

21世纪高等学校计算机教育实用规划教材

C# 程序设计简明教程

江红 余青松 编著



清华大学出版社

014035573

TP312C
2293

21世纪高等学校计算机教育实用规划教材

C#程序设计简明教程

江红 余青松 编著



清华大学出版社
北京



北航

C1722738

TP312C
2293

内 容 简 介

本书集教材、上机指导、练习册于一体,基于 Visual Studio 2012/.NET Framework 4.5 开发和运行环境,阐述 C# 5.0 语言的基础知识,以及使用 C# 5.0 语言的实际开发应用实例,具体内容包括 C# 语言概述,C# 语言基础,程序流程控制,字符串、日期和数组,类和对象,继承和多态,结构、枚举、委托和事件,C# 语言的高级特性(泛型、集合、特性、语言集成查询和多线程处理),文件和 I/O 流,数据库访问,Windows 窗体应用程序,ASP.NET Web 应用程序以及综合应用案例。

本书作者结合多年的程序设计、开发及授课经验,由浅入深、循序渐进地介绍 C# 5.0 程序设计语言,让读者能够较为系统、全面地掌握程序设计的理论和应用。

本书可作为高等学校各专业的计算机程序设计教程,也可作为广大程序设计开发者、爱好者的自学参考书。

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# 程序设计简明教程/江红,余青松编著.--北京:清华大学出版社,2014

21 世纪高等学校计算机教育实用规划教材

ISBN 978-7-302-33139-1

I. ①C… II. ①江… ②余… III. ①C 语言—程序设计—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2013)第 159639 号

责任编辑:魏江江 王冰飞

封面设计:常雪影

责任校对:时翠兰

责任印制:李红英

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 刷 者:北京富博印刷有限公司

装 订 者:北京市密云县京文制本装订厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:21.5

字 数:519 千字

版 次:2014 年 4 月第 1 版

印 次:2014 年 4 月第 1 次印刷

印 数:1~2000

定 价:36.00 元

产品编号:053427-01

出版说明

随着我国高等教育规模的扩大以及产业结构调整的进一步完善,社会对高层次应用型人才的需求将更加迫切。各地高校紧密结合地方经济建设发展需要,科学运用市场调节机制,合理调整和配置教育资源,在改革和改造传统学科专业的基础上,加强工程型和应用型学科专业建设,积极设置主要面向地方支柱产业、高新技术产业、服务业的工程型和应用型学科专业,积极为地方经济建设输送各类应用型人才。各高校加大了使用信息科学等现代科学技术提升、改造传统学科专业的力度,从而实现传统学科专业向工程型和应用型学科专业的发展与转变。在发挥传统学科专业师资力量强、办学经验丰富、教学资源充裕等优势的同时,不断更新教学内容、改革课程体系,使工程型和应用型学科专业教育与经济建设相适应。计算机课程教学在从传统学科向工程型和应用型学科转变中起着至关重要的作用,工程型和应用型学科专业中的计算机课程设置、内容体系和教学手段及方法等也具有不同于传统学科的鲜明特点。

为了配合高校工程型和应用型学科专业的建设和发展,急需出版一批内容新、体系新、方法新、手段新的高水平计算机课程教材。目前,工程型和应用型学科专业计算机课程教材的建设工作仍滞后于教学改革的实践,如现有的计算机教材中有不少内容陈旧(依然用传统专业计算机教材代替工程型和应用型学科专业教材),重理论、轻实践,不能满足新的教学计划、课程设置的需要;一些课程的教材可供选择的品种太少;一些基础课的教材虽然品种较多,但低水平重复严重;有些教材内容庞杂,书越编越厚;专业课教材、教学辅助教材及教学参考书短缺,等等,都不利于学生能力的提高和素质的培养。为此,在教育部相关教学指导委员会专家的指导和建议下,清华大学出版社组织出版本系列教材,以满足工程型和应用型学科专业计算机课程教学的需要。本系列教材在规划过程中体现了如下一些基本原则和特点。

