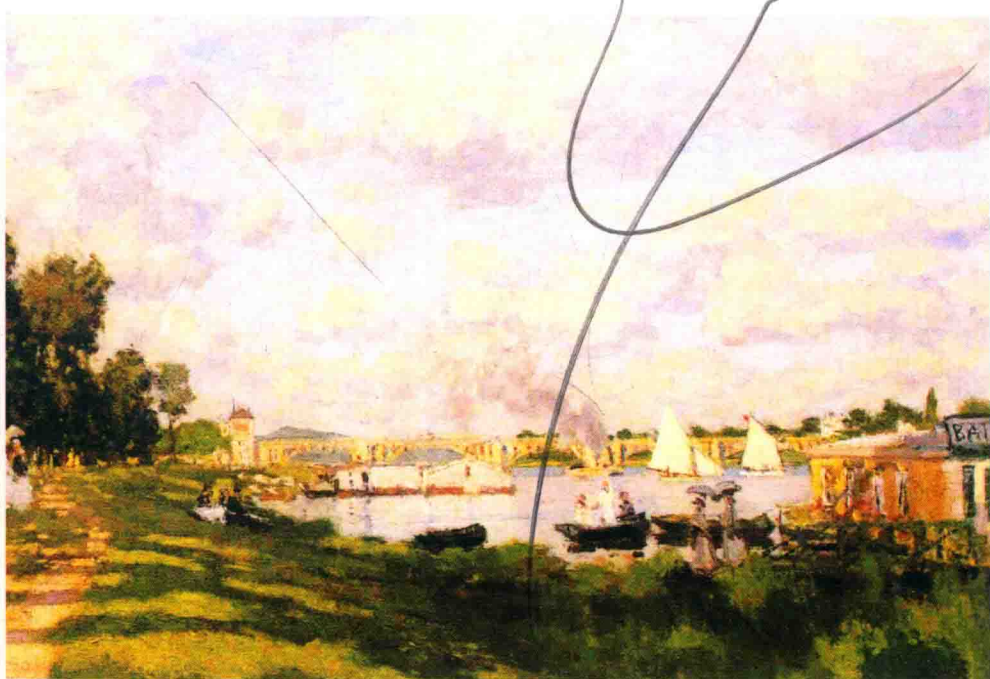




21世纪高等学校计算机
基础实用规划教材

计算机程序设计实践教程 ——C#语言

◎ 李利明 刘卫国 主编



清华大学出版社





21世纪高等学校计算机
基础实用规划教材



计算机程序设计实践教程 ——C#语言

◎ 李利明 刘卫国 主编



清华大学出版社
北京

内 容 简 介

本书是学习 C# 程序设计的教学参考书,全书包含 Visual Studio 2015 集成开发环境、实验指导、常用算法设计、习题选解和模拟试题 5 个部分。前两部分帮助读者熟悉上机环境,方便读者上机操作,通过上机实验更好地掌握 C# 程序设计的基本思想和方法。常用算法设计部分针对初学者学习程序设计的难点,总结常见问题的编程思路,帮助读者提高程序设计能力。习题选解部分通过习题练习,帮助读者复习和掌握课程内容,达到巩固和提高的目的。模拟试题部分包括 3 套模拟试题和参考答案,帮助读者检验学习效果。

本书内容丰富,实用性强,适合作为高等学校计算机程序设计课程的教学用书,也可供各类计算机应用人员阅读参考。

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

计算机程序设计实践教程: C# 语言/李利明,刘卫国主编. —北京:清华大学出版社,2018
(21 世纪高等学校计算机基础实用规划教材)
ISBN 978-7-302-48776-0

I. ①计… II. ①李… ②刘… 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(2017)第 272832 号

责任编辑:刘 星

封面设计:刘 键

责任校对:徐俊伟

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:北京鑫海金澳胶印有限公司

经 销:全国新华书店

开 本:185mm×260mm

印 张:14.75

字 数:372 千字

版 次:2018 年 1 月第 1 版

印 次:2018 年 1 月第 1 次印刷

印 数:1~1000

定 价:39.00 元

产品编号:077575-01

前言

计算机程序设计能够体现问题求解方法,是理解计算机工作过程的有效途径,也是计算思维能力培养的重要载体。因此,计算机程序设计课程的重要性不仅体现在一般意义上的程序设计能力的培养,而且体现在引导学生实现问题求解的思维方式的转换,即学生计算思维能力的培养。在这个过程中,上机实践是十分重要的环节。

C#语言是常用的程序实现工具,程序设计能力需要通过大量的上机实践来培养。许多程序设计方法不是光靠听课和看书就能学到的,而是通过大量的上机实践积累起来的,所以,学习程序设计不能仅限于纸上谈兵,而必须以实践为重。本书是学习C#程序设计的教学参考书,包含5部分内容。

