

Visual C++ 2010 程序设计案例教程

刘冰 张林 蒋贵全 杨勇 编著



- 详细讲述C++编程的基础知识和Visual C++ 2010的特性
- 全面介绍Visual C++ 2010开发技术和构建应用程序过程
- 提供具有实际应用价值的程序设计案例详解

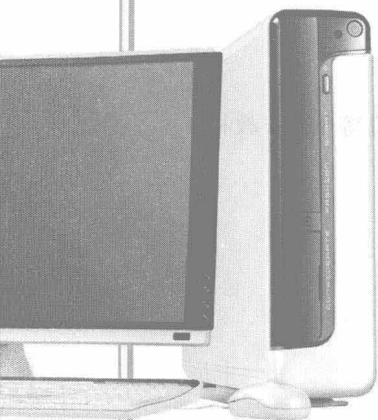


机械工业出版社
China Machine Press

高等院校计算机课程案例教程系列

Visual C++ 2010 程序设计案例教程

刘冰 张林 蒋贵全 杨勇 编著



- 详细讲述C++编程的基础知识和Visual C++ 2010的特性
- 全面介绍Visual C++ 2010开发技术和构建应用程序过程
- 提供具有实际应用价值的程序设计案例详解



机械工业出版社
China Machine Press

本书从实用的角度出发,汲取了国内外程序设计教材的精华,由浅入深、循序渐进地介绍了 Visual C++ 2010 开发环境、C++ 语言基础、面向对象的 C++ 程序设计思想、MFC 基本应用程序、用户界面设计、对话框和常用控件、图形与文本、Visual C++ 2010 的高级应用以及开发实例等知识。为了保证初接触 Visual Studio 2010 的读者能够更好地从事开发工作,各章节都配备了相应的开发实例。

本书内容丰富、实用性强,既可作为普通高等院校计算机专业相关课程的教材或教学参考书,也可作为通信、电子信息、自动化等相关专业的计算机课程教材。对于 .NET 开发初学者,通过本教材的学习可以从入门级水平提升到中级水平。对于具有一定 Visual C++ 编程经验的读者,本书也具有很好的参考价值。

封底无防伪标均为盗版

版权所有,侵权必究

本书法律顾问 北京市展达律师事务所

图书在版编目(CIP)数据

Visual C++ 2010 程序设计案例教程 / 刘冰, 张林, 蒋贵全等编著. —北京: 机械工业出版社, 2012.9

(高等院校计算机课程案例教程系列)

ISBN 978-7-111-39810-3

I. V… II. ①刘… ②张… ③蒋… III. C 语言—程序设计—高等学校—教材
IV. TP312

中国版本图书馆 CIP 数据核字 (2012) 第 223598 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑: 余洁

北京市京瑞印刷有限公司印刷

2013 年 1 月第 1 版第 1 次印刷

185mm × 260mm · 18.5 印张

标准书号: ISBN 978-7-111-39810-3

定价: 36.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

客服热线: (010) 88378991; 88361066

购书热线: (010) 68326294; 88379649; 68995259

投稿热线: (010) 88379604

读者信箱: hzsj@hzbook.com

前 言

作为新一代开发工具, Visual Studio 2010 能够帮助开发团队在最新的平台上构建具有更佳用户体验的应用程序, 进行灵活快速开发, 实现生产效率新突破, 并能使开发团队更好地进行协作。Visual Studio 2010 通过增强编译器后端的代码生成功能来提高生成应用程序的编译速度、质量和性能; 在基于 Visual Studio 2010 的 Visual C++ 2010 开发平台中, 解决方案和项目的构建系统 (build system) 从原先基于 VCBUILD 的构建系统, 迁移到基于 MSBuild (Microsoft Build Engine) 的构建系统, MSBuild 的加入使得“一次编码, 多个平台运行”成为可能; Microsoft Visual Studio Team System 2010 提供完整的工具套件和统一的开发过程, 适用于任何规模的开发团队, 帮助所有团队成员提高自身技能, 使得开发人员、设计人员、测试人员、架构师和项目经理更好地协同工作, 缩短软件或解决方案的交付时间。

本书共分 10 章, 在讲解 Visual C++ 语法的基础上, 通过案例介绍面向对象的程序设计方法。第 1 章主要对 Visual C++ 2010 的新特征和 Visual Studio 2010 集成开发环境进行概述; 第 2 章介绍 Visual C++ 2010 开发环境以及如何创建第一个 MFC 应用程序和控制台应用程序; 第 3 章介绍函数和编译预处理、C++ 的对象、C++ 的类、构造函数与析构函数等内容; 第 4 章介绍 C++ 的基本数据类型、标识符和关键字、运算符和表达式等内容; 第 5 章介绍 MFC 应用程序基本框架以及 MFC 应用程序中包含的 MFC 类, 如应用程序类 CWinApp、主框架类 CFrameWnd、视图类 CView、文档类 CDocument、对话框类 CDialog、菜单类 CMenu 和线程基类 CWinThread; 第 6 章介绍菜单的设计、菜单编程、工具栏设计、状态栏设计等内容; 第 7 章介绍对话框的设计, 静态控件、编辑框、按钮、列表框、组合框和滚动条等标准控件的使用, 以及旋转按钮、进度条、滑块、属性页控件、树形控件等公共控件的使用; 第 8 章介绍图形设备接口、绘图环境、画笔和画刷、文本与字体等内容; 第 9 章介绍打印编程和数据库编程等内容; 第 10 章对图书信息管理系统进行综合示例分析。

