

普通高等教育“十二五”规划教材

PUTONG GAODENG JIAOYU SHIERWU GUIHUA JIAOCAI

C语言程序设计上机实验指导及习题

CYUYAN CHENGXU SHEJI SHANGJI SHIYAN ZHIDAO JI XITI

乔奎贤 杨盛泉 主编



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

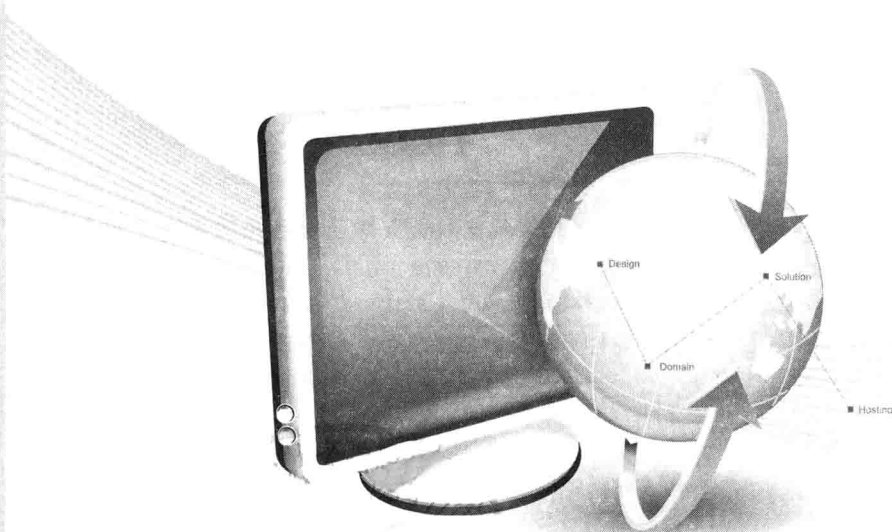
普通高等教育“十二五”规划教材

PUTONG GAODENG JIAOYU SHIERWU GUIHUA JIAOCAI

C语言程序设计上机实验指导及习题

CYUYAN CHENGXU SHEJI SHANGJI SHIYAN ZHIDAO JI XITI

主 编 乔奎贤 杨盛泉
副主编 丁 琦 周江卫 刘海泉
主 审 刘白林



北京航空航天大学出版社
BEIHANG UNIVERSITY PRESS

图书在版编目(CIP)数据

C 语言程序设计上机实验指导及习题/乔奎贤, 杨盛泉主编. —北京: 北京航空航天大学出版社, 2013. 2
ISBN 978-7-5124-1058-9

I. ①C… II. ①乔… ②杨… 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(2013)第 025689 号

版权所有, 侵权必究。

C 语言程序设计上机实验指导及习题

乔奎贤 杨盛泉 主编

责任编辑: 宋丽霞

责任校对: 杨小红

*

北京航空航天大学出版社出版发行

北京市海淀区学院路 37 号(邮编 100191) <http://www.buaapress.com.cn>

发行部电话: (010) 82317024 传真: (010) 82328026

读者信箱: bhpress@263.net 邮购电话: (010) 82316936

北京市彩虹印刷有限责任公司印装 各地书店经销

*

开本: 787×1 092 1/16 印张: 13 字数: 333 千字

2013 年 2 月第 1 版 2013 年 2 月第 1 次印刷

ISBN 978-7-5124-1058-9 定价: 26.00 元

前 言

C 语言程序设计是实践性很强的课程,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的过程中,上机实践是程序设计课程至关重要的教学环节。

本书按照教育部新世纪人才创新项目教材要求编写,是与刘白林主编的《C 语言程序设计基础》配套的上机指导及习题。在参照有关纲要的同时,结合 C 程序设计的特点,本书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《C 语言程序设计基础》教材,包含五部分内容:第一部分详细介绍了 C 语言程序设计上机实验的步骤以及运行环境 Visual C++,并就编程中常见的问题进行解析;第二部分为课程配套实验,按照教材要求,共安排了十个实验,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,便于学生学习掌握每个章节的理论及编程技巧;第三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;第四部分为综合模拟试题,既让学生检验自己的水平、训练学生的综合能力,又有利于学生适应 C 语言程序设计的考试(比如等级考试等);第五部分为各部分的参考答案。