一是Visual Studio 2015集成开发环境。要上机运行一个C#程序,需要C#语言编译系统的支持。这部分介绍了Visual Studio 2015集成开发环境的使用方法,这是上机操作的基础。

二是实验指导。为方便读者上机操作,在这部分设计了12个实验,每个实验都和课程学习内容相配合,以帮助读者通过上机实验加深对课程内容的理解,更好地掌握程序设计的基本思想和方法。实验内容以编写程序练习为主,分为“模仿编程实验”和“独立编程实验”。“模仿编程实验”给出了程序的主体部分,要求将程序补充完整;“独立编程实验”则要求读者独立完成编程练习。

三是常用算法设计。面向对象程序设计的核心是从需要解决的问题中抽象出合适的类,并将数据和对数据的操作方法封装在类的内部。尽管面向对象程序设计的设计思想不同于结构化程序设计,但两者并不是对立的,在类的内部实现仍然要用到结构化程序设计的知识。所以在学习C#程序设计时,算法设计仍然是不能忽视的问题。这部分内容根据程序设计教学基本要求,将常见的程序设计问题进行分类,分别总结每一类问题的算法设计思路,以引导读者掌握基本的程序设计方法和技巧。教学实践表明,这对提高初学者的程序设计能力是很有帮助的。

四是习题选解。这部分以课程学习为线索,编写了十分丰富的习题并给出了参考答案。在使用这些题解时,应重点理解和掌握与题目相关的知识点,而不要死记答案,应在阅读教材的基础上再来做题,通过做题达到强化、巩固和提高的目的。

五是模拟试题。这部分包括3套C#程序设计的模拟试题和参考答案,涵盖了本课程的主要知识点,可以帮助读者了解和检验学习情况。

本书内容丰富,实用性强,适合作为高等学校计算机程序设计课程的教学用书,也可供各类计算机应用人员阅读参考。

本书第1、2、4、5章由李利明编写,第3章由刘卫国编写。此外,周肆清、周欣然、曹岳

辉、蔡旭晖、李小兰、吕格莉、刘胤宏等参与了部分编写工作。清华大学出版社的工作人员对本书的策划、出版做了大量工作，在此表示衷心的感谢。

由于编者水平有限，书中难免存在不足之处，恳请广大读者批评指正。

编 者

2017 年 10 月

目 录

第 1 章 Visual Studio 2015 集成开发环境	1
1.1 Visual Studio 2015 的安装与启动	1
1.2 Visual Studio 2015 主窗口的组成	3
1.3 Visual Studio 2015 下创建 C# 控制台应用程序	5
1.4 Visual Studio 2015 程序调试	7
第 2 章 实验指导	9
实验 1 程序的运行环境和步骤	9
实验 2 C# 语言基础	11
实验 3 程序流程控制(一)	14
实验 4 程序流程控制(二)	18
实验 5 面向对象编程(一)	22
实验 6 面向对象编程(二)	29
实验 7 复杂数据表示与应用	35
实验 8 Windows 窗体与控件	40
实验 9 用户界面设计	45
实验 10 文件操作	50
实验 11 图形与图像处理	53
实验 12 数据库应用	60
第 3 章 常用算法设计	64
3.1 累加与累乘问题	64
3.2 数字问题	69
3.3 数值计算问题	73
3.4 数组的应用	76
3.5 静态方法的应用	81
3.6 解不定方程	83
思考题及答案	85
第 4 章 习题选解	88
习题 1 C# 语言概述	88

习题 2 C# 程序的数据描述	89
习题 3 程序流程控制	93
习题 4 面向对象编程基础	104
习题 5 面向对象高级编程	112
习题 6 复杂数据表示与应用	122
习题 7 Windows 窗体与控件	131
习题 8 用户界面设计	140
习题 9 文件操作	146
习题 10 图形与图像处理	154
习题 11 数据库应用	160
参考答案	167
第 5 章 模拟试题	208
模拟试题 1	208
模拟试题 2	213
模拟试题 3	218
参考答案	224
参考文献	228

第 1 章

Visual Studio 2015 集成开发环境

运行 C# 程序需要相应编译系统的支持。Visual Studio 2015(以下简称 VS 2015)是微软公司提供的—个集成开发环境,即源程序的输入、修改及编译、运行都可以在同一环境下完成,功能齐全,操作方便。VS 2015 可以支持 C/C++、VB、Java、C# 编程,然而—次只能支持—种编程方式。在 VS 2015 安装完成后,第—次运行时会让用户选择常用语言,如果选择 C#,那么就—成了 C# 语言的编程环境,即 VC# 2015 集成开发环境。VS 2015 包含的功能十分丰富,本章只介绍—些常用的操作,以方便读者在 VS 2015 环境下编写 C# 程序。

