判断题	67
计算表达式的值,上机验证其结果题	69
分析程序的输出结果题	70
编程题	74
上机题	75
第3章 Java 语言面向对象的特征	77
3.1 面向对象的基本概念	77
3.1.1 对象和类	77
3.1.2 封装性	79
3.1.3 继承性	79
3.1.4 多态性	80
3.2 类的定义	80
3.2.1 类的定义格式	80

3.2.2 变量	81
3.2.3 方法	84
3.2.4 实例	90
3.2.5 静态变量和静态方法	92
3.2.6 抽象类和抽象方法	95
3.2.7 最终类、最终变量和最终方法	97
3.3 对象	99
3.3.1 对象的定义	99
3.3.2 对象的赋值	100
3.3.3 对象的使用	103
3.4 类的继承和多态	107
3.4.1 父类和子类	107
3.4.2 成员的继承和覆盖	113
3.5 接口和包	119
3.5.1 接口	119
3.5.2 包	124
3.6 常用的工具类简介	126
3.6.1 Java 语言的类库	126
3.6.2 Java 语言基础类库	127
习题	138
问答题	138
选择题	139
判断题	141
填空题	142
分析程序的输出结果题	143
编程题	150
上机题	151
第 4 章 Java Applet 及其应用	153
4.1 Applet 概述	153
4.1.1 一个 Applet 的例子	153
4.1.2 HTML 文件	154
4.1.3 Applet 的特点和工作原理	155
4.2 Applet 与 Applet 类	158
4.2.1 Applet 与 Applet 类的关系	158
4.2.2 Applet 类的主要方法	159

4.3	Applet 程序和 HTML 文件	161
4.3.1	HTML 文件	161
4.3.2	向 Applet 传递参数	162
4.4	Applet 的应用	164
4.4.1	绘制图形	164
4.4.2	演示图像	179
4.4.3	播放声音	182
习题		184
问答题		184
选择题		185
判断题		186
分析程序的输出结果题		187
编程题		189
上机题		189
第5章	图形用户界面设计	191
5.1	图形用户界面概述	191
5.1.1	图形用户界面的构成	191
5.1.2	图形用户界面的实现	192
5.1.3	界面元素的类结构	195
5.2	事件和事件处理	196
5.2.1	事件处理方法	196
5.2.2	事件类和监听者接口	199
5.3	基本控制组件的使用	206
5.3.1	按钮和标签	207
5.3.2	文本框和文本区	209
5.3.3	单复选框和列表	214
5.3.4	下拉列表和滚动条	219
5.3.5	画布	224
5.4	布局设计	226
5.4.1	顺序布局	227
5.4.2	边界布局	229
5.4.3	卡片布局	230
5.4.4	网格布局	233
5.4.5	复杂的网格布局	235
5.5	常用容器组件的使用	239
5.5.1	容器概述	239
5.5.2	容器 Panel	240

5.5.3 窗口和菜单	243
5.5.4 对话框	255
习题	261
问答题	261
选择题	261
判断题	263
分析程序的输出结果题	264
编程题	270
上机题	270
第 6 章 异常和异常处理	271
6.1 异常和异常类	271
6.1.1 异常	271
6.1.2 异常类	272
6.2 异常处理	274
6.2.1 程序运行时异常	275
6.2.2 捕获异常	276
6.2.3 抛出异常	279
6.3 创建异常	281
6.3.1 用户创建自己的异常	281
6.3.2 举例分析	283
习题	289
问答题	289
选择题	289
判断题	290
分析程序的输出结果题	290
简单回答题	294
编程题	295
上机题	295
第 7 章 线程	297
7.1 概念	297
7.2 线程的创建	298
7.2.1 继承 Thread 类	298
7.2.2 实现 Runnable 接口	300
7.2.3 两种方式的比较	302
7.3 线程的状态	302
7.3.1 概述	302

7.3.2	启动	303
7.3.3	终止	303
7.3.4	暂停	304
7.3.5	状态判断	306
7.4	线程的同步	307
7.4.1	非同步实例	307
7.4.2	关键词 synchronized 和锁	309
7.4.3	等待 wait()和唤醒 notify()	311
7.4.4	完整的同步实例	312
7.5	线程的其他特性	314
7.5.1	主线程	314
7.5.2	守护线程	315
7.5.3	线程组	317
7.5.4	线程优先级	319
7.5.5	死锁	319
7.5.6	线程池	321
7.5.7	不再使用的方法	323
习题		324
问答题		324
选择题		325
判断题		325
分析程序的输出结果题		325
上机题		327
第 8 章	Java 语言的集合操作	329
8.1	概念	329
8.2	数组 (Arrays) 类	329
8.3	基本的集合接口	331
8.4	集合 (Set)	332
8.5	列表 (List)	336
8.6	映射 (Map)	338
8.7	Collections 类	342
8.7.1	排序	343
8.7.2	同步	346
8.7.3	只读	346
8.7.4	特殊集合	347
8.8	枚举和迭代	348
8.9	历史集合类	350