本教材在编写过程中,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。由于作者水平有限,书中的不妥和错误在所难免,恳请各位专家、读者不吝指正。

编者

2013 年 1 月

目 录 CONTENTS

第一部分 C 语言程序上机步骤以及运行环境	1
1.1 C 语言程序的上机步骤	3
1.2 Visual C++ 编程工具	6
1.2.1 Visual C++ 语言集成开发环境简介	6
1.2.2 Visual C++ 语言编程过程	6
第二部分 C 语言实验	15
实验报告要求	16
实验 1 熟悉 Visual C++ 语言运行环境	17
实验 2 数据类型、运算符和表达式	19
实验 3 顺序结构程序设计	23
实验 4 选择结构程序设计	26
实验 5 循环结构程序设计	30
实验 6 数 组	33
实验 7 函 数	37
实验 8 指 针	42
实验 9 结构体和共用体	49
实验 10 文 件	56
第三部分 C 语言习题	59
第 1 章 C 语言初步	61
第 2 章 数据类型、运算符与表达式	64
第 3 章 简单 C 程序设计	67
第 4 章 程序结构	72
第 5 章 数 组	77
第 6 章 函 数	81
第 7 章 指 针	85
第 8 章 结构体与共用体	89
第 9 章 文 件	91

第四部分 综合模拟试题	93
综合模拟试题(一)	95
综合模拟试题(二)	99
综合模拟试题(三)	103
综合模拟试题(四)	106
综合模拟试题(五)	109
综合模拟试题(六)	114
第五部分 参考答案	119
一 C 语言实验参考答案	121
二 C 语言习题参考答案	158
三 综合模拟试题参考答案	188
附 录	195
附录 A 常用字符与 ASCII 代码对照表	195
附录 B C 语言中的关键字	196
附录 C 运算符和结合性	196
附录 D 常用的 C 库函数	197

第一部分

C 语言程序上机步骤以及运行环境

1.1 C 语言程序的上机步骤

编写一个 C 语言程序直到完成运行,一般要经过的几个步骤是: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→连接各个目标代码、库函数→产生可执行程序→运行程序。

这里,C 程序要经过 4 个重要的阶段:编辑(edit)、编译(compile)、连接(link)、装入与执行(load & excute)。

一、编辑阶段

第一个阶段是编辑文件,这是用编辑器程序(editor program)完成的。按照 C 语言语法规则编写的 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,并进行必要的修改。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 Turbo C、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,我们假设读者已经知道如何编辑程序。

二、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。这就需要把 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 系统中,预处理程序在编辑器翻译阶段开始之前自动执行。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的某些操作。这些操作通常包含在要编译的文件中,包括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(也称为目标码)。

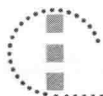
三、连接阶段

目标程序还不能马上交给计算机直接运行,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连接到经编译生成的目标程序中,生成可执行程序。

连接器(linker)将目标码与这些默认功能的代码连接起来,建立可执行程序映像(不再缺少任何代码)。

四、装入执行阶段

下一个阶段是装入与执行。执行之前,要先把 .exe 程序放进内存中,这是由装入器(loader)完成的。装入器读取磁盘中可执行程序的映像文件,并将其放进内存中。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、连接到运行的全过程如图 1.1 所示。