1.1 Visual Studio 2015 的安装与启动

在启动 VS 2015 之前,首先要安装 VS 2015。VS 2015 的安装方法和其他 Windows 程序的安装方法类似。运行 VS 2015 的安装文件 setup.exe,启动安装程序后,根据屏幕提示依次确定有关内容,便可完成系统安装。

启动 VS 2015 的过程十分简单。常用的方法是,在 Windows 桌面选择“开始”| Visual Studio 2015 命令,即可启动 VS 2015 系统。启动成功后,屏幕上出现如图 1-1 所示的 VS 2015 主窗口。

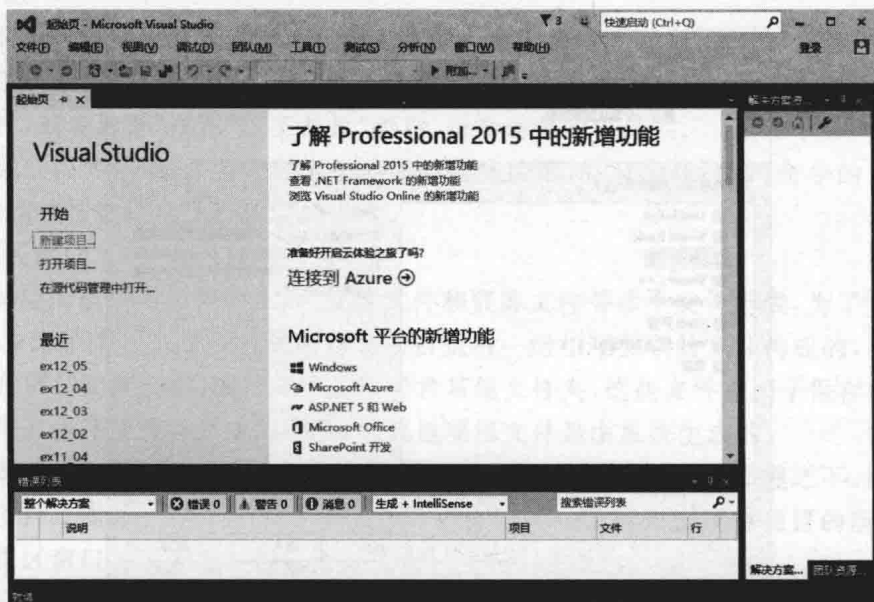


图 1-1 VS 2015 主窗口

在首次使用时,选择默认环境为“Visual C# 开发设置”,这样就可
 在 VS 2015 中配置开发 C# 程序所需要的工具。

如果不是第一次运行 VS,在以前选择了另一个选项,为了使用 C# 环境,要把设置重置为 Visual C# 开发环境:选择“工具”|“导入导出设置”菜单,在弹出的对话框中选择“重置所有设置”选项,如图 1-2 所示。

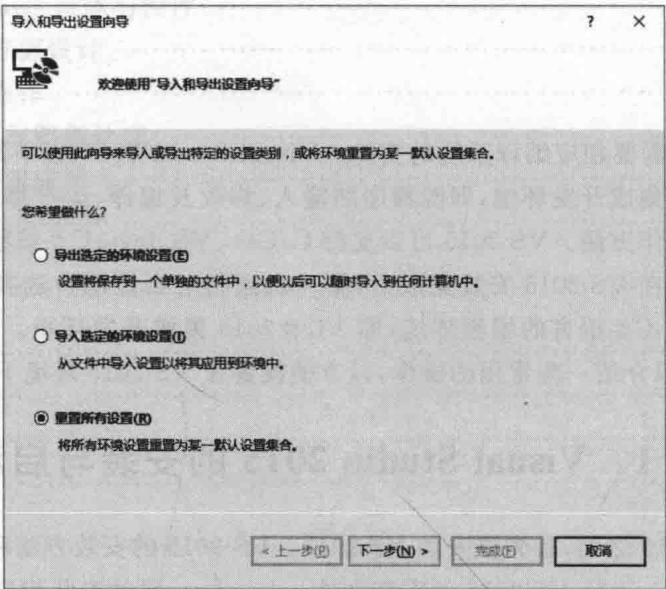


图 1-2 导入和导出设置向导——重置

单击“下一步”按钮,选择是否要保存已有的设置。如果之前对设置进行了定制,就保存设置;否则单击“否”按钮,再次单击“下一步”按钮。在下一个对话框中,选择 Visual C# 选项,如图 1-3 所示。最后单击“完成”按钮,完成设置。

