

```
using System;  
namespace HelloTomorrow  
{  
    class Program  
    {  
        static void Main(string[] args)  
        {  
            Console.WriteLine("Hello tomorrow!");  
        }  
    }  
}
```

叩响 C#之门

◎ 梁斌玉 编著

技术的进步能使普通人完成
过去天才才能完成的任务。
巧妙的讲解能使普通人理解
过去天才才能理解的知识。

```
( class Program  
using System;  
namespace HelloTomorrow  
static void
```



电子工业出版社
PUBLISHING HOUSE OF ELECTRONICS INDUSTRY
<http://www.phei.com.cn>

叩响C#之门

◎ 梁斌玉 编著

電子工業出版社

Publishing House of Electronics Industry

北京·BEIJING

内 容 简 介

本书写给完全没有编程经验的初学者。它从最基本的 C#语法开始,采用逐步构建的学习方法,一步一步地深入到 C#编程的核心概念。当 C#语法基础打扎实后,即进入窗体编程的学习,这样读者就有了比较丰富的编程体验,此时进入委托、线程、程序集等深层次概念的学习,便水到渠成、轻松易懂。本书针对初学者,因为不太久以前作者也是一个初学者,与初学者思维相近,心灵相通,知道初学者的困惑之处,相信学完此书之后,你不但能牢固地掌握 C#的基础语法,更能深刻地体会到 C#的核心思想。

未经许可,不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有,侵权必究。

图书在版编目(CIP)数据

叩响 C#之门 / 梁斌玉编著. —北京: 电子工业出版社, 2009.7
ISBN 978-7-121-09089-9

I. 叩… II. 梁… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2009) 第 100730 号

责任编辑: 顾慧芳

印 刷: 北京天宇星印刷厂

装 订: 三河市皇庄路通装订厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

开 本: 787×1092 1/16 印张: 31 字数: 719.3 千字

印 次: 2009 年 7 月第 1 次印刷

印 数: 4000 册 定价: 59.00 元 (含光盘 1 张)

凡所购买电子工业出版社图书有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系, 联系及邮购电话: (010) 88254888。

质量投诉请发邮件至 zlts@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线: (010) 88258888。

推 荐 序

几个月前，当出版社联系我审校这本书时，我本来是想借故推掉的，但编辑说这本书是一位高中数学老师为初学者写的，花了很长的时间，而且写得很用心，于是，我心动了，不妨看看这本书能否吸引我还是让我感到沉闷。

俗话说：万事开头难。为初学者写书，肩负着带领初学者入门的责任，其难度一点也不亚于为牛人们写提级秘籍。每个人都会经历初学者阶段，然而，随着经验的增长，我们看待问题的角度和层次会逐渐脱离初学者的程度。一方面，这是进步的表现，值得赞许；但另一方面，当初学者找我们探讨问题时，我们看待和描述问题的角度和层次未必是他们能够轻易接受的，其中的差异或许就是所谓的代沟吧，这也正是许多初学者在看高手写的书籍和文章时会感到一头雾水的缘故。于是，我认为，要写好一本入门书籍，首要条件是作者能否置身于初学者的位置，看初学者所看、思初学者所思，只有这样才能深切地体会他们的烦恼，同时又要超越初学者的位置，否则就会变成和他们一起烦闷而不是带领他们走出困惑。

这本书给我的最大震撼莫过于作者以一个初学者的眼光把内容讲得足够详细，又不至于过分高深。在审校的过程中，我也不止一次地感慨：如果给了我，也未必能写得如此深入浅出。是啊，很多时候，当我们非常熟悉某些概念时，我们会倾向于把它们当做理所当然来处理，忽略对它们做进一步的探讨，而这对初学者来说正是造成困惑的一大原因，就这点来说，我也要好好反省了。

