

可下载教学资料

<http://www.tup.tsinghua.edu.cn>

21世纪普通高校计算机公共课程规划教材

# 程序设计基础 (C语言版)(第2版) 实验指导与习题

杨盛泉 丁琦 乔奎贤 主编

刘白林 主审



清华大学出版社

21世纪普通高校计算机公共课程规划教材

# 程序设计基础 (C语言版)(第2版) 实验指导与习题

杨盛泉 丁琦 乔奎贤 主编  
刘白林 主审

清华大学出版社  
北京

## 内 容 简 介

本书是按照教育部新世纪人才创新项目教材编写要求编写的,是与《程序设计基础(C语言版)第2版》配套的实验指导与习题。

本书内容共分4部分:第1部分详细介绍C语言程序设计上机实验的步骤以及Visual C++运行环境,并就编程中常见的问题进行解析;第2部分为课程配套实验,按照教材要求,安排10个实验,每个实验都给出了实验目的以及重点、难点,循序渐进地安排实验题目,从示例到习题,便于学生掌握每个章节的理论及编程技巧;第3部分为课程配套习题,针对教材的每个章节,安排了多种形式的习题,便于学生巩固知识点;第4部分为综合模拟试题,既让学生检验自己的水平、训练学生的综合应用能力,又有利于学生适应C程序设计的考试(比如等级考试等)。

本书侧重实践能力的培养,叙述通俗易懂,可作为本科生的参考书和编程教材,也适合作为计算机软件编程人员和研究生学习C程序设计的入门教材。

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

程序设计基础(C语言版)第2版实验指导与习题/杨盛泉,丁琦,乔奎贤主编. —北京:清华大学出版社,2010.3

(21世纪普通高校计算机公共课程规划教材)

ISBN 978-7-302-22054-1

I. ①程… II. ①杨… ②丁… ③乔… III. ①C语言—程序设计—高等学校—教学参考资料 IV. ①TP312

中国版本图书馆CIP数据核字(2010)第026048号

责任编辑:梁颖 赵晓宁

责任校对:白蕾

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

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)

印 装 者:北京市清华园胶印厂

经 销:全国新华书店

开 本:185×260 印 张:6.75 字 数:156千字

版 次:2010年3月第1版 印 次:2010年3月第1次印刷

印 数:1~4000

定 价:13.00元

---

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:036529-01

# 出版说明

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,以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以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。在制定教材规划时要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。书稿完成后要认真实行审稿程序,以确保出书质量。

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队伍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世纪普通高校计算机公共课程规划教材编委会

联系人:梁颖 liangying@tup.tsinghua.edu.cn

# 前 言

---

C 语言程序设计是实践性很强的过程,任何程序最终都必须在计算机上运行,以检验程序的正确与否。因此,在学习程序设计中,上机实践环节是至关重要的。

本书是按照教育部新世纪人才创新项目教材编写要求编写的,是与《程序设计基础(C 语言版)第 2 版》配套的实验指导与习题。在参照有关纲要的同时,结合 C 程序设计的特点,将实践能力培养放在首位。

本书在内容组织上,力求做到先进、简单、实用。

本书配合主教材,包含 4 部分内容:第 1 部分详细介绍 C 语言程序设计上机实验的步骤以及 Visual C++ 运行环境,并就编程中常见的问题进行解析;第 2 部分为课程配套实验,按照教材要求,安排 10 个实验,每个实验都给出了实验目的以及重点、难点,循序渐进地安排实验题目,从示例到习题,便于学生掌握每个章节的理论及编程技巧;第 3 部分为课程配套习题,针对教材的每个章节,安排了多种形式的习题,便于学生巩固知识点;第 4 部分为综合模拟试题,既可让学生检验自己的水平、训练学生的综合应用能力,又有利于学生适应 C 程序设计的考试(比如等级考试等)。