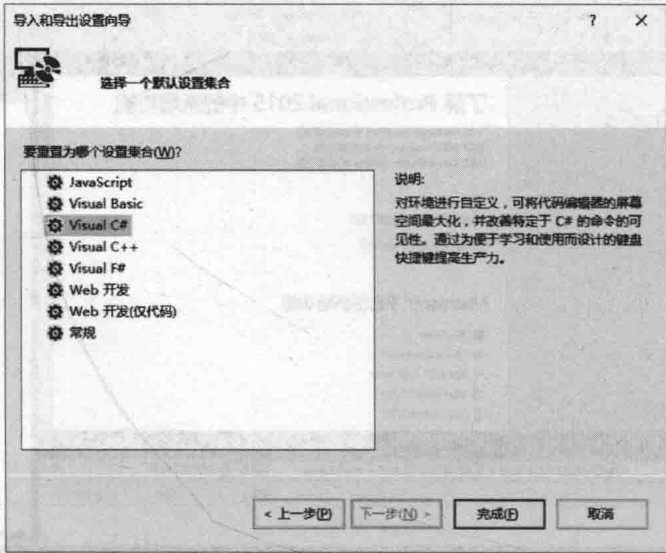


图 1-3 导入和导出设置向导——默认语言设置

VS 2015 环境布局是完全可定制的,其布局如图 1-1 所示,一般用默认设置是比较合适的。

1.2 Visual Studio 2015 主窗口的组成

和其他 Windows 程序一样,VS 2015 主窗口也具有标题栏、菜单栏和工具栏。标题栏的内容是“项目名称-Microsoft Visual Studio”。菜单栏提供了编辑、运行和调试 C# 程序所需要的命令。工具栏是一些命令的快捷按钮,单击工具栏上的按钮,即可执行该按钮所代表的操作。

在 VS 2015 主窗口的右侧是工作区窗口,左侧是起始页,下方是输出窗口。工作区窗口用于显示所设置的工作区的信息;起始页随着打开文件、建立项目等操作,将以分页方式显示文件,并用于输入和修改源程序;输出窗口用于显示程序编译、运行和调试过程中出现的状态信息。

1. 菜单栏

在起始状态下,VS 2015 的菜单栏共有文件(File)、编辑(Edit)、视图(View)、调试(Debug)等 10 个菜单项。在建立项目后,菜单栏共有文件(File)、编辑(Edit)、视图(View)、项目(Project)、生成(Build)、调试(Debug)等 12 个菜单项,每个菜单项都有下拉菜单,下拉菜单中的每个命令执行不同的功能。

“文件”菜单项包含用于对文件进行各种操作的命令,“编辑”菜单项包含所有与文件编辑操作有关的命令,“视图”菜单项包含用于集成开发环境各个工作窗口的显示、打开、切换的各种命令,“项目”菜单项包含用于管理项目和工作区的一系列命令,“生成”菜单项包含用于编译、连接(生成)等的命令,“调试”菜单项包含执行程序、分步执行、设立断点、建立监视项等命令。还有很多其他菜单项,实现各自功能。例如,“窗口”菜单项用于设置 VS 2015 集成开发环境中窗口的属性,“帮助”菜单项提供了详细的帮助信息。

2. 工具栏

VS 2015 集成开发环境提供了十几种工具栏。在一般情况下,系统只显示标准工具栏。要使用其他工具栏,可以通过在主窗口右击菜单栏或工具栏,在弹出的快捷菜单中选择需要显示的工具栏,或者选择“视图”|“工具栏”菜单,选择需要的工具栏。

工具栏包含很多按钮,只要把鼠标指针指向这些按钮,并且稍作停留,命令的名称就显示出来,单击这些按钮就会执行相应的操作。

3. 工作区窗口

一个 C# 应用程序由源程序文件、配置文件和资源文件等多个文件组成,为了更好地管理这些文件,VS 2015 引入了项目的概念。项目是由一组相互关联的文件构成的,项目的所有文件都放在项目文件夹中,项目文件夹也包含其他文件夹,这些文件夹用于保存编译和连接的输出结果。程序员通常只编写源程序,其他项目文件是由系统生成的。

VS 2015 以工作区的形式来组织文件和项目,即项目置于工作区的管理之下,因而工作区通常称为项目工作区。在 VS 2015 中工作区窗口以树状结构列出当前项目的所有文件。用户通过工作区窗口可以方便地操作这些文件。

工作区窗口通常包括 4 个选项卡,即解决方案资源管理器、类视图、属性管理器和团队资源管理器。在窗口底端单击相应图标选项卡可在 4 个选项卡之间切换,用户可以选择不

同的方式操作项目。

1) 解决方案资源管理器

在解决方案资源管理器中列出了项目中的所有文件和文件夹。通常一个大型的应用程序,有可能包含多种类型的项目。为了更好地组织同一个应用程序的多个项目,微软公司提出了解决方案的概念。VS 2015 的解决方案资源管理器提供了管理多个项目的的能力,图 1-4 是包含一个项目的解决方案。当新建一个项目时,除非将项目添加到一个已存在的解决方案,否则系统将会自动生成一个新的解决方案。一个解决方案的一个项目或多个项目的所有信息保存在 .sln 文件中。

