

C语言及程序设计基础 上机指导和习题解答



WUHAN UNIVERSITY PRESS
武汉大学出版社

计算机科学与技术专业规划教材

C语言及程序设计基础 上机指导和习题解答

主编 谭成予

参编 高柯夫



WUHAN UNIVERSITY PRESS

武汉大学出版社

图书在版编目(CIP)数据

C语言及程序设计基础上机指导和习题解答/谭成予主编. —武汉:武汉大学出版社, 2010. 6

计算机科学与技术专业规划教材

ISBN 978-7-307-07749-2

I. C… II. 谭… III. C语言—程序设计—高等学校—教学参考资料
IV. TP312

中国版本图书馆CIP数据核字(2010)第084568号

责任编辑:林莉 责任校对:王建 版式设计:支笛

出版发行:武汉大学出版社 (430072 武昌 珞珈山)

(电子邮件:cbs22@whu.edu.cn 网址:www.wdp.com.cn)

印刷:通山金地印务有限公司

开本:787×1092 1/16 印张:19.75 字数:499千字 插页:1

版次:2010年6月第1版 2010年6月第1次印刷

ISBN 978-7-307-07749-2/TP·358 定价:32.00元

版权所有,不得翻印;凡购买我社的图书,如有缺页、倒页、脱页等质量问题,请与当地图书销售部门联系调换。



前言

计算机是操作性极强的应用型学科,学习程序设计包括理论和实践操作两个环节。作者在武汉大学为一年级本科生教授程序设计时,发现许多学生在课堂教学环境对理论能够正确理解,但在上机实验环境却难以自行动手编程。一旦离开了老师的指导,很多初学者面对实践操作就会有点无所适从,因而编写本书,作为《C 语言及程序设计基础》一书的配套上机实验指导教材,供同学们在实际操作过程中参考。

学习程序设计,设计思想上需要“以算法带动文法”、“学思想用细节”,而实际操作中需要精通编程文法和编译工具的使用。工具和文法是编程的辅助手段,学会对它们的熟练使用,可以帮助程序员顺利地将头脑中的设计思想实现,变成实用的程序。

本书是为水平各不相同的所有程序设计人员编写的,既可作为程序设计的初学者和程序员作为 C 语言实践操作时的指导手册,又可作为讲授《C 语言及程序设计基础》一书的教师的参考书。我们相信本书及配套教材《C 语言及程序设计基础》将为读者提供一种内容丰富而富有挑战性的学习经历。

欢迎访问我们建立的课程资源网站:<http://jpkc.whu.edu.cn/jpkc2005/alprogram/>。

本书系统地介绍 C 语言的实验指导和习题解答。全书共分三个部分,下面简单介绍这些章节的内容:

● 第一部分: C 语言上机指导

这个部分包括从第 1 章到第 6 章的内容。第 1 章到第 4 章分别介绍了在 Visual C++2005、Dev C++、Turbo C 和 Unix/Linux 操作系统中的 C 编译器等常用 C 程序开发环境中编辑、编译、链接、运行和调试的具体步骤和方法。第 5 章介绍了软件测试的常用方法,可帮助读者更好地选择软件测试用例。第 6 章给出了与《C 语言及程序设计基础》中每一章对应的上机指导 1 到上机指导 11 的内容安排。

● 第二部分: C 语言编程高级篇

这个部分包括从第 7 章到第 11 章的内容。第 7 章介绍了在 Turbo C 中完成文本界面设计的基本方法。第 8 章介绍了在 Turbo C 中进行图形图像处理的基本技术。第 9 章介绍了在 Turbo C 中实现中断编程的方法。第 10 章以 Visual C 为编程环境,介绍了使用 Winsock 完成网络通信编程的技术。第 11 章着重描述 C99 标准中新增的功能。

● 第三部分: 习题参考答案

这个部分给出了包括配套教材《C 语言及程序设计基础》各章中超过 95%习题的参考答案,少数几个习题未给出参考答案,可供学生自行扩展功能后作为课堂教学中课程设计的参考题目。

很高兴在此表达对许多人士的感谢。特别感谢高柯夫在本书编写过程给予的大力支持,为本书收集并整理了大量资料,并提供了第二部分中大部分章节的初稿。同时还要感谢武汉大学出版社的工作人员为本书完成的排版工作。