在编写本书的过程中,参考了大量的相关资料,从中汲取了许多宝贵经验,在此谨表谢意。由于编者水平有限,书中的不妥和错误在所难免,恳请各位专家、读者不吝指正。

本书习题答案暂不附上。如有需要,可联系编者。

编者

2010 年 2 月

# 目 录

---

<b>第 1 部分 C 语言上机步骤以及运行环境</b> .....	1
1.1 C 语言程序的上机步骤 .....	1
1.2 Visual C++ 编程工具 .....	3
1.2.1 Visual C++ 语言集成环境简介 .....	4
1.2.2 Visual C++ 语言编程过程 .....	4
<b>第 2 部分 C 语言实验</b> .....	12
2.1 实验报告要求 .....	12
2.2 实验 1 熟悉 VC++ 语言运行环境 .....	12
1. 实验目的 .....	12
2. 实验重点 .....	13
3. 实验难点 .....	13
4. 实验内容 .....	13
5. 讨论与总结 .....	14
2.3 实验 2 数据类型、运算符和表达式 .....	14
1. 实验目的 .....	14
2. 实验重点 .....	14
3. 实验难点 .....	15
4. 实验内容 .....	15
5. 讨论与总结 .....	19
2.4 实验 3 顺序结构程序设计 .....	20
1. 实验目的 .....	20
2. 实验重点 .....	20
3. 实验难点 .....	20
4. 实验内容 .....	20
5. 讨论与总结 .....	22
2.5 实验 4 选择结构程序设计 .....	23
1. 实验目的 .....	23
2. 实验重点 .....	23
3. 实验难点 .....	23

4. 实验内容 .....	23
5. 讨论与总结 .....	26
2.6 实验5 循环结构程序设计 .....	26
1. 实验目的 .....	26
2. 实验重点 .....	26
3. 实验难点 .....	27
4. 实验内容 .....	27
5. 讨论与总结 .....	29
2.7 实验6 数组 .....	29
1. 实验目的 .....	29
2. 实验重点 .....	29
3. 实验难点 .....	30
4. 实验内容 .....	30
5. 讨论与总结 .....	34
2.8 实验7 函数 .....	34
1. 实验目的 .....	34
2. 实验重点 .....	34
3. 实验难点 .....	34
4. 实验内容 .....	34
5. 讨论与总结 .....	39
2.9 实验8 指针 .....	39
1. 实验目的 .....	39
2. 实验重点 .....	39
3. 实验难点 .....	39
4. 实验内容 .....	39
5. 讨论与总结 .....	46
2.10 实验9 结构体和共用体 .....	47
1. 实验目的 .....	47
2. 实验重点 .....	47
3. 实验难点 .....	47
4. 实验内容 .....	47
5. 讨论与总结 .....	53
2.11 实验10 文件 .....	53
1. 实验目的 .....	53
2. 实验重点 .....	53
3. 实验难点 .....	53
4. 实验内容 .....	53
5. 讨论与总结 .....	56



<b>第 3 部分 C 语言习题</b>	57
3.1 C 语言初步	57
1. 选择题	57
2. 填空题	58
3. 简答题	58
4. 编程题	59
3.2 数据类型、运算符与表达式	59
1. 选择题	59
2. 填空题	60
3. 简答题	61
4. 编程题	61
3.3 简单 C 程序设计	61
1. 选择题	61
2. 填空题	64
3. 程序改错题	65
4. 编程题	66
3.4 程序结构	66
1. 选择题	66
2. 填空题	68
3. 程序分析题	69
4. 程序设计题	69
3.5 数组	70
1. 选择题	70
2. 填空题	71
3. 程序分析题	72
4. 程序设计题	73
3.6 函数	73
1. 选择题	73
2. 填空题	74
3. 程序分析题	75
4. 程序设计题	76
3.7 指针	76
1. 选择题	76
2. 填空题	77
3. 程序分析题	78