本书的编者都是长期在高校从事软件教学的教师, 有丰富的教学经验和科研开发能力。另外, 在编写本书的过程中编者还参阅了大量国内外有关 C++ 程序设计的教材和资料。本书的顺利出版, 要感谢重庆邮电大学计算机科学与技术学院的相关领导和老师给予的大力支持和帮助, 还要感谢计算机教育界许多同行的关心。

为方便教师, 本书有配套电子教案和习题答案, 如有需要可与编者联系 (cobly1837@sina.com) 或登录华章网站 (www.hzbook.com) 下载。

目前, 国内外有关 Visual C++ 方面的资料很多, 新理论、新技术层出不穷。由于 Visual C++ 程序设计方法、设计思想发展迅速, 加上时间仓促, 编者水平有限, 书中难免存在不妥和错误之处, 恳请读者批评指正, 并提出宝贵意见, 以便进一步完善。

感谢您阅读本书, 请将您的宝贵建议和意见发送至: liubing@cqupt.edu.cn。

教学建议

教学章节	教学要求	课时
第1章 Visual C++ 2010 开发平台概述	了解 Visual C++ 2010 的新特征 掌握 Visual Studio 2010 的安装 掌握 Visual Studio 2010 的配置要求 掌握部署工具	2
	了解资源编辑器 了解生成和调试工具 掌握 Ribbon 界面设计可视化 了解重新启动管理器	2
第2章 Visual C++ 2010 MFC 程序设计快速入门	了解 Visual C++ 2010 开发环境 创建第一个 MFC 应用程序 创建控制台应用程序	2
第3章 面向对象的 C++ 程序设计	掌握面向对象的基本概念 掌握 C++ 的类和对象的定义	2
	掌握 C++ 的类 掌握成员函数的概念 掌握友元函数和友元类 掌握基类成员的访问控制	4
	掌握方法的重写与重载 掌握构造函数与析构函数	2~4
第4章 C++ 语言基础	掌握 C++ 的数据类型 掌握标识符和关键字 掌握运算符和表达式	6
	掌握数组 掌握指针和引用	2~4
	了解字符串处理函数 掌握 C++ 的标准输入输出	4~8
第5章 MFC 基本应用程序	掌握 MFC 应用程序 编写简单的 MFC 程序 了解 MFC 的类	4~8
第6章 用户界面设计	掌握菜单设计 掌握工具栏设计 掌握状态栏设计	8~12
第7章 对话框和常用控件	掌握对话框设计 掌握控件设计	6
	了解标准控件的使用 掌握公共控件的使用	4
第8章 图形与文本	了解图形设备接口 掌握绘图环境 掌握画笔和画刷	4~6
	掌握文本与字体 了解位图 掌握图标和光标	4~6

(续)

教学章节	教学要求	课时
第9章 Visual C++ 2010 的高级应用	掌握打印编程	4
	掌握数据库编程	4
第10章 综合示例分析	掌握图书信息管理系统创建	8~16

说明:

- 1) 建议课堂教学全部在多媒体机房内完成,实现“讲-练”结合。
- 2) 建议教学分为核心知识技能模块(前9章的内容)和技能提高模块(第10章的内容),其中核心知识技能模块建议教学学时为64~82,技能提高模块建议教学学时为8~16,不同学校可以根据各自的教学要求和计划学时数对教学内容进行取舍。

