



博学·大学公共课系列

大学程序设计(C) 实践手册

主 审 陆 铭
主 编 夏 耘 吉顺如 王学光

 复旦大学出版社
www.fudanpress.com.cn

TP312/3034D

2008



博学 · 大学公共课系列

大学程序设计(C) 实践手册

主 审 陆 铭
主 编 夏 耘 吉顺如 王学光
副主编 陈章进 臧劲松

编写人员 (按姓氏笔画排列)

王学光	王淮亭	刘丽霞	刘 琴
吉顺如	宋兰华	邹启明	陈海燕
陈章进	金惠芳	夏 耘	黄春梅
臧劲松			

复旦大学出版社

图书在版编目(CIP)数据

大学程序设计(C)实践手册/夏耘,吉顺如,王学光主编. —上海:
复旦大学出版社,2008.8
(复旦博学·大学公共课系列)
ISBN 978-7-309-06089-8

I. 大… II. ①夏…②吉…③王… 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(2008)第086884号

大学程序设计(C)实践手册

夏 耘 吉顺如 王学光 主编

出版发行 复旦大学出版社 上海市国权路579号 邮编 200433
86-21-65642857(门市零售)
86-21-65100562(团体订购) 86-21-65109143(外埠邮购)
fupnet@fudanpress.com http://www.fudanpress.com

责任编辑 黄 乐

出品人 贺圣遂

印 刷 上海复文印刷厂
开 本 787×1092 1/16
印 张 16.5
字 数 422千
版 次 2008年8月第一版第一次印刷
印 数 1—7 000

书 号 ISBN 978-7-309-06089-8/T·323
定 价 33.00元

如有印装质量问题,请向复旦大学出版社发行部调换。

版权所有 侵权必究

内 容 提 要

本书以上海市高等学校计算机等级考试二级（C程序设计）考试大纲为标准，将C语言实验作为切入点，倡导基于DIY的程序设计教学法，使读者通过本书设置的循序渐进的实验，从体验程序、调试程序、编写部分程序功能，到编写完整的程序并能开发微型应用系统，有效解决高校学生长期以来C程序上课能听、下课不能解题、谈编程色变的问题。

本书第1至第9章以实验为中心，每章分成7个部分，分别是：实验目的、实验内容、实验的理论知识、典型例题分析、课后练习、自我测试、常见问题及解决方法。第10章对开发微型应用系统所涉及的软件开发技术展开了讨论，同时给出了案例和大量的实训题，从而使读者全面掌握C程序设计的思想和方法。

本书注重基本概念的系统化，叙述简明扼要，对实验中需要的理论进行了梳理，对重点逐一进行了点拨。全书内容精炼，结构合理，重点突出，特别是，针对读者在实验中遇到的不少典型问题给出了十分清楚和详细的阐述。

本书由上海市部分院校计算机基础教学的一线教师共同编写完成，适合各本科、大专院校的师生及计算机编程爱好者使用。

前 言

PREFACE

编写程序是一项系统而繁琐的工作,它不仅需要程序设计人员具有一定的功底,更需有良好的编程习惯和风格。良好的编程习惯和风格不仅可以使程序代码更易于读懂和修改,更重要的是,它可以使程序的结构更加合理,有助于提高程序的执行效率。大型程序的编写工作还需要程序设计人员具有团队协作精神。因此,多年来,程序设计课程一直作为大学生的计算机基础课程。

编程语言 C 以其小巧、灵活、高效等特点成为当今软件开发的主流,近年来不少高校将 C 程序设计语言作为大学生的入门编程语言,教育部考试中心及大部分省市也将 C 程序设计纳入计算机等级考试的科目。

本书将实验作为切入点,旨在倡导基于 DIY(Do It Yourself)的程序设计教学法,读者通过本书设置的循序渐进的实验,能从亲自体验程序到调试程序、编写部分程序到最后编写完整的程序并开发微型应用系统。读者在实验中会遇到不少问题,本书在每个实验之后及时对本次实验中可能出现的问题进行归纳总结,使读者在实验中经历发现问题—解决问题—归纳总结这一过程,解决学生长期以来的 C 程序上课能听、下课不能解题、谈编程色变的问题。

