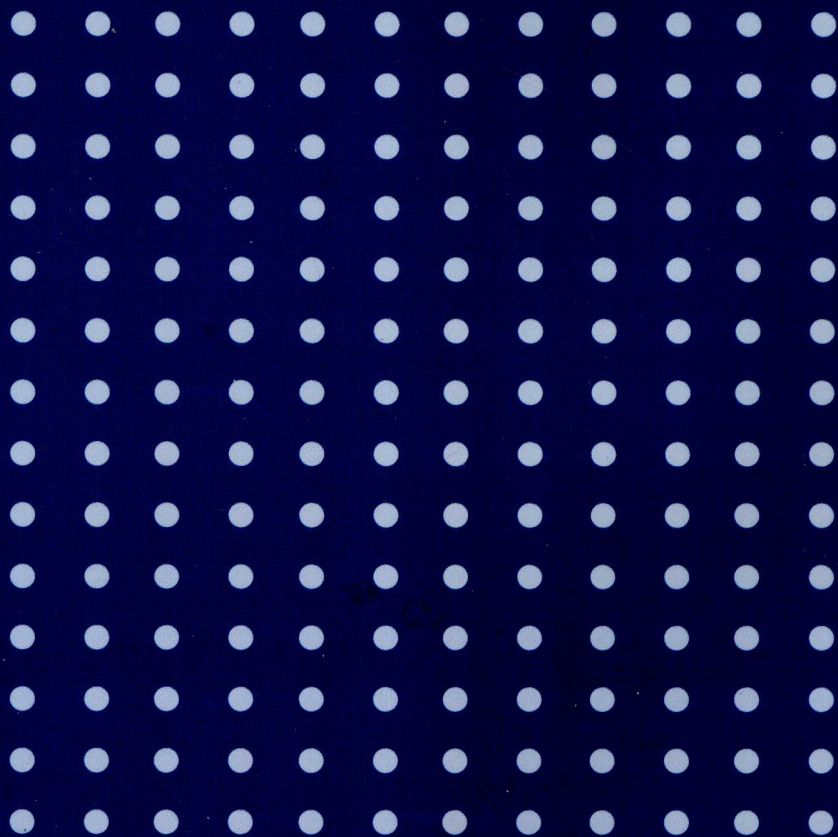


重点大学计算机专业系列教材

C# 程序设计教程 (第2版)

李春葆 谭成予 曾平 曾慧 编著



清华大学出版社



重点大学计算机专业系列教材

C# 程序设计教程 (第2版)

李春葆 谭成予 曾平 曾慧 编著

福建师范大学
图书馆
藏书印

1042938



T1042938

清华大学出版社

北京

内 容 简 介

本书以 Visual C# 2005 版本为平台介绍 C# 程序设计方法, 主要内容包括 C# 可视化编程过程、C# 语言基础、窗体和控件设计、面向对象程序设计、图形设计、文件操作、C# 高级特性、ADO.NET 数据库访问技术、XML 应用和 Web 应用程序设计等。

本书内容翔实, 循序渐进地介绍各个知识点, 既可作为各类高等院校计算机及相关专业“C# 程序设计”课程的教学用书, 也可作为计算机应用人员和计算机编程爱好者的参考书。

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

C# 程序设计教程/李春葆等编著.--2 版.--北京: 清华大学出版社, 2013

重点大学计算机专业系列教材

ISBN 978-7-302-29603-4

I. ①C… II. ①李… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 179830 号

责任编辑: 魏江江 张为民

封面设计: 常雪影

责任校对: 时翠兰

责任印制: 沈 露

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

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: 北京嘉实印刷有限公司

经 销: 全国新华书店

开 本: 185mm×260mm

印 张: 30

字 数: 730 千字

版 次: 2010 年 1 月第 1 版

2013 年 1 月第 2 版

印 次: 2013 年 1 月第 1 次印刷

印 数: 1~3000

定 价: 49.00 元

产品编号: 044698-01

出版说明

随着国家信息化步伐的加快和高等教育规模的扩大,社会对计算机专业人才的需求不仅体现在数量的增加上,而且体现在质量要求的提高上,培养具有研究和实践能力的高层次的计算机专业人才已成为许多重点大学计算机专业教育的主要目标。目前,我国共有 16 个国家重点学科、20 个博士点一级学科、28 个博士点二级学科集中在教育部部属重点大学,这些高校在计算机教学和科研方面具有一定优势,并且大多以国际著名大学计算机教育为参照系,具有系统完善的教学课程体系、教学实验体系、教学质量保证体系和人才培养评估体系等综合体系,形成了培养一流人才的教学和科研环境。

