

21世纪高等学校规划教材 | 计算机科学与技术



C#程序设计 经典教程（第二版）

罗福强 杨剑 张敏辉 主编



清华大学出版社

014056292

21世纪高等学校规划教材 | 计算机科学与技术

TP312C
2029-2

C#程序设计经典教程

(第二版)

罗福强 杨剑 张敏辉 主编

图书馆

清华大学出版社



北航

C1744923

TP312C

2029-2

010282983

内 容 简 介

C#是微软公司推出的新一代编程语言。它在C++的基础之上重新打造,成为一门全新的完全面向对象的程序设计语言,能够提供更高的可靠性和安全性,不仅能用于开发传统的控制台应用程序和Windows应用程序,还可用于开发Web应用程序、WPF应用程序、Silverlight应用程序、Azure云应用程序、Windows 8应用程序等。

本书共14章,分为两部分,第1部分为C#基础,共8章,重点介绍了C#的基本语法、面向对象的程序设计方法、基于事件驱动的程序设计方法以及程序的调试和异常处理方法。第2部分共6章,在C#基础之上,比较全面地展示了C#的各种应用技术,包括Windows程序设计技术、数据库编程技术、文件操作与编程技术、XML与LINQ高级数据访问技术、网络应用编程技术和多媒体编程技术等。本书内容丰富,可操作性强,叙述简洁流畅,没有晦涩的术语,所有实例精心设计,能够使学生轻松、愉快地掌握C#的基本语法、编程方法和应用技巧。

本书可作为高等院校计算机相关专业学生的教材,也可作为初、中级读者和培训班学员学习的参考用书。

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#程序设计经典教程/罗福强等主编.--2版.--北京:清华大学出版社,2014

21世纪高等学校规划教材·计算机科学与技术

ISBN 978-7-302-36003-2

I. ①C… II. ①罗… 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(2014)第065917号

责任编辑:刘向威 薛 阳

封面设计:傅瑞学

责任校对:李建庄

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:北京市清华园胶印厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:26.25 字 数:658千字

版 次:2012年2月第1版 2014年8月第2版 印 次:2014年8月第1次印刷

印 数:1~2000

定 价:49.50元

产品编号:057392-01

出版说明

随着我国改革开放的进一步深化,高等教育也得到了快速发展,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,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投入力度,通过教育改革合理调整和配置了教育资源,优化了传统学科专业,积极为地方经济建设输送人才,为我国经济社会的快速、健康和可持续发展以及高等教育自身的改革发展做出了巨大贡献。但是,高等教育质量还需要进一步提高以适应经济社会发展的需要,不少高校的专业设置和结构不尽合理,教师队伍整体素质亟待提高,人才培养模式、教学内容和教学方法需要进一步转变,学生的实践能力和创新精神亟待加强。

教育部一直十分重视高等教育质量工作。2007年1月,教育部下发了《关于实施高等学校本科教学质量与教学改革工程的意见》,计划实施“高等学校本科教学质量与教学改革工程”(简称“质量工程”),通过专业结构调整、课程教材建设、实践教学改革、教学团队建设等多项内容,进一步深化高等学校教学改革,提高人才培养的能力和水平,更好地满足经济社会发展对高素质人才的需要。在贯彻和落实教育部“质量工程”的过程中,各地高校发挥师资力量强、办学经验丰富、教学资源充裕等优势,对其特色专业及特色课程(群)加以规划、整理和总结,更新教学内容、改革课程体系,建设了一大批内容新、体系新、方法新、手段新的特色课程。在此基础上,经教育部相关教学指导委员会专家的指导和建议,清华大学出版社在多个领域精选各高校的特色课程,分别规划出版系列教材,以配合“质量工程”的实施,满足各高校教学质量和教学改革的需要。

为了深入贯彻落实教育部《关于加强高等学校本科教学工作,提高教学质量的若干意见》精神,紧密配合教育部已经启动的“高等学校教学质量与教学改革工程精品课程建设工作”,在有关专家、教授的倡议和有关部门的大力支持下,我们组织并成立了“清华大学出版社教材编审委员会”(以下简称“编委会”),旨在配合教育部制定精品课程教材的出版规划,讨论并实施精品课程教材的编写与出版工作。“编委会”成员皆来自全国各类高等学校教学与科研第一线的骨干教师,其中许多教师为各校相关院、系主管教学的院长或系主任。

