



普通高等教育“十一五”国家级规划教材 计算机系列教材

C程序设计 实验指导与习题集

衣治安 主 编
倪红梅 刘延军 副主编

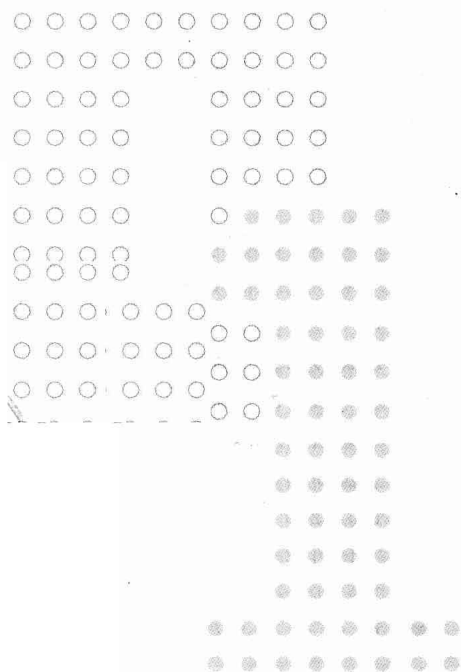
清华大学出版社



普通高等教育“十一五”国家级规划教材 计算机系列教材

衣治安 主 编
倪红梅 刘延军 副主编

C程序设计 实验指导与习题集



清华大学出版社
北京

内 容 简 介

本书是与马瑞民等编写的《C 程序设计教程》(清华大学出版社,马瑞民、衣治安主编)教材配套的辅助教材。全书共分为三个部分。第一部分是实验指导,安排了 20 个实验,除了第 1 个和第 20 个实验是介绍运行环境外,其他的每个实验都提供了 4~8 个调试程序题以及 3~6 个编程题,它们覆盖相关章节的知识点。每个调试程序后面的分析讨论都能够帮助学生举一反三。该部分的实验都不提供参考答案,以便锻炼学生独立学习的能力。第二部分是习题集,通过选择、填空、编程等题型让学生掌握 C 语言的基本语法和运行程序的能力。第三部分是习题解答,提供第二部分全部内容的参考答案。

本书适合作为高等学校各专业的本科生实验教材,实验学时建议为 30~40 学时。本书也可作为软件技术人员的参考书。

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

C 程序设计实验指导与习题集 / 衣治安主编. —北京:清华大学出版社, 2011.2
(计算机系列教材)

ISBN 978-7-302-24631-2

I. ①C… II. ①衣… 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 013507 号

责任编辑:张瑞庆 薛 阳

责任校对:焦丽丽

责任印制:李红英

出版发行:清华大学出版社

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

社 总 机:010-62770175

投稿与读者服务:010-62795954,jsjic@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015,zhiliang@tup.tsinghua.edu.cn

地 址:北京清华大学学研大厦 A 座

邮 编:100084

邮 购:010-62786544

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185×260

印 张:12.5

字 数:285 千字

版 次:2011 年 2 月第 1 版

印 次:2011 年 2 月第 1 次印刷

印 数:1~3000

定 价:19.90 元

产品编号:041114-01

本书是与马瑞民等编写的《C 程序设计教程》相配套的实验指导书和习题集,其中内容和章节的编排顺序也是一致的,按照知识的循序渐进思路进行编写,其中包含了作者以及同事十几年从事 C 语言教学的经验和总结。

根据中国高等学校计算机基础教育课程体系(CFC 2004、CFC 2006 和 CFC 2008)的精神,本着提高学生计算机应用能力和注重实践教学的宗旨,书中实验指导部分占了很大的比重,大幅度压缩了习题部分的篇幅。对于实验指导部分配有大量的分析讨论问题,希望学生在完成每个题的时候都能够达到举一反三的目的。同时对于该部分无论是改错、填空还是编程等题型,都没有给出答案,一方面是因为题目可以通过上机来完成,另外有些答案也不唯一,可以给学生以想象和思考的空间,希望能够锻炼学生。全书共设计了 20 个实验,其中全部的程序都是在 Microsoft Visual C++ 6.0 中调试通过,为了照顾到一些仍然使用 Turbo C 环境的读者,在实验 20 中给出了在 Turbo C 中运行 C 程序的基本步骤。对于习题部分,都给出了参考答案。