谭成予

2010 年 3 月于武汉大学

目 录

第一部分 C 语言上机指导

第 1 章 在 Visual C++ 2005 中编写 C 程序	3
1.1 Visual Studio 2005 简介	3
1.1.1 Visual Studio 2005 简介	3
1.1.2 Visual C++ 2005 简介	3
1.2 Visual Studio 2005 安装	4
1.2.1 Visual Studio 2005 要求的系统配置	4
1.2.2 安装 Visual Studio 2005	4
1.2.3 安装 MSDN	7
1.3 集成开发环境 IDE 简介	7
1.3.1 启动进入 Visual C++ 2005	8
1.3.2 工具栏选项	8
1.3.3 项目和解决方案	9
1.3.4 设置 Visual C++ 2005 的选项	10
1.4 使用 IDE 编写 C 程序	10
1.4.1 创建新项目 and 源程序	10
1.4.2 编辑已存在的项目及源程序	14
1.4.3 编译并构建解决方案	15
1.4.4 运行解决方案	15
1.5 Visual C++ 2005 中的调试工具	16
1.5.1 程序故障	16
1.5.2 调试器	16
1.5.3 设置断点	17
1.5.4 设置跟踪点	18
1.5.5 启动调试模式	19
1.5.6 检查和修改变量的值	20
第 2 章 在 Dev C++ 中编写 C 程序	22
2.1 Dev C++ 简介	22
2.2 Dev C++ 安装	22
2.2.1 Dev C++ 要求的系统配置	22
2.2.2 安装 Dev C++	23



2.3 使用 Dev C++ 编写 C 程序	29
2.3.1 启动进入 Dev C++	29
2.3.2 创建新的工程及源程序	30
2.3.3 编辑已存在的工程及源程序	33
2.3.4 编译和连接	33
2.3.5 运行程序	35
2.4 Dev C++ 中调试工具	35
2.4.1 设置与调试有关的选项	35
2.4.2 调试工具	35
第 3 章 Turbo C 2.0 编译系统	37
3.1 Turbo C 2.0/3.0 简介	37
3.2 Turbo C 2.0 安装	37
3.2.1 Turbo C 2.0 要求的系统配置	37
3.2.2 安装 Turbo C 2.0	37
3.3 Turbo C 2.0 主界面简介	38
3.3.1 启动进入 Turbo C 2.0	38
3.3.2 菜单栏	38
3.3.3 编辑区	39
3.3.4 信息提示区	39
3.3.5 快捷提示区	40
3.3.6 退出 Turbo C	40
3.3.7 Turbo C 的工作准备	40
3.4 使用 Turbo C 2.0 编写 C 程序	40
3.4.1 创建新的源程序	40
3.4.2 编辑已存在的源程序	41
3.4.3 编译和连接	41
3.4.4 运行程序	42
3.5 Turbo C 2.0 中调试工具	43
3.5.1 断点调试模式	43
3.5.2 单步调试模式	43
3.5.3 查看并修改变量值	43
3.5.4 设置监视窗口	43
3.5.5 终止调试模式	44
第 4 章 在 Unix/Linux 中编写 C 程序	45
4.1 Unix/Linux 简介	45
4.2 cc 编译命令和 gcc 编译器	46
4.2.1 cc 编译命令	46
4.2.2 gcc 编译器	49

4.3 在 Unix/Linux 中编写 C 程序	50
4.3.1 创建并编辑源程序文件	50
4.3.2 编译和连接	51
4.3.3 运行程序	51
第 5 章 软件测试	52
5.1 软件测试的基本概念	52
5.1.1 软件测试和程序调试的区别	52
5.1.2 软件测试的基本概念	52
5.2 软件测试的基本方法	53
5.2.1 白盒法	54
5.2.2 黑盒法	56
5.3 软件测试的实施	56
第 6 章 上机实验安排	58
上机指导 1 使用常用 C 编译环境编写 C 程序	58
上机指导 2 数据、类型和运算	60
上机指导 3 顺序结构程序设计	62
上机指导 4 流程控制	65
上机指导 5 函数	68
上机指导 6 程序测试与调试	73
上机指导 7 数组	74
上机指导 8 指针	76
上机指导 9 结构、联合、枚举和 typedef	78
上机指导 10 流与文件	84
上机指导 11 综合程序设计	88