按照教育部的要求,“编委会”一致认为,精品课程的建设工作从开始就要坚持高标准、严要求,处于一个比较高的起点上。精品课程教材应该能够反映各高校教学改革与课程建设的需要,要有特色风格、有创新性(新体系、新内容、新手段、新思路,教材的内容体系有较高的科学创新、技术创新和理念创新的含量)、先进性(对原有的学科体系有实质性的改革和发展,顺应并符合21世纪教学发展的规律,代表并引领课程发展的趋势和方向)、示范性(教材所体现的课程体系具有较广泛的辐射性和示范性)和一定的前瞻性。教材由个人申报或各校推荐(通过所在高校的“编委会”成员推荐),经“编委会”认真评审,最后由清华大学出版

社审定出版。

目前,针对计算机类和电子信息类相关专业成立了两个“编委会”,即“清华大学出版社计算机教材编审委员会”和“清华大学出版社电子信息教材编审委员会”。推出的特色精品教材包括:

- (1) 21 世纪高等学校规划教材·计算机应用——高等学校各类专业,特别是非计算机专业的计算机应用类教材。
- (2) 21 世纪高等学校规划教材·计算机科学与技术——高等学校计算机相关专业的教材。
- (3) 21 世纪高等学校规划教材·电子信息——高等学校电子信息相关专业的教材。
- (4) 21 世纪高等学校规划教材·软件工程——高等学校软件工程相关专业的教材。
- (5) 21 世纪高等学校规划教材·信息管理与信息系统。
- (6) 21 世纪高等学校规划教材·财经管理与应用。
- (7) 21 世纪高等学校规划教材·电子商务。
- (8) 21 世纪高等学校规划教材·物联网。

清华大学出版社经过三十多年的努力,在教材尤其是计算机和电子信息类专业教材出版方面树立了权威品牌,为我国的高等教育事业做出了重要贡献。清华版教材形成了技术准确、内容严谨的独特风格,这种风格将延续并反映在特色精品教材的建设中。

清华大学出版社教材编审委员会

联系人:魏江江

E-mail: weijj@tup.tsinghua.edu.cn

前言

C#是由微软公司推出的完全面向对象的计算机高级语言。它简单、安全、灵活、功能强大,能够快速地开发各种应用软件。它解决了存在于许多程序设计语言中的问题,如安全问题、可靠性问题、与其他语言协调的能力、跨平台的兼容性等。相对于C++来说,C#更容易被理解和接受;相对于Java来说,C#更容易使用,开发软件的效率更高。C#既能用于开发传统的DOS应用程序、Windows应用程序和组件程序,还能用于开发Web应用程序、XML Web服务程序、WPF应用程序、WCF服务程序等,特别是它将Web技术、Silverlight动画编程技术、XNA游戏编程技术和Azure云平台技术紧密结合,使得它成为如今云计算时代的一种高效的程序设计语言。

本书在2012年2月发行了第1版,受到广大师生的欢迎。结合广大师生的反馈意见以及新的教学和应用开发经验,我们迅速组织编写了本教材第2版。第2版保持第1版的优点,基本思路不变:第一,面向应用型本科院校学生,立足于把C#的语法讲透彻、讲清楚,文字叙述要简练;第二,紧紧围绕面向对象程序设计思想和可视化的Windows程序设计方法展开教学内容;第三,书中所有案例均精心设计,不仅代码完整,还贴近学生实际生活;第四,坚持零起点原则,学生可以在没有C/C++基础的情况下使用本书;第五,坚持应用为纲,全书分为基础篇和应用实战篇,特别在应用部分,全面展示了C#在各应用领域的编程技巧。

针对第1版,本书第2版在修订时,首先深入分析了C#程序设计的知识结构和技能结构,然后设计教学内容,使得全书的重点更加突出。在内容组织上,根据教学过程中的实际反馈,把比较难以理解的内容独立出来,把程序设计时实际上不常用的技术独立出来,从而使得全书的难点更加清晰。例如,针对类的方法成员和构造函数,第1版把方法的声明、调用、参数传递以及重载,构造函数的声明、调用和重载等集中安排,实际教学过程中发现学生越学越迷糊,因此第2版把相关内容拆分到4.3、4.4、4.5这三节中,先安排最简单的内容(即方法和构造函数的简单声明和调用),再介绍参数传递,最后讨论重载问题。这样,更有利于分层次教学。