本书注重基本概念的系统化,叙述简明扼要,书中对实验中需要的理论作了讲解,对重点逐一进行了点拨,针对部分难点和重点,采用理论链接的方式,给出了相关知识和理论的分析。本书内容精炼,结构合理,重点突出,对读者可能遇到的难点作了十分清楚和详细的阐述。

本书第 1 至第 9 章以实验为中心,每章分成 7 个部分,分别是:实验目的、实验内容(分为基本实验内容和拓展实验)、实验的理论知识、典型例题分析、课后练习(倡导学生自主学习,采用小组讨论形式对实验中的知识进行探索)、自我测试(通过测试让学生发现学习中存在的问题,在教师和同学的帮助下及时解决问题)、常见问题及解决方法(能帮助读

者及时解决本次实验中可能出现的问题,同时对这些问题所涉及的知识进行归纳总结)。

本书的第10章对开发微型应用系统所涉及的软件开发技术展开了讨论,同时给出了案例和大量的实训题。作者相信,读者按第10章的实验要求和实训题具体要求进行训练一定会全面掌握程序设计的方法,部分读者还将成为程序设计的行家,广大读者分析问题、解决问题的能力必将得到提高。

本书整理了课上的教案,注重训练环节,体现了在理论指导下,让学生动手、动脑的基本思想方法,提出理性思维和理性实践,按照建构主义的学习理论,提倡学生作为学习的主体在与客观环境(指所学内容)的交互过程中构建自己的知识结构。本书引导学生在解题编程中探索其中带规律性的认识,将感性认识升华到理性高度,这样学生就能举一反三。

本书可供各层面学生、教师、自学应试者阅读。

本书附光盘1张,包含C语言程序设计实验课件、实验中所使用的源程序和范例、自我测试系统、课外资料、C库函数、C语言语法图、ASCII码表、实验报告(样本)、实训报告(样本)。

本书由上海大学计算中心、上海理工大学计算机与电气工程学院、上海电机学院计算机基础教学部、华东政法大学信息科学技术学院的一线教师共同编写,在编写过程中,组织了集体统稿、定稿,并得到了清华大学、交通大学、复旦大学、华东师范大学、华东理工大学、上海大学、上海理工大学等校计算中心各位老师的帮助,在此一并致谢。

上海理工大学计算机与电气工程学院夏耘老师、上海电机学院计算机基础教学部吉顺如老师和华东政法大学信息科学技术学院王学光老师任本书主编,上海大学计算中心陈章进老师、上海理工大学计算机与电气工程学院臧劲松老师任本书副主编,上海大学计算中心陆铭老师担任本书主审。参加本书编写的还有上海大学计算中心的宋兰华老师、邹启明老师,上海理工大学计算机与电气工程学院的黄春梅老师、刘丽霞老师,上海电机学院计算机基础教学部的王淮亭老师和华东政法大学信息科学技术学院的金惠芳老师、刘琴老师、陈海燕老师。

由于时间仓促和水平有限,本书中难免还存在一些不妥之处,敬请广大读者批评指正。

编 者

2008年5月

目 录

CONTENTS

第 0 章 绪论

0.1	程序设计(C) 的课程目标	/ 1
0.2	程序设计(C)课程的学习方法	/ 2
0.3	程序设计(C)课程的学习环境	/ 2
0.4	程序设计(C)课程的实验环境	/ 3
0.4.1	Visual C++6.0 编程环境	/ 3
0.4.2	Turbo C 2.0 编程环境	/ 10

第 1 章 基础知识和基本实验技能

1.1	实验目的	/ 23
1.2	实验内容	/ 23
1.2.1	C 程序运行环境及简单程序的运行	/ 23
1.2.2	数据类型、运算符和表达式	/ 26
1.3	实验的理论知识	/ 28
1.4	典型例题分析	/ 33
1.5	课后练习与讨论	/ 39
1.6	自我测试	/ 41
1.7	常见问题及解决方法	/ 41