本书的重点是对实验教学内容进行了一些探索,大幅度增加实验中的试题量,大量增加常用的算法,希望学生通过实验能够对课堂内容有更深的了解。在实验部分,本书共涉及 142 道程序题和 79 道编程题,覆盖面是比较广泛的,目的就是为了满足不同学生的学习要求以及教师的实验安排。其中大量的程序题是按照先易后难的原则,首先给出程序,让读者在上机练习的过程中发现题中的错误,或者根据分析讨论中的提示来对程序进行完善和补充。作者试图通过这样的方法让读者能够进行更多的思考,掌握程序设计思路的多样性、程序设计方法的灵活性,同时也能够避免以往学生常犯的错误。本书编写的出发点就是让学生成为程序设计的主人,而不仅仅只是一个实践者,希望通过对本书的应用,让读者能够学到 C 程序设计的精髓,能够灵活应用 C 语言完成自己的任务。

本书涉及的题量较多,也比较全面,各位教师可以根据自己学校的授课情况选择其中的部分实验内容,同时根据具体的教学进度,调整其中的实验次数,或根据学生的具体情况进行整合。书中内容的设置能够满足不同层次学生不同进度的学习需要。读者在应用本书实验指导部分的时候一定要多练习、多实践,努力在“做中学”,在知识的积累过程中获得快乐的感受。而在应用习题部分一定要多读、多分析,不要着急参看答案以及上机操作。这部分内容重点考核学生的“走程序”能力和对基础知识的掌握情况,希望能够更好地帮助学生提高读程序和编程序的能力。

本书是在多年 C 语言教学的基础上,以提高学生综合编程能力和程序设计素养为出

发点编写的,本书由衣治安主编,倪红梅、刘延军参编,衣治安编写了实验1~15,刘延军编写了实验16~20,倪红梅编写了习题集及习题解答。在本书编写及以往应用过程中,东北石油大学计算机基础教育系的教师们提出了许多中肯的意见,在此表示衷心感谢。由于编者水平有限,书中一定存在许多不足,恳请读者提出批评意见和建议。

编 者

2010年11月

F O R E W O R D

第一部分 C 程序设计实验指导 /1

- 实验 1 用 Visual C++ 运行 C 程序 /2
- 实验 2 简单的 C 程序 /9
- 实验 3 顺序结构程序设计 /14
- 实验 4 选择结构程序设计 /19
- 实验 5 while、do...while 构成的循环 /24
- 实验 6 for 语句构成的循环 /29
- 实验 7 多重循环程序设计 /36
- 实验 8 一维数组 /41
- 实验 9 二维数组 /48
- 实验 10 字符型数据 /56
- 实验 11 变量做函数参数 /62
- 实验 12 数组做函数参数 /67
- 实验 13 递归函数 /75
- 实验 14 指针和函数 /78
- 实验 15 用指针变量处理一维数组 /85
- 实验 16 用指针变量处理二维数组 /93
- 实验 17 结构体和链表 /99
- 实验 18 文件 /106
- 实验 19 C 语言的其他应用 /112
- 实验 20 用 Turbo C 环境运行 C 程序 /119

第二部分 C 程序设计习题集 /129

- 第 1 章 概述 /130
- 第 2 章 顺序结构程序设计 /133
- 第 3 章 选择结构程序设计 /136
- 第 4 章 循环结构程序设计 /140
- 第 5 章 数组 /144
- 第 6 章 函数 /147
- 第 7 章 指针 /151
- 第 8 章 结构体与链表 /157
- 第 9 章 文件 /163

第 10 章 C 语言涉及的其他知识 /165

第三部分 C 程序设计习题解答 /173

第 1 章 概述 /174

第 2 章 顺序结构程序设计 /175

第 3 章 选择结构程序设计 /176

第 4 章 循环结构程序设计 /178

第 5 章 数组 /180

第 6 章 函数 /183

第 7 章 指针 /186

第 8 章 结构体与链表 /188

第 9 章 文件 /189

第 10 章 C 语言涉及的其他知识 /190

参考文献 /191

第一部分 C 程序设计实验指导

实验 1 用 Visual C++ 运行 C 程序

运行一个 C 程序,一般要经过编辑、编译、连接以及运行 4 个步骤。所谓的编辑就是把用户编写的 C 程序代码利用某种编辑软件输入到计算机,并形成扩展名是 c 的源程序的过程,这样形成的源程序是一个纯文本文件。把这样的源程序经过编译可以得到二进制的目标代码,文件扩展名是 obj,这个文件还不能被执行。把目标代码经过连接操作形成一个扩展名是 exe 的可执行文件。通过执行该文件,则能够得到相应的结果。

运行 C 程序的环境有很多,目前比较常用的有 Turbo C 和 Microsoft Visual C++ (简称 VC)环境,前者是 DOS 运行方式,后者是 Windows 环境。本实验将介绍在 VC 环境下运行 C 程序的方法,同时本教材的所有程序都是在该环境下调试通过的。VC 的功能非常强大,而本实验只是介绍其中编辑运行 C 程序的简单方法,按照英文界面介绍,需要的时候把中文大概的意思标注出来,如果读者的环境不是完全一致也没有太大的影响。