同时,为了尽量避免对使用本书第1版的教师造成教学影响,第2版在修订时针对原来的教学案例或上机任务基本保持不变,当然新增了部分更加经典的案例。在每章的习题部分,为了能更有效地培养或提高学生的编程能力,因此增加了一些编程任务。针对复杂的编程任务,本书还给出了适当的提示信息。

本书第2版以Visual Studio .NET 2012和.NET Framework 4.5为蓝本,深入介绍了C#语言及其应用。全书共分为14章,基本上覆盖了C#的主要领域,在讲解C#语法的基础上,以面向对象的程序设计方法和基于事件的程序设计方法为核心,重点阐述Windows程序设计技术、数据库访问与编程技术、文件操作与编程技术、XML和LINQ高级数据访问技术、网络应用编程技术、多媒体编程技术等实用技术。

因此,经过修订,本书第2版具有以下三个特点:第一,知识结构完整,根据循序渐进的认识规律设计编写内容及顺序,重点和难点突出;第二,提供了大量的案例,所有案例程序都是完整的,都是通过 Visual Studio .NET 2012 调试的,并给出了运行效果,针对部分复杂案例还进行了详细的分析,这样有助于读者理解程序算法;第三,全书配备了丰富的、符合教学实际的、能真正培养学生编程能力的编程任务和实训任务。

本书可作为高等院校 Visual C# .NET 课程的教材或参考资料,也可供软件开发人员参考使用。

本书入选电子科技大学成都学院 CDIO 教改项目,我们在软件工程专业入门计算机语言教学中引入 C#,经过实验取得了较好的教学效果。参与本书编写工作的有:罗福强、杨剑、张敏辉和熊永福等。罗福强编写了第1、4、5、6、9和10章,杨剑编写了第2、3、7、8、11和12章,张敏辉编写了第14章,熊永福编写了第13章和所有上机实验。本书由罗福强负责全书统稿、修改和审校工作。本书在立项、教学实验和编写过程中得到国腾集团云计算研究中心主任兼电子科技大学成都学院云计算科学与技术系主任武志学博士的大力支持和帮助,特别是在教学内容、教学方法方面提供大量的指导意见。在此,我们对武主任表示真挚的感谢。

由于时间仓促,书中难免有不妥之处,殷切期望读者提出中肯的意见。

编者

2014年5月

目 录

第 1 章 C# 概述	1
1.1 C# 简介	1
1.1.1 C# 语言的发展	1
1.1.2 C# 语言的特点	2
1.2 C# 程序的开发环境	3
1.2.1 Visual Studio .NET 的基本操作	3
1.2.2 Visual Studio .NET 解决方案和项目文件的组织结构	6
1.2.3 Visual Studio .NET 2012 的帮助	7
1.3 C# 程序入门	8
1.3.1 一个简单的 C# 控制台应用程序	9
1.3.2 一个简单的 Win 32 应用程序	12
1.3.3 一个具有输入功能的 Win 32 应用程序	14
1.3.4 一个简单的 Web 应用程序	17
1.3.5 C# 程序的特点	19
习题	20
上机实验 1	20
第 2 章 C# 程序设计基础	22
2.1 常量与变量	22
2.1.1 常量	23
2.1.2 变量	24
2.2 C# 的数据类型	26
2.2.1 简单类型	27
2.2.2 枚举型 enum	28
2.2.3 结构型 struct	29
2.2.4 数据类型转换	31
2.3 运算符与表达式	33
2.3.1 算术运算符与表达式	33
2.3.2 赋值运算符与表达式	35
2.3.3 关系运算符与表达式	36
2.3.4 逻辑运算符与表达式	36
2.4 数组和字符串	38

2.4.1 一维数组	38
2.4.2 多维数组	41
2.4.3 数组型的数组	43
2.4.4 字符串	44
习题	46
上机实验 2	47
第 3 章 C# 程序的流程控制	50
3.1 C# 程序的分支语句	50
3.1.1 if 语句	51
3.1.2 多分支 if...else if 语句	52
3.1.3 switch 语句	55
3.1.4 分支语句的嵌套	58
3.2 C# 程序的循环语句	59
3.2.1 while 语句	60
3.2.2 do...while 语句	61
3.2.3 for 语句	63
3.2.4 foreach 语句	65
3.2.5 循环语句的嵌套	67
3.3 跳转语句	68
3.3.1 break 语句	68
3.3.2 continue 语句	70
习题	71
上机实验 3	73
第 4 章 面向对象程序设计入门	75
4.1 面向对象的基本概念	75
4.1.1 对象	76
4.1.2 事件与方法	76
4.1.3 类与对象	77
4.1.4 抽象、封装、继承与多态	78
4.2 类的定义与使用	80
4.2.1 类的声明和实例化	80
4.2.2 类的可访问性	83
4.2.3 值类型与引用类型	84
4.3 类的成员及其定义	86
4.3.1 类的常量与字段成员	86
4.3.2 类的属性成员	87
4.3.3 类的方法成员	89