第 2 章 顺序程序设计实验

2.1	实验目的	/ 45
-----	------	------

2.2	实验内容	/ 45
2.2.1	设计并运行简单的 C 程序	/ 45
2.2.2	拓展与实践	/ 48
2.3	实验的理论知识	/ 50
2.4	典型例题分析	/ 53
2.5	课后练习与讨论	/ 54
2.6	自我测试	/ 56
2.7	常见问题及解决方法	/ 58

第 3 章 选择结构程序设计实验

3.1	实验目的	/ 62
3.2	实验内容	/ 62
3.2.1	选择结构程序设计	/ 62
3.2.2	拓展与实践	/ 66
3.3	实验的理论知识	/ 72
3.4	典型例题分析	/ 74
3.5	课后练习与讨论	/ 76
3.6	自我测试	/ 79
3.7	常见问题及解决方法	/ 83

第 4 章 循环结构程序设计实验

4.1	实验目的	/ 86
4.2	实验内容	/ 86
4.2.1	循环结构程序设计	/ 86
4.2.2	拓展与实践	/ 89
4.3	实验的理论知识	/ 92
4.4	典型例题分析	/ 94
4.5	课后练习与讨论	/ 97
4.6	自我测试	/ 102
4.7	常见问题及解决方法	/ 106

第 5 章 数组实验

5.1	实验目的	/ 109
5.2	实验内容	/ 109

5.2.1 数组程序设计	/ 109
5.2.2 拓展与实践	/ 113
5.3 实验的理论基础	/ 114
5.4 典型例题分析	/ 118
5.5 课后练习与讨论	/ 127
5.6 自我测试	/ 131
5.7 常见问题及解决方法	/ 135

第6章 函数实验

6.1 实验目的	/ 140
6.2 实验内容	/ 140
6.2.1 函数程序设计	/ 140
6.2.2 拓展与实践	/ 144
6.3 实验的理论基础	/ 146
6.4 典型例题分析	/ 149
6.5 课后练习与讨论	/ 154
6.6 自我测试	/ 158
6.7 常见问题及解决方法	/ 162

第7章 指针实验

7.1 实验目的	/ 168
7.2 实验内容	/ 168
7.2.1 指针程序设计	/ 168
7.2.2 拓展与实践	/ 172
7.3 实验的理论基础	/ 173
7.4 典型例题分析	/ 175
7.5 课后练习与讨论	/ 180
7.6 自我测试	/ 185
7.7 常见问题及解决方法	/ 189

第8章 结构体与链表实验

8.1 实验目的	/ 191
8.2 实验内容	/ 191
8.2.1 结构体与链表程序设计	/ 191

8.2.2 拓展与实践	/ 196
8.3 实验的理论基础	/ 198
8.4 典型例题分析	/ 200
8.5 课后练习与讨论	/ 202
8.6 自我测试	/ 206
8.7 常见问题及解决方法	/ 210

第9章 文件实验

9.1 实验目的	/ 212
9.2 实验内容	/ 212
9.2.1 文件程序设计	/ 212
9.2.2 拓展与实践	/ 216
9.3 实验的理论基础	/ 217
9.4 典型例题分析	/ 221
9.5 课后练习与讨论	/ 223
9.6 自我测试	/ 227
9.7 常见问题及解决方法	/ 230

第10章 项目设计实验

10.1 实验目的	/ 232
10.2 实验内容	/ 232
10.3 实验的理论知识	/ 243

附录

上海市高等学校计算机等级考试二级(C 程序设计)考试大纲	/ 250
------------------------------	-------

参考书目

/ 255

0.1 程序设计(C)的课程目标

如何让计算机为我所用,是每一个本门课程的学习者首先要问的问题。自然,要与计算机沟通,就要让计算机明白你的意图,与计算机沟通的工具是程序设计语言。程序设计语言将工程、生活、工作中遇到的具有计算、分析和判断的问题,按计算机能接受的文字和句法进行描述,经过编译形成可执行代码,在给定运算数据时,计算机能提供相应的结果。目前计算机上的所有软件都是用某种程序设计语言编制而成的。作为当代大学生,自然应了解程序设计语言这个与计算机沟通的基本工具,从而能更好地驾驭计算机。

