



21世纪高等学校计算机  
应用技术规划教材

# C# 程序设计教程 (第2版)



© 蒙祖强 编著

清华大学出版社

21 世纪高等学校计算机应用技术规划教材

# C# 程序设计教程

(第 2 版)

蒙祖强 编著

清华大学出版社  
北京

## 内 容 简 介

本书将目标驱动和内容驱动相结合,深入浅出地介绍了 C# 语言的基础知识和多种应用程序的开发方法与技术。内容包括程序设计语言和程序设计方法的相关概念、Visual Studio 2015 集成开发环境、C# 语言的基本数据类型和语法体系、面向对象编程方法、异常处理技术、窗体应用程序设计和开发方法、目录和文件的读写操作、ActiveX 控件和自定义组件的开发技术、多线程技术、数据库开发技术、ASP.NET Web 应用开发方法、基于数据控件的应用程序开发技术、Excel 数据读写在 Web 开发中的应用以及各类应用程序的部署和发布方法等。每章均配有一定数量的练习题,并以电子资源的方式提供了全部的参考答案(包括上机题的实例程序),以便学生练习和辅助教学。

本书所有实例(包括习题中的上机题程序)的源代码以及教学用的全部 PPT 课件、教学大纲、习题答案等教学资源均可在清华大学出版社网站(<http://www.tup.com.cn>)上下载。

本书主要面向应用型本科院校、大专院校计算机专业及相近专业的学生,也适用于 C# 爱好者、初学者,还可以作为有关培训机构的培训教材。

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/蒙祖强编著.—2 版.—北京:清华大学出版社,2019

(21 世纪高等学校计算机应用技术规划教材)

ISBN 978-7-302-52999-6

I. ①C… II. ①蒙… III. ①C 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(2019)第 093925 号

责任编辑:陈景辉 黄 芝

封面设计:刘 键

责任校对:胡伟民

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

邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto: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:31.5

字 数:790 千字

版 次:2010 年 10 月第 1 版 2019 年 9 月第 2 版

印 次:2019 年 9 月第 1 次印刷

印 数:22501~24000

定 价:79.90 元

产品编号:078410-01

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。

本系列教材立足于计算机公共课程领域,以公共基础课为主、专业基础课为辅,横向满足高校多层次教学的需要。在规划过程中体现了如下一些基本原则和特点。

(1) 面向多层次、多学科专业,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,反映各层次对基本理论和原理的需求,同时加强实践和应用环节。

(2) 反映教学需要,促进教学发展。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,为学生的知识、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教学质量和教学改革成果的教材。

(4) 主张一纲多本,合理配套。基础课和专业基础课教材配套,同一门课程可以有针对不同层次、面向不同专业的多本具有各自内容特点的教材。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。在制定教材规划时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、评审确定主题。书稿完成后要真实实行审稿程序,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。建立一支高水平教材编写梯队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应用技术规划教材

联系人:魏江江 weijj@tup.tsinghua.edu.cn

# 前言

C#是微软公司基于.NET平台推出的一种全新的、完全面向对象的高级程序设计语言。它充分吸收了C/C++的优点,继承了Visual Basic的高效性和C++的强大功能,基于.NET Framework的有力支撑提供了实现跨平台应用开发的强有力的集成开发工具和方法,具有良好的可靠性和安全性。用微软公司的话来说,“C#是从C和C++派生来的一种简单、现代、面向对象和类型安全的编程语言”。

C#看起来与Java有着惊人的相似,几乎与Java有相同的语法,也是先编译成中间代码,然后再加载到内存运行,但在底层实现中却有着本质的区别。Java程序编译后形成字节代码需要在Java虚拟机(JVM)上运行。C#程序编译成中间代码后则是通过.NET Framework中的公共语言运行时(Common Language Runtime,CLR)来执行,它借鉴了Delphi的一些原理,与COM(组件对象模型)直接集成,同时.NET Framework还提供内容丰富、功能强大的类库供C#调用,这使得C#变成一种功能十分强大的开发工具,可以实现几乎所有类型应用程序的开发。