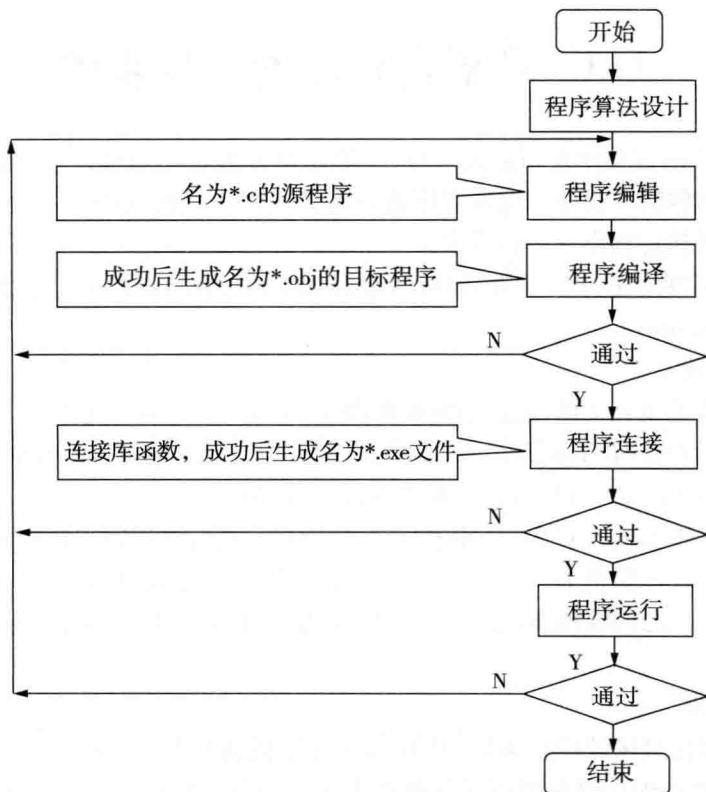


图 1.1 C 语言编辑运行步骤

程序不是一次就能够运行,上述每个阶段都可能因为各种错误而失败。例如,可能除数为 0(计算机上的非法操作,和算术运算中一样),这会使计算机打印出错信息。程序员需返回编辑阶段,进行必要的修改并继续其余阶段,确定修改之后能否顺利工作。

除了较简单的情况,一般的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。我们要在学习程序设计过程中逐步培养调试程序的能力,这不可能靠几句话讲清楚,要靠自己在上机中不断摸索总结,可以说是一种经验积累。

程序中的错误大致可分为三类:

- 程序编译时检查出来的语法错误;
- 连接时出现的错误;
- 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、大括号不匹配、语句少分号等。连接错误一般由未定义或未指明要连接的函数,或者函数调用不匹配等因素引起,对系统函数的调用必须要通过 include 说明。

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,找出相应错误所在。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带来的。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,并且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出现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,到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,难以一下子找出错误,可以进一步把该块细分为更小的块,再按照上述步骤检查。在确定了大致出错位置后,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,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不正确的数据。比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。

下面我们以 Visual C++ 6.0 为上机平台,对 C 程序的编译、连接和调试作简单介绍。建议在开始学习上机时,把注意力放在程序的编译、连接和运行上,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,它增加了面向对象编程的功能,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Visual C++被广泛地应用于各种编程。

1.2.1 Visual C++语言集成开发环境简介

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,已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,它不仅可以用来写系统程序,也可以用来写应用程序。利用Visual C++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,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,主要的原因在于其具有以下特点:

- (1)开发效率高,成本低。
- (2)面向对象的开发工具,代码的可重用性好,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

下面我们简要介绍在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++ 6.0。
- (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,可参照输出窗格中给出的提示进行修改,然后重复(3)至

(6),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(3)至(7),直至结果正确为止。

一、Visual C++ 6.0 安装与启动

现在常用的 Visual C++ 6.0 版本,虽然已有公司推出汉化版,但只是把菜单汉化了,并不是真正的中文版 Visual C++ 6.0,而且汉化的用词也不准确,因此许多人都使用英文版。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,并按照屏幕提示进行安装即可。