然而程序设计语言的学习与自然语言的学习不同,不能把程序设计语言看成是自然语言。很多学生都会问教师,什么时候应该学习一门程序设计语言,以及如何才能学好它?对这样的问题,三言两语很难解释清楚。理想情况下,语言的学习应该遵从实际的需要。当需要使用该语言解决实际问题时,才应该花费宝贵的时间学习它。但是,很多学生其实并不了解自己是不是需要某种特定的程序设计语言作为工具,或者并不知道自己未来会不会需要这样的程序设计语言来解决实际问题。显然,如果要同学们在需要使用该语言解决实际问题时再来学习它,估计大多数同学都会反对这样的观点。另一方面,一旦面临找工作的压力,如果这个没有学,那个也没有学,履历看上去就会有所欠缺了,更何况还有不同专业的教学计划的鸿沟呢——现实情况是,一旦教学计划定下来,什么应该学,什么可以不学其实并没有多少选择的余地!对于非计算机专业的学生来说,从一门未来极可能很少甚至不会使用的程序设计语言中学到什么东西,才能够获得尽可能多的知识和经验呢?类似于自然语言的学习,如果某个单词或语法结构在未来的学习和工作中永远不会被用到,学习它的意义显然几乎为零。程序设计语言也一样,记住 C 语言的全部语法规则细节但却从来不用它,同样也是毫无意义的。

所以,该课程的目标是让学生掌握计算机科学的原理而不是编程细节,在学习中不过分强调程序设计的细节,而从计算角度看计算机科学中的主要贡献,并着重对计算能力的理解和着重解决计算机科学中会遇到的可能影响其他学科的问题。借助程序设计的知识载体,获得基本问题的求解过程和基本思路,同时通过该课程的学习建立算法意识,培养良好的思维习惯。最后,从学科方法论上打基础,培养计算思维能力和分析问题、解决问题的能力。

该课程将“项目设计”作为完整的一章呈现给读者,力图通过完整的软件开发项目流程让读者了解软件开发的全过程,培养确定数据结构的能力,公共代码共享、模块接口、程序调试能力,Bug 的查找和排除、程序优化的能力以及小软件的研制能力。

0.2 程序设计(C)课程的学习方法

1. 养成良好的学习习惯

程序设计的入门学习并不难,但却是一个十分重要的过程,因为程序设计思想就是在这时形成的,良好的程序设计习惯也在这个阶段养成。

2. 学习程序设计要注重理解一些重要的概念

程序设计本身并不复杂,翻开一本程序设计学习的书籍,看到的无非就是变量、函数、条件语句、循环语句等概念。但要真正能进行程序设计,需要深入理解这些概念。因此,在程序入门阶段还是应该重视概念的学习。

3. 自己动手编写程序

程序设计入门阶段要经常自己动手编写程序,亲自动手进行程序设计是创造性思维应用的体现,是培养逻辑思维的好方法。因此一定要多动手编写程序,而且要从小程序开发开始,逐渐提高开发程序的规模。

4. 阅读、借鉴别人设计好的程序

多看别人设计好的程序代码,包括教材上的例题程序。在读懂别人的程序后,要想他为什么这么设计?能不能将程序修改完成更多的功能?这样可以学到别人优秀的东西,帮助自己提高编程水平。

5. 程序设计学习的重点

重点放在思路、算法、编程构思和程序实现上。语句只是表达工具,要求课堂上积极思考,尽量当堂学懂,并做到灵活应用。学会利用计算机编程手段分析问题和解决问题。

6. 养成良好的编程习惯

强调程序的可读性,变量要加注释,程序构思要有说明,学会如何调试程序,对运行结果要做正确与否的分析。

0.3 程序设计(C)课程的学习环境

程序设计的学习需要学生在一定情境下,借助于他人的帮助,如人与人之间的协作、交流、利用必要的信息等等,通过意义的建构而获得。理想的学习环境应当包括情境、协作、交流和意义建构四个部分。

情境,学习环境中的情境必须有利于学习者对所学内容的意义建构。在程序设计的学习中需要学习者具备“自立、自信、自尊、自强”,做到不放弃、不抛弃。在教学内容的设计中,本手册将创设有利于学习者建构意义的情境。