目 录

前言

教学建议

第1章 Visual C++ 2010 开发平台

概述 1

1.1 Visual C++ 2010 概述 1

1.2 Visual Studio 2010 集成开发

环境 1

1.2.1 安装 Visual Studio 2010 1

1.2.2 配置 Visual Studio 2010 5

1.2.3 解决方案和项目 8

1.2.4 生成和调试工具 10

1.2.5 部署工具 11

1.2.6 Ribbon 界面设计可视化 12

1.2.7 重新启动管理器 12

1.3 本章小结 13

1.4 习题 13

第2章 Visual C++ 2010 MFC 程序设计

快速入门 14

2.1 Visual C++ 2010 开发环境概述 14

2.2 创建第一个 MFC 应用程序 17

2.3 创建控制台应用程序 20

2.3.1 用 AppWizard 创建一个 CLR

控制台应用程序 20

2.3.2 用 AppWizard 创建一个 Win32

控制台应用程序 22

2.4 本章小结 23

2.5 习题 23

第3章 面向对象的 C++ 程序设计 25

3.1 函数和编译预处理 25

3.1.1 函数的声明 25

3.1.2 函数的参数和返回值 26

3.1.3 函数的调用 28

3.1.4 函数参数的传递 29

3.2 C++ 的对象 31

3.2.1 对象的定义 31

3.2.2 对象的引用 32

3.2.3 对象的赋值 33

3.3 C++ 的类 34

3.3.1 类的定义 34

3.3.2 类的作用域 35

3.3.3 成员函数 36

3.3.4 友元函数和友元类 38

3.3.5 基类成员的访问控制 41

3.3.6 方法的重写与重载 42

3.4 构造函数与析构函数 43

3.4.1 构造函数 43

3.4.2 析构函数 46

3.5 本章小结 47

3.6 习题 47

第4章 C++ 语言基础 48

4.1 C++ 基本数据类型 48

4.2 标识符和关键字 49

4.2.1 标识符 49

4.2.2 关键字 49

4.3 运算符和表达式 50

4.3.1 C++ 运算符 50

4.3.2 表达式 54

4.4 数组 57

4.4.1 一维数组与二维数组 57

4.4.2 字符数组与字符串数组 60

4.4.3 对象数组 61

4.5 指针和引用 62

4.5.1 指针变量的定义与赋值 62

4.5.2 数组指针与指针数组 64

4.5.3 动态内存分配 64

4.5.4 引用 66

4.6 字符串处理函数 67

4.7 C++ 标准输入输出	69	7.1.3 通用对话框	123
4.8 本章小结	71	7.2 标准控件	130
4.9 习题	71	7.2.1 静态控件	130
第5章 MFC 基本应用程序	73	7.2.2 编辑框	131
5.1 MFC 应用程序概述	73	7.2.3 按钮	132
5.1.1 简单的 MFC 程序	73	7.2.4 列表框	133
5.1.2 MFC 应用程序的分析	76	7.2.5 组合框	133
5.2 MFC 的类	77	7.2.6 分组框	134
5.2.1 MFC 应用程序中包含的 MFC 类	77	7.2.7 滚动条	135
5.2.2 应用程序类 CWinApp	78	7.2.8 标准控件的应用	136
5.2.3 主框架类 CFrameWnd	80	7.3 公共控件	143
5.2.4 视图类 CView	81	7.3.1 旋转按钮	143
5.2.5 文档类 CDocument	82	7.3.2 进度条	144
5.2.6 对话框类 CDialog	83	7.3.3 滑块	146
5.2.7 菜单类 CMenu	85	7.3.4 属性页控件	148
5.2.8 线程基类 CWinThread	87	7.3.5 图像列表和列表控件	151
5.3 本章小结	89	7.3.6 树形控件	155
5.4 习题	89	7.4 本章小结	164
第6章 用户界面设计	90	7.5 习题	165
6.1 菜单的类型	90	第8章 图形与文本	166
6.2 菜单编程	90	8.1 图形设备接口	166
6.2.1 菜单设计	91	8.1.1 图形设备接口概述	166
6.2.2 创建弹出式菜单	94	8.1.2 颜色设置	167
6.2.3 创建动态菜单	96	8.1.3 映射模式与坐标系	167
6.2.4 创建基于对话框的菜单	99	8.1.4 GDI 图像处理	169
6.3 工具栏设计	100	8.2 绘图环境	170
6.3.1 CMFCToolBar 类	100	8.2.1 DC 结构	171
6.3.2 工具栏编辑器的使用	102	8.2.2 CDC 类	172
6.3.3 创建工具栏	103	8.3 画笔和画刷	173
6.4 状态栏设计	106	8.3.1 使用库存对象	173
6.4.1 CMFCStatusBar 类	106	8.3.2 画笔	174
6.4.2 创建状态栏	107	8.3.3 画刷	176
6.5 本章小结	110	8.3.4 创建和使用画笔、画刷	178
6.6 习题	110	8.4 文本与字体	179
第7章 对话框和常用控件	111	8.5 位图	183
7.1 对话框设计	111	8.5.1 位图资源的创建	184
7.1.1 模态对话框	113	8.5.2 位图的使用	186
7.1.2 非模态对话框	120	8.5.3 位图使用示例	190
		8.6 图标和光标	214

8.7 本章小结	215	第 10 章 综合示例分析	229
8.8 习题	215	10.1 图书信息管理系统	229
第 9 章 Visual C++ 2010 的高级		10.1.1 数据库表设计	229
应用	217	10.1.2 程序实现	231
9.1 打印编程	217	10.2 本章小结	283
9.2 数据库编程	224	10.3 习题	283
9.3 本章小结	228	参考文献	287
9.4 习题	228		

第 1 章

Visual C++ 2010开发平台概述

1.1 Visual C++ 2010 概述

Visual C++ 2010 开发环境为项目管理与配置（包括更好地支持大型项目）、源代码编辑、源代码浏览和调试工具提供强大的支持，是开发过程中不可缺少的工具，是提高代码生产力的基本保障。

在 Visual C++ 2010 中，增强的 MFC 为程序员开发出环境优美、安全稳定的应用程序奠定了坚实的基础。Visual C++ 2010 中的 MFC 新功能包括：

（1）便于生成 Windows 7 应用程序

Visual Studio 2010 提供了为本机 Windows API 编写代码所需的 Windows SDK 组件。此外该版本的 Visual Studio 还可以在增强的 MFC 的帮助下更轻松地编写 Windows 应用程序。可通过 MFC 库访问主要的 Windows 7 功能，而不必直接写入本机 API。只需重新编译，现有的 MFC 应用程序就可以在 Windows 7 上运行使用，而且新应用程序可以充分利用新增的功能。

（2）增加了重新启动管理器

对于大多数的程序编写者来说，可能遇到过由于应用程序的意外关闭或重新启动，将花了很大功夫编写的文档化为乌有的情况，这是很郁闷的事情。在 Visual C++ 2010 中增加了重新启动管理器的功能，一旦遇到上述情况，便可以恢复到发生意外情况时的状态，不至于让自己的工作成果因为故障而丢失。

重新启动管理器是 Windows Vista 中引入的一项有用的功能，可以使应用程序在终止前执行保存操作。应用程序可以调用此功能，然后在重新启动时还原其状态。在 Visual C++ 2010 中，可以在 MFC 应用程序中充分利用重新启动管理器，更妥善地处理应用程序崩溃和重新启动的情况。使用重新启动管理器时，分两种情况：一是向现有的应用程序中添加重新启动管理器；二是在新创建的 MFC 应用程序中使用此功能。