在现今的数据时代,数据的有效管理、分析、处理以及良好的呈现方式是一项基本的应用需求。Visual Studio 2015很好地迎合了这种应用需求的发展。作为Visual Studio的强力支撑语言,C#必将得到微软的进一步加强和完善,在数据管理、分析和数据呈现等方面发挥着不可替代的作用,受到更多程序员的青睐。可以说,要想掌握软件开发的未来,就要先掌握基于.NET平台的C#开发方法。

本书主要面向应用研究型本科院校、大专院校计算机专业及相近专业的学生,也适用于C#爱好者、初学者,还可以作为有关培训机构的培训教材。

针对上述的读者定位,本书采用目标驱动和内容驱动相结合的行文方式,其中以内容驱动为主、目标驱动为辅。具体讲,总体上是按照C#语言教学内容逐层深入统稿全书,先讲容易的、基础的内容,然后讲解复杂的、深入的内容,这与目前大多教材的行文方式相同;但在局部上则采用目标驱动的方法,即针对一个较大的知识点,一般都先设定一个具体的目标(要解决的具体问题),然后编写一个简要的、容易实现的、能达到该目标(解决问题)的应用程序,该程序涉及的知识尽可能覆盖该知识点的所有内容。这样,即使读者不知道“为什么”,但他知道“怎么做”,由此可以快速获得对该知识点的感性认识,实现对知识点学习的快速入门,这对理解和掌握随后要讲解的内容大有裨益。可见,本书的行文方式有效吸收了内容驱动和目标驱动的优点,摒弃了它们的缺点,能让读者以最快的速度掌握C#语言的核心内容。

本书第一版已经销售了两万余册,深受广大师生和读者的喜爱,其中有些师生和读者来信咨询相关学习问题,有些读者提出了宝贵的意见等。这些都是作者出版第二版的重要动力来源。在清华大学出版社有关领导和编辑的关心和指导下,历经一年多的编写和完善,本书第二版终于跟读者见面了。与第一版相比,第二版融入了作者这几年的实际项目开发经

验,包含了项目开发过程中常用的方法和技术。增加和修改的部分主要体现在以下几个地方:对内容体系结构进行了适当调整;在第2~5章中进一步丰富和完善了C#的语法部分;在第6章中增加了对许多常用控件的介绍,使得针对窗体的编程变得更为灵活;在第10、11章中全面地介绍了数据库应用开发的理论和方法;在第12章中系统地介绍了Visual Studio提供的数据显示和数据操纵控件,为复杂数据的管理和可视化提供了有效的解决方案;在第13章中详细介绍了Excel数据读写技术及其在Web应用开发中的实现方法。此外,凡是涉及数据库应用开发的部分,基本上都同时给出了面向C/S模式和B/S模式的实现方法。

通过对本书的学习,读者不但可以较为全面地掌握C#的理论基础知识,而且还可以深入掌握项目开发中常用的技术和方法,基本具备开发中等规模软件系统的能力。

此外,为了辅助教学和方便学生的学习,每章均配有一定数量的练习题,并以电子资源的形式提供了全部的参考答案(包括上机题的实例程序)。

全书由蒙祖强执笔,杨柳审阅。此外,参与本书编写、资料整理或调试程序的还有覃华、杨丽娜、黄柏雄、秦亮曦、唐天兵、张锦雄、李虹利、郭英明、李富星、陈凤、杨坚、林敏鸿、韦人予、唐嘉骏等。

本书所有实例(包括习题中的上机题程序)的源代码以及教学用的全部PPT课件、教学大纲、习题答案等教学资源均可在清华大学出版社网站(<http://www.tup.com.cn>)上下载。