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

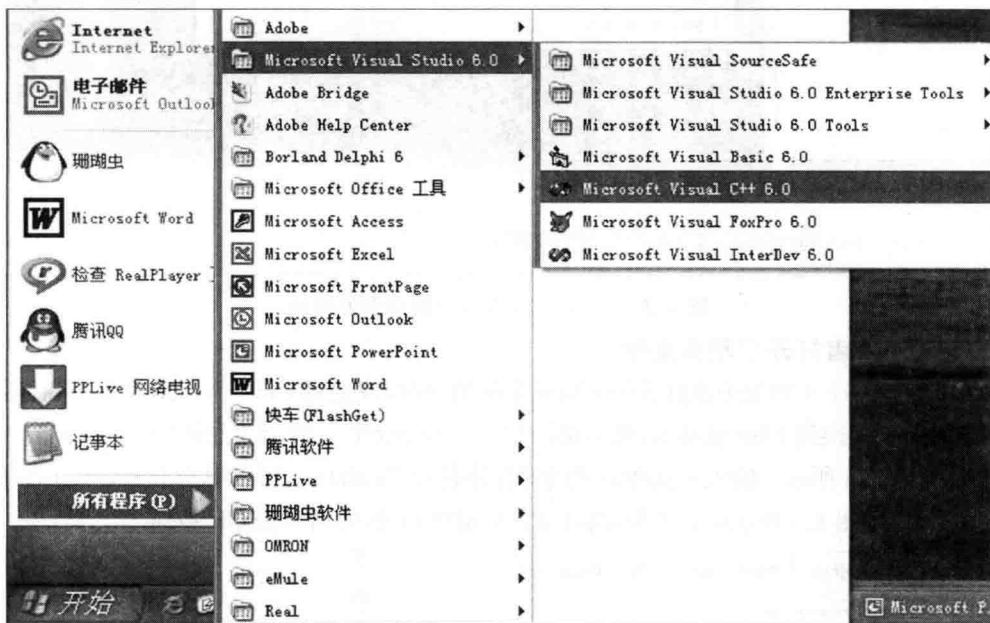


图 1.2 启动 Visual C++ 6.0

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,然后出现 Visual C++ 6.0 的主窗口,如图 1.3 所示,窗口中间出现“每日一贴”提示信息窗口。

在 Visual C++ 6.0 主窗口的顶部是主菜单栏,其中包括 9 个菜单项:File(文件)、Edit(编辑)、Insert(插入)、Project(项目)、Build(构建)、Tools(工具)、Window(窗口)和 Help(帮助)。以上各项括号中是中文显示,以便读者在使用软件时对照。

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)。

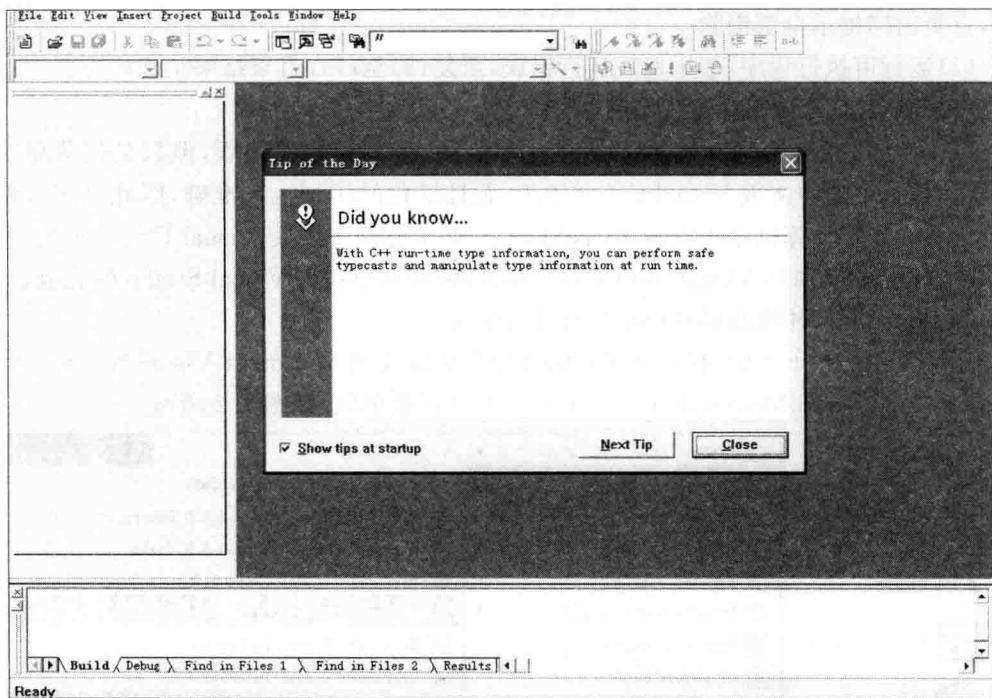
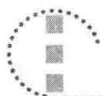