重点大学计算机学科的教学与科研氛围是培养一流计算机人才的基础,其中专业教材的使用和建设则是这种氛围的重要组成部分,一批具有学科方向特色优势的计算机专业教材作为各重点大学的重点建设项目成果得到肯定。为了展示和发扬各重点大学在计算机专业教育上的优势,特别是专业教材建设上的优势,同时配合各重点大学的计算机学科建设和专业课程教学需要,在教育部相关教学指导委员会专家的建议和各重点大学的大力支持下,清华大学出版社规划并出版本系列教材。本系列教材的建设旨在“汇聚学科精英、引领学科建设、培育专业英才”,同时以教材示范各重点大学的优秀教学理念、教学方法、教学手段和教学内容等。

本系列教材在规划过程中体现了如下一些基本组织原则和特点。

(1) 面向学科发展的前沿,适应当前社会对计算机专业高级人才的培养需求。教材内容以基本理论为基础,反映基本理论和原理的综合应用,重视实践和应用环节。

(2) 反映教学需要,促进教学发展。教材要能适应多样化的教学需要,正确把握教学内容和课程体系的改革方向。在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

(3) 实施精品战略,突出重点,保证质量。规划教材建设的重点依然是专业基础课和专业主干课;特别注意选择并安排了一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现重点大学

计算机专业教学内容和课程体系改革成果的教材。

(4) 主张一纲多本,合理配套。专业基础课和专业主干课教材要配套,同一门课程可以有多本具有不同内容特点的教材。处理好教材统一性与多样化的关系;基本教材与辅助教材以及教学参考书的关系;文字教材与软件教材的关系,实现教材系列资源配套。

(5) 依靠专家,择优落实。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主编。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

教材编委会

FOREWORD

前言

C#是微软公司结合 C/C++ 和 Java 等特点而设计的一种新的程序设计语言,它基于 .NET Framework 通用平台,C# 程序开发人员可以直接使用 .NET Framework 中完整且丰富的类库设计出跨平台的软件系统。C# 具有简单易学、使用方便、能采用可视化设计方法开发复杂软件系统等特点。

本书从介绍 C# 的基本概念出发,由简单到复杂,循序渐进地介绍 C# 面向对象的程序设计方法。在内容介绍上力求翔实和全面,并且细致解析每个知识点和其他知识点的联系。书中列举了大量的应用示例,读者通过上机模仿可以大大提高使用 C# 开发控制台应用程序、Windows 窗体应用程序和 Web 应用程序的能力。

全书分为 16 章,第 1 章为 C# 概述;第 2 章为简单的 C# 程序设计;第 3 章为 C# 程序设计基础;第 4 章为 C# 控制语句;第 5 章为数组和集合;第 6 章为面向对象程序设计;第 7 章为继承和接口设计;第 8 章为 Windows 窗体应用程序设计;第 9 章为用户界面设计;第 10 章为图形设计;第 11 章为文件操作;第 12 章为错误调试和异常处理;第 13 章为 C# 高级特性;第 14 章为 ADO.NET 数据库访问技术;第 15 章为 XML 及其应用;第 16 章为 Web 应用程序设计。各章提供了一定数量的练习题,供读者选用。附录给出了练习题中单项选择题的参考答案。

本书的读者对象需具备简单的编程经验,并对 Windows 的基本操作有所了解。本书的源程序代码、练习题中编程题的代码和上机实验题参考答案可以从清华大学出版社网站(www.tup.com.cn)免费下载。

本书可以作为各类高等院校的计算机专业和非计算机专业 C# 编程的教材,尤其适合作为高职高专院校数据库应用方向学生的 C# 程序设计教材,也可供具有 Windows 初步知识的计算机编程爱好者参阅。

书中 C# 编程环境采用 Visual C# 2005 中文版,在教学中也可以使用 Visual C# 2008 或 Visual C# 2010。

由于编者水平所限,书中若有不当之处,敬请读者指正。

编 者
2012 年 6 月

告 白

CONTENTS

目录

第 1 章 C# 概述	1
1.1 什么是 C#	1
1.2 .NET Framework 概述	2
1.2.1 什么是 .NET 平台	2
1.2.2 什么是 .NET Framework	2
1.2.3 .NET 程序编译过程	5
1.2.4 C# 与 .NET Framework	6
1.3 配置 Visual C# 开发环境和启动退出操作	7
1.3.1 配置 Visual C# 开发环境	7
1.3.2 Visual C# 的启动	8
1.3.3 Visual C# 的退出	11
1.4 Visual C# 的集成开发环境	11
1.4.1 Visual C# 菜单栏	11
1.4.2 Visual C# 工具栏	13
1.4.3 Visual C# 工具箱	14
1.4.4 解决方案资源管理器	15
1.4.5 属性窗口	17
1.4.6 窗体设计器	18
1.4.7 代码编辑窗口	19
1.4.8 其他窗口	20
1.4.9 帮助系统	21
1.5 编辑器设置	21
练习题 1	23
第 2 章 简单的 C# 程序设计	25
2.1 创建第一个 Windows 控制台应用程序	25
2.1.1 控制台应用程序示例	25