由于作者水平有限,书中疏漏和不妥之处在所难免,恳请广大读者批评指正。

蒙祖强

2019年5月



<b>第 1 章 C# 程序设计基础</b> .....	1
1.1 程序设计语言 .....	1
1.1.1 程序设计语言的定义 .....	1
1.1.2 程序设计语言的分类 .....	1
1.2 程序设计方法 .....	3
1.2.1 结构化程序设计方法 .....	3
1.2.2 面向对象程序设计方法 .....	4
1.3 C# 程序设计语言概述 .....	6
1.3.1 C# 语言的起源与发展 .....	6
1.3.2 C# 语言的特点 .....	6
1.4 C# 集成开发环境 .....	8
1.4.1 Visual Studio 的发展历史 .....	8
1.4.2 Visual Studio 2015 集成开发环境 .....	9
1.4.3 控制台应用程序的开发步骤 .....	10
1.4.4 窗体(Windows)应用程序的开发步骤 .....	11
1.5 习题 .....	13
<b>第 2 章 基本数据类型</b> .....	15
2.1 一个简单的程序——华氏温度到摄氏温度的转换 .....	15
2.1.1 创建控制台应用程序 .....	15
2.1.2 代码解释 .....	16
2.2 基本数据类型 .....	16
2.2.1 数值类型 .....	17
2.2.2 字符类型和字符串类型 .....	18
2.2.3 布尔类型与对象类型 .....	18
2.3 变量与常量 .....	18
2.3.1 标识符与命名规则 .....	18
2.3.2 变量 .....	20
2.3.3 常量 .....	21
2.3.4 类型转换 .....	22
2.3.5 装箱与拆箱 .....	24
2.4 基本运算 .....	25
2.4.1 算术运算 .....	25
2.4.2 关系运算与逻辑运算 .....	26

2.4.3	条件运算 .....	28
2.4.4	赋值运算 .....	28
2.4.5	运算符的优先级 .....	29
2.5	复合数据类型 .....	30
2.5.1	结构类型 .....	30
2.5.2	枚举类型 .....	31
2.6	数组的定义和使用 .....	32
2.6.1	数组的定义 .....	32
2.6.2	数组的引用 .....	34
2.6.3	二维数组 .....	34
2.6.4	多维数组 .....	35
2.7	习题 .....	35
<b>第3章</b>	<b>选择结构和循环结构 .....</b>	<b>38</b>
3.1	一个简单的选择结构程序——分段函数的实现 .....	38
3.1.1	创建C#控制台应用程序 .....	38
3.1.2	选择结构解析 .....	39
3.2	if语句——二分支选择语句 .....	39
3.2.1	if...语句 .....	40
3.2.2	if...else...语句 .....	41
3.2.3	if...else if...else...语句 .....	42
3.3	switch语句——多分支选择语句 .....	43
3.4	一个简单的循环结构程序——等差数列求和 .....	46
3.4.1	创建C#控制台应用程序 .....	46
3.4.2	循环结构解析 .....	46
3.5	while语句和do...while语句 .....	47
3.5.1	while语句 .....	47
3.5.2	do...while语句 .....	48
3.6	for语句和foreach语句 .....	49
3.6.1	for语句 .....	50
3.6.2	foreach语句 .....	51
3.7	跳转语句 .....	54
3.7.1	break语句和continue语句 .....	54
3.7.2	goto语句 .....	56
3.7.3	return语句 .....	57
3.8	习题 .....	58
<b>第4章</b>	<b>面向对象编程方法 .....</b>	<b>62</b>
4.1	一个简单的程序——虚数类的定义与应用 .....	62
4.1.1	编写虚数类的代码 .....	62

4.1.2	程序结构解析	64
4.2	类和对象	65
4.2.1	类和对象的定义	65
4.2.2	对象的访问方法及访问控制	66
4.2.3	类的构造函数和析构函数	69
4.2.4	类的属性	72
4.2.5	类的静态成员	73
4.2.6	成员方法的四种参数类型	75
4.3	类的继承、重载与多态	78
4.3.1	继承	78
4.3.2	重载	80
4.3.3	类的多态	81
4.4	运算符的重载	83
4.4.1	一元运算符重载	83
4.4.2	二元运算符重载	85
4.4.3	类型转换运算符重载	85
4.5	接口及其实现	86
4.5.1	接口的声明	86
4.5.2	接口的实现	87
4.6	方法的委托	88
4.6.1	一个简单的方法委托程序	89
4.6.2	委托类型的声明和实例化	90
4.6.3	委托的引用	91
4.6.4	委托的组合	94
4.7	泛型类	96
4.7.1	泛型类的定义	96
4.7.2	泛型数组类——List < T >类	98
4.8	常用的几个类	102
4.8.1	String 类	102
4.8.2	DateTime 类	105
4.8.3	Math 类和 Random 类	108
4.9	命名空间	110
4.9.1	命名空间的声明	110
4.9.2	命名空间的导入	112
4.10	习题	113
<b>第 5 章</b>	<b>异常处理</b>	<b>119</b>
5.1	一个产生异常的简单程序	119
5.1.1	程序代码	119