4.3.4	类的构造函数	91
4.4	方法的参数传递	95
4.4.1	按值传参	95
4.4.2	按引用传参	96
4.4.3	输出参数	97
4.4.4	引用类型的参数	99
4.4.5	数组型参数	100
4.5	方法的重载	102
4.5.1	方法的重载	102
4.5.2	构造函数的重载	104
4.6	对象的生命周期	106
4.6.1	对象的生命周期	106
4.6.2	析构函数	107
	习题	108
	上机实验 4	109
第 5 章	面向对象的高级程序设计	112
5.1	静态成员与静态类	112
5.1.1	类的静态成员	113
5.1.2	静态构造函数	115
5.1.3	静态类	116
5.2	类的继承性	117
5.2.1	派生类的声明	117
5.2.2	构造函数	118
5.2.3	密封类	122
5.3	类的多态性	123
5.3.1	使用 new 重新定义类的成员	123
5.3.2	用 virtual 和 override 定义类的成员	123
5.3.3	访问基类的成员	125
5.4	抽象类	129
5.4.1	抽象类及其抽象成员	129
5.4.2	重载抽象方法	131
5.5	接口	133
5.5.1	接口的声明	133
5.5.2	接口的实现	134
5.5.3	接口的继承性	134
5.5.4	多重接口实现	135
5.5.5	访问接口的成员	136
5.5.6	抽象类与接口的比较	141

5.6 嵌套类、分部类与命名空间	141
5.6.1 嵌套类	141
5.6.2 分部类	143
5.6.3 命名空间	144
习题	146
上机实验 5	148
第 6 章 集合、索引器与泛型	153
6.1 集合	153
6.1.1 集合概述	153
6.1.2 ArrayList	154
6.1.3 哈希表 Hashtable	159
6.1.4 栈和队列	161
6.2 索引器	162
6.2.1 索引器的定义	163
6.2.2 索引器的使用	164
6.2.3 索引器的重载	164
6.2.4 接口中的索引器	166
6.2.5 索引器与属性的比较	167
6.3 泛型	167
6.3.1 泛型概述	167
6.3.2 泛型集合	168
6.3.3 自定义泛型	170
6.3.4 泛型的高级应用	174
习题	178
上机实验 6	178
第 7 章 程序调试与异常处理	180
7.1 程序错误	180
7.1.1 程序错误分类	180
7.1.2 调试程序错误	182
7.2 程序的异常处理	185
7.2.1 异常的概念	185
7.2.2 异常处理	186
7.2.3 try...catch 语句	188
7.2.4 finally 语句	190
7.2.5 throw 语句与抛出异常	191
习题	192
上机实验 7	193

第 8 章 基于事件驱动的程序设计技术	195
8.1 基于事件的编程思想	195
8.2 委托	197
8.2.1 委托概述	197
8.2.2 委托的声明、实例化与使用	197
8.2.3 多路广播与委托的组合	200
8.3 事件	201
8.3.1 事件的声明	202
8.3.2 订阅事件	203
8.3.3 触发事件	204
8.4 基于事件的 Windows 编程	207
8.4.1 Windows 应用程序概述	208
8.4.2 Windows 窗体与事件驱动编程	210
习题	215
上机实验 8	215
第 9 章 Windows 程序的界面设计	221
9.1 窗体与控件概述	222
9.1.1 Windows 窗体	222
9.1.2 窗体的控件	223
9.2 按钮与文本显示、编辑控件	225
9.2.1 按钮控件	225
9.2.2 文本显示控件	226
9.2.3 文本编辑控件	227
9.2.4 应用实例——用户登录	229
9.3 列表与选择控件	231
9.3.1 RadioButton 控件	231
9.3.2 CheckBox 控件	231
9.3.3 ListBox 控件	232
9.3.4 ComboBox 控件	233
9.3.5 其他常用控件	233
9.3.6 应用实例——添加个人收支明细	234
9.4 图形显示控件	237
9.4.1 PictureBox 控件	237
9.4.2 ImageList 控件	238
9.4.3 应用实例——关于我们	238
9.5 容器控件	239
9.5.1 GroupBox 控件	239