2. 程序分析题 .....	80
3. 程序设计题 .....	81
3.9 文件.....	81
1. 选择题 .....	81
2. 填空题 .....	82
<b>第4部分 综合模拟测试题 .....</b>	<b>83</b>
4.1 模拟试题1 .....	83
4.2 模拟试题2 .....	86
4.3 模拟试题3 .....	89
<b>参考文献 .....</b>	<b>93</b>

# 第 1 部分

## C 语言上机步骤以及运行环境

程序设计是实践性很强的过程,任何程序最终都必须在计算机上运行,以检验程序的正确与否。通过上机练习可以加深理解 C 语言的有关概念,巩固理论知识;另一方面也可以培养程序调试的能力与技巧。因此,在学习程序设计中,一定要重视上机实践环节。

### 1.1 C 语言程序的上机步骤

编写一个 C 程序直到完成运行,一般要经过以下几个步骤:开机进入 C 语言编辑环境→输入与编辑源程序→对源程序进行编译→产生目标代码→链接各个目标代码、库函数→产生可执行程序→运行程序。

这里 C 程序经过 4 个重要的阶段,即编辑(Edit)、编译(Compile)、链接(Link)和装入执行(Load & Excute)。

#### 1. 编辑阶段

第一个阶段是编辑文件,是用编辑程序(editor program)完成的。按照 C 语言语法规则而编写的 C 程序称为源程序。源程序由字母、数字及其他符号等构成,在计算机内部用相应的 ASCII 码表示,并保存在扩展名为 C 的文件中。

程序员用编辑器输入 C 程序,进行必要的修改,然后将程序存放在磁盘之类的辅助存储设备中。个人计算机上的 TC、Borland C 和 Microsoft Visual C++ 等 C 程序开发软件包都有自己的编辑器,它们与编程环境紧密集成。这里,假设读者已经知道如何编辑程序。

#### 2. 编译阶段

源程序是无法直接被计算机运行的,因为计算机的 CPU 只能执行二进制的机器指令。这就需把 ASCII 码的源程序先翻译成机器指令,然后计算机的 CPU 才能运行翻译好的程序。

在 C 语言系统中,预处理程序在编译器翻译开始之前自动执行。C 预处理器采用预处理指令(Preprocess Directive)表示程序编译之前要进行某些操作。这些操作通常包括在要编译的文件中包含其他文本文件和进行各种文本替换。编辑器在将程序翻译为机器语言代码之前调用预处理器。

程序员发出编译(Compile)程序的命令后,编译器将 C 程序预处理并翻译为机器语言代码(也称为目标程序)。

#### 3. 链接阶段

目标程序并不能交计算机直接运行,因为在源程序中,输入、输出以及常用函数运算并不是用户自己编写的,而是直接调用系统函数库中的库函数。因此,必须把“库函数”的处理

过程链接到经编译生成的目标程序中,从而生成可执行程序。

链接器(Linker)将目标码与这些默认功能的代码链接起来,建立执行程序映像(不再缺少任何代码)。

#### 4. 装入执行阶段

下一个阶段是装入与执行。在执行之前,要先把 exe 程序放入内存中,这是由装入器(Loader)完成的,装入器读取磁盘中执行程序的映像文件,并将其放入内存中。最后,计算机在 CPU 控制下逐条指令地执行程序,最终得到结果。