协作,应该贯穿于整个学习活动过程中。随着程序设计内容的增加,越来越多的工作需要团队合作来完成。团队协作模式最大的特点在于更加强调团队中个人的创造性发挥和团队整体的协同工作。在程序设计中,对怎样才能保证每个人的程序都能整合到一起、同时还能保证程序正常运行,对怎样减少在连调中产生的 bug,怎样才能修正一个 bug 后而不会由此产生新的更多的 bug,这些都是协作中需要解决的问题。协作在一定的意义上是协商的过程。协商主要有自我协商和相互协商。自我协商是指自己和自己反复商量什么是比较合理的;相互协商是指学习小组内部之间的商榷、讨论和辩论直至完成任务。

交流,是协作过程中最基本的方式或环节。比如学习小组成员之间必须通过交流来商讨如何修正 bug,如何达到程序的目标,怎样更多地获得教师或他人的指导和帮助等等。其实,协作学习的过程就是交流的过程,在这个过程中,每个学习者的想法都为整个学习群体所共享。交流对于推进每个学习者的学习进程,是至关重要的手段。

意义建构,是最终目标。其建构的意义是指事物的性质、规律以及事物之间的内在联系。在学习过程中帮助学生建构意义就是要帮助学生程序设计技巧、规律以及程序与算法之间的内在联系达到较深刻的理解。

0.4 程序设计(C)课程的实验环境

0.4.1 Visual C++ 6.0 编程环境

VC++ 6.0 是 Microsoft 公司出品的基于 Windows 环境的 C/C++ 开发工具,它是 Microsoft Visual Studio 套装软件的一个组成部分。C 语言源程序可以在 VC++ 6.0 集成环境中进行编译、连接和运行。

1. VC++ 6.0 主窗口

从 Visual Studio 的光盘中运行 VC++ 6.0 安装程序(Setup.exe),完成安装后,就可以从桌面上顺序选择“开始/程序/Microsoft Visual Studio/Microsoft Visual C++ 6.0”或双击桌面上的 VC++ 6.0 快捷图标来启动。启动后的 VC++ 6.0 主窗口如图 0-1 所示。

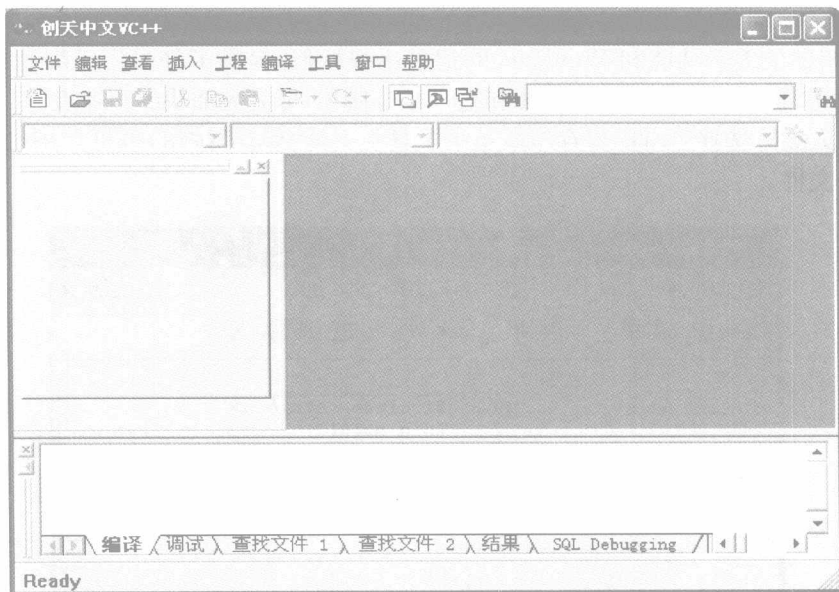


图 0-1 VC++ 6.0 主窗口

2. 输入和编辑 C 源程序

在 Visual C++ 主窗口的主菜单中选择“文件/新建”,屏幕上出现“新建对话框”,如图 0-2 所示。单击此对话框的“文件”选项卡,选择“C++ Source File”选项建立新的 C++ 源程序文件,然后在对话框右边的目录文本框中输入准备编辑的源程序文件的存储路径(如:

