



普通高等教育“十二五”规划教材

C 语言程序设计 上机指导与习题解答 (第二版)

主 编 任正云 李素若 赖 玲
副主编 琚 辉 胡玉荣 张 牧 肖 衡
主 审 田 原



中国水利水电出版社
www.waterpub.com.cn

普通高等教育“十二五”规划教材

C 语言程序设计上机指导与习题解答

(第二版)

主 编 任正云 李素若 赖 玲

副主编 琚 辉 胡玉荣 张 牧 肖 衡

主 审 田 原



中国水利水电出版社
www.waterpub.com.cn

内 容 提 要

本书是中国水利水电出版社出版的《C 语言程序设计》(第二版)的配套教材。修订后的内容分五个部分: C 语言上机指导、上机实验安排、主教材习题和参考解答、计算机等级考试(二级 C 语言)高频考点分析和上机典型题解、附录。

第一部分介绍了 Visual C++ 6.0 集成环境下的上机方法; 第二部分介绍了程序设计实验的一般步骤, 并给出了一份上机报告的样例, 精心设计了 11 套上机实验题, 每套实验都给出了实验目的、要求和内容; 第三部分对教材中的每道习题都给出了详细的解答, 有些容易出错和容易混淆的习题还给出了不同的解题方法, 帮助读者提高对 C 语言的基本概念、算法和各种规则的理解; 第四部分是“计算机等级考试(二级 C 语言)高频考点分析和上机典型题解”; 第五部分“附录”给出了最近三年的全国计算机等级考试二级 C 语言的笔试试题和参考答案与解析, 供读者学习。

本书内容丰富, 注重实践; 突出重点、分散难点, 知识结构合理、覆盖面广, 既可作为高校学生学习 C 语言程序设计课程的辅导教材, 也可作为 C 语言自学者的参考教材。

图书在版编目(CIP)数据

C 语言程序设计上机指导与习题解答 / 任正云, 李素若, 赖玲主编. -- 2 版. -- 北京: 中国水利水电出版社, 2011.11

普通高等教育“十二五”规划教材
ISBN 978-7-5084-9070-0

I. ①C… II. ①任… ②李… ③赖… 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(2011)第 210378 号

策划编辑: 杨庆川 责任编辑: 李 炎 封面设计: 李 佳

书 名	普通高等教育“十二五”规划教材 C 语言程序设计上机指导与习题解答(第二版)
作 者	主 编 任正云 李素若 赖 玲 副主编 琚 辉 胡玉荣 张 牧 肖 衡 主 审 田 原
出版发行	中国水利水电出版社 (北京市海淀区玉渊潭南路 1 号 D 座 100038) 网址: www.waterpub.com.cn E-mail: mchannel@263.net(万水) sales@waterpub.com.cn
经 售	电话: (010) 68367658(发行部)、82562819(万水) 北京科水图书销售中心(零售) 电话: (010) 88383994、63202643、68545874 全国各地新华书店和相关出版物销售网点
排 版	北京万水电子信息有限公司
印 刷	三河市鑫金马印装有限公司
规 格	184mm×260mm 16 开本 19 印张 492 千字
版 次	2007 年 3 月第 1 版第 1 次印刷 2011 年 11 月第 2 版 2011 年 11 月第 1 次印刷
印 数	0001—4000 册
定 价	32.00 元

凡购买我社图书, 如有缺页、倒页、脱页的, 本社发行部负责调换

版权所有·侵权必究

第二版前言

本书出版以来,经过多年的使用,得到了广大教师和读者的肯定,特别是本书的主教材《C 语言程序设计》(任正云、李素若主编,中国水利水电出版社出版)在 2010 年被教育部高等学校高职高专计算机类专业教学指导委员会评为优秀教材。根据教育部高等学校计算机基础课程教学指导委员会《高等学校计算机基础核心课程教学实施方案》的要求,针对应用型人才培养的目标,结合主教材《C 语言程序设计》和实际需要,吸取广大读者的宝贵意见,本书在第一版的基础之上进行了如下几个方面的改进。

(1) 只介绍了 Visual C++ 6.0 集成环境下的上机方法。

(2) 根据主教材《C 语言程序设计》(第二版)(任正云、李素若、赖玲主编)编排体系的要求,重新对全书所有的习题给出了详细的解答,对于部分题目除了给出答案外,还给出了分析过程。这部分题目一般都是容易出错和容易混淆的,通过进一步讲解可以帮助读者提高对 C 语言的基本概念、算法和各种规则的理解。