（3）使用任务对话框

为了改善用户体验，在 Visual C++ 2010 中，使用任务对话框（task dialog）代替原先广泛使用的消息框（message box）。与消息框相比，任务对话框不仅可以向用户显示消息，还可以显示自定义的按钮、命令行选项以及更加丰富的辅助信息等。新增的 CTaskDialog 类提供了很多简单实用的函数，可以轻松上手，使用任务对话框来改善应用程序的用户体验，使自己的应用程序的界面风格与 Vista 系统保持一致。

1.2 Visual Studio 2010 集成开发环境

1.2.1 安装 Visual Studio 2010

Visual C++ 2010 是 Visual Studio 2010 的一部分。正如 Windows 7 有旗舰版、家庭高级版、家庭初级版一样，Visual C++ 2010 也有更多的子版本，本书选择的是旗舰版，旗舰版的功能最全面。

Visual Studio 2010 具体安装步骤如下：

1) 将下载的 ISO 文件装入虚拟光驱，运行“Autorun.exe”程序，开始 Visual Studio 2010 的

第一步，如图 1-1 所示。



图 1-1 启动安装界面

- 2) 复制安装过程中需要的文件。
- 3) 正式开始安装，单击“下一步”按钮，如图 1-2 所示。

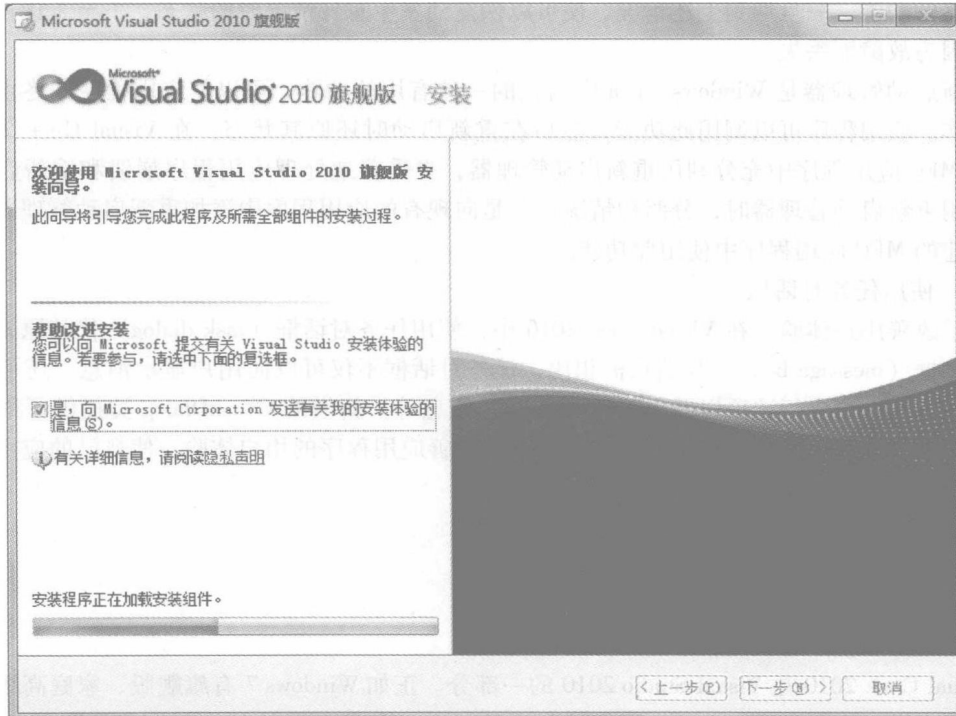


图 1-2 Visual Studio 2010 安装向导界面