第二部分 C 语言编程高级篇

第 7 章 文本界面设计	93
7.1 文本方式的控制	93
7.1.1 文本方式控制	93
7.1.2 文本方式颜色控制	94
7.1.3 字符显示亮度控制	95
7.1.4 清屏函数	96
7.1.5 光标操作	96
7.2 窗口设置和文本输出函数	97
7.2.1 窗口设置函数	97
7.2.2 控制台文本输出函数	97
7.2.3 状态查询函数	98
7.3 文本移动和存取函数	99



7.3.1 文本移动	100
7.3.2 文本存取	100
7.4 文本方式创建亮条式菜单	100
第8章 图形图像处理	106
8.1 图形图像的基本知识	106
8.1.1 图形显示的坐标	106
8.1.2 像素	107
8.1.3 有关坐标位置的函数	107
8.2 图形方式的控制	107
8.2.1 图形系统的初始化	107
8.2.2 退出图形状态	110
8.2.3 独立图形运行程序的建立	110
8.2.4 恢复显示方式和清屏函数	111
8.2.5 图形方式下的颜色控制函数	111
8.2.6 图形窗口和图形屏幕函数	112
8.3 图形函数	115
8.3.1 基本图形函数	115
8.3.2 封闭图形的填充	117
8.3.3 设定线型	120
8.4 图形方式下的文本输出	121
8.4.1 文本输出函数	121
8.4.2 格式化输出字符串函数	122
8.4.3 定义文本字型	122
8.5 动画技术	125
8.5.1 动态开辟图视口的方法	125
8.5.2 利用显示页和编辑页交替变化	125
8.5.3 利用画面存储再重放技术	126
8.5.4 利用对图像动态存储器进行操作	127
8.6 电子时钟	128
第9章 中断技术	140
9.1 中断的基本概念	140
9.1.1 BIOS	140
9.1.2 中断和异常	141
9.1.3 BIOS 功能调用	143
9.2 鼠标和键盘中断	144
9.2.1 鼠标的 INT33H 功能调用	144
9.2.2 常用鼠标功能函数	147
9.3 键盘编程	151

9.3.1 键盘扫描码	151
9.3.2 键盘缓冲区	154
9.3.3 键盘操作函数 bioskey()	154
第 10 章 网络通信编程	155
10.1 Winsock 编程基础	155
10.1.1 常用协议报头	155
10.1.2 Winsock 基础	157
10.1.3 套接字选项	160
10.1.4 名字解析	162
10.2 串口编程和并口编程	163
10.2.1 基本概念	163
10.2.2 串行接口和串行通信	163
10.2.3 并行接口和并行通信	165
10.2.4 串/并口的输入输出函数	166
10.3 实现 Ping 命令	166
第 11 章 C99 标准	182
11.1 C99 简介	182
11.1.1 C99 和 C89 的差异	182
11.1.2 对 C99 的支持	183
11.2 新的内置数据类型	184
11.2.1 _Bool	184
11.2.2 _Complex 和 _Imaginary	184
11.2.3 long long int 类型	185
11.3 扩展的整数类型	185
11.4 注释、变量定义和运算的修改	185
11.4.1 单行注释	185
11.4.2 分散代码和声明	185
11.4.3 在 for 循环中定义变量	186
11.4.4 复合赋值	186
11.5 用 restrict 修饰的指针	187
11.6 对数组的增强	187
11.6.1 变长数组	187
11.6.2 类型修饰符在数组声明中的应用	188
11.6.3 柔性数组结构成员	189
11.7 对函数的修改	189
11.7.1 inline	189
11.7.2 不再支持隐含的 int	190
11.7.3 删除了隐含的函数声明	190



11.7.4 对返回值的约束	190
11.7.5 __func__ 预定义标识符	190
11.8 预处理命令的修改	191
11.8.1 变元表	191
11.8.2 _Pragma 操作符	191
11.8.3 内置的编译指令 (Pragmas)	191
11.8.4 增加的内置宏	192
11.9 C99 中的新库	192