C 语言程序经过编辑、编译、链接到运行的全过程如图 1.1 所示。

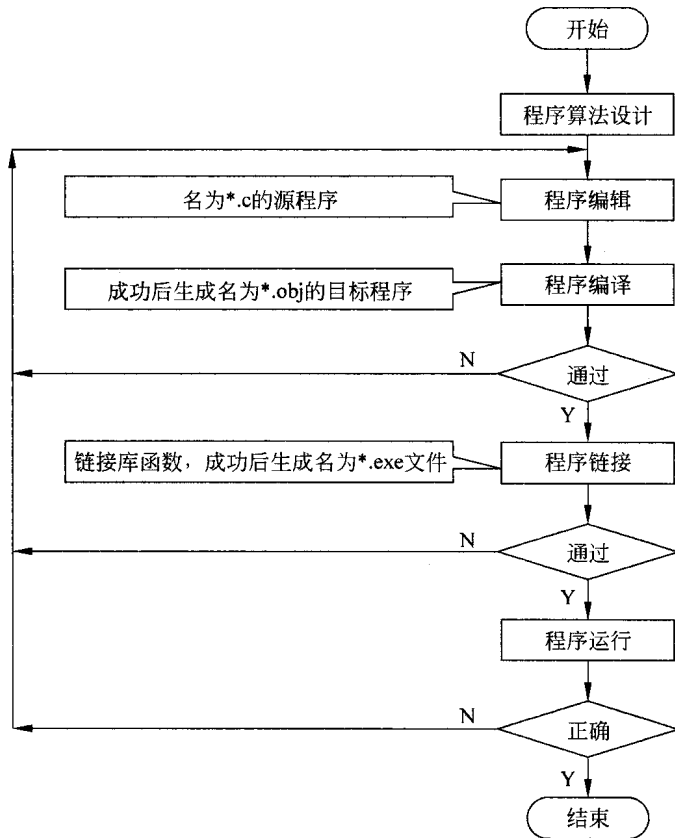


图 1.1 C 语言编辑运行步骤

程序不是一次就能够顺利运行的。上述每个阶段都可能因为各种错误而失败。例如,除数可能为 0(计算机上的非法操作与算术运算中一样),这样就会使计算机程序结果出错。然后,程序员需返回到编辑阶段,进行必要的修改并继续操作其余阶段,确定修改之后能否顺利工作。

除了较简单的情况,一般程序很难一次就能做到完全正确。在上机过程中,根据出错现象找出错误并改正称为程序调试。要在学习程序设计过程中,逐步培养调试程序的能力。这不可能靠几句话就能讲清楚,要靠自己在上机中不断摸索总结,也可以说是一种经验积累。

程序中的错误大致可分为 3 类：

- (1) 程序编译时检查出来的语法错误。
- (2) 链接时出现的错误。
- (3) 程序执行过程中的错误。

编译错误通常是编程者违反了 C 语言的语法规则,如保留字输入错误、括号不匹配、语句缺少分号等。链接错误一般由未定义或未指明要链接的函数,或者函数调用不匹配等因素引起。

对于编译连接错误,C 语言系统会提供出错信息,包括出错位置(行号)、出错提示信息。编程者可以根据这些信息找出错误所在。有时系统提示的一大串错误信息并不表示真的有这么多的错误,往往是因为前面的一两个错误引起的连锁反应。所以当你纠正了几个错误后,不妨再编译连接一次,然后根据最新的出错信息继续纠正。

有些程序通过了编译连接并能够在计算机上运行,但得到的结果不正确,这类在程序执行过程中的错误往往最难改正。错误的原因一部分是程序书写错误带来的,例如应该使用变量  $x$  的地方写成了变量  $y$ ,虽然没有语法错误,但意思完全错了;另一部分可能是程序的算法不正确,解题思路不对。还有一些程序有时计算结果正确,有时不正确,这往往是编程时,对各种情况考虑不周所致。解决运行错误的首要步骤就是错误定位,即找到出错的位置,才能予以纠正。通常我们先设法确定错误的大致位置,然后通过 C 语言提供的调试工具找出真正的错误。

为了确定错误的大致位置,可以先把程序分成几大块,并在每一块的结束位置手工计算一个或几个阶段性结果,然后用调试方式运行程序,在每一块结束时,检查程序运行的实际结果与手工计算是否一致,通过这些阶段性结果来确定各块是否正确。对于出错的程序块,可逐条仔细检查各语句,找出错误所在。如果出错块程序较长,难以快速找出错误,可以进一步把该块细分成更小的块,按照上述步骤进一步检查。在确定大致出错位置后,如果无法直接看出错误,可以通过单步运行相关位置的几条语句,逐条检查,这样一定能找出错误的语句。