(1) 面向工程型与应用型学科专业,强调计算机在各专业中的应用。教材内容坚持基本理论适度,反映基本理论和原理的综合应用,强调实践和应用环节。

(2) 反映教学需要,促进教学发展。教材规划以新的工程型和应用型专业目录为依据。教材要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。规划教材建设仍然把重点放在公共基础课和专业基础课的教材建设上;特别注意选择并安排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工程型和应用型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。基础课和专业基础课教材要配套,同一门课程可以有多种具有不同内容特点的教材。处理好教材统一性与多样化,基本教材与辅助教材,教学参考书,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,择优选用。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和建设力度,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。

21 世纪高等学校计算机教育实用规划教材编委会

联系人:魏江江 weijj@tup.tsinghua.edu.cn

前 言

程序设计是大专院校计算机、电子信息、工商管理等相关专业的必修课程,C#作为一门新的程序设计语言,集中了C、C++、Java等语言的优点,是一门现代的、优越的、具有广阔发展前景的程序设计语言。

本书集教材、上机指导、练习册于一体,主要基于Visual Studio 2012/.NET Framework 4.5开发和运行环境,阐述C# 5.0语言的基础知识,以及使用C# 5.0语言的实际开发应用实例,具体包括C#语言概述,C#语言基础,程序流程控制,字符串、日期和数组,类和对象,继承和多态,结构、枚举、委托和事件,C#语言高级特性(泛型、集合、特性、语言集成查询和多线程处理),文件和I/O流,数据库访问,Windows窗体应用程序,ASP.NET Web应用程序以及综合应用案例。

本书各章节所涉及的所有源程序代码和相关素材,以及供教师参考的教学电子文稿均可以通过清华大学出版社网站(www.tup.tsinghua.edu.cn)下载,也可以通过 hjiang@cc.ecnu.edu.cn 直接与作者联系。

本书由华东师范大学江红和余青松共同编写。由于时间和编者学识有限,书中不足之处在所难免,敬请诸位同行、专家和读者指正。

编 者

2013年12月

目 录

第 1 章 C# 语言	1
1.1 C# 语言概述	1
1.1.1 C# 语言简介	1
1.1.2 C# 语言的特点和开发应用范围	2
1.2 C# 语言的编译和运行环境	2
1.2.1 C# 语言与 .NET Framework	2
1.2.2 C# 语言的运行环境	3
1.2.3 C# 语言的开发环境	3
1.3 简单 C# 程序的创建、编译和运行	4
1.3.1 简单 C# 程序的创建	4
1.3.2 编译和运行结果	5
1.4 C# 程序的基本结构	5
1.4.1 程序结构	5
1.4.2 命名空间	6
1.4.3 类和对象	8
1.4.4 Main 方法	9
1.4.5 注释	10
1.5 控制台输入/输出	11
1.5.1 System.Console 类	12
1.5.2 复合格式	12
1.6 Visual Studio 快速入门	14
1.6.1 集成开发环境界面	14
1.6.2 创建解决方案和项目	14
1.6.3 设计器/编辑器	16
1.6.4 生成和调试工具	18
1.6.5 安装和部署工具	19
1.6.6 帮助系统	19
习题 1	19
实验 1	20