D:\C 源程序),在对话框右侧的文件文本框中输入准备编辑的 C 源程序文件名(如: sy1_1.c)。后缀.c 表示建立的是 C 源程序,若不加后缀,则默认的文件后缀为 .cpp,表示建立的是 C++源程序。

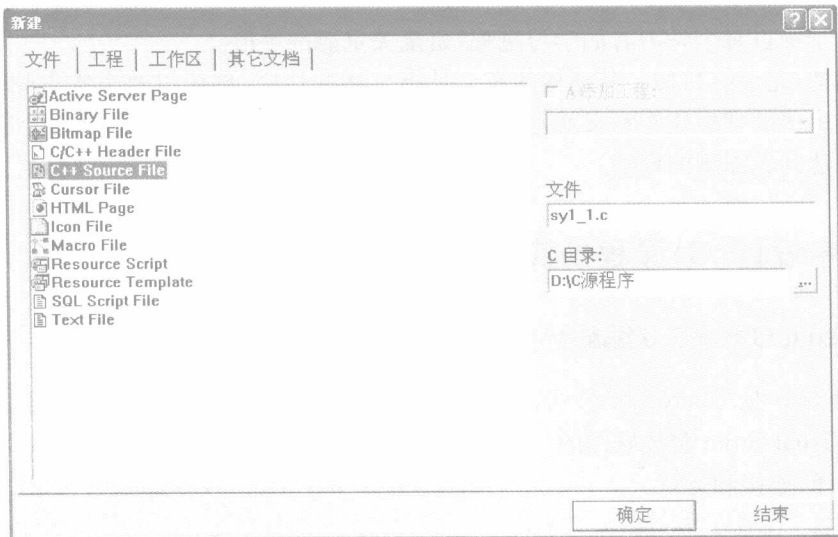


图 0-2 【新建】对话框

单击“确定”按钮,返回 Visual C++主窗口,此时窗口的标题栏中显示当前编辑的源程序文件名 sy1_1.c,如图 0-3 所示。可以看到光标在程序编辑窗口闪烁,表示程序编辑窗口已激活,可以输入和编辑源程序了。

VC 编辑器的编辑功能和 Windows 的记事本很相似,提供了许多用于编写代码的功能,如关键字加亮、自动调整格式等。鼠标和键盘配合使用,可大大加快编写速度。

程序输入完毕选择“文件/保存”,或单击工具栏上的“保存”按钮,也可以用<Ctrl+S>快捷键来保存文件。

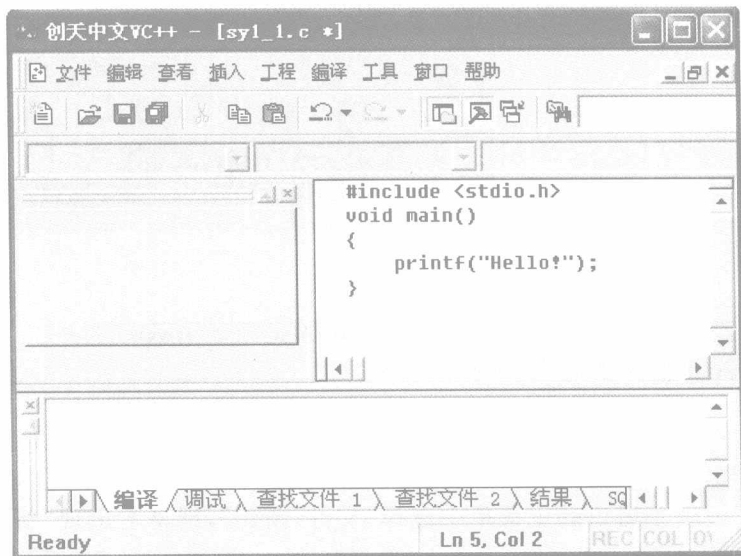


图 0-3 编辑窗口

3. 编译、连接和运行

程序编写完毕后,选择菜单“编译/编译”命令,或单击工具栏上的“编译”图标,也可以按<Ctrl+F7>键,开始编译。但在正式编译之前,VC 会先弹出如图 0-4 所示的对话框,询问是否建立一个默认的项目工作区。VC 必须有项目才能编译,所以这里必须回答【是】。