当程序出现计算结果有时正确有时不正确的情况时,其原因一般是算法对各种数据处理情况考虑不全面。解决办法最好多选几组典型的输入数据进行测试,除了普通的数据外,还应包含一些边界数据和错误的输入数据。例如,确定正常的输入数据范围后,分别以最小值、最大值、比最小值小的值和比最大值大的值,多方面运行检查自己的程序。

下面以 VC++ 6.0 为上机平台,对 C 程序编译、链接和调试作简单介绍。建议一开始学习上机时,把注意力放在程序的编译、链接和运行上,以能运行为目标。

## 1.2 Visual C++ 编程工具

C++ 语言是在 C 语言的基础上发展而来的,它增加了面向对象的编程,成为当今最流行的一种程序设计语言。Visual C++ 是微软公司开发的,面向 Windows 编程的 C++ 语言工具。它不仅支持 C++ 语言的编程,也兼容 C 语言的编程。由于 VC++ 被广泛地用于各种编程,因此使用面很广。

### 1.2.1 Visual C++ 语言集成环境简介

现在 C++ 语言有两大非常有名且应用非常广泛的开发工具,一种就是 Microsoft 的 Visual C++,另一种是 Borland 的 C++ Builder (BCB),它们的功能都非常强大,而且各有各的优缺点。Visual C++ 6.0 是微软公司的重要产品——Visual Studio 工具集的组成部分。它用来在 Windows (包括 Windows 95、Windows 98、Windows NT、Windows 2000、Windows XP 等)环境下开发应用程序,是一种功能强大、行之有效的可视化编程工具,成为广大软件开发人员的首选。

Visual C++ 是美国 Microsoft 公司推出的功能强大的应用程序开发工具,是一款功能超群、使用方便、易于开发复杂应用系统的 C++ 或者传统 C 程序开发工具,它不仅可以用来编写系统程序,也可以用来编写应用程序。利用 Visual C++ 所提供的丰富而完善的开发工具,可以轻松地开发大型 C++ 或者 C 应用系统。Visual C++ 采用的图形界面使得程序员能够迅速方便地开发出相互独立的对象,而这些对象可供程序员共享或重复使用。Visual C++ 自问世以来受到了应用软件开发人员的重视,其主要的原因在于它具有以下特点:

- (1) 开发效率高,成本低。
- (2) 面向对象的开发工具,代码的可重复性好,开发的软件易于维护。
- (3) 客户机/服务器计算模式的前端工具,对数据库的应用开发有着特殊的支持,特别适合做信息系统的开发。
- (4) 提供了丰富的对象、控件和函数,为开发人员提供了良好的用户界面,并为编制功能强大的应用软件创造了便利条件。

### 1.2.2 Visual C++ 语言编程过程

下面简要介绍在 Windows 环境下使用 Visual C++ 6.0 编辑与运行 C++ 程序的步骤:

- (1) 启动 Visual C++。
- (2) 创建一个新的 C 程序文件。
- (3) 编辑 C 程序代码。
- (4) 保存 C 程序文件。
- (5) 源程序文件的编译、连接。
- (6) 如果程序有语法错误,可参照“输出”窗口中给出的提示进行修改,然后重复前面的步骤,直至出错提示全部消除。
- (7) 运行可执行程序,如果发现运行错误,重复前面的步骤,直至结果正确为止。

#### 1. Visual C++ 6.0 安装与启动

现在常用的版本是 Visual C++ 6.0,虽然有些公司推出汉化版,但只是把菜单汉化了,并不是真正的中文版 Visual C++ 6.0,而且汉化的用词不准确,因此许多人还是使用英文版。如果计算机中未安装 Visual C++ 6.0,则应先安装 Visual C++ 6.0。Visual C++ 是 Microsoft Visual Studio 的一部分,因此需要找到 Visual Studio 的光盘,执行其中的 setup.exe,并按照屏幕上的提示进行安装即可。