第三部分 习题参考答案

第 1 章 程序设计概述	195
第 2 章 数据、类型和运算	197
第 3 章 简单程序设计	203
第 4 章 流程控制	209
第 5 章 函数	223
第 6 章 程序设计方法概述	238
第 7 章 数组	244
第 8 章 指针	261
第 9 章 结构、联合、枚举和 typedef	276
第 10 章 流与文件	287
第 11 章 问题求解策略和算法设计	293
参考文献	305

第一部分 C语言上机指导

第一部分的主题是 C 语言上机指南，详细地介绍编辑、编译 C 程序的上机过程，以及软件测试的基本方法，并给出上机实验安排和上机实验的要求。

第一部分的主要内容包括以下方面：

- 常见的四种 C 编译环境：Visual C++2005、Dev C++、Turbo C2.0 和 Unix/Linux 操作系统中 C 程序的编写。
- 软件测试的基本方法。
- 本书配套教材的 11 次实验安排。



第1章 在 Visual C++ 2005 中编写 C 程序



本章将介绍 Visual C++2005 的集成开发环境, 以及在 Visual C++2005 中编辑、编译和连接并执行 C 程序的方法。

1.1 Visual Studio 2005 简介

1.1.1 Visual Studio 2005 简介

Visual Studio 2005 是 Microsoft 出品的一套完整的开发工具集, 用于生成 ASP.NET Web 应用程序、XML Web Services、桌面应用程序和移动应用程序。Visual Studio 2005 提供的版本包括: 标准版、专业版和 Team System 版。其中, Visual Studio 2005 Team System 是一个高效、集成且可扩展的软件开发生命周期工具平台, 可以帮助软件团队提高整个软件开发过程中的通信和协作能力。

Visual Studio 2005 的标准版和专业版包含多种核心语言: Visual Basic、Visual C++、Visual C#和 Visual J#以及 Visual Web Developer。这些核心语言全都使用相同的集成开发环境 (IDE), 利用此 IDE 可以共享工具且有助于创建混合语言解决方案。另外, 这些语言利用了 .NET Framework 的功能, 通过此框架可使用简化 ASP Web 应用程序和 XML Web Services 开发的关键技术。

1.1.2 Visual C++ 2005 简介

Visual C++ 2005 是 Visual Studio 2005 中的一个集成开发工具, 它可用于创建基于 Microsoft Windows 和基于 Microsoft.NET 的应用程序。它既可以用做集成开发系统, 也可以用做一组独立的工具。

Visual C++ 2005 包含下列组件:

- (1) Visual C++ 2005 编译器工具。该编译器包含一些新功能, 支持面向虚拟计算机平台 (如公共语言运行库 (CLR)) 的开发人员。现在已经有面向 x64 和 Itanium 的编译器。
- (2) Visual C++ 2005 库。其中包括行业标准活动模板库 (ATL)、Microsoft 基础类 (MFC) 库, 以及各种标准库, 如标准 C++库和 C 运行时库 (CRT), 并新增了 C++支持库。
- (3) Visual C++ 2005 开发环境。提供对项目管理与配置、源代码编辑、源代码浏览和调试工具的强大支持。该环境还支持 IntelliSense, 在编写代码时可以提供智能化且特定于上下文的建议。



1.2 Visual Studio 2005 安装

1.2.1 Visual Studio 2005 要求的系统配置

Visual C++ 2005 的不同版本对硬件和系统配置的要求差别较小, 总体要求如下:

- (1) 处理器要求: 600 MHz CPU, 建议 1 GHz CPU。
- (2) 内存要求: 192 MB 内存, 建议 256M 内存。
- (3) 可用硬盘空间: 不含 MSDN 时, 系统驱动器上需要 1 GB 的可用空间, 安装驱动器上需要 2 GB 的可用空间。包含 MSDN 时, 系统驱动器上需要 1 GB 的可用空间, 完整安装 MSDN 的安装驱动器上需要 3.8 GB 的可用空间, 默认安装 MSDN 的安装驱动器上需要 2.8 GB 的可用空间。
- (4) 显示器: 800×600, 256 色; 建议 1024×768, 增强色 16 位。
- (5) 支持的操作系统: Windows® 2000 Service Pack 4、Windows XP Service Pack 2、Windows Server 2003 Service Pack 1 或更高版本。对于 64 位计算机, 要求 Windows Server 2003 Service Pack 1 x64 版本、Windows XP Professional x64 版本。