需要特别强调的,本书给出的程序并非唯一正确的解答,对同一个题目可以编出多种程序,我们给出的只是其中的一种,甚至不一定是最佳的一种。对于有些题目,我们给出了两种答案,供读者参考和比较,以启发思路。读者在使用本书时,千万不要照抄照搬,我们只是提供了一种参考答案,读者完全可以编写出更好的程序。

(3) 列举了近几年来全国计算机等级考试(二级 C 语言)高频考点,对高频考点进行了详细分析。

(4) 列举了近几年来全国计算机等级考试(二级 C 语言)上机题例题,对每种类型的题目给出了举例和解答,同时给出了部分上机题习题并附有答案和解析。

(5) 本书附录收录了近 3 年的全国计算机等级考试(二级 C 语言)笔试试题(共 5 套),并给出了每套考试试题的解析和答案,供读者参考和学习。

本书由任正云、李素若、赖玲任主编,琚辉、胡玉荣、张牧、肖衡任副主编。其中第一、二部分由李素若、赖玲共同编写;第三部分中的第 1、2、3、4、5、6 章主要由任正云编写,第 8、9、10 章由赖玲编写,第 7 章由李素若、琚辉共同编写;第四部分由任正云编写;第五部分由赖玲编写。全书由任正云、李素若统稿,李敏、胡玉荣、张牧、肖衡负责全书的校对,在本书的编写过程中还得到了主审田原教授和相关专家的指导,参与本书大纲讨论和相关内容编写的还有严永松、游明坤、陈万华、李俊梅、李祥琴、袁鹰和胡波等,在此一并表示衷心的感谢。

由于作者水平有限,书中难免还会存在不足之处,再次恳请广大读者批评指正。

编者

2011 年 8 月

第一版前言

C 语言是国内外广泛使用的计算机语言。许多高校都开设了“C 语言程序设计”课程。初学者对程序设计都有一个感觉，看别人写的程序，好像挺明白的，一旦要自己编写一个程序，就感到无从下手。这是因为程序设计是一门对实践环节要求很高的课程，初学者要想真正学会 C 语言程序设计，最重要的要抓住两个关键环节：一是多做程序设计的习题，多编程；二是多上机，写在纸上的程序看上去没有错误，但只有上机调试才能验证是否正确。因此，我们编写了这本上机指导与习题集解答。

本书包括三个部分。

第一部分是“C 语言上机指导”。介绍了 Turbo C 2.0、Turbo C++ 3.0 和 Visual C++ 6.0 集成环境下的上机方法。其中主要介绍了 Turbo C 2.0 的详细操作过程，特别是对 Turbo 2.0 的调试过程做了很详细的说明，给予初学者很大的帮助。考虑到现在正在学习“C 语言程序设计”的部分读者将来会继续学习“C++面向对象程序设计”，因而在这部分中介绍了 Turbo C++ 3.0 和 Visual C++ 6.0 中的部分功能及 C 语言程序在这两个环境下的编辑、编译、连接、运行和调试的方法。本书的附录Ⅲ中有 Turbo C 2.0 编译出错信息说明，当编译出错时可以查阅相关出错信息的详细解释。

第二部分是“上机实验安排”。在这部分中提出了上机实验的要求和步骤，实际上也是给读者灌输软件开发过程中的一般步骤，同时给出了试验报告的书写样本供读者参考。在实验的总体安排上主要跟教材《C 语言程序设计》（任正云、李素若主编）相配合，根据教材中每章的内容来设计实验。在每次实验中首先拿出 3~4 个案例来引导读者去上机实践，来加深对本章内容的理解，紧接着又设计 4~5 个上机习题供读者练习，使初学者能循序渐进地完成本次实验。

第三部分是“《C 语言程序设计》习题和参考解答”。在这部分中包括了水利电力出版社出版的《C 语言程序设计》（任正云、李素若主编）一书的全部习题和参考答案。对于部分题目除了给出答案外，还给出了分析过程。这部分题目一般都是容易出错和容易混淆的，通过进一步讲解可以帮助读者提高对 C 语言的基本概念、算法和各种规则的理解。

需要特别强调的，本书给出的程序并非惟一正确的解答，对同一个题目可以编出多种程序，我们给出的只是其中的一种，甚至不一定是最佳的一种。对于有些题目，我们给出了两种答案，供读者参考和比较，以启发思路。读者在使用本书时，千万不要照抄照搬，我们只是提供了一种参考答案，读者完全可以编写出更好的程序。

本书附录 I、II 中将部分读者很关心的全国计算机等级考试大纲和近 3 年的全国计算机等级考试二级笔试试题（C 语言）列出，并将每次考试试题的参考答案也列出供读者参考和学习。