2.1.2 控制台应用程序项目的组成	27
2.1.3 控制台应用程序中的基本元素	30
2.2 创建第一个 Windows 应用程序	31
2.2.1 Windows 应用程序示例	31
2.2.2 Windows 应用程序项目的组成	34
2.2.3 Windows 应用程序中的基本元素	35
练习题 2	35
第3章 C# 程序设计基础	37
3.1 标识符	37
3.2 C# 中数据类型	38
3.2.1 值类型	39
3.2.2 引用类型	41
3.3 C# 中的变量和常量	43
3.3.1 变量	43
3.3.2 值类型变量和引用类型变量的区别	44
3.3.3 常量	47
3.4 类型转换	48
3.4.1 隐式转换	48
3.4.2 显式转换	49
3.4.3 装箱和拆箱	50
3.5 结构类型和枚举类型	51
3.5.1 结构类型	51
3.5.2 枚举类型	53
3.6 C# 运算符和表达式	55
3.6.1 算术运算符	55
3.6.2 字符串运算符	56
3.6.3 赋值运算符	56
3.6.4 关系运算符	56
3.6.5 逻辑运算符	57
3.6.6 位运算符	57
3.6.7 条件运算符	58
3.6.8 其他运算符	59
3.6.9 运算符的优先级	60
3.7 C# 中常用类和结构	60
3.7.1 String 类	61
3.7.2 Math 类	64
3.7.3 Convert 类	64
3.7.4 DateTime 结构	65

练习题 3	67
第 4 章 C# 控制语句	70
4.1 选择控制语句	70
4.1.1 if 语句	70
4.1.2 if...else 语句	71
4.1.3 if...else if 语句	72
4.1.4 switch 语句	73
4.2 循环控制语句	75
4.2.1 while 语句	75
4.2.2 do...while 语句	76
4.2.3 for 语句	76
4.3 跳转语句	77
4.3.1 break 语句	77
4.3.2 continue 语句	78
4.3.3 goto 语句	79
练习题 4	80
第 5 章 数组和集合	85
5.1 一维数组	85
5.1.1 一维数组的定义	85
5.1.2 一维数组的动态初始化	86
5.1.3 一维数组的静态初始化	87
5.1.4 访问一维数组中的元素	87
5.1.5 一维数组的越界	88
5.2 二维数组	88
5.2.1 二维数组的定义	89
5.2.2 二维数组的动态初始化	89
5.2.3 二维数组的静态初始化	89
5.2.4 访问二维数组中的元素	90
5.3 交错数组	91
5.3.1 交错数组的定义和初始化	91
5.3.2 访问交错数组中的元素	91
5.4 Array 类	92
5.4.1 Array 类的属性和方法	92
5.4.2 Array 类中方法的使用	93
5.5 ArrayList 类	94
5.5.1 ArrayList 类的对象的定义	94
5.5.2 ArrayList 类的属性	95

5.5.3 ArrayList 类的方法	95
5.6 List<T>类	96
5.6.1 List<T>类的对象的定义	96
5.6.2 List<T>类的属性	96
5.6.3 List<T>类的方法	97
练习题 5	99
第 6 章 面向对象程序设计	102
6.1 面向对象程序设计概述	102
6.1.1 面向对象的基本概念	102
6.1.2 面向对象的优点	103
6.2 类	104
6.2.1 类的声明	104
6.2.2 类的成员	105
6.2.3 分部类	105
6.2.4 类和结构类型的差异	106
6.3 对象	106
6.3.1 定义类的对象	106
6.3.2 访问对象的字段	107
6.3.3 调用对象的方法	107
6.4 构造函数和析构函数	109
6.4.1 构造函数	109
6.4.2 析构函数	111
6.5 静态成员	111
6.5.1 静态字段	112
6.5.2 静态方法	112
6.6 属性	115
6.6.1 什么是属性	115
6.6.2 属性声明及其使用	115
6.7 方法	116
6.7.1 什么是方法	116
6.7.2 方法的定义	116
6.7.3 方法的返回值	116
6.7.4 方法的参数	117
6.7.5 方法的重载	118
6.8 嵌套类	119
6.8.1 嵌套类的声明	119
6.8.2 this 关键字	119
6.8.3 嵌套类和包含类的关系	121