安装 Visual C++ 后,在开始菜单中找到 Microsoft Visual Studio 6.0 中的程序组,可以看到 Visual C++ 6.0 的菜单项,如图 1.2 所示。

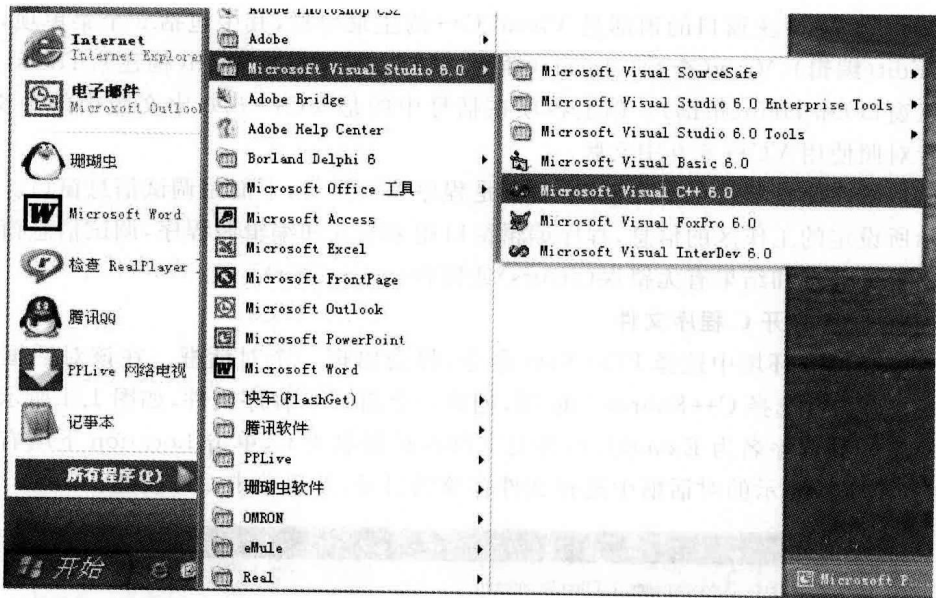


图 1.2 Visual C++ 6.0 启动

单击 Visual C++ 6.0 的图标,即可启动运行 Visual C++ 6.0 软件。此时屏幕上短暂显示 Visual C++ 6.0 的版权页后,出现 Visual C++ 6.0 主窗口,如图 1.3 所示,窗口中间出现“每日一帖”提示信息窗口。