本书由李素若、任正云共同主编，胡玉荣、张牧、肖衡担任副主编。其中第一部分中的第 1 章主要由肖衡编写，第 2 章主要由张牧编写；第三部分中的第 1、2、4、6、9 章主要由任正云编写，第 3、5 章主要由胡玉荣编写，第 8 章主要由张牧编写；除以上章节以外的其他

章节全部由李素若编写。全书由李素若统稿，在本书的编写过程中得到了主审田原副教授和相关专家的指导，参与本书大纲讨论和相关内容编写的还有严永松、游明坤、文斌、陈万华、李俊梅、李祥琴、袁鹰、胡波和刘青筱等，在此一并表示衷心的感谢。

由于作者水平有限，书中不妥或错误之处在所难免，敬请读者指正。

编 者

2006年12月

目 录

第二版前言

第一版前言

第一部分 C 语言上机指导

第 1 章 Visual C++ 6.0 上机操作	1	1.5 执行程序	3
1.1 使用 VC++ 6.0 调试程序一般步骤	1	1.6 关闭程序工作区	4
1.2 启动 VC++	1	1.7 命令行参数处理	5
1.3 新建或打开 C 程序文件	2	1.8 程序调试	5
1.4 程序保存	2		

第二部分 上机实验安排

第 1 章 程序设计实验的一般步骤	8	2.2 数据类型、运算符和表达式实验	16
1.1 问题分析	8	2.3 最简单的 C 程序设计实验	18
1.2 设计测试计划	9	2.4 逻辑结构程序设计实验	19
1.3 设计程序方案	9	2.5 循环结构实验	22
1.4 编写程序	10	2.6 数组实验	25
1.5 上机调试	10	2.7 函数实验	28
1.6 实验总结与实验报告内容	11	2.8 指针实验	31
1.7 实验报告样例	11	2.9 结构体、共用体实验	34
第 2 章 实验安排	14	2.10 位运算实验	39
2.1 熟悉 TC 环境下运行 C 程序实验	14	2.11 文件实验	40

第三部分 主教材习题和参考解答

第 1 章 C 语言程序设计基础	43	第 6 章 函数	81
第 2 章 C 语言数据类型与基本操作	47	第 7 章 数组	91
第 3 章 顺序结构程序设计	57	第 8 章 指针	101
第 4 章 选择结构程序设计	63	第 9 章 结构体和共用体	111
第 5 章 循环结构程序设计	71	第 10 章 文件	126

第四部分 计算机等级考试（二级 C 语言）高频考点分析与上机典型题解

第 1 章 等级考试高频考点分析	136	1.4 指针	142
1.1 C 语言程序设计基础	136	1.5 数组	143
1.2 C 语言的基本结构	138	1.6 字符串	146
1.3 函数	141	1.7 结构体、共用体和用户定义类型	148

1.8 文件	150	2.3.1 字符串的问题	161
第2章 上机题例题选讲	151	2.3.2 数组的遍历	164
2.1 填空题	151	2.3.3 二维数组按列存放的问题	165
2.1.1 字符串的处理问题	151	2.3.4 数学问题	166
2.1.2 数学问题	152	2.3.5 结构体问题	167
2.1.3 文件的操作	152	2.3.6 排序问题	168
2.1.4 结构体的问题	153	2.3.7 其他问题	169
2.1.5 数组和指针的问题	156	2.4 上机题练习题	171
2.2 改错题	157	2.4.1 填空题	171
2.2.1 基本语法的错误	157	2.4.2 改错题	185
2.2.2 函数返回值的类型与声明的数据 类型不匹配	157	2.4.3 程序设计题	198
2.2.3 数据的强制转换问题	158	2.5 上机题参考答案	209
2.2.4 结构体的灵活应用问题	159	2.5.1 填空题	209
2.2.5 数组与指针的关系问题	160	2.5.2 改错题	212
2.3 程序设计题	161	2.5.3 程序设计题	215

第五部分 附录

2009年3月全国计算机等级考试二级C语言 笔试试题	223	2009年9月全国计算机等级考试二级C语言 笔试试题参考答案与解析	271
2009年9月全国计算机等级考试二级C语言 笔试试题	231	2010年3月全国计算机等级考试二级C语言 笔试试题参考答案与解析	276
2010年3月全国计算机等级考试二级C语言 笔试试题	239	2010年9月全国计算机等级考试二级C语言 笔试试题参考答案与解析	282
2010年9月全国计算机等级考试二级C语言 笔试试题	248	2011年3月全国计算机等级考试二级C语言 笔试试题参考答案与解析	289
2011年3月全国计算机等级考试二级C语言 笔试试题	256	参考文献	294
2009年3月全国计算机等级考试二级C语言 笔试试题参考答案与解析	265		