2) 类视图

类视图用来显示当前工作区中所有类和类的成员。类视图选项卡提供了项目中所有类及其成员的层次列表。通过单击列表左侧的加号(+)或减号(-)图标可以扩展或折叠列表。双击列表开头靠近文件夹或书本形状图标的文字也可以扩展或折叠列表。

在层次列表的每个项目前面都有一个特殊的图标。例如,保护成员或私有成员的图标是一个钥匙,全局变量是一个紫色图标。当双击某个类或成员的图标时,在编辑窗口将打开对应的代码。

用户在某一个列表项目名上右击时,将弹出一个快捷菜单,从中可以选择要执行的命令。右击的项目名不同,快捷菜单中的命令也就不同。

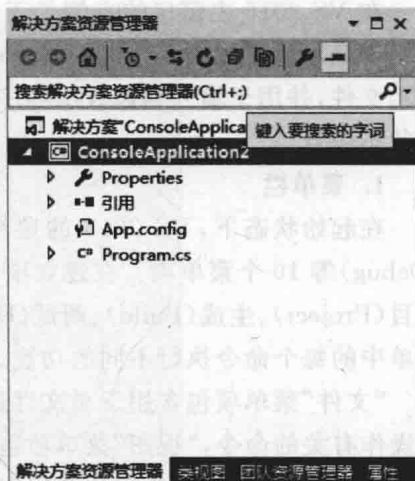


图 1-4 解决方案资源管理器

3) 属性管理器

在属性管理器中,可以对程序编译进行一些设置,一般采用默认设置。

4) 团体资源管理器

团队资源管理器用于管理分配给用户和团队的工作,并与其他团队成员协调工作。

4. 文档窗口

在 VS 2015 中可以编辑多种不同类型的文件,如 C# 源程序文件、文本文件和 HTML 文件等,每种类型的文件都具有一个默认的编辑器。当用户在解决方案资源管理器中双击相应的文件时,将使用默认的编辑器打开文件。也可在解决方案资源管理器中右击相应的文件,选择“打开方式”命令,将弹出打开方式窗口,可以在列表中选择其他的编辑器或添加新的编辑器。

VS 2015 编辑器除了具有复制、查找、替换等一般文本编辑器的功能外,还具有很多特色功能,如根据 C# 语法将不同元素按照不同颜色显示,根据合适长度自动缩进等。文本编辑器还具备自动提示的功能。当用户输入程序代码时,文本编辑器会显示对应的成员函数和变量,用户可以在成员列表中选择需要的成员,这样既可以减少输入工作量,又可以避免手动输入错误。

5. 输出窗口

输出窗口主要用于显示编译、调试结果以及文件的查找信息等。

1.3 Visual Studio 2015 下创建 C# 控制台应用程序

1. 创建项目

在 VS 2015 主窗口中选择“文件”|“新建”|“项目”命令,这时屏幕出现“新建项目”对话框,在其左边的模板选择区域选择 Visual C# 选项,在中间的项目类型选择区选择“控制台应用程序”选项,如图 1-5 所示。现假设建立一个名为 ex1_01 的项目,则在“新建项目”对话框下方的名称编辑框输入 ex1_01,位置编辑框指出项目文件存放的位置,此处输入 e:\,然后单击“确定”按钮。