如果你是一个 C# 初学者，我相信，你一定会在这本书里找到共鸣，也会从作者身上看到值得借鉴的经验，但更重要的是，你一定要有自己的思考，并且要勤于动手试验自己的想法。我衷心希望这本书能够引你入门，成为一块让你站得更高的砖块。

软件工程师 李永伦

博客地址 <http://allenlooplee.cnblogs.com>

2009 年 7 月 14 日于上海

前 言

这是一部花费五年时间写成的著作，千锤百炼写就、心血凝结而成。

这是一部面向初学者的著作，因为写此书之前我也是一个初学者。正因为如此我懂得初学者在想什么，懂得初学者需要什么与初学者思维相近，心灵相通，故能针对初学者的困惑之处着笔。看完这本书你并不能马上成为绝顶高手，但能帮你快速地掌握编程的基本概念和基本思想，为深入学习打下扎实的基础。我希望你看完这本书后，再也不用为不知如何踏入编程的殿堂而彷徨，再也不用为书籍中繁多的名词术语而迷茫，能够轻松地阅读别人的代码，踏上高手之路。

之所以说是心血凝结而成，是因为这部著作不但耗费了巨大的时间和精力，而且融入了我多年的学习体会和教学思想。

本书的特点

1. 知初学者之需 我是一名高中数学教师，在教学过程中我发现，让学生替我讲述往往会收到很好的效果。究其原因，并非学生讲得多么精彩，而是学生间的思维层次相同，该学生重点讲解之处就是其他学生困惑之处。因此，我在编写本书时，尽力回想自己学习时的体验，以己推人，了解初学者的困惑。幸好，我在学习 C#之初做了大量的笔记，写下了不少困惑和感受，这对我编写本书大有裨益。

2. 逐步构建知识 学习知识有两种基本方法，一种方法是先对整体概述，然后逐步细化。如果学习者对所学领域有一定认识，生活中有一定的体验，就适合这种方法。另一种方法是由最简单的知识出发，逐步添加内容，最后构建成一个整体。如果学习者对所学领域非常陌生，就适合这种方法。编程，现实生活中没有体验，对初学者来说是非常陌生和抽象的事物，所以本书并没有像其他教材那样开篇即对 .NET 框架展开大篇幅论述（成堆的陌生概念会让人望而生畏，我刚接触 C#时就是这种感觉），而是简单介绍后就采用逐步构建的方法来学习。对于每个知识点，先指导学习者编写一些简单小程序，获得感性体验，然后透过现象看本质，讲解其背后蕴涵的道理。随着内容的不断添加，最终构建完整的知识体系。

3. 例子典型简单 要想知道苹果的味道，只需要尝一口即可，要想学好编程，就要不断练习。所以本书通过大量贴切的例子来讲解概念。这些例子精心构思，巧妙设计，为了一个例子，我常常琢磨数月，以确保它典型、简单，一目了然。

4. 结构编排合理 C#语言体系庞大，知识繁杂，概念互相交叉。为此，我在结构编排上花费了很多心思，理清了概念间的顺序，让这些概念按逻辑顺序逐一展现。并且编排中也考虑了读者的兴趣，内容难易交错，跌宕起伏，读者一路学习下来，会很有成就感。

5. 语言简洁通顺 啰里啰唆的行文会妨碍读者理解，因此每段话都我都追求逻辑连贯，含义准确，千锤百炼，字字珠玑，绝无半句废话妨碍读者思维。为了达到这个目的，一些重要段落常常修改数周方才定稿，一些难讲的概念常常花费数月进行构思。我认为花这么多时间是值得的，因为我一个人多花点时间，就可以使成千上万的读者节省很多时间。

为什么写一本书我要花费如此多的心血？这可能和我是追求完美主义的偏执性格有关，看到不完善的地方就不安心，就有一种非改不可的冲动。时至今日每读一遍本书，我都会修改很多词句，觉得还不够十全十美。今天偶然读到一句话：“追求完美是浪费生命，追求卓越才是追求成功。”掩卷回思，感悟颇深，感叹道：此书虽非完美无瑕，应该也可以称得上卓越了。于是决定先出版这本书，奉献给读者，这样总比放在我的电脑里强百倍，你说是不是？