第一部分 C 语言上机指导

第 1 章 Visual C++ 6.0 上机操作

C++语言是在 C 语言的基础上发展而来，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++ 6.0（简称 VC++ 6.0）是由微软公司开发的，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++ 6.0 被广泛地用于各种编程，使用面很广。这里简要地介绍如何在 Visual C++ 6.0 下运行 C 语言程序。

1.1 使用 VC++ 6.0 调试程序一般步骤

在使用 Visual C++ 6.0 调试程序之前，必须先了解项目和项目工作区两个概念。项目（Project）是一组相互关联的文件，是将一个应用程序中需要的所有文件组成一个有机的整体，一般包括源文件、头文件、资源文件等。而项目文件必须在某个项目工作区（Workspace）的管理之下，一个项目工作区可以包含一个以上的项目。

使用 Visual C++ 6.0 调试程序的大致步骤是：

- （1）创建一个项目工作区（Workspace）。
- （2）在项目工作区中新建一个项目（Project）。

也可以将上面两步合并，在建立新项目的同时创建项目工作区。

- （3）在项目中建立源文件、头文件等（或将这些文件添加到项目中）。
- （4）编辑项目中的源文件、头文件。
- （5）连接、编译应用程序。
- （6）运行应用程序。

注意：如果不先创建项目工作区（Workspace）和项目（Project），而直接创建源文件，当编译该源文件时会提醒没有项目工作区（Workspace）和项目（Project），单击“确定”按钮会自动创建一个默认的项目工作区（Workspace）和项目（Project）。

1.2 启动 VC++

Visual C++ 6.0 是一个庞大的语言集成工具，经安装后将占用几百兆磁盘空间。从“开始”→“程序”→Microsoft Visual Studio 6.0→Microsoft Visual C++ 6.0，可启动 Visual C++ 6.0，屏幕上将显示如图 1.1 所示的窗口。