图 1.3 Visual C++ 6.0 集成开发环境

二、创建或者打开 C 程序文件

在 Visual C++ 环境中选择 File→New 菜单项(或按快捷键 Ctrl+N), 出现 New 对话框。在该对话框中切换到 Files 选项卡, 然后选择 C++ Source File 项, 准备创建一个新的 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, 可以单击 Location 下端的...按钮, 出现图 1.5 所示的对话框, 选择文件存放的目录, 然后单击 OK 按钮。

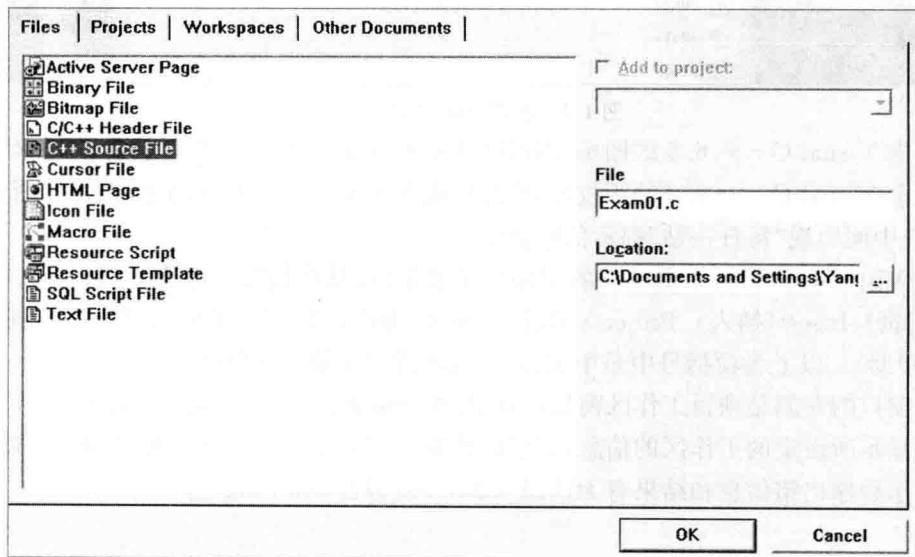


图 1.4 创建一个 C 语言程序文件



图 1.5 选择 C 语言程序文件存放目录

这里要注意的是,在选择文件名时,应该指定扩展名 .c, 否则系统将按 C++ 扩展名 .cpp 保存。

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

三、在编辑窗口中编写 C 程序

准备了一个字符大小写的程序,在程序编辑窗口中输入它的代码,如图 1.6 所示。

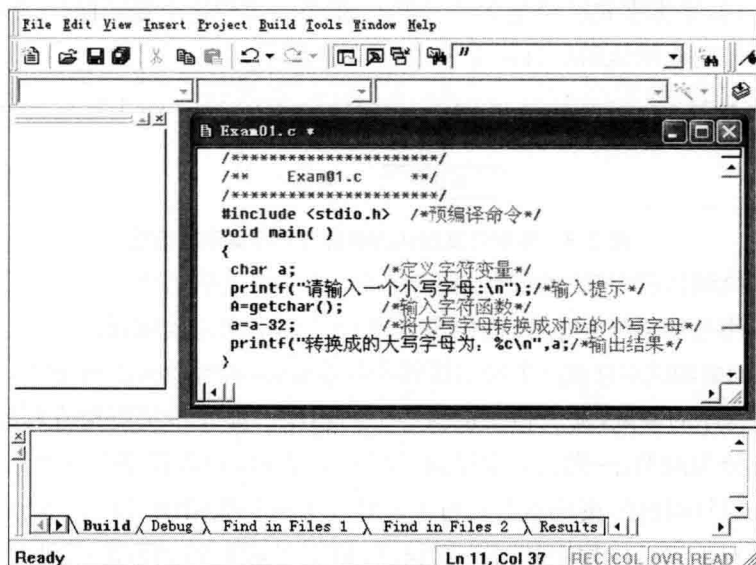
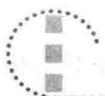


图 1.6 输入程序代码

为了看到更全面的后续调试程序过程,在输入本程序代码的过程中我们故意弄了两个错误:一个是将变量 a 在使用的时候故意写成大写字母 A; 另一个错误是将 printf() 函数的右括号故意漏掉。输入完毕后保存程序,执行主菜单栏 File→Save 命令即可。



四、编译、连接程序