4) 选择“我已阅读并接受许可条款”单选按钮,单击“下一步”按钮,如图1-3所示。

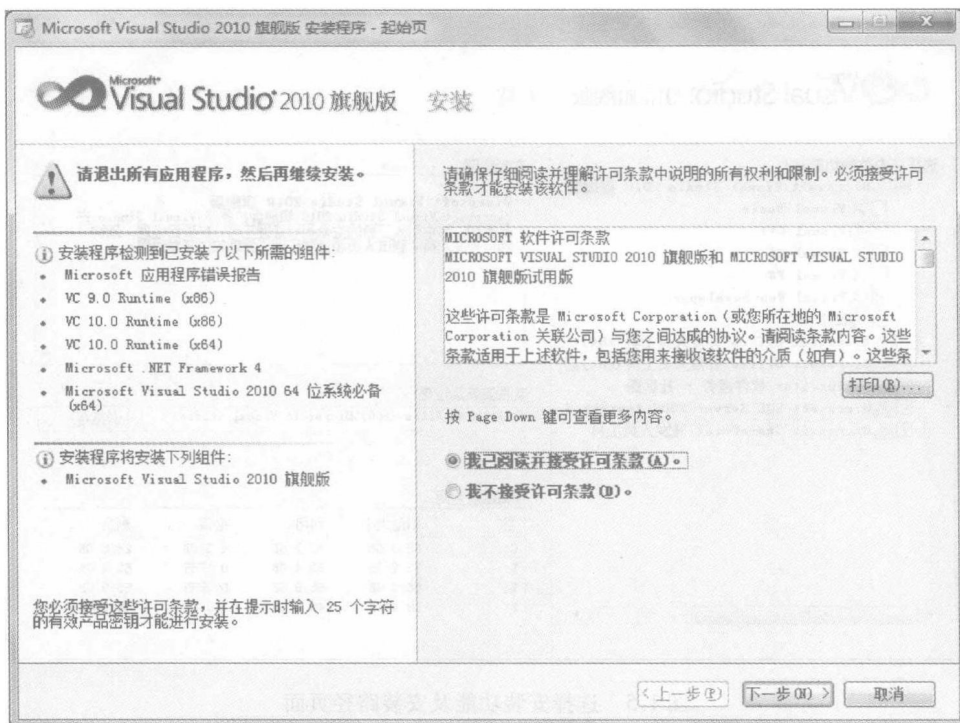


图 1-3 Visual Studio 2010 安装起始页界面

5) 选择完全安装或者自定义安装,完全安装需要 7.5G 的磁盘空间,如图1-4所示。

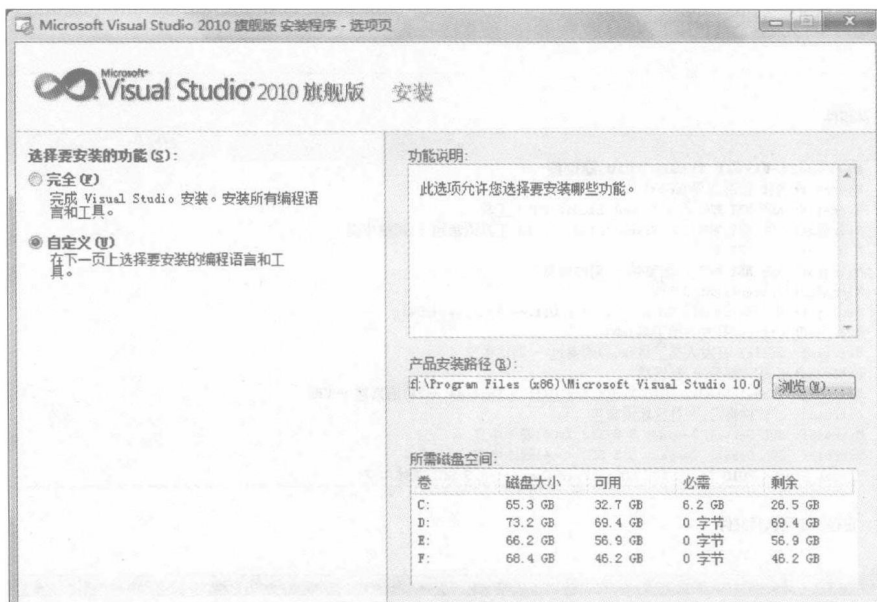


图 1-4 Visual Studio 2010 安装选项页界面

6) 选择要安装的功能和安装路径, 如图 1-5 所示。

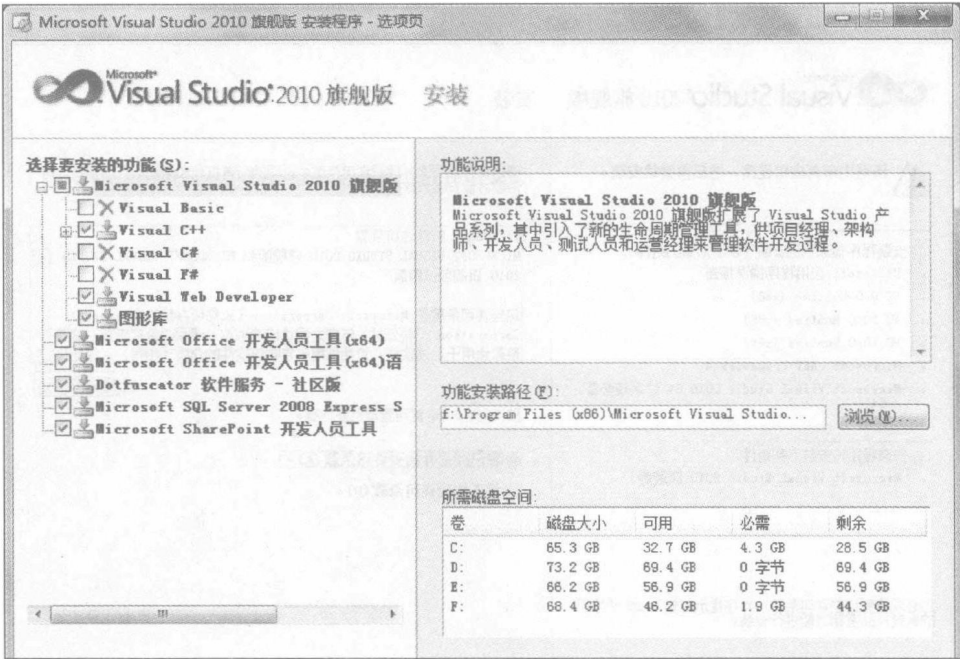


图 1-5 选择安装功能及安装路径页面

7) 开始安装 Visual Studio 组件, 如图 1-6 所示。

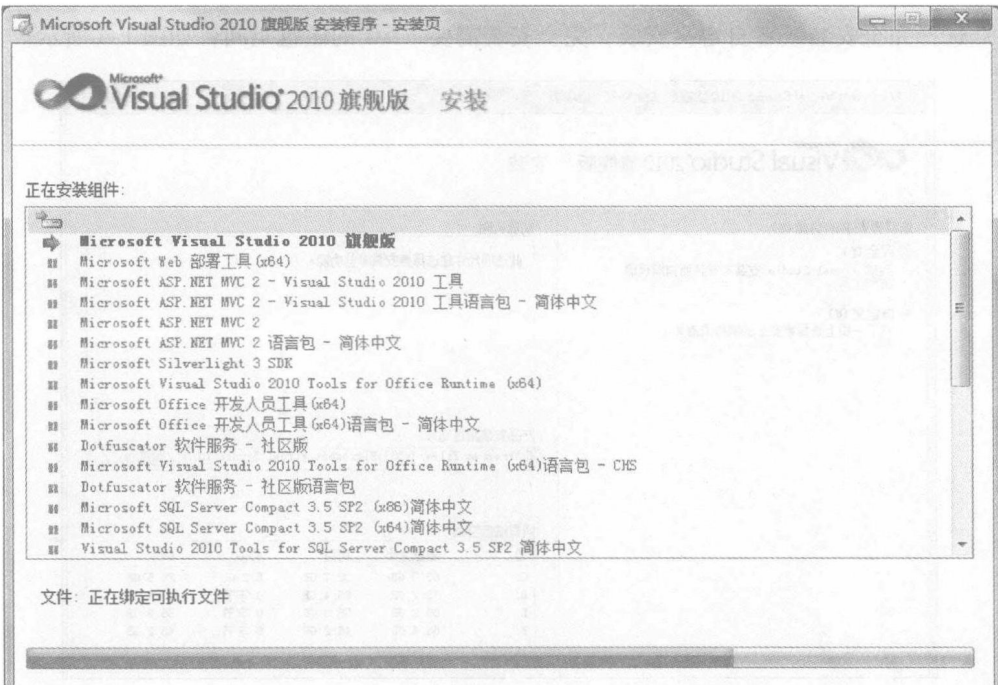


图 1-6 Visual Studio 2010 安装组件页面

8) 看到图 1-7 就意味着 Visual Studio 2010 安装完成。

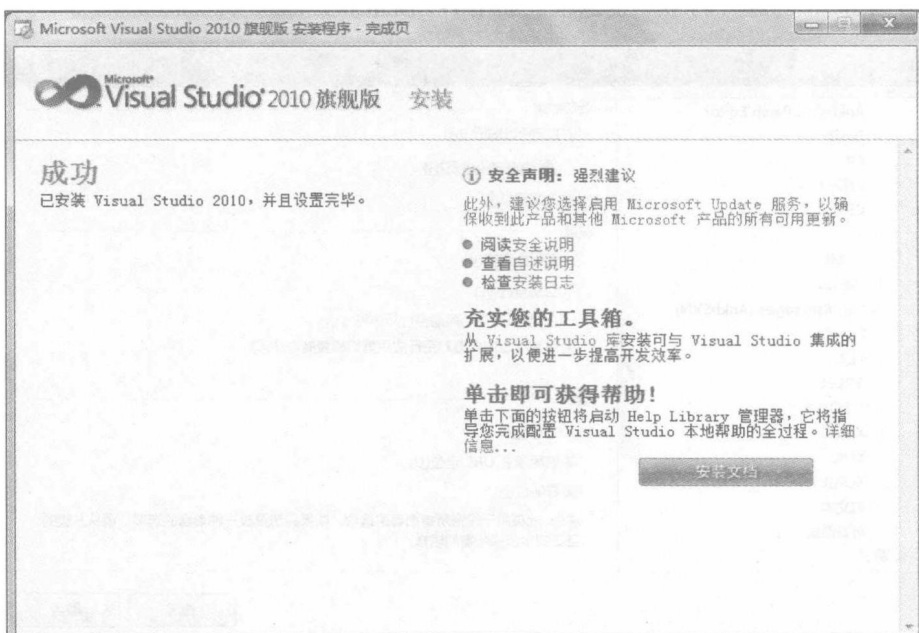


图 1-7 Visual Studio 2010 安装完成界面

1.2.2 配置 Visual Studio 2010

1) 第一次运行 Visual Studio 2010 需要进行环境设置, 启动 Visual Studio 2010 旗舰版, 如图 1-8所示。