第 2 章 C# 语言基础	23
2.1 语句	23
2.2 表达式	24
2.2.1 表达式的组成	24
2.2.2 表达式的书写规则	24
2.3 运算符	24
2.4 变量	25
2.4.1 标识符	25
2.4.2 变量的命名规则	25
2.4.3 变量的声明和赋值	25
2.4.4 赋值运算	26
2.5 常量	27
2.5.1 文本常量	27
2.5.2 用户声明常量	27
2.6 数据类型	27
2.6.1 类型系统	28
2.6.2 值类型和引用类型	28
2.6.3 装箱和拆箱	29
2.7 数值数据类型和算术运算	30
2.7.1 整型	30
2.7.2 浮点类型	31
2.7.3 decimal 类型	33
2.7.4 算术运算符	33
2.7.5 Math 类和数学函数	35
2.7.6 Random 类和随机函数	36
2.7.7 位运算符	37
2.8 字符类型	38
2.8.1 字符数据类型	38
2.8.2 字符常量	38
2.8.3 System.Char 类成员	38
2.9 数据类型转换	39
2.9.1 隐式转换	39
2.9.2 显式转换	40
2.9.3 Convert 类提供的类型转换方法	40
2.9.4 checked 关键字	41
习题 2	42
实验 2	43

第 3 章 程序流程控制(一)	45
3.1 布尔数据类型	45
3.2 运算符	45
3.2.1 逻辑运算符	45
3.2.2 关系和类型测试运算符	46
3.2.3 条件运算符	47
3.2.4 其他运算符	48
3.2.5 运算符优先级	48
3.3 顺序结构	50
3.4 选择结构	50
3.4.1 if 语句	50
3.4.2 switch 语句	58
习题 3	60
实验 3	61
第 4 章 程序流程控制(二)	65
4.1 循环结构	65
4.1.1 for 循环	65
4.1.2 while 循环	67
4.1.3 do...while 循环	69
4.1.4 foreach 循环	70
4.1.5 循环的嵌套	71
4.2 跳转语句	72
4.2.1 goto 语句	72
4.2.2 break 语句	73
4.2.3 continue 语句	73
4.2.4 return 语句	74
4.3 异常处理	74
4.3.1 C# 异常处理机制	74
4.3.2 创建和引发异常	76
习题 4	78
实验 4	80
第 5 章 字符串、日期和数组	83
5.1 字符串处理	83
5.1.1 C# 字符串概述	83
5.1.2 字符串运算符	84
5.1.3 String 类和 StringBuilder 类	84

5.1.4	正则表达式	85
5.2	日期和时间处理	86
5.3	数组	88
5.3.1	数组概述	88
5.3.2	一维数组	89
5.3.3	多维数组	90
5.3.4	交错数组	92
5.3.5	作为对象的数组	92
5.3.6	数组的应用举例	94
习题 5		100
实验 5		102
第 6 章	类和对象	106
6.1	面向对象的基本概念	106
6.2	类和对象	107
6.2.1	类的声明	108
6.2.2	对象的创建和使用	108
6.2.3	访问修饰符	109
6.3	类的成员	110
6.3.1	数据成员	110
6.3.2	函数成员	112
6.3.3	静态成员和实例成员	125
6.4	嵌套类	126
6.5	分部类	126
习题 6		127
实验 6		129
第 7 章	继承和多态	131
7.1	继承和多态的基本概念	131
7.2	派生类	132
7.2.1	派生类的声明和使用	132
7.2.2	关键字 this 和 base	132
7.2.3	虚方法、重写方法和隐藏方法	134
7.3	抽象类和抽象方法	135
7.3.1	抽象类	135
7.3.2	抽象方法	136
7.4	密封类和密封方法	137
7.4.1	密封类	137
7.4.2	密封方法	138

7.5	接口	138
7.5.1	接口声明	138
7.5.2	分部接口	139
7.5.3	接口成员	139
7.5.4	接口实现	139
7.5.5	接口继承	141
习题 7		142
实验 7		143
第 8 章	结构、枚举、委托和事件	146
8.1	结构	146
8.1.1	结构概述	146
8.1.2	结构的声明	147
8.1.3	结构的调用	147
8.1.4	结构成员	148
8.1.5	嵌套结构和分部结构	148
8.2	枚举	149
8.2.1	枚举概述	149
8.2.2	枚举的声明	149
8.2.3	枚举的使用	150
8.3	委托	151
8.3.1	委托的声明	151
8.3.2	委托的实例化和调用	152
8.3.3	匿名方法委托	154
8.3.4	多播委托	155
8.4	事件	155
8.4.1	事件处理机制	155
8.4.2	事件的声明	157
8.4.3	事件的订阅和取消	158
8.4.4	静态事件和实例事件	158
8.4.5	.NET Framework 事件模型	158
8.4.6	应用举例：实现事件的步骤	159
习题 8		160
实验 8		161
第 9 章	C# 语言的高级特性	163
9.1	泛型	163
9.1.1	泛型的基本概念	163
9.1.2	泛型的定义和使用	163