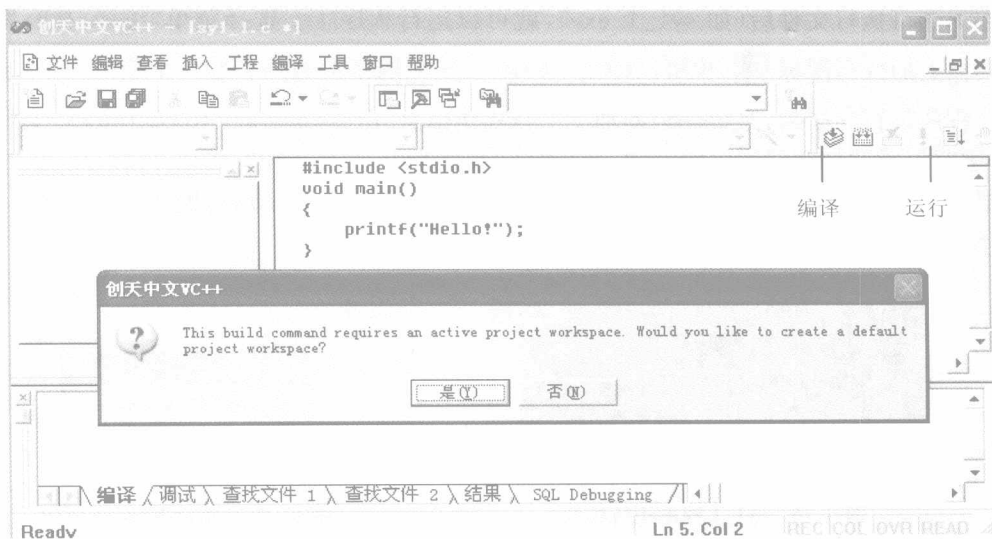


图 0-4

在进行编译时,编译系统检查源程序中的语法,并在主窗口下部的调试信息窗口输出编译的信息,如果有语法错,就会指出错误的位置和性质,并统计错误和警告的个数,如图 0-5 所示。



图 0-5

如果编译没有错误,在得到目标程序(如 sy1_1. obj)后,就可以对程序连接了,单击<F7>键或工具栏图标,生成应用程序的. EXE 文件(如 sy1_1. exe)。

以上介绍的是分别进行程序的编译与连接,实际应用中也可以直接按<F7>键一次完成编译与连接。

在得到可执行文件后(如 sy1_1. exe),就可以运行程序了。选择菜单“编译/执行”,或单击工具栏上的执行图标,也可以使用<Ctrl+F5>快捷键,程序将在一个新的 DOS 窗口中运行。程序运行完毕后,系统会自动加上一行提示信息“Press any key to continue”,如图 0-6 所示,按照提示按任意键即关闭 DOS 运行窗口返回 VC++6.0 开发环境。

图 0-7 是“编译连接”工具栏,它提供了常用的编译、连接以及运行操作命令。表 0-1 则是编译、连接以及运行命令的功能列表。

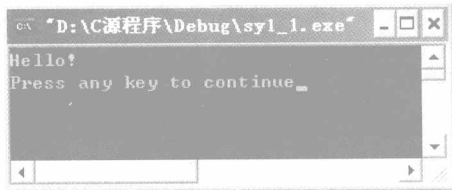


图 0-6 sy1_1 程序运行结果



图 0-7 “编译连接”工具栏

表 0-1 编译连接工具栏按钮命令及功能描述

按 钮 命 令	功 能 描 述
Compile 	编译 C 或 C++ 源代码文件
Build 	生成应用程序. EXE 文件
Stop Build 	停止编连
Execute 	执行应用程序
Go 	单步执行
Insert/Remove Breakpoint 	插入或消除断点

4. 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。若需要执行第二个程序时,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。