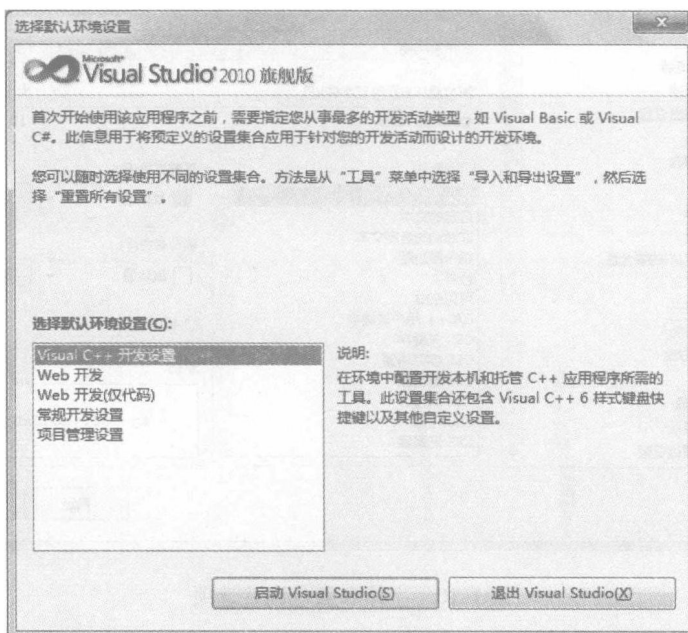


图 1-8 启动 Visual Studio 2010 运行环境设置界面

2) 常见的设置。通过菜单“工具→选项”调出“选项”对话框，选择“文本编辑器”下方的“所有语言”，勾选“行号”复选框，如图 1-9 所示。

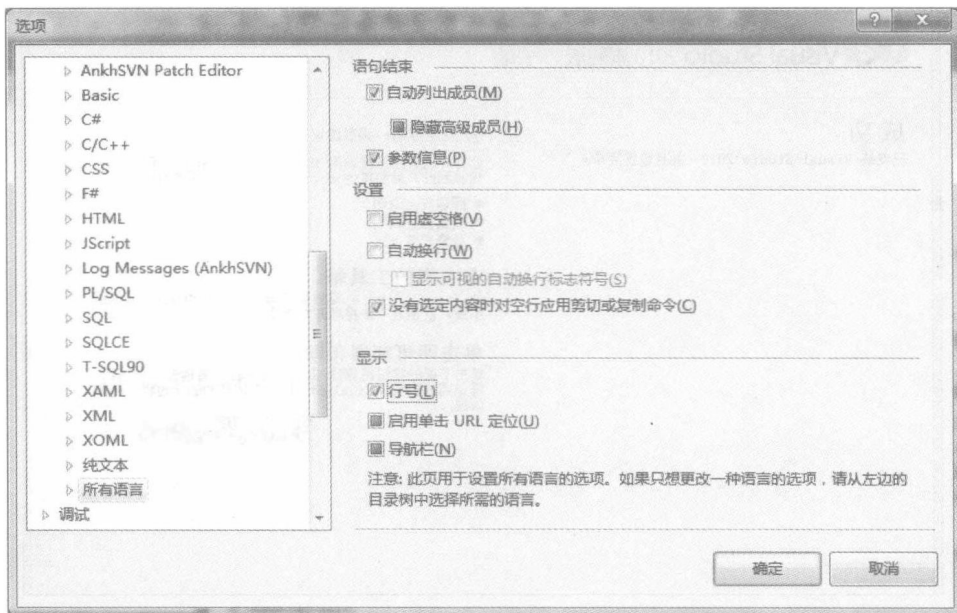


图 1-9 “选项”对话框

3) 如果想调整字体或者选择颜色，可以在“选项”对话框中单击“环境”下方的“字体和颜色”，出现如图 1-10 所示对话框，并在其中进行选择。

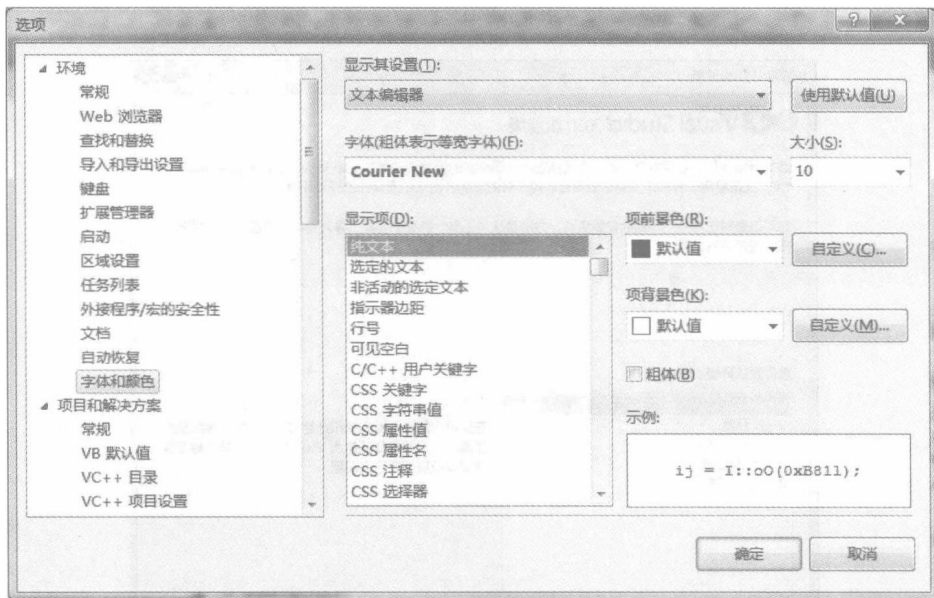


图 1-10 字体和颜色调整选项

4) 单击工具栏中的空白区域，把生成工具栏调出来，如图 1-11 所示。

这样就可以直接在工具栏上面选择编译项目、编译整个解决方案、运行程序和调试程序了。

选择生成工具栏旁边的小三角形可以自定义，还可以添加一些按钮。

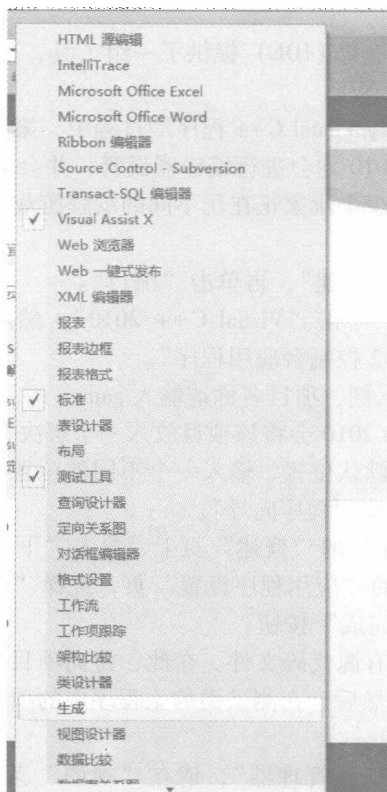


图 1-11 生成工具栏

5) 单击自定义界面下面的添加按钮就可以添加新的按钮了，现在把“开始执行（不调试）”添加到工具栏中，如图 1-12 所示。

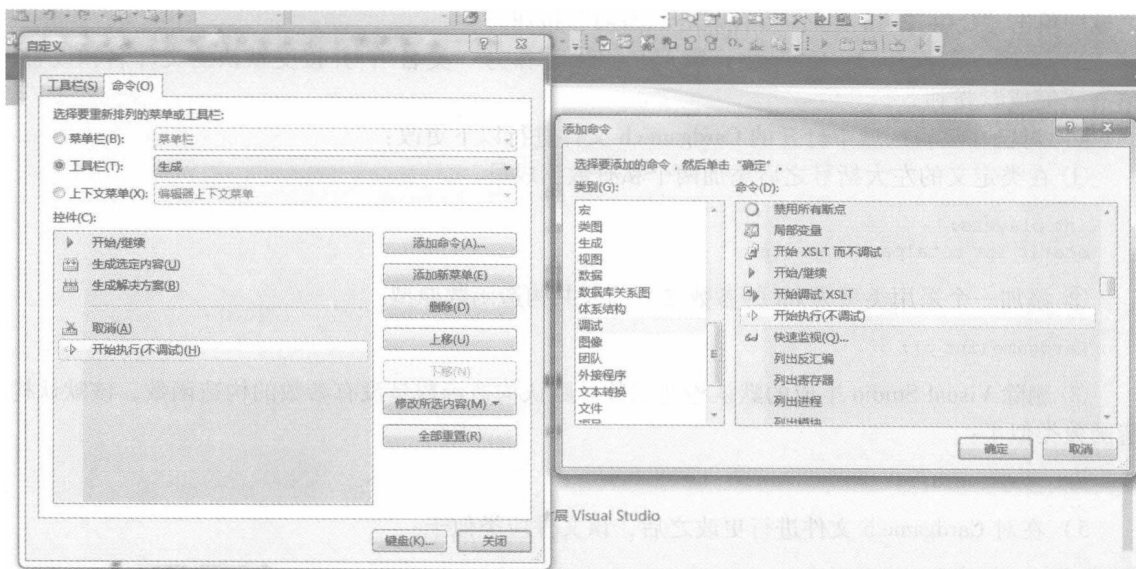


图 1-12 自定义对话框

1.2.3 解决方案和项目