9.5.2	Panel 控件	239
9.5.3	TabControl 控件	240
9.5.4	应用实例——添加收支项目	240
9.6	对话框	244
9.6.1	对话框概述	244
9.6.2	消息框	246
9.6.3	通用对话框	248
9.6.4	应用实例——简单的文本编辑器	250
9.7	菜单、工具栏和状态栏	251
9.7.1	菜单	251
9.7.2	工具栏	253
9.7.3	状态栏	254
9.7.4	应用实例——个人理财系统的主窗口设计	255
9.8	SDI 和 MDI 应用程序	257
9.8.1	创建 SDI 应用程序	257
9.8.2	创建 MDI 应用程序	257
9.8.3	应用实例——个人理财的 MDI 设计	258
	习题	259
	上机实验 9	260
第 10 章	C# 数据库编程技术	265
10.1	数据库与 ADO.NET 概述	265
10.1.1	数据库概述	265
10.1.2	SQL 概述	268
10.1.3	ADO.NET 概述	269
10.1.4	ADO.NET 访问数据库的一般步骤	271
10.2	Connection 与 Command 对象的使用	271
10.2.1	Connection 对象	271
10.2.2	Command 对象	272
10.2.3	应用实例——实现用户登录	273
10.2.4	应用实例——实现收支类别的添加	275
10.3	DataReader 对象的使用	276
10.3.1	DataReader 对象	276
10.3.2	应用实例——实现收支项目的添加	277
10.3.3	应用实例——实现收支明细的添加	280
10.4	DataAdapter 与 DataSet 对象的使用	283
10.4.1	DataAdapter 与 DataSet 对象	283
10.4.2	DataGridView 控件	286
10.4.3	应用实例——实现收支明细的查询	287

习题	289
上机实验 10	289
第 11 章 文件操作与编程技术	292
11.1 文件的输入/输出	292
11.1.1 文件 I/O 与流	292
11.1.2 读写文本文件	293
11.1.3 读写二进制文件	295
11.1.4 对象的序列化	297
11.2 文件操作控件	300
11.2.1 SaveFileDialog 控件	301
11.2.2 OpenFileDialog 控件	302
11.2.3 FolderBrowserDialog 控件	304
11.2.4 应用实例——简易的写字板程序	305
习题	308
上机实验 11	309
第 12 章 高级数据访问与处理技术	313
12.1 XML 编程	313
12.1.1 XML 概述	313
12.1.2 XML 文档的创建	315
12.1.3 XML 文档的查询	321
12.1.4 XML 文档的编辑	324
12.2 LINQ 编程	327
12.2.1 LINQ 概述	327
12.2.2 LINQ 的查询子句	330
12.2.3 LINQ to XML 的应用	334
12.2.4 LINQ to SQL 的应用	336
习题	343
上机实验 12	344
第 13 章 网络应用编程技术	346
13.1 网络编程基础	346
13.1.1 计算机网络概述	346
13.1.2 计算机网络的通信协议	347
13.1.3 System.Net 概述	348
13.2 Socket 编程	352
13.2.1 Socket 编程概述	352
13.2.2 TCP 应用编程	355

13.2.3	UDP 应用编程	360
13.3	E-mail 与 FTP 应用编程	363
13.3.1	E-mail 的发送	364
13.3.2	E-mail 的接收	366
13.3.3	FTP 的下载	371
	习题	375
	上机实验 13	375
第 14 章	多媒体编程技术	378
14.1	GDI+ 绘图	378
14.1.1	GDI+ 概述	378
14.1.2	System.Drawing 命名空间	380
14.1.3	创建 Graphics 对象	380
14.1.4	颜料、钢笔和画笔	381
14.1.5	点、线和图形	383
14.1.6	图像和文本	390
14.1.7	坐标系统及变换	393
14.2	Windows Media Player 组件的使用	399
14.2.1	Windows Media Player 组件的介绍	399
14.2.2	Windows Media Player 组件的使用	401
	习题	404
	上机实验 14	404
	参考文献	406

第1章

C#概述

总体要求

- 了解 C# 语言的特点及其发展。
- 了解简单的三种应用程序(控制台应用程序、Windows 窗体应用程序和 Web 应用程序)的结构及其特点。
- 初步掌握 Visual Studio .NET 2012 的操作方法,包括创建项目、编辑程序、生成和调试应用程序的方法。