5.1.2	异常处理过程分析 .....	120
5.2	异常的捕获与处理 .....	120
5.2.1	异常的概念 .....	120
5.2.2	try-catch 结构 .....	121
5.2.3	try-catch-catch 结构 .....	123
5.2.4	try-catch-finally 结构 .....	124
5.3	异常的抛出及自定义异常 .....	126
5.3.1	抛出异常 .....	126
5.3.2	用户自定义异常 .....	128
5.4	习题 .....	129
<b>第 6 章</b>	<b>窗体应用程序设计 .....</b>	<b>133</b>
6.1	一个简单的文本编辑器 .....	133
6.1.1	创建文本编辑器程序的步骤 .....	133
6.1.2	程序结构解析 .....	136
6.2	组件的公共属性、事件和方法 .....	136
6.2.1	Object 类 .....	136
6.2.2	Control 类 .....	138
6.3	常用的控件 .....	142
6.3.1	按钮类控件 .....	142
6.3.2	文本类控件 .....	145
6.3.3	列表类控件 .....	156
6.3.4	其他常用控件 .....	173
6.4	常用的对话框 .....	180
6.4.1	打开和保存文件对话框 .....	180
6.4.2	字体对话框和颜色对话框 .....	183
6.4.3	文件夹浏览对话框 .....	184
6.5	消息对话框 .....	185
6.5.1	模式对话框与非模式对话框 .....	185
6.5.2	基于 MessageBox 类的消息对话框 .....	188
6.6	菜单和工具栏的设计 .....	190
6.6.1	主菜单 .....	190
6.6.2	弹出式菜单 .....	192
6.6.3	工具栏 .....	194
6.7	实例——多文档界面编辑器 .....	195
6.7.1	创建 MDI 应用程序框架 .....	195
6.7.2	设计菜单和工具栏 .....	196
6.7.3	编写事件处理函数 .....	198
6.8	习题 .....	202

<b>第 7 章 目录和文件操作</b> .....	204
7.1 一个简单的文件读写程序 .....	204
7.1.1 创建 C# 窗体应用程序 .....	204
7.1.2 程序结构解析 .....	208
7.2 目录管理 .....	208
7.2.1 目录存在的判断 .....	208
7.2.2 目录的创建和删除 .....	208
7.2.3 当前工作目录的获取 .....	209
7.2.4 目录相关信息的获取 .....	209
7.2.5 目录大小的获取 .....	211
7.3 文件管理 .....	212
7.3.1 文件的复制、移动和删除 .....	213
7.3.2 文件信息的获取和设置 .....	213
7.4 文本文件的读写 .....	215
7.4.1 读文本文件 .....	215
7.4.2 写文本文件 .....	217
7.5 二进制文件的读写 .....	218
7.5.1 写二进制文件 .....	219
7.5.2 读二进制文件 .....	220
7.6 习题 .....	223
<b>第 8 章 ActiveX 控件和自定义组件开发</b> .....	225
8.1 一个简单的 ActiveX 控件的开发 .....	225
8.1.1 创建 ActiveX 控件程序 .....	225
8.1.2 生成和调用 ActiveX 控件 .....	227
8.2 ActiveX 控件 .....	228
8.2.1 什么是 ActiveX 控件 .....	228
8.2.2 ActiveX 控件开发实例 .....	229
8.3 自定义组件 .....	239
8.3.1 创建自定义组件 .....	239
8.3.2 使用自定义组件 .....	241
8.4 习题 .....	244
<b>第 9 章 多线程</b> .....	246
9.1 一个简单的多线程应用程序 .....	246
9.1.1 创建控制台多线程应用程序 .....	246
9.1.2 程序说明 .....	248
9.2 线程及其实现方法 .....	248