Visual Studio 2010 集成开发环境 (IDE) 提供了一组工具, 可帮助编写和修改代码, 以及检测和更正错误。

在使用 Visual Studio 2010 编写 Visual C++ 程序的过程中, 第一个任务是选择项目的类型。对于每个项目类型, Visual Studio 2010 均会进行编译器设置, 并会生成起始代码。下面我们将创建一个基本程序, 该程序跟踪有多少个玩家正在玩不同的纸牌游戏。

1. 创建项目

1) 在“文件”菜单上选择“新建”, 再单击“项目”。

2) 在“项目类型”区域中, 单击“Visual C++ 2010”。然后, 在“Visual Studio 2010 已安装的模板”窗口中, 单击“Win32 控制台应用程序”。

3) 输入项目名称。对于此示例, 项目名称请输入 game。

当创建项目时, Visual Studio 2010 会将该项目放入一个解决方案。默认情况下, 解决方案的名称与项目名称相同。可以接受默认位置、输入一个不同的位置或者浏览到要保存项目的目录。单击“确定”按钮启动“Win32 应用程序向导”。

4) 在“Win32 应用程序向导”的“概述”页上, 单击“下一步”按钮。

5) 在“应用程序类型”下的“应用程序设置”页, 选择“控制台应用程序”。在“附加选项”下选择“空项目”。单击“完成”按钮。

现在有了项目, 但项目还没有源代码文件。在此, 将向项目中添加一个类, 并且 Visual Studio 2010 将添加 .h 和 .cpp 文件。然后, 在测试类的主程序中的项目上添加一个新的源代码文件。

2. 向项目添加类

1) 如果未显示“解决方案资源管理器”, 请在“视图”菜单上选择“解决方案资源管理器”。

2) 在“解决方案资源管理器”中, 右击“头文件”文件夹, 选择“添加”, 然后单击“类”。

在“Visual C++ 2010”类别中, 单击“C++ 2010”, 在“Visual Studio 2010 已安装的模板”区域中单击“C++ 2010 类”, 然后单击“添加”按钮。

3) 在“一般 C++ 类向导”中, 输入 Cardgame 作为“类名”, 并接受默认的文件名和设置。单击“完成”按钮。

4) 对已在编辑区域中打开的 Cardgame.h 文件进行以下更改:

① 在类定义的左大括号之后添加两个私有数据成员。

```
int players;
static int totalparticipants;
```

② 添加一个采用类型为 int 的参数之一的公共构造函数原型。

```
Cardgame(int p);
```

③ 删除 Visual Studio 生成的默认构造函数。默认构造函数是没有参数的构造函数, 该默认构造函数类似于:

```
Cardgame(void);
```

5) 在对 Cardgame.h 文件进行更改之后, 该文件应类似于:

```
#pragma once
class Cardgame
{
    int players;
```