6.9	索引器	122
6.9.1	什么是索引器	122
6.9.2	声明索引器	122
6.9.3	使用其他值进行索引	123
6.10	委托	124
6.10.1	什么是委托	124
6.10.2	定义和使用委托	125
6.10.3	委托对象封装多个方法	127
6.10.4	使委托与匿名方法关联	128
6.11	事件	129
6.11.1	事件处理机制	129
6.11.2	事件的创建和使用	130
6.12	运算符重载	134
6.12.1	运算符重载概述	134
6.12.2	一元运算符重载	134
6.12.3	二元运算符重载	135
6.13	类对象的转换	136
6.13.1	关键字 is	136
6.13.2	关键字 as	137
6.14	命名空间	138
6.14.1	命名空间概述	138
6.14.2	使用命名空间	138
	练习题 6	141
第 7 章	继承和接口设计	147
7.1	继承	147
7.1.1	什么是继承	147
7.1.2	派生类的声明	148
7.1.3	基类成员的可访问性	149
7.1.4	按次序调用构造函数和析构函数	149
7.1.5	使用 sealed 修饰符来禁止继承	151
7.2	多态性	152
7.2.1	隐藏基类方法	152
7.2.2	重写	152
7.3	抽象类	155
7.3.1	抽象类的特性	155
7.3.2	抽象方法	156
7.3.3	抽象属性	157
7.4	接口	158

7.4.1	接口的特性	158
7.4.2	接口的定义	158
7.4.3	接口的成员	159
7.4.4	接口的实现	160
7.4.5	接口映射	163
7.4.6	接口实现的继承	164
7.4.7	重新实现接口	166
7.5	接口在集合排序中的应用	168
7.5.1	ArrayList 类的排序方法	168
7.5.2	Comparable 接口	168
7.5.3	Comparer 接口	170
	练习题 7	174
第 8 章	Windows 窗体应用程序设计	180
8.1	窗体设计	180
8.1.1	创建 Windows 窗体应用程序的过程	180
8.1.2	窗体类型	181
8.1.3	窗体的常用属性	181
8.1.4	窗体的常用事件	184
8.1.5	窗体的常用方法	184
8.1.6	多个窗体之间的调用	185
8.1.7	窗体上各事件的引发顺序	190
8.1.8	焦点与 Tab 键次序	190
8.2	常用控件的设计	191
8.2.1	控件概述	191
8.2.2	富文本框控件	192
8.2.3	分组框控件	194
8.2.4	面板控件	194
8.2.5	复选框控件	195
8.2.6	单选按钮控件	197
8.2.7	图片框控件	198
8.2.8	组合框控件	199
8.2.9	列表框控件	201
8.2.10	带复选框的列表框控件	204
8.2.11	定时器控件	205
8.2.12	滚动条控件	206
8.2.13	月历控件	208
8.2.14	日期/时间控件	208
8.2.15	超链接标签控件	210

8.3	多文档窗体	211
8.4	窗体设计的事件机制	214
8.4.1	什么是事件处理程序	214
8.4.2	在 Windows 窗体中创建事件处理程序	214
8.4.3	在执行时为 Windows 窗体创建事件处理程序	215
8.4.4	将多个事件连接到 Windows 窗体中的单个事件处理程序	216
	练习题 8	218
第 9 章	用户界面设计	221
9.1	菜单设计	221
9.1.1	菜单的基本结构	221
9.1.2	创建下拉式菜单	223
9.1.3	弹出式菜单设计	228
9.1.4	为菜单项添加访问热键和快捷键	230
9.2	通用对话框	231
9.2.1	打开文件对话框	231
9.2.2	保存文件对话框	233
9.2.3	颜色对话框	235
9.2.4	字体对话框	235
9.3	图像列表框控件	236
9.3.1	建立 ImageList 控件	236
9.3.2	ImageList 控件的属性	237
9.3.3	Images 集合的属性和方法	238
9.4	树形视图控件	238
9.4.1	TreeView 控件概述	239
9.4.2	建立 TreeView 控件	239
9.4.3	TreeView 控件的属性	240
9.4.4	TreeView 控件的事件和方法	240
9.4.5	Nodes 集合和 TreeNode 对象	241
9.4.6	执行时 TreeView 控件的基本操作	242
9.5	列表视图控件	244
9.5.1	ListView 控件概述	244
9.5.2	建立 ListView 控件	245
9.5.3	ListView 控件的属性	246
9.5.4	ListView 控件的事件和方法	246
9.5.5	Items 集合和 ListViewItem 对象	247
9.5.6	Columns 集合和 ColumnHeader 对象	248
9.5.7	执行时 ListView 控件的基本操作	249
9.6	工具栏控件	251