9.2.1	线程的概念	248
9.2.2	线程的实现方法	249
9.2.3	线程的优先级	254
9.3	线程的同步控制	255
9.3.1	为什么要同步控制	255
9.3.2	使用 ManualResetEvent 类	257
9.3.3	使用 AutoResetEvent 类	262
9.4	线程池	264
9.5	线程对控件的访问	268
9.6	习题	272
<b>第 10 章</b>	<b>数据库开发技术</b>	<b>274</b>
10.1	一个简单的 C# 数据库应用程序	274
10.1.1	创建数据库和数据表	274
10.1.2	创建数据库应用程序	276
10.1.3	程序结构解析	278
10.2	数据库系统与 ADO.NET 概述	279
10.2.1	数据库系统	279
10.2.2	ADO.NET 概述	280
10.3	SQL 语言简介	280
10.3.1	Select 语句	281
10.3.2	Insert 语句	282
10.3.3	Update 语句	283
10.3.4	Delete 语句	283
10.4	ADO.NET 对象	283
10.4.1	ADO.NET 体系结构	283
10.4.2	Connection 对象	284
10.4.3	Command 对象	284
10.4.4	DataReader 对象	290
10.4.5	DataAdapter 对象	291
10.4.6	DataSet 对象	294
10.5	数据库操作举例	299
10.5.1	数据检索	299
10.5.2	数据添加	301
10.5.3	数据更新	303
10.5.4	数据删除	307
10.6	习题	309

<b>第 11 章 ASP.NET Web 应用开发</b> .....	311
11.1 一个简单的 ASP.NET Web 应用程序 .....	311
11.1.1 创建 Web 应用程序 .....	311
11.1.2 程序结构解释 .....	314
11.2 关于 ASP.NET .....	316
11.3 ASP.NET 控件和对象 .....	317
11.3.1 ASP.NET 控件 .....	317
11.3.2 ASP.NET 常用对象 .....	322
11.4 ASP.NET 数据库应用程序 .....	329
11.4.1 数据库的连接和数据浏览 .....	329
11.4.2 对数据库的增、删、查、改操作 .....	330
11.5 Web 服务的应用 .....	333
11.5.1 什么是 Web 服务 .....	333
11.5.2 Web 服务的创建 .....	334
11.5.3 Web 服务的调用 .....	337
11.6 习题 .....	342
<b>第 12 章 基于数据控件的应用程序开发</b> .....	343
12.1 数据显示控件 .....	343
12.2 DataGridView 控件的结构 .....	343
12.3 DataGridView 控件的属性和方法 .....	344
12.3.1 DataGridView 控件的常用属性 .....	344
12.3.2 DataGridView 控件的常用事件 .....	355
12.4 对 DataGridView 控件加载数据 .....	356
12.4.1 数据绑定 .....	356
12.4.2 数据添加 .....	362
12.5 DataGridView 控件的应用举例 .....	365
12.5.1 在控件中查找 .....	365
12.5.2 在控件中批量删除 .....	367
12.5.3 在控件中使用复选框和单选框 .....	369
12.5.4 控件列的隐藏和添加 .....	376
12.5.5 控件中隔行换色 .....	378
12.5.6 行背景色随鼠标移动变色 .....	378
12.5.7 与导航控件结合使用 .....	381
12.5.8 使用控件操纵数据 .....	383
12.6 GridView 控件的属性和事件 .....	388
12.6.1 一个简单的例子 .....	388
12.6.2 GridView 控件的常用属性 .....	393