9.2 集合	164
9.2.1 C#集合和数据结构概述	164
9.2.2 列表类集合类型	165
9.2.3 字典类集合类型	165
9.2.4 队列集合类型	165
9.2.5 堆栈集合类型	165
9.3 特性	165
9.3.1 特性的基本概念	165
9.3.2 特性的使用	166
9.3.3 预定义通用特性类	167
9.3.4 自定义特性类	168
9.3.5 使用反射访问特性	168
9.4 语言集成查询	169
9.4.1 相关语言要素	170
9.4.2 LINQ 基本操作	173
9.4.3 LINQ 查询操作	173
9.4.4 标准查询运算符	174
9.5 多线程处理概述	175
9.5.1 进程和线程	175
9.5.2 线程的优缺点	175
9.5.3 创建和启动新线程	176
习题 9	177
实验 9	178
第 10 章 文件和 I/O 流	181
10.1 文件和流操作概述	181
10.2 磁盘、目录和文件的基本操作	181
10.2.1 磁盘的基本操作	181
10.2.2 目录的基本操作	183
10.2.3 文件的基本操作	185
10.3 文本文件的写入和读取	187
10.3.1 文本文件的写入	187
10.3.2 文本文件的读取	188
10.4 二进制文件的写入和读取	189
10.4.1 二进制文件的写入	189
10.4.2 二进制文件的读取	190
10.5 随机文件访问	191
10.6 通用 I/O 流类	192
习题 10	193

实验 10	193
第 11 章 数据库访问	197
11.1 ADO.NET 概述	197
11.1.1 ADO.NET 的基本概念	197
11.1.2 ADO.NET 的结构	197
11.1.3 .NET Framework 数据提供程序	198
11.1.4 ADO.NET DataSet	199
11.2 使用 ADO.NET 连接和操作数据库	200
11.2.1 使用数据提供程序访问数据库的步骤	200
11.2.2 范例数据库 PNorthwind.mdf	202
11.2.3 查询数据库表数据	203
11.2.4 插入数据库表数据	205
11.2.5 更新数据库表数据	206
11.2.6 删除数据库表数据	206
11.2.7 使用存储过程访问数据库	207
11.3 使用 DataAdapter 和 DataSet 访问数据库	208
11.3.1 使用 DataAdapter 和 DataSet 访问数据库的步骤	208
11.3.2 查询数据库表数据	209
11.3.3 维护数据库表数据	210
习题 11	211
实验 11	212
第 12 章 Windows 窗体应用程序(一)	214
12.1 Windows 窗体应用程序	214
12.1.1 Windows 窗体应用程序概述	214
12.1.2 使用 Visual Studio 开发 Windows 窗体应用程序	216
12.1.3 窗体和控件概述	219
12.2 常用的 Windows 窗体控件	219
12.2.1 标签、文本框和命令按钮	219
12.2.2 单选按钮、复选框和分组	221
12.2.3 列表选择控件	223
12.2.4 图形存储和显示控件	226
12.2.5 Timer 控件	228
习题 12	230
实验 12	231
第 13 章 Windows 窗体应用程序(二)	236
13.1 通用对话框	236