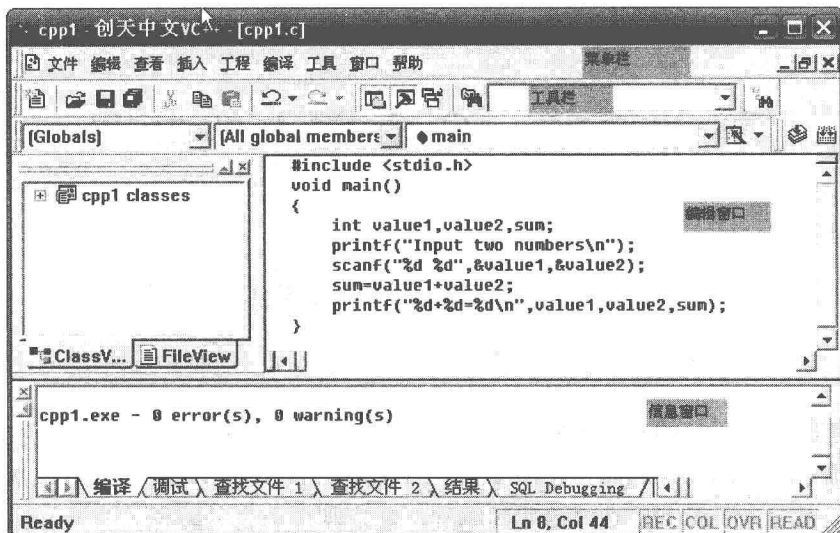


图 1.1 VC++ 6.0 窗口

1.3 新建或打开 C 程序文件

选择“文件”菜单中的“新建”命令，单击如图 1.2 所示的“文件”标签，选中“C++ Source File”，单击“确定”按钮，然后在编辑窗口中输入程序。

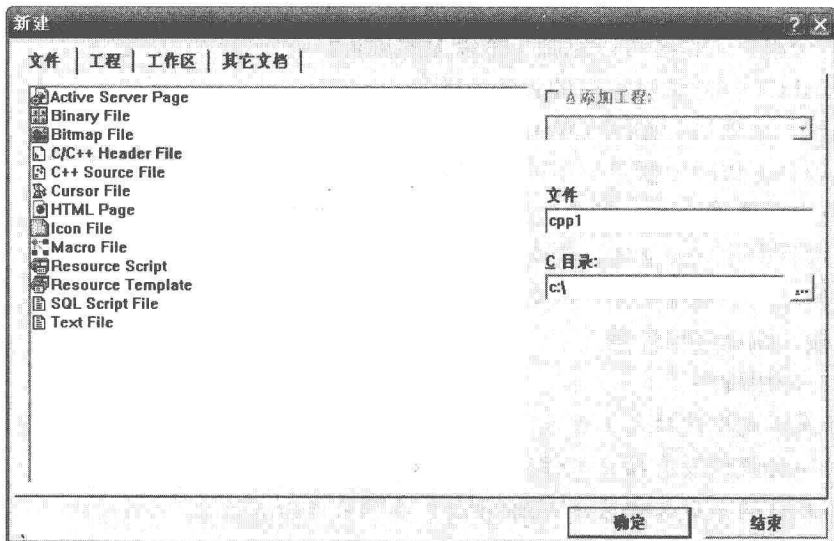


图 1.2 “新建”对话框

如果程序已经输入过，可选择“文件”菜单中的“打开”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。

1.4 程序保存

在打开的 VC++ 界面上，可直接在编辑窗口输入程序，由于完全是 Windows 界面，输入

及修改可借助鼠标和菜单进行,十分方便。当输入结束后,保存文件时,应指定扩展名为“.C”,否则系统将按 C++扩展名“.CPP”保存。程序保存对话框如图 1.3 所示。

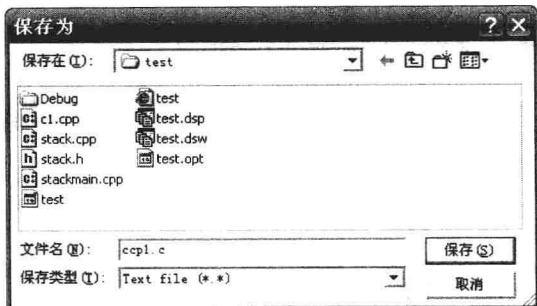


图 1.3 程序保存对话框

1.5 执行程序

首先要生成可执行文件。使用 VC++ “编译”菜单中的“构件”命令,如图 1.4 所示,也可使用快捷键 F7。在编译、连接过程中 VC++ 将保存该新输入的程序,并生成一个同名的工作区。保存文件时需填入文件名,如“4-1.C”。假如不指定扩展名.C, VC++ 会把扩展名自动定义为.CPP,即 C++程序。

如果程序没有错误,将在如图 1.5 所示的信息窗口中显示内容:

0 error(s), 0 warning(s)

表示没有任何错误。有时出现几个警告性信息(warning),不影响程序执行。假如有致命性错误(error),如图 1.6 所示,双击某行出错信息,程序窗口中会指示对应的出错位置,可根据信息窗口的提示分别予以纠正。然后用“编译”菜单中的“执行”命令(或快捷键 Ctrl+F5)执行程序。