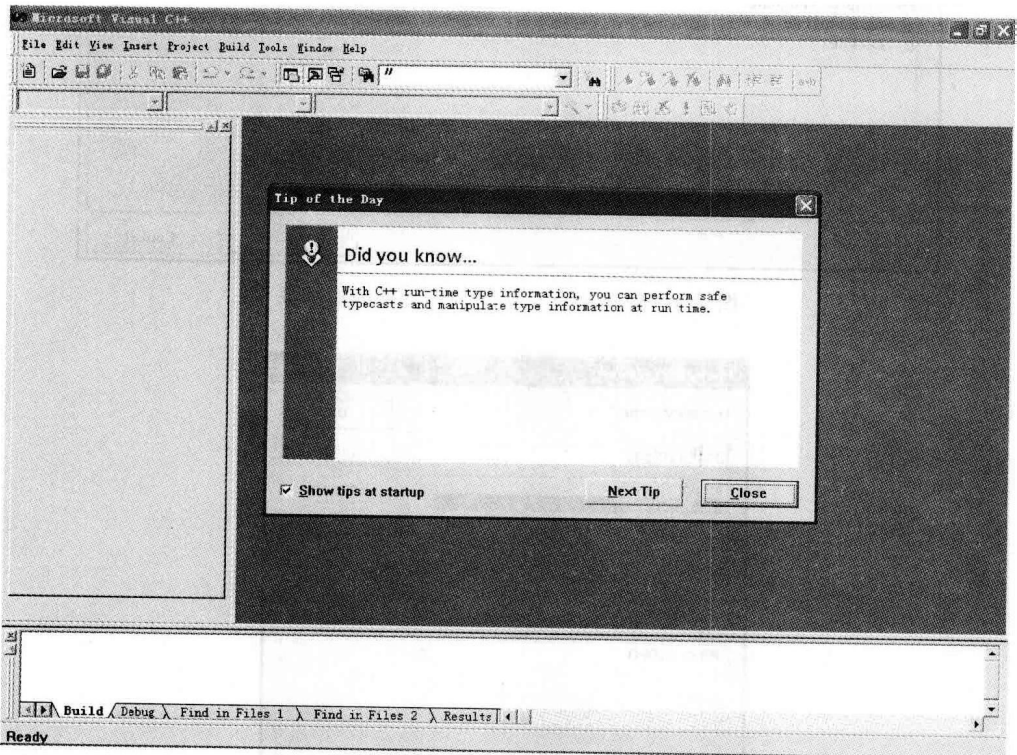


图 1.3 Visual C++ 6.0 开发集成环境

Visual C++ 6.0 主窗口的顶部是 Visual C++ 的主菜单栏,其中包括 9 个菜单项,即 File(文件)、Edit(编辑)、View(查看)、Insert(插入)、Project(项目)、Build(构建)、Tools(工具)、Window(窗口)和 Help(帮助)。以上各项在括号中的是 VC++ 6.0 中文版中的中文显示,便于读者对照使用 VC++ 6.0 中文版。

主窗口的左侧是项目工作区窗口,右侧是程序编辑窗口,下面是调试信息窗口。工作区窗口显示所设定的工作区的信息,程序编辑窗口用来输入和编辑源程序,调试信息窗口用来显示程序出错信息和结果有无错误(errors)或警告(warnings)。

## 2. 创建或者打开 C 程序文件

在 Visual C++ 环境中选择 File|New 命令,将会出现一个对话框。在该对话框中选择 Files 选项卡,然后选择 C++ Source File 项,创建一个新的 C 程序文件,如图 1.4 所示。输入本次要编辑的源程序名为 Exam01.c,并且文件的扩展名为 C,单击 Location 下端右边的...按钮,在如图 1.5 所示的对话框中选择文件存放的目录,然后单击 OK 按钮。