相关知识

- 了解计算机软件、计算机语言及分类的知识。
- 熟悉 Windows 系统基础知识及操作。

学习重点

- C# 程序的结构、特点。
- C# 程序的建立、编辑、生成和运行等基本上机操作过程。

学习难点

- 控制台应用程序与 Win 32 应用程序的区别。

1.1 C# 简介

1.1.1 C# 语言的发展

在过去的三十年里,C 和 C++ 已经成为在商业软件的开发领域中使用最广泛的语言。它们为程序员提供了十分灵活的操作,不过同时也牺牲了一定的效率。与 Visual Basic 等语言相比,同等级别的 C/C++ 应用程序往往需要更长时间来开发。由于 C/C++ 语言的复杂性,许多程序员都试图寻找一种新的语言,希望能在功能与效率之间找到一个更为理想的权衡点。

目前有些语言,以牺牲灵活性的代价来提高效率。可是这些灵活性正是 C/C++ 程序员所需要的。这些解决方案对编程人员的限制过多(如屏蔽一些底层代码控制的机制),其所提供的功能难以令人满意。这些语言无法方便地同原来的系统交互,也无法与当前的网络编程很好地结合。

对于 C/C++ 用户来说,最理想的解决方案无疑是在快速开发的同时又可以调用底层平

台的所有功能。他们想要一种和最新的网络标准保持同步并且能和已有的应用程序良好整合的环境。另外,一些 C/C++ 开发人员还需要在必要的时候进行一些底层的编程。

C#(读作 C Sharp)是微软对这一问题的解决方案。C#是一种最新的、面向对象的编程语言。它是一种简单但功能强大的编程语言,使得程序员可以快速地编写各种基于 Microsoft .NET 平台的应用程序。

它是从 C 和 C++ 语言演化而来。它在语句、表达式和运算符方面使用了许多 C++ 功能。它在类型安全性、版本转换、事件和垃圾回收等方面进行了相当大的改进和创新。它提供对常用 API^①(例如 .NET Framework、COM+ 等)的访问。

C# 自推出以来,已得到不断地改进和优化,通常同 .NET Framework 一起,随新版的 Visual Studio .NET 的发布而更新。目前,Visual Studio .NET 主要版本包括 2003 版、2005 版、2008 版、2010 版、2012 版和 2013 版。.NET Framework 主要版本包括 1.1 版、2.0 版、3.0 版、3.5 版、4.0 版、4.5 版和 5.0 版。由于微软公司淡化了 C# 的版本问题,因此通常借助 .NET Framework 的版本号来区别 C# 的版本。本书以 .NET Framework 4.5 和 Visual Studio .NET 2012 为范本,所有案例均在 Visual Studio .NET 2012 中经过调试运行无误。

1.1.2 C# 语言的特点

C# 是一种简洁、类型安全的面向对象的语言,开发人员可以使用它来构建在 .NET Framework 上运行的各种安全、可靠的应用程序,包括控制台应用程序、Windows 窗体应用程序、Web 应用程序、WPF 应用程序、Silverlight 应用程序、WCF 服务、分布式组件、数据库应用程序、微软云平台 Azure 应用程序、Windows 8 应用程序等。

作为一种面向对象的语言,C# 支持封装、继承和多态性的概念。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。C# 程序的生成过程比 C 和 C++ 简单,比 Java 更为灵活。没有单独的头文件,也不要求按照特定顺序声明方法和类型。C# 源文件可以定义任意数量的类、结构、接口和事件。

相对其他计算机程序设计语言来说,C# 具有如下优点。

(1) C# 是一种精确、简单、类型安全、面向对象的语言。正是由于 C# 面向对象的卓越设计,使它成为构建各种应用程序组件的理想之选——无论是高级的商业对象还是系统级的应用程序。

(2) C# 具有生成持久系统级组件的能力,提供 COM+ 或其他技术平台支持以集成现有代码,提供垃圾回收和类型安全以实现应用程序可靠性,提供内部代码信任机制以保证应用程序的安全性。

(3) C# 利用 .NET Framework 的通用类型系统能够与其他程序设计语言交互操作。C# 应用程序能跨语言、跨平台互相调用。使用 C# 语言可实现具有不同专业技术背景的人员协同工作,完成软件系统的设计和开发。

(4) C# 支持 MSMQ(微软消息队列服务)、COM+ 组件服务、WCF 服务和 .NET

^① API 即 Application Programming Interface,应用程序编程接口。