13.1.1	OpenFileDialog 对话框	236
13.1.2	SaveFileDialog 对话框	237
13.1.3	FontDialog 对话框	237
13.1.4	通用对话框应用举例	238
13.2	菜单和工具栏	240
13.2.1	MenuStrip 控件	240
13.2.2	ContextMenuStrip 控件	240
13.2.3	ToolStrip 控件	240
13.2.4	菜单和工具栏应用举例	240
13.3	多重窗体	243
13.3.1	添加新窗体	243
13.3.2	调用其他窗体	243
13.3.3	多重窗体应用举例	243
13.4	多文档界面	244
13.4.1	创建 MDI 父窗体	245
13.4.2	创建 MDI 子窗体	245
13.4.3	处理 MDI 子窗体	245
13.5	图形绘制	246
13.5.1	GDI+ 图形绘制概述	246
13.5.2	绘制字符串	248
13.5.3	绘制图形	248
习题 13		251
实验 13		251
第 14 章	ASP.NET Web 应用程序(一)	254
14.1	ASP.NET Web 应用程序概述	254
14.2	创建 ASP.NET Web 应用程序	255
14.3	ASP.NET Web 页面	257
14.3.1	ASP.NET Web 页面概述	257
14.3.2	创建 ASP.NET 页面	258
14.4	ASP.NET Web 服务器控件	259
14.4.1	ASP.NET Web 服务器控件概述	259
14.4.2	使用标准服务器控件创建 Web 页面	261
14.5	验证服务器控件	264
14.5.1	验证服务器控件概述	264
14.5.2	使用验证服务器控件创建 Web 页面	264
14.6	数据服务器控件	267
14.6.1	数据服务器控件概述	267
14.6.2	使用数据服务器控件创建 Web 页面	267

14.7 创建 ASP.NET Web 页面并访问数据库	270
习题 14	271
实验 14	271
第 15 章 ASP.NET Web 应用程序(二)	277
15.1 ASP.NET 页面会话状态和页面导航	277
15.1.1 ASP.NET Web 应用程序上下文	277
15.1.2 ASP.NET Web 应用程序事件	279
15.1.3 ASP.NET Web 页面导航	281
15.2 ASP.NET Web 应用程序的布局和导航	282
15.2.1 ASP.NET Web 母版页	282
15.2.2 ASP.NET Web 导航控件	283
15.2.3 应用举例:设计 ASP.NET Web 站点	284
15.3 ASP.NET 主题和外观	289
15.3.1 ASP.NET 主题和外观概述	289
15.3.2 定义主题	289
15.3.3 定义外观	289
15.3.4 定义 CSS 样式	290
15.3.5 在页面中使用主题	291
15.3.6 应用举例:使用 ASP.NET 主题和外观自定义 Web 站点	292
习题 15	295
实验 15	295
第 16 章 综合应用案例	299
16.1 多窗口文本编辑器系统设计	299
16.1.1 系统基本功能	299
16.1.2 功能模块设计	299
16.1.3 系统的实现	300
16.2 ASP.NET 网上书店系统设计	303
16.2.1 系统总体设计	303
16.2.2 数据库设计	303
16.2.3 功能模块设计	304
16.2.4 系统的实现	305
实验 16	317
附录 A 部分习题参考解答	318
参考文献	325

第 1 章

C# 语言

C# 语言源于 C 语言家族,是一种简洁的、类型安全的、面向对象的编程语言,主要用来构建在 .NET Framework 上运行的各种安全、可靠的应用程序。

本章要点:

- ◆ C# 语言及其特点
- ◆ C# 语言的编译和运行环境
- ◆ 简单 C# 程序的创建、编译和运行
- ◆ C# 程序的基本结构
- ◆ 控制台输入/输出
- ◆ Visual Studio 快速入门

1.1 C# 语言概述

1.1.1 C# 语言简介

C# (C Sharp) 语言(本书中多处简写为 C#)是一种简洁的、类型安全的、面向对象的编程语言。C# 起源于 C 语言家族,因此具有 C++ 的功能。C# 已经分别由 ECMA International 和 ISO/IEC 组织接受并确立为 ECMA-334 标准和 ISO/IEC 23270 标准。