“文件”菜单提供关闭程序工作区功能,如图 0-8(a),执行“关闭工作区”菜单功能,然后在如图 0-8(b)对话框中选择【否】按钮。如果选择【是】按钮将同时关闭源程序窗口。

5. 程序的调试

程序调试的任务是发现和改正程序中的错误,使程序能正常运行。编译系统能检查程序的语法错误。语法错误分为两类:一类是致命错误,以 error 表示,如果程序中有这类错误,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 表示,这类

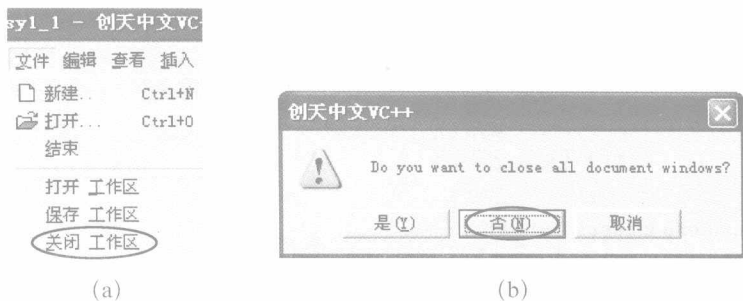


图 0-8 关闭程序工作区

错误不影响生成目标程序和可执行程序,但可能影响运行的结果,因此也应当改正,使程序既无 error,也无 warning。

在图 0-9 下方的调试窗口中可以看到编译的信息,指出源程序有 1 个 error 和 0 个 warning。用鼠标移动调试窗口右侧的滚动条,可以看到程序出错的位置和性质。用鼠标双击调试信息窗口的报错行,则在程序窗口中出现一个粗箭头指向被报错的程序行,提示出错的位置。根据出错内容提示信息(missing ';' before '}'),经检查程序,发现在第 4 行末漏写了分号。注意,在分析编译报错信息时,要检查出错点的上下行。当所有出错点均改正后,再进行编译调试,直至编译信息为: 0 error(s), 0 warning(s), 表示编译成功。

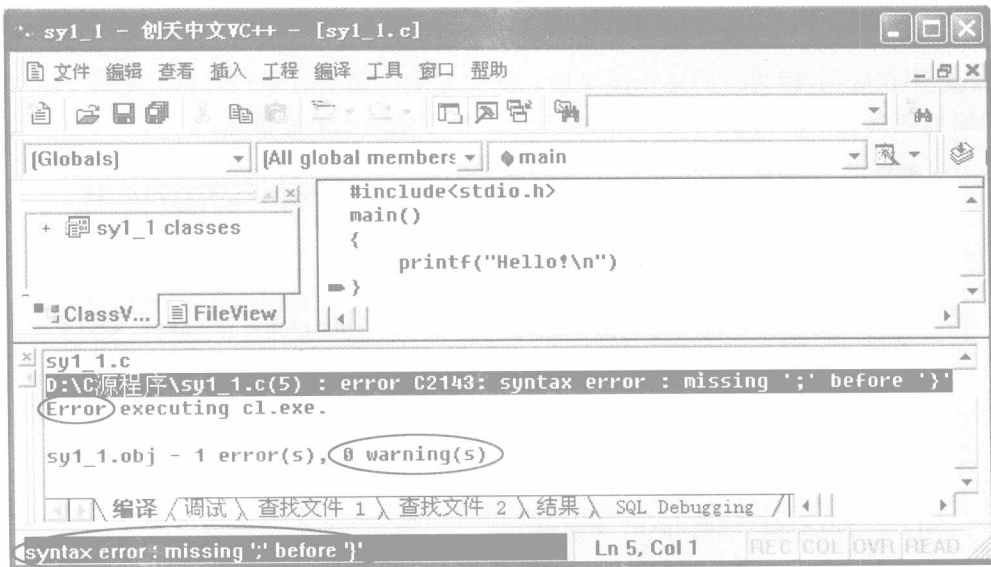


图 0-9

(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 0-10 所示,选择菜单“编译/开始调试/Run to Cursor”,或按<Ctrl+F10>,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>,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。