将程序保存后,就可以编译、连接 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,选择 Build 子菜单,如图 1.7 所示。系统执行对源程序编译、连接的任务,如果期间有错误,系统会停下来,提示用户出错。Build 将编译、连接合成到一起自动依次执行,用户也可以单独先执行编译命令 Compile。

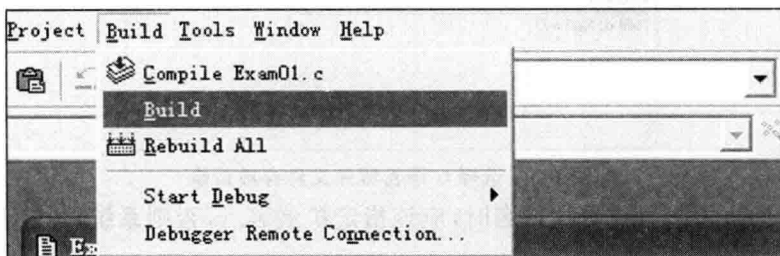


图 1.7 Build 菜单项

执行 Compile Exam01.c(编译 Exam01.c)命令,出现一个提示对话框,内容是:This build command requires an active project workspace. Would you like to create a default project workspace? 此编译命令要求一个有效的项目工作区,你是否同意建立一个默认的项目工作区(如果你事先已经建立了工作区,则不会出现这个提示框)。单击“是”按钮,表示同意由系统建立默认的项目工作区,如图 1.8 所示。

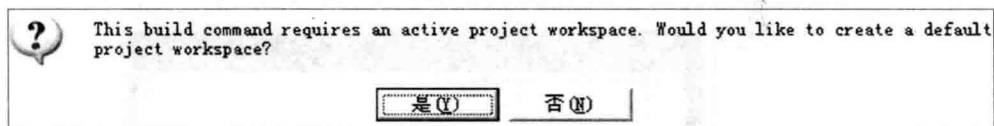


图 1.8 要求建立默认的项目工作区提示对话框

屏幕如果继续出现“将改动保存到 Exam01.c”,单击“是”按钮。

也可以不用选择菜单的方法,而用快捷键 Ctrl+F7 来完成编译。

在屏幕下面的调试信息窗口中指出源程序有无错误,本例显示 2 error(s),0 warning(s)。我们现在开始程序的调试,发现和改正程序中的错误。编译系统能检查程序中的语法错误。语法错误分为两类:一类是致命错误,以 error 表示,如果程序有这类错误,就通不过编译,无法形成目标程序,更谈不上运行了;另一类是轻微错误,以 warning(警告)表示,这类错误不影响生成目标程序和可执行程序,但有可能影响运行的结果,因此也应当改正,使程序既无 error,又无 warning。本例编译显示的窗口如图 1.9 所示。