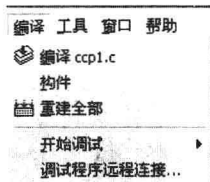


图 1.4 编译菜单

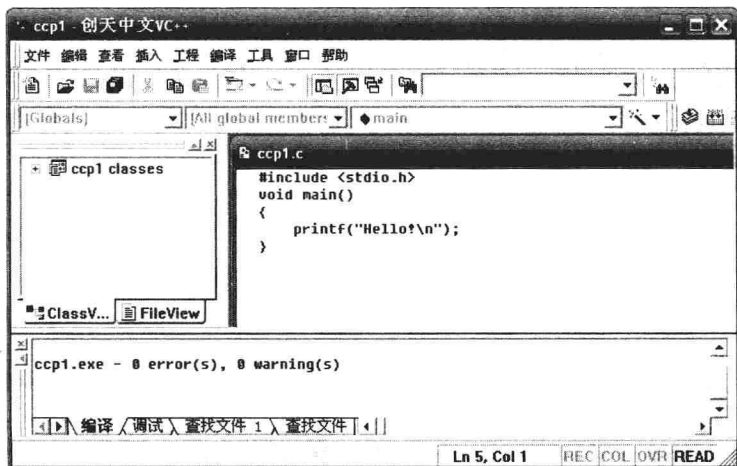


图 1.5 编译连接正确

当运行 C 程序后, VC++ 将自动弹出程序运行窗口,如图 1.7 所示。按任意键将关闭该窗口。

对于编译、连接、执行操作, VC++还提供了一组工具按钮, 如图 1.8 所示。

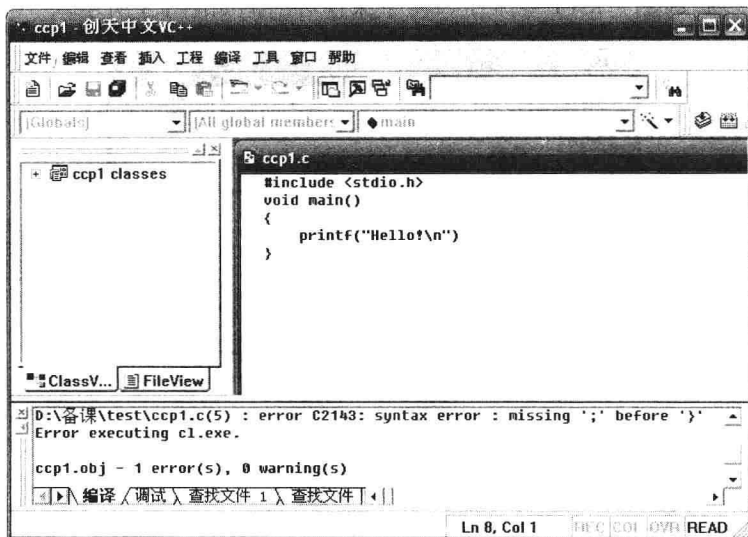


图 1.6 编译连接出错



图 1.7 程序运行窗口

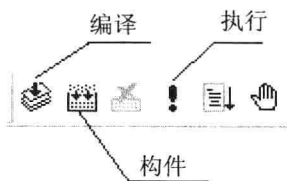


图 1.8 编译连接执行工具按钮

1.6 关闭程序工作区

当一个程序编译连接后, VC++系统自动产生相应的工作区, 以完成程序的运行和调试。若想执行第二个程序时, 必须关闭前一个程序的工作区, 然后通过新的编译连接, 产生第二个程序的工作区。否则, 运行的将一直是前一个程序。“文件”菜单提供关闭程序工作区功能, 如图 1.9 所示, 执行“关闭工作区”命令, 然后在如图 1.10 所示的对话框中单击“否”按钮。如果单击“是”按钮将同时关闭源程序窗口。

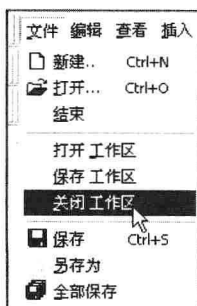


图 1.9 “文件”菜单

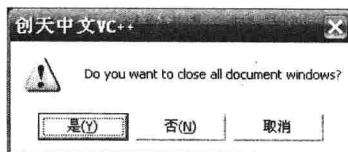


图 1.10 文件关闭对话框

1.7 命令行参数处理

VC++是一个基于窗口操作的 C++系统，没有提供命令行参数功能，需要在 Windows 的“MS-DOS 方式”窗口里以命令方式实现。具体参考如下步骤：

- (1) 正确编译连接，生成可执行程序。
- (2) 通过“我的电脑”或“资源管理器”找到所运行的 C 源程序（设为 a.c）。
- (3) 进入 debug 文件夹（它包含 a.c 程序的可执行文件 a.exe）。
- (4) 执行“开始”菜单的“运行”命令，填入 a，然后单击“确定”按钮。
- (5) 在打开的“MS-DOS 方式”窗口中输入：a 参数 1 参数 2 ..，带参数运行程序。

1.8 程序调试

VC++是一个完全基于 Windows 的系统，它的调试过程通过鼠标比较容易进行。

1. 程序执行到中途暂停以便观察阶段性结果

方法一：使程序执行到光标所在的那一行暂停。

- (1) 在需暂停的行上单击鼠标，定位光标。

(2) 如图 1.11 所示，分别单击菜单“编译”→“开始调试”→Run to Cursor 命令，或按 Ctrl+F10 组合键，程序将执行到光标所在行暂停。如果把光标移动到后面的某个位置，再按 Ctrl+F10 组合键，程序将从当前的暂停点继续执行到新的光标位置，第二次暂停。

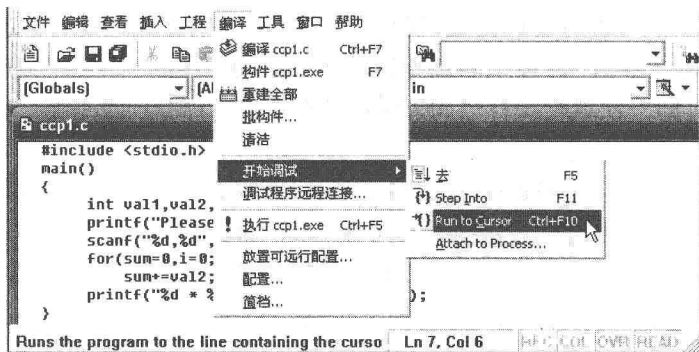


图 1.11 执行到光标所在行暂停

方法二：在需暂停的行上设置断点。

- (1) 在需设置断点的行上单击鼠标，定位光标。

(2) 按“编译微型条”中最右面的按钮，如图 1.12 所示，或按 F9 键。被设置了断点的行前面会有一个红色圆点标志。



图 1.12 工具栏中断点设置按钮

与 TC 一样，不管是通过光标设置还是断点设置，其所在的程序行必须是程序执行的必经之路，亦即不应该是分支结构中的语句，因为该语句在程序执行中受到条件判断的限制，有