8.10 使用原则	351
习题	352
问答题	352
选择题	352
判断题	352
分析程序的输出结果题	353
上机题	355
第9章 Java 输入/输出	357
9.1 流的概念	357
9.2 输入/输出类	358
9.2.1 基本类	358
9.2.2 类别和功能	359
9.3 标准输入/输出	361
9.4 文件读/写	364
9.4.1 字节流方式	364
9.4.2 字符流方式	367
9.4.3 文件缓冲类	369
9.5 流的包装	373
9.5.1 概念	373
9.5.2 连接多个输入流	375
9.5.3 基本数据流	377
9.5.4 管道操作	380
9.6 文件类	383
9.7 随机访问文件	389
习题	391
问答题	391
选择题	392
判断题	392
分析程序的输出结果题	393
上机题	394
第10章 Java 网络编程	395
10.1 概述	395
10.2 URL 访问	397
10.3 InetAddress 类	400
10.4 NetworkInterface 类	401
10.5 Socket 应用	403

10.6 安全套接字应用.....	409
10.7 UDP 数据报	416
10.8 组播.....	419
10.9 对 IPv6 的支持	422
习题.....	425
问答题	425
选择题	425
判断题	426
分析程序的输出结果题	426
上机题	430
参考文献.....	431

第 1 章

CHAPTER

Java 语言概述

Java 语言是由美国 Sun Microsystems 公司开发的一种面向对象的程序设计语言。Java 是从 C++ 程序语言发展而来的,但比 C++ 语言简单。该语言具有简单、高效、健壮、安全、与平台无关、可移植性好和多线程等特点。它是当前网络编程中的首选语言。

1.1 Java 语言的历史

Java 语言是在 C++ 语言已广泛使用,Internet(因特网)发展迅速的历史背景下出现的,因此,Java 语言源于 C++ 语言,又与 Internet 同步发展。

1991 年,美国 Sun Microsystems 公司成立了一个开发面向家电(电视机、烤面包箱等)的编程软件小组,项目名称为“Green”,软件名称为“Oak”(一种橡树名)。当时 C++ 语言已经比较成熟,自然成为他们开发软件的首选语言。后来,发现 C++ 语言比较繁琐,并且安全性较差,不能满足要求,于是决定开发一种新的语言,当时取名为“Oak”。

Oak 语言刚出世后几经磨难,是 WWW 挽救了它。Sun 公司的一位创始人认为 Oak 语言的安全性和分布式系统非常适于 Internet,于是采用了在 Internet 上免费使用的策略。后来经过继续发展和完善,1995 年,Oak 语言改名为 Java 语言。

Java 语言发布后,发展顺利。Sun 公司很快公布了完整的 Java 技术规范,在 WWW 厂商支持下,用 Java 编写的 Applet 小程序在浏览器内部嵌入也获得了成功。不久,又推出一套全新的基于 Java 语言的开放式多媒体格式和应用程序接口(application program interface,API);1995 年底,又推出开放式、与平台无关的对象 Java 脚本语言(Java Script),从而为 Java 语言的进一步推广使用开辟了美好前景。

由于 Java 语言具有较好的安全性和可移植性及与平台无关等特性,因此,它将为 Internet 开辟一个新时代;而 Internet 的发展,又促进了 Java 语言的开发利用。

1.2 Java 语言的特点

Java 语言是一种卓越的程序设计语言。它具有下述特点。

1. 面向对象

Java 语言是一种面向对象的程序设计语言。面向对象是当前软件开发的先进技术和重要方法。面向对象的方法是基于信息隐藏和数据抽象类型的概念,利用类和对象的机制将数据和方法封装在一起,通过统一的接口与外界交互;通过类的继承机制实现代码重用。Java 语言只支持单继承,但它却支持多接口。Java 语言还支持方法重载和动态调用。总之,Java 语言支持面向对象方法中的 3 个基本特性:封装性、继承性和多态性。面向对象方法反映了客观世界中现实的实体在程序中的独立性和继承性。这种方法有利于提高程序的可维护性和可重用性,还有利于提高软件开发效率和程序的可管理性。虽然,C++ 语言是面向对象的,但是它为了兼容 C 语言,不得不保留一些面向过程的成分;而 Java 语言去掉了 C++ 语言中非面向对象的成分,因此,它是一个完全的面向对象的程序设计语言。

2. 简单性

与面向对象程序设计的 C++ 语言相比较,Java 语言更加简单、高效,生成的可执行目标代码也短小。Java 语言中取消了 C++ 语言中很少使用的、难以理解的、容易混淆的和复杂麻烦的特性和内容。例如,

(1) C++ 语言中,结构、联合和类的概念重合使用,Java 语言中只保留了类的概念。

(2) Java 语言中取消了 goto 语言,增强结构化,Java 语言支持带标号的 break 语句和 continue 语句。

(3) 运算符重载是 C++ 语言的特点之一,由于它实际使用的少,作用也不大,Java 语言不支持运算符重载。

(4) C++ 语言支持单继承和多继承;Java 语言只支持单继承,不支持多继承,但它支持多接口。因为多继承给编程者带来了复杂性,在实际使用中发现多继承难以使用,容易混淆,又浪费资源。因此,取消多继承,使得 Java 语言更简单。

(5) 指针是 C 语言的特点,在 C++ 语言中指针比 C 语言使用少了,因为指针使用困难,容易混淆,因此,Java 语言中取消了指针。

此外,Java 语言还取消了 C++ 语言中的预处理功能和内存空间的动态申请和释放功能,增添了内存空间自动回收功能。这些措施都使 Java 语言更简单、更方便。

总之,Java 是移除累赘和难以理解的部分之后的 C++ 语言,它是更纯粹的面向对象的程序设计语言。