“文王拘而演周易，仲尼厄而作春秋”，批阅五载，增删十次，借物明志。

本书的主要内容

本书分为三个部分，共 23 章。第 I 部分为 C#语法基础，包括第 1 章至第 12 章，深入浅出地介绍了 C#的变量、运算符和表达式、流程控制、数组、函数、面向对象的编程，以及.NET 框架的基本概念等 C#编程的初步知识。第 II 部分为 Windows 编程，包括第 13 章至第 15 章，详细描述了 Windows 窗体编程的方方面面。第 III 部分为 .NET 进阶，包括第 16 章至第 23 章，阐述了 C#的诸多特性，如：委托与事件、绘图、文件和流、字符串和正则表达式、程序集和反射、多线程、集合，以及泛型。

本书光盘的内容

Project\第 I 部分 C#语言

该文件夹中包含了 3 个 Visual Studio 2005 的解决方案。它们对应着本书第 I 部分的相关案例。

Project\第 II 部分 Windows 编程

该文件夹中包含了本书第 II 部分从第 13 章到第 15 章的案例。这些案例均为 Visual Studio 2005 的解决方案。

Project\第 III 部分 .NET 进阶

该文件夹中包含了本书第 III 部分从第 16 章到 23 章的案例。这些案例均为 Visual Studio

2005 的解决方案。

Project\资源

该文件夹中包含了一些资源文件，如图片、图表等文件。

（注意：如果您用 Visual Studio 2005 之前版本的 Visual Studio，打开上面这些解决方案的时候会出问题。）

联系方式

虽然我力求完美，但智识短浅，谬误之处难以避免，欢迎读者批评指正。联系方式为 beginnerClassroom@163.com，博客地址为 <http://www.cnblogs.com/BeginnerClassroom>。

最后感谢李永伦先生对本书的审阅，以及为本书早日出版而付出辛勤劳动的所有人士表示衷心的感谢。

梁斌玉

2008 年 12 月

目 录

第 I 部分 C#语法基础

第 1 章 初识编程	2
1.1 C#简介	2
1.2 Visual Studio 开发环境	3
1.3 认识程序	4
1.3.1 创建项目	4
1.3.2 编写代码	5
1.3.3 运行程序	7
1.4 编译	7
1.5 二进制	8
1.5.1 二进制转换为十进制	9
1.5.2 十进制转换为二进制	10
1.6 数据和内存	13
第 2 章 变量	15
2.1 整型变量	16
2.1.1 声明整型变量	16
2.1.2 int 型变量的取值范围	20
2.1.3 short 型	21
2.1.4 long 型	21
2.1.5 无符号型	21
2.1.6 整数类型小结	21
2.1.7 溢出	22
2.2 实数型变量	23
2.3 字符	24
2.3.1 字符和字符串	25
2.3.2 字符型变量和字符串变量	26
2.3.3 字符串的输出	27
2.3.4 字符串的读取	27
2.3.5 ASCII 编码和 Unicode 编码	28
2.3.6 转义字符	32
2.3.7 @控制符	33