可能因条件的不满足而不被执行。这时程序将一直执行到结束或下一个断点为止。

2. 设置需观察的结果变量

按照上面的操作,使程序执行到指定位置时暂停,目的是为了查看有关变量的中间结果。在图 1.13 中,左下角窗口中系统自动显示了有关变量的值,其中 val1 和 val2 的值分别是 3 和 4,而变量 i、sum 的值是不正确的,因为它们还未被赋值。图中左侧的箭头表示当前程序暂停的位置。如果还想增加观察变量,可在图中右下角的 Name 框中填入相应变量名。

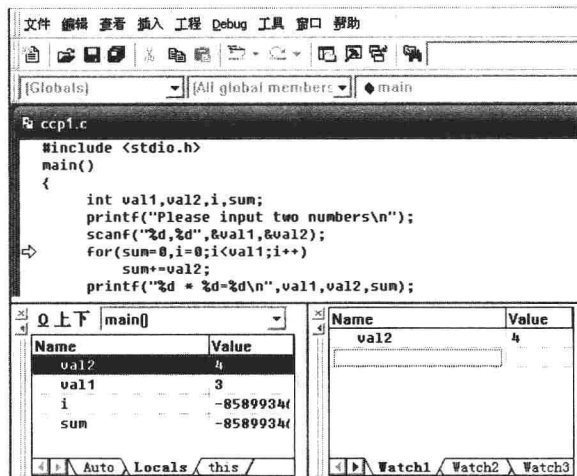


图 1.13 调试过程中观察变量的值

3. 单步执行

当程序执行到某个位置时发现结果已经不正确了,说明在此之前肯定有错误存在。如果能确定一小段程序可能有错,先按上面步骤暂停在该小段程序的头一行,再输入若干个观察变量,然后单步执行,即一次执行一行语句,逐行检查下来,看看到到底是哪一行造成结果出现错误,从而能确定错误的语句并予以纠正。

单步执行可按“调试”工具栏中 Step Over 按钮  或 F8 键,如图 1.14 所示。如果遇到自定义函数调用,想进入函数进行单步执行,可按 Step Into 按钮  或 F11 键。当想结束函数的单步执行,可按 Step Out 按钮  或 Shift+F11 组合键。对不是函数调用的语句来说, F11 键与 F8 键作用相同。但一般对系统函数不要使用 F11 键。



图 1.14 单步调试

4. 断点的使用

使用断点也可以使程序暂停。但一旦设置了断点,不管是否还需要调试程序,每次执行程序都会在断点上暂停。因此调试结束后应取消所定义的断点。方法是先把光标定位在断点所在行,再按“编译微型条”中最右面的按钮或 F9 键,该操作是一个开关,按一次是设置,按两次是取消设置。如果有多个断点想全部取消,可执行“编辑”菜单中的“断点”命令,屏幕上会显示 Breakpoints 对话框,如图 1.15 所示,对话框下方列出了所有断点,单击 Remove All 按钮,将取消所有断点。断点通常用于调试较长的程序,可以避免使用 Run to Cursor (运行程

序到光标处暂停)或 Ctrl+F10 组合键功能时,经常要把光标定位到不同的地方。而对于长度为上百行的程序,要寻找某位置并不太方便。

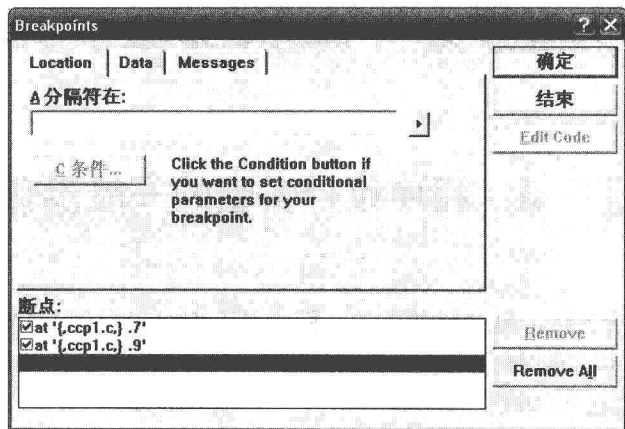


图 1.15 Breakpoints (断点) 对话框

如果一个程序设置了多个断点,按一次 Ctrl+F5 组合键会暂停在第一个断点,再按一次 Ctrl+F5 组合键会继续执行到第二个断点暂停,依次执行下去。

5. 停止调试

使用 Debug 菜单的 Stop Debugging 命令,或按 Shift+F5 组合键可以结束调试,从而回到正常的运行状态。

第二部分 上机实验安排

第1章 程序设计实验的一般步骤

程序设计不能纸上谈兵。通过调试程序，程序员不但可以发现并纠正程序中存在的语法错误和逻辑错误，还可以判断自己编写的程序是否完成了预期的功能。如果程序能够完成预期的功能，则说明程序通过了调试。