9.6.1	建立 ToolStrip 控件	251
9.6.2	ToolStrip 控件的属性	252
9.6.3	Items 集合和 ToolStripButton 对象	252
9.7	状态栏控件	253
9.7.1	建立 StatusStrip 控件	253
9.7.2	StatusStrip 控件的属性	254
	练习题 9	254
第 10 章	图形设计	257
10.1	绘图概述	257
10.1.1	绘图的基本知识	257
10.1.2	什么是 GDI+	258
10.1.3	Graphics 类	259
10.2	绘图的基本步骤	259
10.3	绘制图形	261
10.3.1	绘制直线	261
10.3.2	绘制矩形	262
10.3.3	绘制多边形	263
10.3.4	绘制圆和椭圆	263
10.3.5	绘制弧线	264
10.3.6	绘制饼形	265
10.3.7	绘制非闭合曲线	265
10.3.8	绘制闭合曲线	266
10.3.9	绘制贝济埃曲线	267
10.4	创建画图工具	268
10.4.1	创建画笔	268
10.4.2	创建笔刷	269
10.4.3	创建字体	271
10.5	绘制文本	271
	练习题 10	273
第 11 章	文件操作	275
11.1	文件和 System.IO 模型概述	275
11.1.1	文件类型	275
11.1.2	文件的属性	276
11.1.3	文件访问方式	277
11.2	System.IO 模型	277
11.2.1	什么是 System.IO 模型	277
11.2.2	文件编码	278

11.2.3 C# 的文件流	278
11.3 文件夹和文件操作	278
11.3.1 文件夹操作	278
11.3.2 文件操作	279
11.4 FileStream 类	281
11.5 文本文件的操作	282
11.5.1 StreamReader 类	282
11.5.2 StreamWriter 类	284
11.6 二进制文件操作	286
11.6.1 BinaryWriter 类	286
11.6.2 BinaryReader 类	287
11.6.3 二进制文件的随机查找	289
11.7 序列化和反序列化	290
11.7.1 序列化	290
11.7.2 反序列化	291
练习题 11	293
第 12 章 错误调试和异常处理	295
12.1 错误分类	295
12.1.1 语法错误	295
12.1.2 执行错误	296
12.1.3 逻辑错误	297
12.2 程序调试	297
12.2.1 调试工具	298
12.2.2 如何开始调试	298
12.2.3 设置断点	298
12.2.4 定位错误	299
12.2.5 调试过程	299
12.2.6 显示调试信息	299
12.2.7 单步跟踪	302
12.3 异常处理	302
12.3.1 异常概述	302
12.3.2 异常处理语句	303
12.3.3 常用的异常类	306
练习题 12	306
第 13 章 C# 高级特性	309
13.1 泛型	309
13.1.1 什么是泛型	309

13.1.2	泛型的声明和使用	310
13.2	反射	312
13.2.1	反射概述	312
13.2.2	反射中常用的类	312
13.2.3	反射的应用示例	315
13.3	多线程程序设计	318
13.3.1	多线程的概述	318
13.3.2	线程命名空间	319
13.3.3	Thread 类及其应用	319
13.3.4	线程优先级和线程调度	323
13.3.5	线程互斥	327
13.3.6	线程同步	331
	练习题 13	333
第 14 章	ADO.NET 数据库访问技术	335
14.1	数据库概述	335
14.1.1	关系数据库的基本结构	335
14.1.2	结构化查询语言	337
14.2	ADO.NET 模型	341
14.2.1	ADO.NET 简介	341
14.2.2	ADO.NET 体系结构	342
14.2.3	ADO.NET 数据库的访问流程	344
14.3	ADO.NET 的数据访问对象	344
14.3.1	OleDbConnection 对象	344
14.3.2	OleDbCommand 对象	347
14.3.3	OleDbDataReader 对象	353
14.3.4	OleDbDataAdapter 对象	355
14.4	DataSet 对象	360
14.4.1	DataSet 对象概述	360
14.4.2	Dataset 对象的属性和方法	361
14.4.3	Tables 集合和 DataTable 对象	362
14.4.4	Columns 集合和 DataColumn 对象	365
14.4.5	Rows 集合和 DataRow 对象	366
14.4.6	Relations 集合和 DataRelation 对象	368
14.5	数据绑定	369
14.5.1	数据绑定概述	369
14.5.2	BindingNavigator 控件	374
14.6	DataView 对象	377
14.6.1	DataView 对象概述	377