2.3.8	+运算符	33
2.4	变量的格式化输出	34
2.5	变量的命名规则	37
2.6	常量	38
2.7	关键字 (Keyword)	39
2.8	深化练习: 交换两个变量的值	40
第 3 章	运算符和表达式	43
3.1	算术运算符	44
3.2	自增、自减运算符	46
3.3	赋值运算符	48
3.4	优先级	49
3.5	类型转换	50
3.5.1	隐式转换	50
3.5.2	显式转换	52
3.5.3	字符串和数值间的转换	53
第 4 章	流程控制	55
4.1	算法的概念	56
4.2	顺序结构	57
4.3	逻辑表达式	59
4.3.1	关系运算符	59
4.3.2	逻辑运算符	61
4.3.3	布尔变量	63
4.4	选择结构	64
4.4.1	if 语句	64
4.4.2	switch 语句	70
4.4.3	条件表达式	73
4.4.4	深化练习: 三个变量排序	74
4.5	循环结构	75
4.5.1	while 语句	75
4.5.2	do-while 语句	80
4.5.3	for 语句	82
4.6	break 语句和 continue 语句	84
4.7	循环语句深化练习: 分形图	85
第 5 章	枚举、结构体和数组	88
5.1	枚举	88
5.2	结构体	92
5.3	数组	94
5.3.1	一维数组	94

5.3.2	foreach 循环语句	97
5.3.3	二维数组	97
5.3.4	可变数组	99
第 6 章	函数	101
6.1	为什么需要函数	101
6.2	定义函数	102
6.3	返回值	105
6.4	参数	107
6.4.1	值传递	107
6.4.2	地址传递	108
6.4.3	引用型参数 (ref)	109
6.4.4	输出型参数 (out)	111
6.4.5	参数匹配	112
6.5	递归调用	113
6.6	变量的作用域	116
6.6.1	局部变量	116
6.6.2	程序块中的局部变量	117
第 7 章	面向对象编程：类和对象（一）	119
7.1	背景	119
7.2	面向对象的基本概念	120
7.3	定义类	122
7.4	声明对象	125
7.5	属性	127
7.6	构造函数	129
7.7	析构函数和垃圾回收	132
7.8	实例演练：Time 类	133
第 8 章	面向对象编程：类和对象（二）	137
8.1	以对象为成员	137
8.2	静态成员	138
8.2.1	静态变量	138
8.2.2	静态函数	141
8.3	常量成员	142
8.3.1	const 常量	142
8.3.2	readonly 常量	144
8.4	重载 (Overload)	146
8.4.1	函数重载	146
8.4.2	构造函数的重载	148
8.4.3	运算符重载	149

8.5	this 关键字	153
8.6	索引	154
8.7	值类型和引用类型	157
8.7.1	值类型变量	158
8.7.2	引用型变量	159
8.7.3	引用型变量和垃圾回收器	160
8.8	引用符和对象的区别	160
8.9	声明对象数组	162
8.10	类视图	162
第 9 章	面向对象编程：继承	163
9.1	继承概述	163
9.2	由基类创建派生类	164
9.3	protected 成员	167
9.4	虚函数的重写	170
9.5	普通函数的隐藏	172
9.6	base 关键字	173
9.7	抽象类和抽象函数	174
9.8	密封类和密封函数	175
9.9	派生类的构造函数	176
9.10	万类之源：Object	180
第 10 章	面向对象编程：多态性	182
10.1	派生类的对象和基类的关系	182
10.2	多态性的概念	182
10.3	多态性的应用	183
10.4	is 运算符	186
10.5	向下类型转换	187
10.6	接口	188
10.7	类关系图	192
第 11 章	.NET 框架基本概念	193
11.1	如何理解.NET 框架	193
11.2	C#和.NET 的关系	194
11.3	中间语言	194
11.4	强数据类型	195
11.5	类型的判定	197
11.5.1	sizeof 运算符	197
11.5.2	typeof 运算符	197
11.5.3	GetType()函数	197
11.5.4	is 运算符	198