12.6.3	行编程与列编程 .....	396
12.6.4	GridView 控件的常用事件 .....	401
12.7	GridView 控件的数据库应用 .....	404
12.7.1	使用 SqlDataSource 对象绑定数据 .....	404
12.7.2	使用 DataSet 对象绑定数据 .....	407
12.8	应重视的问题 .....	420
12.8.1	重复加载问题 .....	420
12.8.2	重复提交问题 .....	421
12.9	习题 .....	425
<b>第 13 章</b>	<b>Excel 数据读写在 Web 开发中的应用 .....</b>	<b>427</b>
13.1	Excel 表的结构 .....	427
13.2	Excel 数据读写方法 .....	428
13.2.1	OleDb 方法 .....	428
13.2.2	COM 组件方法 .....	431
13.2.3	NPOI 方法 .....	437
13.2.4	三种方法的比较 .....	444
13.3	构造不规则 Excel 表 .....	444
13.3.1	字体、样式的设置方法 .....	444
13.3.2	构造不规则 Excel 表的方法 .....	445
13.4	Excel 数据的导入与导出 .....	449
13.4.1	Excel 数据导入和导出的原理 .....	449
13.4.2	面向 Web 数据库应用的数据导入与导出 .....	450
13.5	习题 .....	456
<b>第 14 章</b>	<b>应用程序的发布 .....</b>	<b>458</b>
14.1	关于应用程序的发布 .....	458
14.2	由手工复制文件来发布程序 .....	458
14.2.1	窗体应用程序的发布 .....	459
14.2.2	使用 WinRAR 发布程序 .....	463
14.3	IIS 安装与 Web 应用程序发布 .....	465
14.3.1	在 Windows 7 系统中安装与发布 .....	465
14.3.2	在 Windows XP 系统中安装与发布 .....	475
14.4	使用 .NET 项目来发布程序 .....	477
14.4.1	InstallShield Limited Edition 的下载和安装 .....	478
14.4.2	制作应用程序的安装程序 .....	478
14.5	习题 .....	489
<b>参考文献</b>	.....	<b>490</b>

# 第 1 章

## C#程序设计基础

**主要内容：**本章介绍与 C# 程序开发相关的基本概念和常识，内容包括程序设计语言的定义及其分类、程序设计的方法、C# 语言的起源和特点、C# 集成开发环境、C# 应用程序开发步骤等。

**教学目标：**了解程序设计语言、程序设计方法的相关概念，掌握 C# 应用程序开发的基本步骤。

### 1.1 程序设计语言

#### 1.1.1 程序设计语言的定义

语言(自然语言)是人类在长期劳动过程中形成和发展的、用于思维和传递信息的工具。人之所以具有智能，在很大程度上是因为人能够运用语言进行思考，完成对信息的加工和处理。计算机作为拟人的机器，要实现对人类智能的模拟，也必须拥有自己的语言，这种语言就是程序设计语言。

程序设计语言(Programming Language)是一套遵循既定规则的符号系统；一个计算机程序实际上就是由一些符号按若干规则构成的符号串。程序设计语言包含三方面的内容，即语法、语义和语用。语法就是符号串构成的规则，它表示程序的结构或形式；语义表示语法单位和程序的意义，离开语义，语言只不过是一些符号的集合；语用表示程序与其使用的关系，这种关系将语言的基本概念和语言的外界联系起来。

#### 1.1.2 程序设计语言的分类

在计算机诞生后的发展过程中，程序设计语言也经历了从无到有、从低级到高级的发展历程。相应地，程序设计语言可分为低级程序设计语言和高级程序设计语言，而低级程序设计语言又可以进一步分为机器语言和汇编语言。

##### 1. 机器语言

机器语言是指直接用二进制代码指令表达的计算机语言。它实际上是由 0 和 1 构成的字符串。机器能直接识别和执行的只有机器语言，其他语言要经过编译器翻译为相应的机器语言后才能被执行。

一般来说,用机器语言编写的计算机程序具有较高的执行效率。但机器语言依赖于具体的机型,即不同的机器,其机器语言是不尽相同的,因此其移植性非常差。而且用机器语言编写计算机程序(0-1串)的过程十分烦琐、费时、易出差错,程序调试也十分困难,因此使用机器语言来编写程序是不现实的。

## 2. 汇编语言

针对机器语言存在的缺点,人们对它进行了符号化,使用比较接近自然语言的符号串来表示相应的二进制指令,从而大大减少了直接编写二进制代码带来的烦琐,可使用相对直观、易记的符号串来编写计算机程序,这便促成了汇编语言的形成和发展。