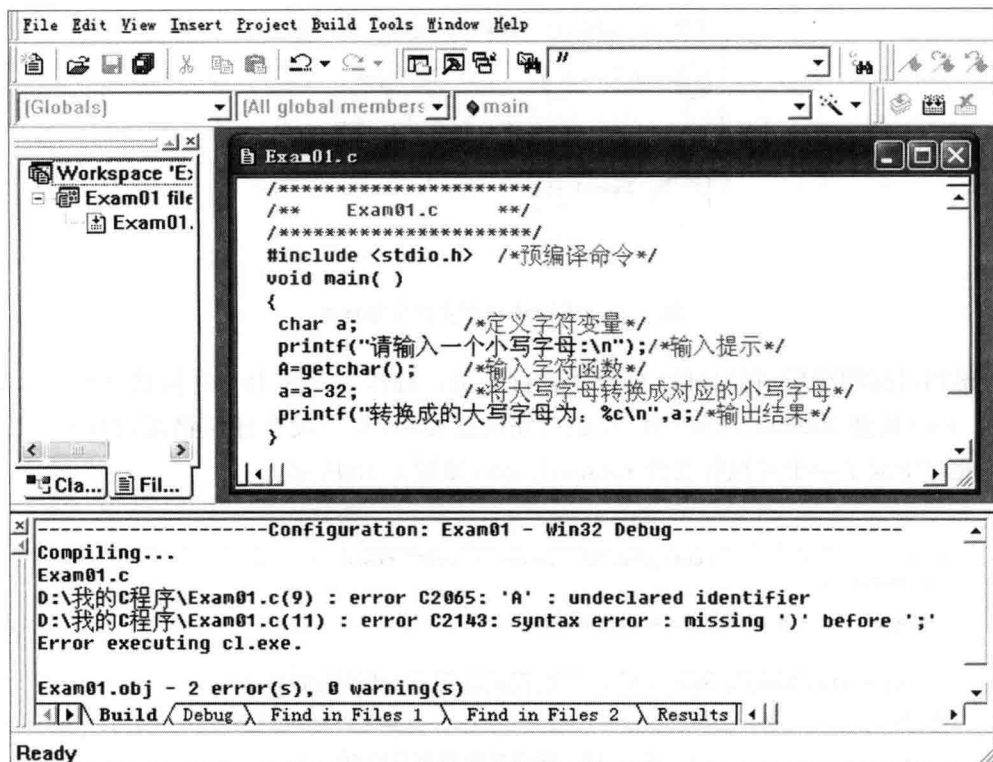


图 1.9 编译显示的窗口

编译、连接生成的结果出现在底部的输出窗口中,如果提示生成成功,则可执行下一个动作,否则单击调试信息窗口右侧的向上箭头,可以看到出错的位置和性质,参照出错提示,进行修改。

在本例中,我们根据出错提示信息将 `A=getchar()`; 修改成 `a=getchar()`; ,然后将最后一个语句的右括号加上。再仔细阅读程序,应该没有问题了。

选择 Compile Exam01. c 项重新编译,此时编译信息告诉我们:0 error(s),0 warning(s)。既没有致命错误(error),也没有警告错误(warning),编译成功,如图 1.10 所示。

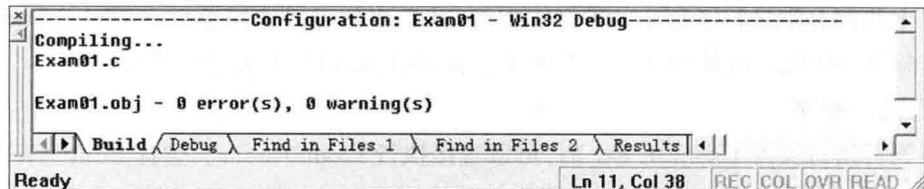


图 1.10 编译成功显示的信息

源程序经过正确编译后,产生了一个 Exam01. obj 文件,如图 1.11 所示。