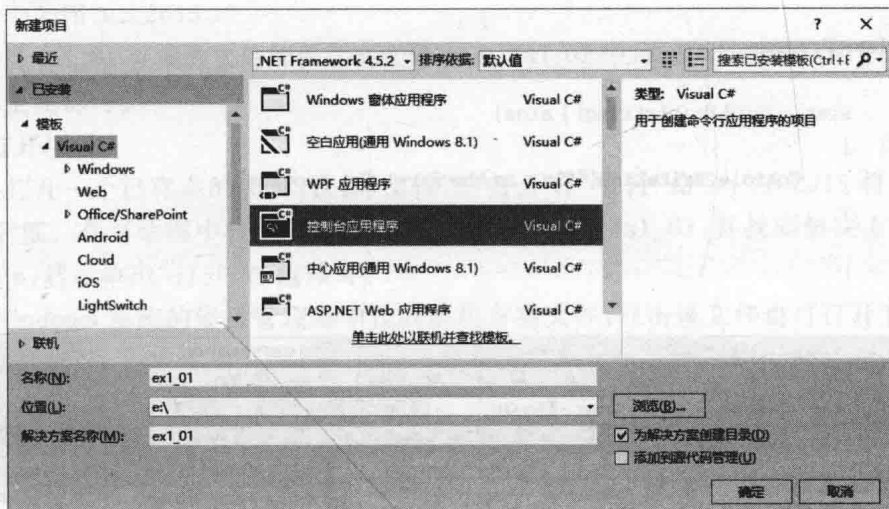


图 1-5 “新建项目”对话框

进入 VS 2015 主窗口,系统建立了一个解决方案的框架。解决方案资源管理器中以树状形式显示解决方案的结构,如图 1-6 所示。

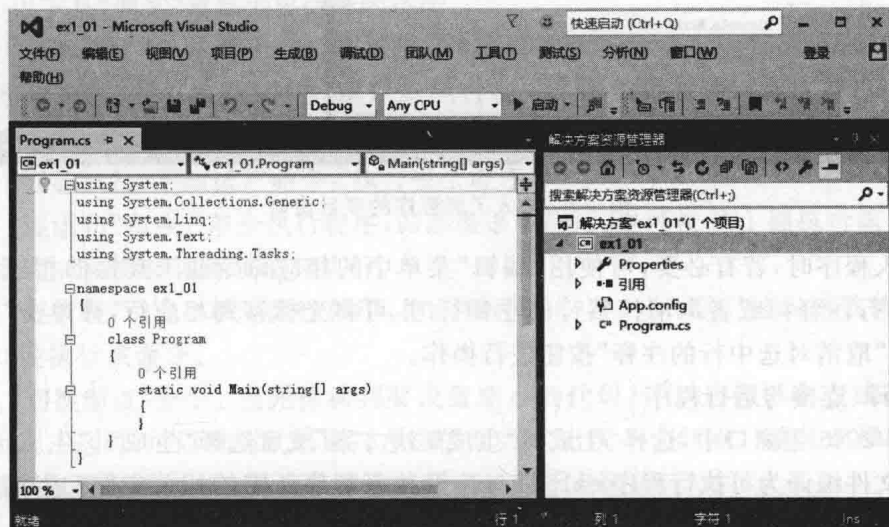


图 1-6 ex1_01 空白项目

在图 1-6 所示的解决方案管理器窗口中出现了一个名为 ex1_01 的默认解决方案,在其下方有项目 ex1_01 及其有关信息项的图标。一个解决方案可包含一个或多个项目,对于初学者来说,最好让一个解决方案只包含一个项目。VS 2015 总是将项目置于某个解决方案中进行加工处理。此时 VS 2015 在文件夹 e:\中生成 ex1_01 文件夹,在 e:\ex1_01 中生成多个文件,其中 ex1_01.csproj 就是项目文件,ex1_01.sln 就是解决方案文件。

2. 输入 C# 源程序代码

在图 1-7 所示的界面,将光标移到左边的代码编辑窗口,在 Main 函数内输入如下程序代码(黑体字部分):

```
namespace ex1_01
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("This is the first C# program");
        }
    }
}
```

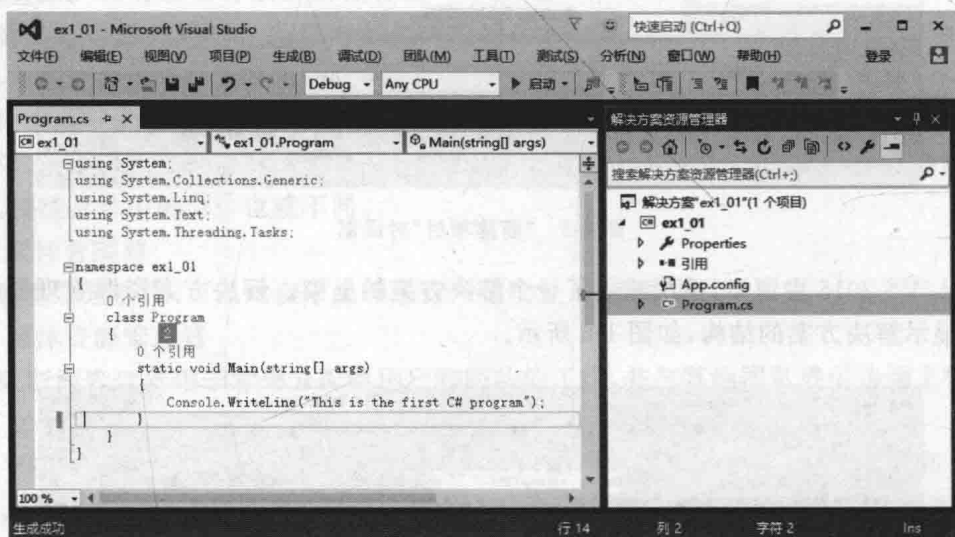


图 1-7 输入了源程序的项目窗口

在输入程序时,若有必要,可使用“编辑”菜单中的相应命令或工具栏的相应按钮。例如,要将某行注释掉或者取消掉某行的注释行时,可将光标移到相应行,再单击“注释选中行”按钮或“取消对选中行的注释”按钮进行操作。