1.2.2 安装 Visual Studio 2005

1. 安装 Visual Studio 2005 开发工具

(1) 将 Visual Studio 2005 安装盘放入计算机的光盘驱动器中, 光盘运行后会自动进入安装程序界面, 如图 1-1 所示。该界面中包含 3 个安装选项供用户选择, 分别是: 安装 Visual Studio 2005、更改或移除产品文档和检查 Service Release。应用 Visual Studio 2005 开发环境开发程序, 只需安装前两项即可。这里选择“安装 Visual Studio 2005”, 出现如图 1-2 所示的 Visual Studio 2005 安装向导界面。



图 1-1 Visual C++2005 安装程序界面



图 1-2 Visual C++2005 安装向导界面



(2) 在如图 1-2 所示的 Visual Studio 2005 安装向导界面中, 单击“下一步”, 安装程序会跳转到“Visual Studio 2005 安装程序——起始页”界面。该界面左侧显示关于 Visual Studio 2005 安装程序的所需组件信息, 右侧显示用户许可协议。选中“我接受许可协议中的条款”, 单击“下一步”按钮, 安装程序会跳转到“Visual Studio 2005 安装程序——选项页”界面, 如图 1-3 所示。用户可对自己需要的功能进行选择, 并对产品安装路径进行设置, 通常用户选择要安装的功能为默认值, 产品默认路径为“C:\Program Files\Microsoft Visual Studio 8\”。



图 1-3 Visual Studio 2005 安装程序——选项页

(3) 依据图 1-3 的提示, 确认安装目标磁盘具有足够的空闲空间, 然后单击“安装”按钮。安装程序出现如图 1-4 所示的“Visual Studio 2005 安装程序——安装页”界面, 该界面中显示正在安装组件。

(4) Visual Studio 2005 安装完成后, 将弹出如图 1-5 所示的“Visual Studio 2005 安装程序——安装成功”界面。



图 1-4 Visual Studio 2005 安装程序——安装页

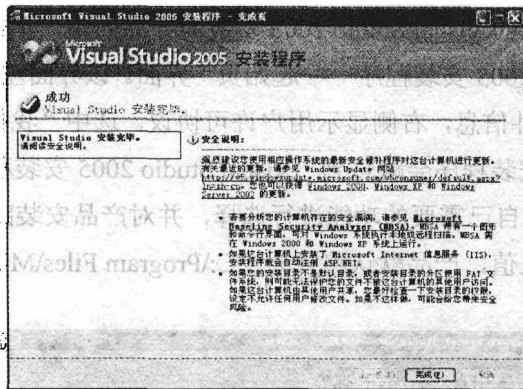


图 1-5 Visual Studio 2005 安装程序——安装成功

(5) 单击“完成”按钮，结束安装 Visual Studio 2005 开发工具的过程。

2. 启动并配置 Visual Studio 2005 默认环境

(1) 启动 Visual Studio 2005。

在 Windows 操作系统中启动 Visual Studio 2005 开发环境的方法主要有两种：

①单击“开始”菜单，依次选择“程序”/Microsoft Visual Studio 2005/Microsoft Visual Studio 2005 选项，便可启动 Visual Studio 2005。

②如果在安装 Visual Studio 2005 开发环境的时候在桌面上创建了快捷方式，可直接双击该快捷方式图标，即可启动 Visual Studio 2005 开发环境。

(2) 配置 Visual Studio 2005 默认环境。

第一次运行 Visual Studio 2005 开发环境时，系统会弹出“选择默认环境设置”对话框，在该对话框中，用户可以根据自己的实际情况，选择适合自己的开发语言，这里选择的是“Visual C++开发设置选项”，即使用 C++语言进行开发，然后单击“启动 Visual Studio (S)”按钮即可。

(3) 修改 Visual Studio 2005 默认环境。

在使用 Visual Studio 2005 开发环境时，如果希望修改默认环境设置，可单击“工具”菜单项中的“导入和导出设置”，将弹出如图 1-6 所示的“导入和导出设置向导”界面。