11.6	命名空间	199
11.7	装箱和拆箱	202
11.8	对象的相等	204
第 12 章	异常	208
12.1	异常概述	208
12.2	捕获异常——try-catch 结构	209
12.3	finally 块——try-catch-finally 结构	210
12.4	抛出异常——throw 语句	211
12.5	.NET 中的异常类	212
12.6	异常类的属性	214
12.7	自定义异常	215
第 II 部分 Windows 编程		
第 13 章	Windows 窗体编程（一）	218
13.1	窗体（Form）	218
13.2	按钮（Button）	223
13.3	标签（Label）	226
13.4	超链接标签（LinkLabel）	227
13.5	文本框（TextBox）	228
13.6	单选按钮（RadioButton）	230
13.7	复选框（CheckBox）	231
13.8	数字输入框（NumericUpDown）	233
13.9	群组框（GroupBox）	235
13.10	面板（Panel）	237
第 14 章	Windows 窗体编程（二）	245
14.1	鼠标事件处理	245
14.2	键盘事件处理	249
14.3	图形框（PictureBox）	252
14.4	列表框（ListBox）	254
14.5	复选列表框（CheckListBox）	256
14.6	下拉式列表框（ComboBox）	259
14.7	列表视图（ListView）	261
14.8	树状视图（TreeView）	266
14.9	控件布局	270
第 15 章	Windows 窗体编程（三）	277
15.1	菜单	277
15.1.1	主菜单（MainMenu）	277

15.1.2 关联菜单 (ContextMenu)	284
15.2 工具栏 (ToolStrip)	285
15.3 对话框	289
15.3.1 另存为对话框 (SaveFileDialog)	289
15.3.2 打开文件对话框 (OpenFileDialog)	291
15.3.3 字体对话框 (FontDialog)	293
15.3.4 颜色对话框 (ColorDialog)	294
15.3.5 自定义对话框	295
15.4 多文档界面	296
15.5 Windows 窗体类的层次结构	302

第Ⅲ部分 .NET 进阶

第 16 章 委托与事件	306
16.1 委托	306
16.2 多播委托	311
16.3 匿名函数	312
16.4 事件处理机制	313
16.4.1 事件处理机制的原理	313
16.4.2 自定义事件	316
第 17 章 绘图	321
17.1 概述	321
17.2 像素和坐标系	322
17.3 颜色	322
17.4 Graphics 类	324
17.5 画笔 (Pen)	325
17.6 画刷 (Brush)	326
17.7 路径 (Path)	329
17.8 OnPaint()方法	331
17.9 坐标变换	332
17.9.1 平移变换	332
17.9.2 旋转变换	334
17.9.3 伸缩变换	335
17.9.4 在滚动窗口中绘图	336
17.10 字体 (Font)	341
17.11 显示图像	344
17.11.1 Image 类	344
17.11.2 Bitmap 类	345
17.11.3 Metafile 类	346

第 18 章	文件和流	348
18.1	文件系统	348
18.1.1	File 类和 FileInfo 类	348
18.1.2	关于文件的异常	350
18.1.3	Directory 类和 DirectoryInfo 类	351
18.1.4	Path 类	353
18.1.5	Environment 类	354
18.2	基于流的文件操作	356
18.2.1	FileStream 类	357
18.2.2	关于流的异常	359
18.2.3	用流读写文本文件	363
18.2.4	用流读写二进制文件	365
第 19 章	字符串和正则表达式	368
19.1	String 类	368
19.1.1	复制字符串	369
19.1.2	比较字符串	371
19.1.3	验证字符串首尾	373
19.1.4	定位字符或子串	374
19.1.5	截取子串	377
19.1.6	拆分字符串	378
19.1.7	更改大小写	378
19.1.8	修改字符串	379
19.1.9	插入格式化变量	379
19.2	StringBuilder 类	380
19.3	Char 类	384
19.4	正则表达式	385
19.4.1	通配符	385
19.4.2	可选字符集	386
19.4.3	或	387
19.4.4	非	388
19.4.5	数量限定符	388
19.4.6	定位符	393
19.4.7	Regex 类的函数	395
19.4.8	深化练习：验证用户输入	397
第 20 章	特性、程序集和反射	398
20.1	特性	398
20.1.1	使用特性	398
20.1.2	自定义特性	400