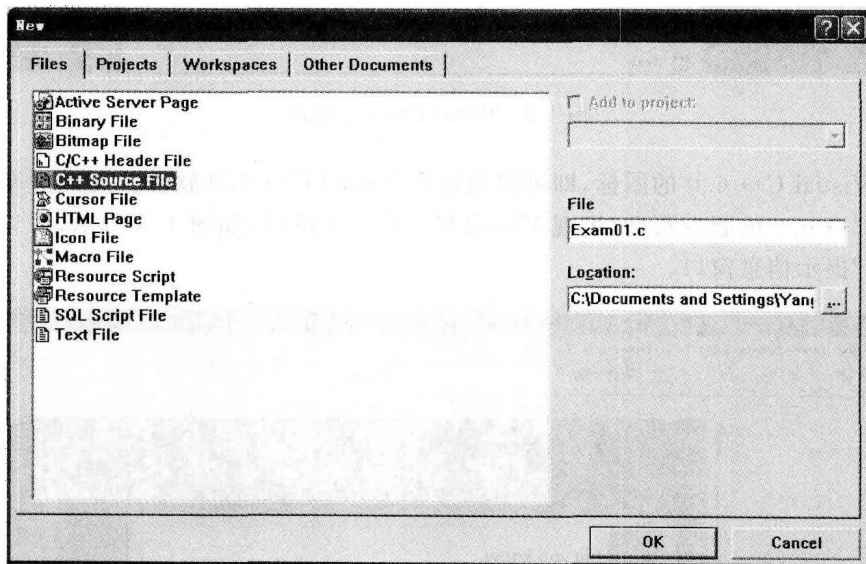


图 1.4 创建一个 C 语言程序文件的对话框

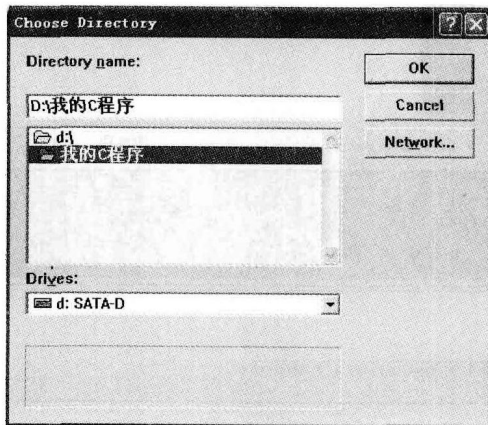


图 1.5 选择 C 语言程序文件存放目录的对话框



这里读者要注意的是,选择文件名时,应该指定扩展名为 C,否则系统将按 C++ 扩展名 CPP 进行保存。

如果程序已经输入过,可选择 File|Open 命令,并在查找范围中找到正确的文件夹,调入指定的程序文件。

### 3. 在编辑窗口中编写 C 程序

在程序编辑窗口中输入,一个字符大小写的程序,如图 1.6 所示。

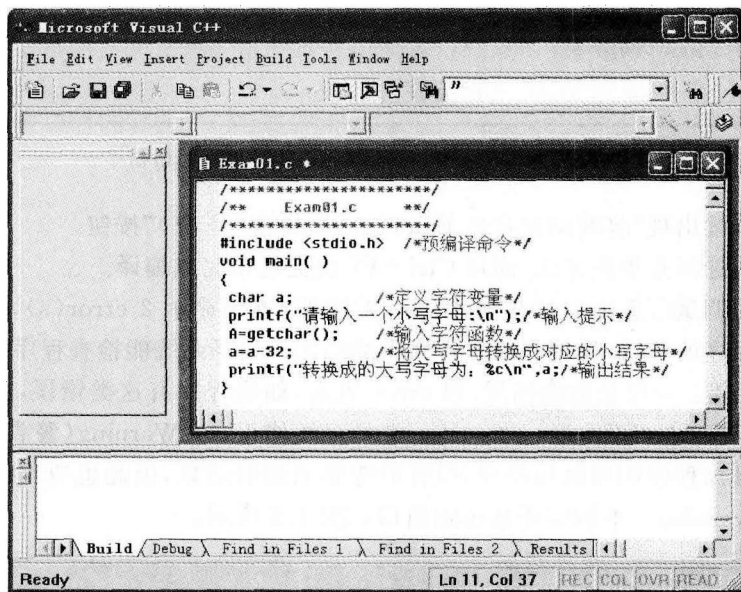


图 1.6 输入程序代码

为了看到更全面的调试程序过程,在输入本程序代码的过程中设置两个错误:一个是将变量 a 故意写成大写的字母 A;另一个错误是将 printf 函数的右括号故意漏掉。输入完毕后,选择 File|Save 命令,保存程序。

### 4. 编译链接程序

将程序保存后,就可以编译、链接 C 语言程序了。只有通过编译、连接,代码才能变成机器能够执行的同名 .exe 文件。

在系统的 Build 菜单项中,如图 1.7 所示,有 Build 选项,选择该选项。系统将执行对源程序编译、链接的任务,如果期间有错误,系统会停下来,提示用户出错。Build 将编译、链接合成到了一起自动依次执行。用户也可以单独先执行编译命令 Compile。

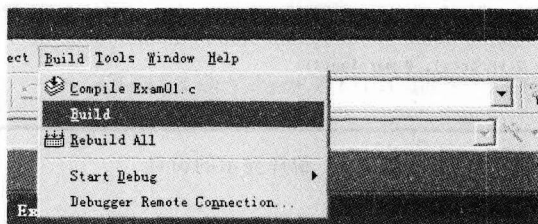


图 1.7 Build 菜单项