3. 编译、连接与运行程序

在 VS 2015 主窗口中,选择“生成”|“生成解决方案”或者选择“生成”|“生成…”菜单项可将项目文件编译为可执行程序 ex1_01.exe 及执行程序所需的相关文件。生成操作也可按快捷键 F6 或者 Shift+F6 完成。文件 ex1_01.exe 及相关文件均位于文件夹 e:\ex1_01\ex1_01\bin\debug 中。

当编译或连接过程中出现错误,会在输出窗口提示错误信息,用户可根据错误提示信息对源程序进行修改,再重新生成直至没有错误信息提示。

编译、连接成功后,选择“调试”|“开始执行”命令运行 ex1_01.exe 文件,输出结果如图 1-8 所示。

执行操作也可用快捷键 F5 或 Ctrl+F5 或“标准”工具栏或“调试”工具栏中的“启动”按钮  完成。

操作完成后,选择“文件”|“关闭解决方案”命令关闭当前解决方案,或选择“文件”|“退出”命令关闭 VS 2015。

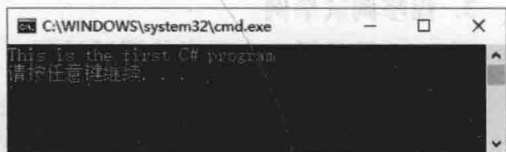


图 1-8 ex1_01.exe 文件运行结果

与项目 ex1_01 有关的文件都保存在文件夹 e:\ex1_01 中,要复制该项目,将 e:\ex1_01 文件夹整体复制即可。

4. 打开项目

若要打开一个已存在的项目,选择“文件”|“打开”|“项目”命令,VS 2015 将弹出“打开项目”对话框。在对话框中找到项目所在的文件夹,如 e:\ex1_01,并找到解决方案文件,如 ex1_01.sln,然后单击“打开”按钮即可。

在 Windows 系统的资源管理器中找到解决方案文件,双击该文件也可打开文件所描述的项目。

1.4 Visual Studio 2015 程序调试

VS 2015 设计了一个非常方便的程序调试环境,在调试程序时可以选择调试工具对程序进行调试。下面先介绍调试工具,再介绍几种调试程序的方法。

1. “调试”菜单

当文件编辑区已经打开文件时,在“调试”菜单中选择有关命令可以以分步、跟踪方式执行程序。以下是“调试”菜单主要命令的功能。

(1) “启动调试”命令或 F5 键:按断点(中断程序执行的位置点)分步执行程序。每执行一次“启动调试”命令或按 F5 键,程序就执行到一个断点处暂停。程序员可以检查当前变量或表达式的值。如果没有设置断点,则无停顿地执行完整个程序语句。

(2) “开始执行(不调试)”命令:执行整个程序。

(3) “逐语句”命令:单步执行程序,即逐条语句执行,也可按 F11 键执行该命令。当调用函数时,进入函数体内逐条语句执行。

(4) “逐过程”命令:单步执行程序,把函数调用作为一步,即不进入函数体内执行,也可按 F10 键执行该命令。

(5) “切换断点”命令:当光标移到要设置断点的代码行时,选择该命令或按 F9 键,设置一个断点。这时,可以看到代码窗口左边的断点区出现一个红点。重复此操作则取消断点。设置断点更简单的方法是直接单击代码窗口的断点区。