20.2	程序集	404
20.2.1	认识程序集	405
20.2.2	程序集的结构	412
20.2.3	私有程序集和共享程序集	413
20.2.4	程序集的特性	413
20.3	反射	416
20.3.1	获取类型信息 (Type 类)	416
20.3.2	获取程序集信息 (Assembly 类)	418
20.3.3	动态加载类型	419
第 21 章	多线程	422
21.1	线程的概念	422
21.2	Thread 类	423
21.3	线程的优先级	427
21.4	线程的插入	431
21.5	线程的状态	433
21.6	线程的同步	434
21.6.1	线程同步的概念	434
21.6.2	互锁 (Interlocked 类)	436
21.6.3	管程 (Monitor 类)	438
21.6.4	互斥体 (Mutex 类)	443
21.6.5	死锁	446
21.7	线程池	449
第 22 章	集合	453
22.1	集合概述	453
22.2	Array 类	454
22.3	ArrayList 类	455
22.4	Stack 类	459
22.5	Queue 类	461
22.6	SortedList 类	463
22.7	Hashtable 类	465
第 23 章	泛型	473
23.1	泛型的概念	473
23.2	泛型集合类	476
23.2.1	List<T> 类	477
23.2.2	Stack<T> 类	477
23.2.3	Queue<T> 类	478
23.3	类型约束	478

第 I 部分 C#语法基础

- 第 1 章 初识编程
- 第 2 章 变量
- 第 3 章 运算符和表达式
- 第 4 章 流程控制
- 第 5 章 枚举、结构体和数组
- 第 6 章 函数
- 第 7 章 面向对象编程：类和对象（一）
- 第 8 章 面向对象编程：类和对象（二）
- 第 9 章 面向对象编程：继承
- 第 10 章 面向对象编程：多态性
- 第 11 章 .NET 框架基本概念
- 第 12 章 异常

初 识 编 程

欢迎同学们来到奇妙的编程世界！自从 1946 年世界第一台电子计算机诞生以来，软件技术获得了飞速的发展，形成了众多经典的计算机语言。2000 年 6 月，微软推出了 .NET 战略（下一代互联网软件和服务战略），软件设计达到了一个全新的境界。C#语言是 .NET 平台的主打语言，寄托着微软对未来的希望，C#都有些什么特点呢？那么我们就带着这些疑问走进 C#的世界吧。

1.1 C#简介

20 世纪 70 年代，人们设计出了高效灵活的 C 语言，后来又在 C 语言中加入了面向对象的技术形成 C++语言。最近几年，C++一直是最有生命力的程序设计语言，它为我们带来了强大的功能和高度的灵活性，但正是由于 C++的复杂和灵活，使我们必须忍受艰苦的学习过程和漫长的开发周期，而且内存泄露和缓冲区溢出的痼疾也使很多初学者头痛不已。因此，许多程序员一直在寻求一种新的语言，以图在开发能力和学习难度之间取得更好的平衡。

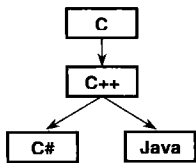


图 1-1 C#语言源于 C++

C#语言的出现解决了这种问题。C#语言源于 C++（如图 1-1 所示），吸取了 C++的长处，摒弃了 C++的缺点，增加了很多易用的新特性，同时剔除了 C++中一些复杂和容易出错的语法，让碍事的东西统统走开，这使 C#在功能强大的同时保持简单易学，鱼和熊掌兼得。C#语言的特点如下。

1. 简洁流畅的语法

随着 C++的不断扩充和发展，语法冗余成了 C++挥之不去的弊端，类似的功能往往有多种表示方法，它们之间只有细微的区别，这些难以记忆的东西不但增加学习难度，而且非常容易出错。C#进行了大刀阔斧的改革，相似的功能只保留最常用的一种，让鸡毛蒜皮的东西走开。简洁流畅成了 C#吸引人们目光的资本，易学易用成了 C#鹤立鸡群的特征。