如果不能掌握程序调试的一般方法和技巧，就不能说学会了程序设计。

程序设计实验的目的不仅仅是为了验证程序，而是加深对 C 语言知识的理解，从而掌握编写 C 程序的方法。因此，程序设计实验要做的工作可不是单单指上机这段时间要完成的工作。在上机之前，应该完成一系列准备工作，而在上机之后，还要对实验过程中出现的问题进行分析和总结，逐步积累经验。

完成一个题目的程序设计实验的一般步骤是：问题分析、设计测试计划、设计程序方案、编写程序、上机调试、实验总结等。

1.1 问题分析

在解决一个实际问题之前，必须分析了解还需要解决的问题是什么，从而确定解决问题的输入和输出数据及其格式，以及计算方法等。

经过分析，要从以下几个方面对所要解决的问题进行说明。

1. 问题定义

对问题进行必要的描述，描述要正确、详细。

2. 定义输入、输出

对解决问题需要的输入数据和输出内容进行描述。

3. 说明计算公式

解决有些问题需要使用一个或多个公式，要对这些公式进行描述。

4. 定义计算需求

由于计算机所表示数据的范围是有限的，不同数据类型的表示范围也是不相同的，因此必须搞清楚问题中所有数据的类型、范围和精度，并对其进行描述。

【例1】已知一个学生的“高等数学”的期中和期末成绩，总评成绩的计算方法是：总评成绩=30%×期中成绩+70%×期末成绩。编写程序输入期中和期末成绩，输出总评成绩。

下面将根据题目要求写出经过问题分析以后的说明。

问题定义：为一个学生计算总评成绩。

输入：期中成绩和期末成绩。

公式：总评成绩=30%×期中成绩+70%×期末成绩

输出：总评成绩。

计算要求：所有成绩为整数。

限制：输入的期中成绩和期末成绩必须小于等于 100 并且大于等于 0。若用户输入有误，就提示用户，并提供再次输入的机会。

问题分析的结果是写出问题说明的文档，作为实验报告的第一部分内容。

1.2 设计测试计划

为了验证程序必须设计测试计划，测试计划是调试的依据。对于程序是否能够达到预期的目的，必须根据测试计划输入不同的数据，从而验证计算结果。程序测试计划可以在编写程序之前，程序说明完成之后就设计。因为设计一个测试计划有助于程序员理清自己的思路，可能会帮助程序员发现一些必须处理的特殊情况，以便构造一个易于验证的设计方案。

测试计划用一张表来表示，表中应该包括一组数据和根据相应的输入数据计算得到的正确结果，结果一般是手工填写的（如果可能的话）。

测试数据一般包括容易计算的数据和特殊情况数据。

测试数据要能够覆盖各种可能产生的情况。

下表是根据例 1 的问题说明编写的测试计划。

测试计划：

期中成绩	期末成绩	总评成绩	备注
88	98	95	
76	67	69	
46	75	66	
-10 (55)	66	62	先输入-10，再输入 55
54	102 (99)	85	先输入 102，再输入 99

第四行的数据说明先输入-10，正确的程序应该报错，并提供再次输入的机会。

程序设计完成之后，程序员应该回顾一下测试计划，并根据程序增加一些数据样本，从而保证程序的每一行代码都能够测试到。

1.3 设计程序方案

根据问题分析的结果设计程序方案。设计程序方案时，必须考虑三个问题，即算法、驱动程序、调试输出。

（1）算法是对求解一个特定问题的描述。解决同一个问题时，不同的人设计的算法也可以不相同。一个算法必须定义完善、明确和有效。描述算法的方法有自然语言和流程图等。

除了一些系统程序必须设计成永久运行外，大多数程序都是基于算法实现的。

对于初学程序设计的人，即使对于非常简单的算法，也应该用自然语言或流程图进行描述，并在此基础上再编写程序。

（2）驱动程序负责接收输入和产生输出，应该使用友好的界面与用户进行交流，通知用户程序正在进行的操作。发现错误输入时，应该向用户显示一条恰当的错误提示。