(6) “新建断点”命令:在函数内部设置断点。

2. “调试”工具栏

“调试”工具栏提供“调试”菜单主要命令的快捷方式,共包括 12 个按钮,如图 1-9 所示。用户可以使用该工具栏上的按钮及其快捷键调试程序。



图 1-9 “调试”工具栏

3. 程序调试举例

图 1-10 显示了一个程序的调试过程。

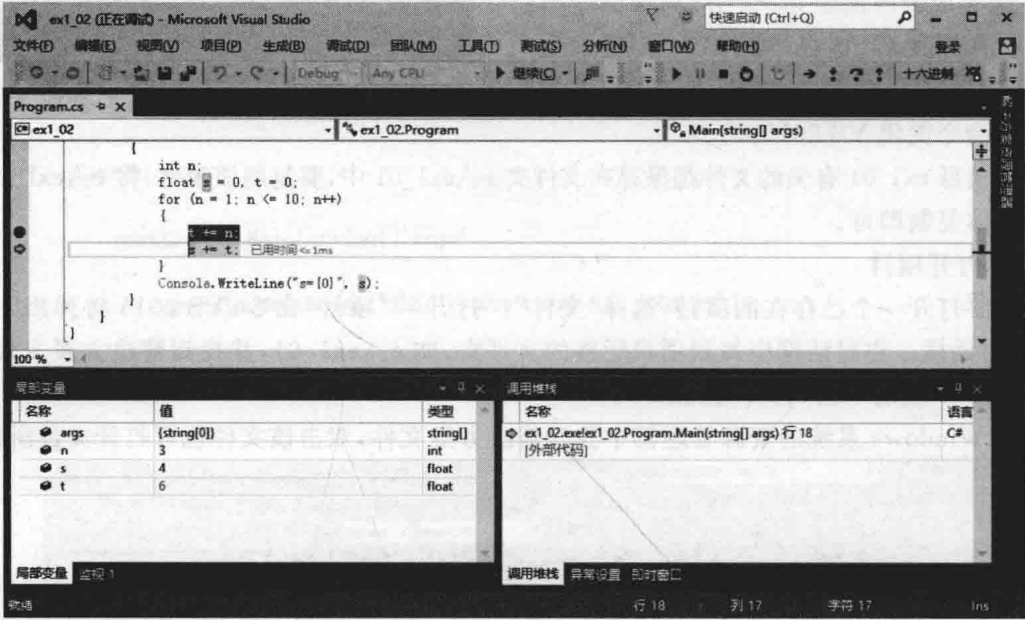


图 1-10 程序的调试

1) 设置断点

在程序中某个关键的语句处设置断点,最简捷的方法是,单击要设置断点的语句行左边的断点区,这时,行首断点区出现一个红色的圆点。再次单击将取消断点。

2) 开始调试

选择“启动调试”命令或按 F5 键,程序进入调试状态。若进入调试状态之前程序没有经过编译,系统首先会进行编译。没有编译错误的程序才可进行调试。

进入调试后,程序执行到第一个断点处暂停,这时可以查看有关的中间结果。在图 1-10 中,左下角窗口中系统自动显示了有关变量的值,其中,n、s、t 的值分别是 3、4、6。图中左侧断点区的箭头表示当前程序暂停的位置。

再继续按 F5 键,则执行到下一断点处暂停,若无下一断点,则将执行完整个程序。若再继续按 F11 键则接着进行“逐语句”调试。

3) 停止调试

在调试过程中,可选择“调试”|“停止调试”命令或按 Shift+F5 键结束调试,回到初始状态。

第2章

实验指导

学习程序设计,上机实验是十分重要的环节。为了方便读者上机练习,本章设计了12个实验。这些实验和课堂教学紧密配合,读者可以根据实际情况从每个实验中选择部分内容作为上机练习,各个实验后面的实验思考题可以作为实验内容的补充。

为了达到理想的实验效果,读者务必做到:

(1) 实验前认真准备,要根据实验目的和实验内容,复习好实验中要用到的概念、语句,想好编程的思路,做到胸有成竹,提高上机效率。

(2) 实验过程中积极思考,要深入分析实验现象、程序的执行结果以及各种屏幕信息的含义、出现的原因并提出解决办法。

(3) 实验后认真总结,要总结本次实验有哪些收获,还存在哪些问题,并写出实验报告。实验报告应包括实验目的、实验内容、流程图、程序清单、运行结果以及实验的收获与体会等内容。

实验1 程序的运行环境和步骤

一、实验目的

1. 熟悉 VS 2015 集成开发环境的使用方法。
2. 掌握运行一个 C# 程序的基本步骤。
3. 熟悉 C# 程序的结构特征和程序书写规则。

二、模仿编程实验

1. 在 VS 2015 环境下,练习程序的编辑、生成和运行。

```
using System;
namespace ex01_01
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Hello, C#!");
        }
    }
}
```

2. 下面是一个减法程序,程序运行时等待用户从键盘输入两个整数,然后求它们的差,上机运行该程序。

```
using System;
namespace ex01_02
{
    class Program
    {
        static void Main(string[] args)
        {
            int a, b, c;
            Console.Write("a:");
            a = int.Parse(Console.ReadLine());
            Console.Write("b:");
            b = int.Parse(Console.ReadLine());
            c = a - b;
            Console.WriteLine("a - b = {0}", c);
        }
    }
}
```

3. 计算并输出半径为 5 的圆的面积。请补充程序,并上机运行该程序。

```
using System;
namespace ex01_03
{
    class Program
    {
        static void Main(string[] args)
        {
            double p;
            p = ①;
            Console.WriteLine("半径为 5 的圆的面积: {0}", p);
        }
    }
}
```

4. 分析下面的程序,改正其中的错误。

```
using System;
namespace ex01_04
{
    class Program
    {
        static void Main(string[] args)
        {
            int i = 15, j = 6;
            s = i + j;
            Console.WriteLine("s = {0}", s);
        }
    }
}
```

/* 变量 s 没有定义 */