汇编语言是对机器语言中二进制指令进行符号化表示而形成的一种程序设计语言。表示二进制指令的符号通常称为助记符,用助记符编写的程序称为汇编语言程序。汇编语言程序不是用机器语言编写的,因而也不能被机器直接执行。同样,需要将汇编语言程序“翻译”成机器语言,然后机器才能执行它。这种翻译过程称为汇编,这种汇编任务是由称为汇编程序的软件来完成。汇编后形成的机器语言程序称为目标程序,这时被汇编的汇编语言程序又称为源程序。这个过程如图 1.1 所示。

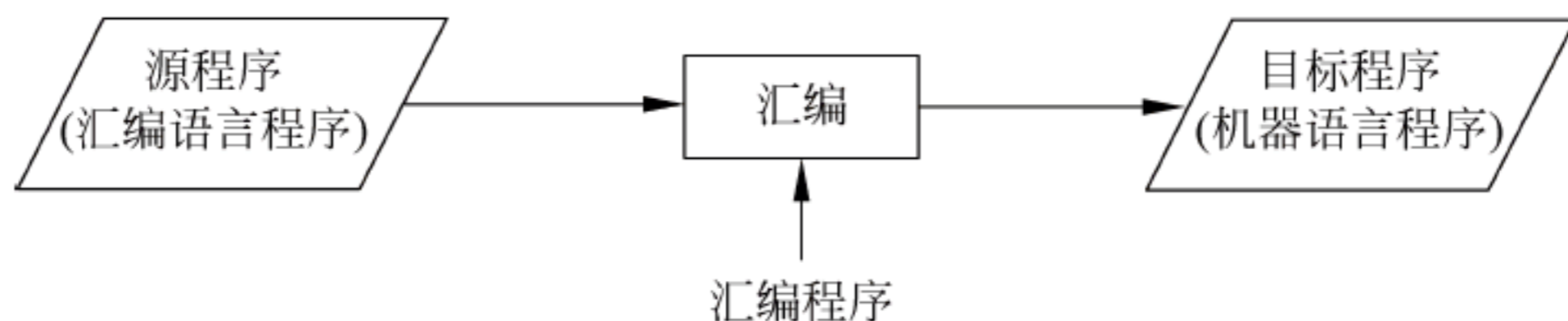


图 1.1 汇编过程

虽然汇编语言比晦涩难懂的机器语言有所改进,也具有机器语言执行速度快等优点,但这种改进是远远不够的。实际上,汇编语言中的指令与机器语言指令几乎是一一对应,其改进之处主要是用助记符来表示机器语言指令,因此编写汇编语言程序仍然需要对机器的组成(主要是寄存器、存储器等)有清晰的了解,汇编语言程序仍然依赖于具体的机型,移植性不强,在编写复杂程序时依然显得烦琐、费时、易错,调试也困难。

## 3. 高级语言

高级语言(即高级程序设计语言)出现于 20 世纪 50 年代中期,此后得到迅速的发展和完善,至今已有上千种高级语言,常使用的也有上百种,如 C/C++、Pascal、FORTRAN、COBOL、Java、BASIC 以及本书要介绍的 C# 等都是常用的高级程序设计语言。

高级语言是由接近自然语言(英语)的词汇(记号)和语法(规则)构成的符号系统。其“高级”之处在于:①它较好地克服了机器语言和汇编语言的不足,采用近似自然语言的符号和语法,大大提高了编程的效率和程序的可读性;②它不依赖于具体机型的指令系统,程序具有很高的可移植性;③编写代码时不需考虑具体的细节,如数据存放到哪里、从哪个存储单元读取数据等,从而使程序员能够把更多的精力集中在问题求解本身的设计当中。

显然,高级语言不能被机器直接执行,而需要专门的程序将由高级语言编写的程序“翻译”成机器语言程序,然后才能被执行。这种翻译和执行的方式有两种:一种是翻译一句执行一句,这种翻译称为解释,相应的翻译程序称为解释程序,如 BASIC 语言就是采用解释执