选中“重置所有设置”，单击“下一步”按钮，将弹出如图 1-7 所示的“导入和导出设置

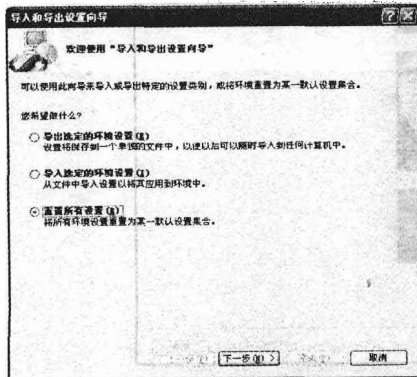


图 1-6 导入和导出设置向导

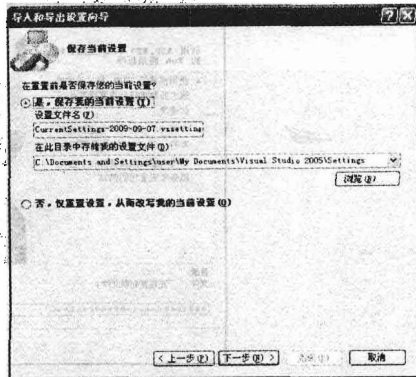


图 1-7 导入和导出设置向导——保存当前设置



向导——保存当前设置”界面，单击“下一步”按钮，将弹出如图 1-8 所示的“导入和导出设置向导——选择一个默认设置集合”界面，这时可选择你希望使用的语言环境，然后点击“完成”按钮即可。

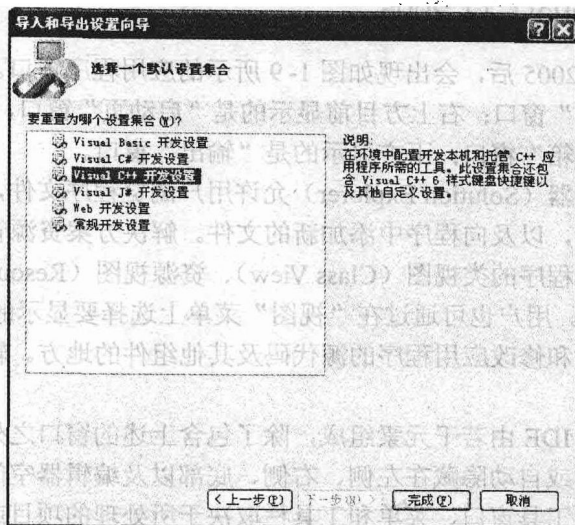


图 1-8 导入和导出设置向导——选择一个默认设置集合

1.2.3 安装 MSDN

MSDN（微软开发网络，Microsoft Development Network）库提供了关于 Visual C++2005 的所有功能以及更多其他主题的详尽的参考资料。当安装 Visual C++2005 时，将出现安装部分或全部文档的选项，如果磁盘空间够用，建议安装 MSDN 库。安装 MSDN 库的过程较为简单，读者可按照安装向导的提示进行即可。

启动 Visual C++2005 之后，可以通过按 F1 按钮或者“帮助”菜单浏览 MSDN 库，MSDN 库不仅提供参考文档，而且在处理代码中的错误时也会是很有用的工具。

1.3 集成开发环境 IDE 简介

随 Visual C++2005 一起提供的 IDE（集成开发环境，Integrated Development Environment）是一个用于创建、编译、连接和测试 C/C++ 程序的完全独立的环境，所有程序的开发和执行都是在 IDE 内完成的。

Visual C++2005 的集成开发环境 IDE 的基本部件包括编辑器、编译器、连接器和库。这些都是编写和执行 C/C++ 所必需的基本工具。

（1）编辑器。编辑器给用户创建和编辑 C/C++ 源代码的交互式环境，该编辑器可通过文字的颜色来区分不同的语言元素，有助于提高代码的可读性。

（2）编译器。编译器负责检测并报告编译过程中的错误，并将源代码转换为目标代码。编译器输出的目标代码通常使用以 .obj 为扩展名的名称。

（3）连接器。连接器将编译器根据源代码文件生成的各种模块组合成可执行的整体。连