C# 采用与 C、C++ 或 Java 一致的花括号({和})语法,简单易学。C# 语法简化了 C++ 的诸多复杂性,同时又提供了 Java 所不具备的许多强大的功能,例如可为 null 的值类型、枚举、委托、lambda 表达式和直接内存访问等。

C# 作为微软 .NET Framework 的主要语言,其主要发展历史如表 1-1 所示。

表 1-1 C# 主要发展历史

发布时间	开发工具	开发平台	C# 版本
2003 年 4 月	Visual Studio .NET 2003	.NET Framework 1.1	1.2
2005 年 11 月	Visual Studio 2005	.NET Framework 2.0	2.0
2007 年 11 月	Visual Studio 2008	.NET Framework 3.5	3.0
2010 年 4 月	Visual Studio 2010	.NET Framework 4.0	4.0
2012 年 5 月	Visual Studio 2012	.NET Framework 4.5	5.0

本书主要基于 Visual Studio 2012/.NET Framework 4.5,讲述 C# 5.0 语言基础知识,以及使用 C# 5.0 语言的实际开发应用实例。需要说明的是,本书中多处省略了软件版本号。

1.1.2 C#语言的特点和开发应用范围

1. C#语言的特点

C#是一种现代的、面向对象的、类型安全的编程语言。C#具有下列特点：

(1) 简单。C#简化了C/C++中许多复杂的特性。例如,采用“==”比较操作,从而避免了C语言中与赋值操作“=”的混淆错误。

(2) 面向对象。C#支持数据封装、继承、多态和接口。所有的变量和方法,包括Main方法(应用程序的入口点),都封装在类定义中。

(3) 类型安全(type-safe)。C#是强类型语言,即每个变量和对象都必须具有声明类型。

(4) 现代。C#包括许多现代先进语言的特性。例如,C#支持属性、泛型、Lambda表达式、垃圾回收和异常处理等。

(5) 相互兼容性。C#提供对COM和基于Windows的应用程序的支持。

(6) 可伸缩性和可升级性。C#的设计中充分考虑到版本控制(versioning)的需要,以确保C#程序和库能够以兼容的方式逐步演进。

2. C#语言的开发应用范围

C#语言主要用来构建在.NET Framework上运行的各种安全、可靠的应用程序。使用C#,可以创建下列类型的应用程序和服务:

(1) 桌面应用。包括控制台(console)应用程序、Windows窗体应用程序等。

(2) Web应用。包括ASP.NET应用程序、Web服务等。

(3) WPF(Windows Presentation Foundation,Windows呈现基础)应用程序。

(4) 面向服务的应用程序 WCF(Windows Communication Foundation)。

(5) 工作流程的应用程序 WF(Windows Workflow Foundation)。

(6) 移动应用。

(7) Office平台应用程序。

(8) Windows服务。

1.2 C#语言的编译和运行环境

1.2.1 C#语言与 .NET Framework

用C#语言编写的程序在.NET Framework上运行。.NET Framework是Windows的一个组件,包括一个称为公共语言运行时(Common Language Runtime,CLR)的虚拟运行环境和一组统一的类库/Framework Class Library,FCL)。

用C#编写的源代码被编译为中间语言(Intermediate Language,IL)。IL代码与资源(例如位图和字符串)一起作为一种称为程序集的可执行文件存储在磁盘上,通常具有的扩展名为.exe(应用程序)或.dll(库)。执行C#程序时,程序集将加载到CLR中,然后根据程序集清单中的信息执行不同的操作。如果符合安全要求,CLR执行实时编译(Just-In-Time (JIT) Compilation)将IL代码转换为本机机器指令,并执行。CLR还提供与自动垃圾回