运行一个 C 程序的具体步骤如下。

1. 打开 VC 窗口

首先在“开始”菜单的“程序”项下找到 Microsoft Visual C++ 6.0 选项,单击该选项可以打开如图 1-1 所示的 VC 主窗口。

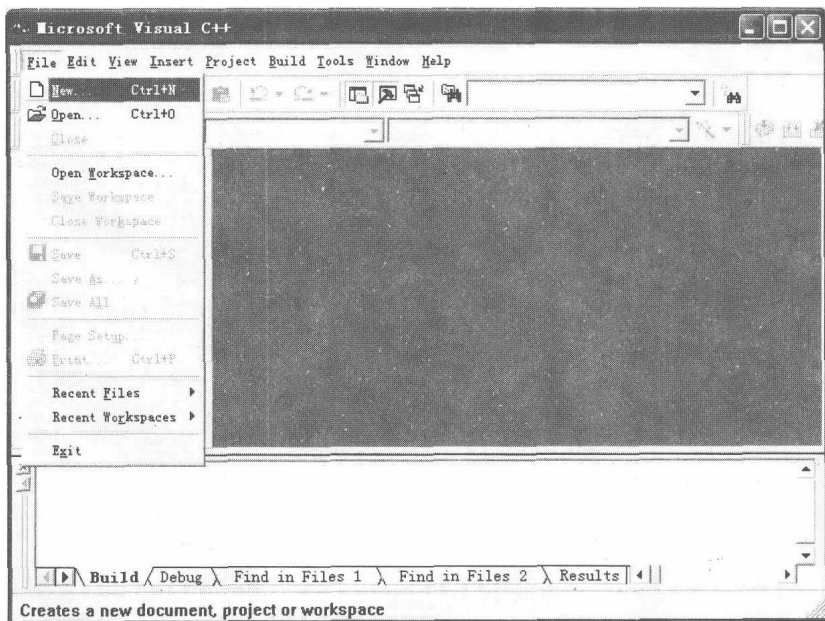


图 1-1 Microsoft Visual C++ 主窗口

为了以后应用的统一,本教材中的所有程序都保存在 E 盘的 VCLIST 文件夹下,因此,先在 Windows 中建立该文件夹后再进行后面的操作。

准备编辑一个 C 源程序。在窗口的 File 菜单中单击 New 命令,此时弹出一个 New 对话框(如图 1-2 所示)。该对话框有 4 个选项卡,默认状态下当前显示的应该是 Projects (工程)选项卡。请单击如图 1-2 所示鼠标所指的 Files 标签,显示文件选项卡的内容,然后选择其中的 C++ Source File 列表项(图 1-2 中反白显示的项),表明要编辑的是 C(或 C++)源程序。在选项卡右侧中部 File 文本框处输入文件名(此处输入的是 1-1.c)。

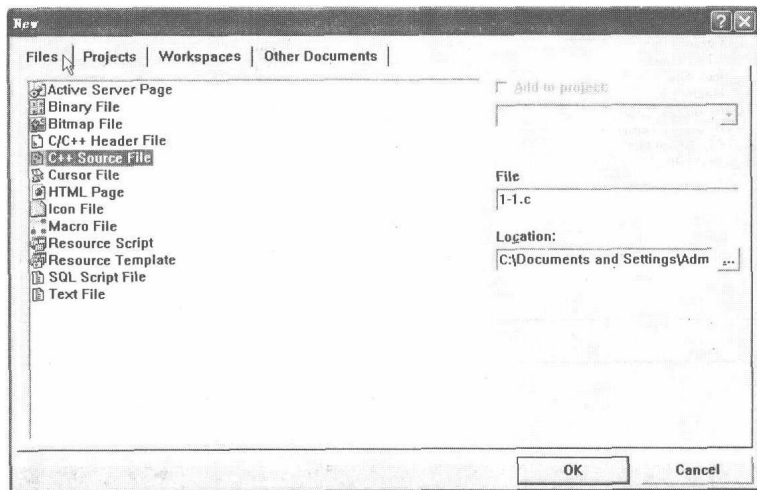


图 1-2 New 对话框的 Files 选项卡

注意,一定要为文件输入扩展名 c,以防止不必要的麻烦。因为如果读者不给源程序指定扩展名,系统将默认指定扩展名是 cpp,按 C++ 程序处理。在右下侧的 Location(位置)文本框中显示的是“C:\Documents and Settings\...”,表明源程序目前默认的存放位置。如要改变存放位置,单击其后的按钮 ...,会弹出如图 1-3 所示的 Choose Directory(选

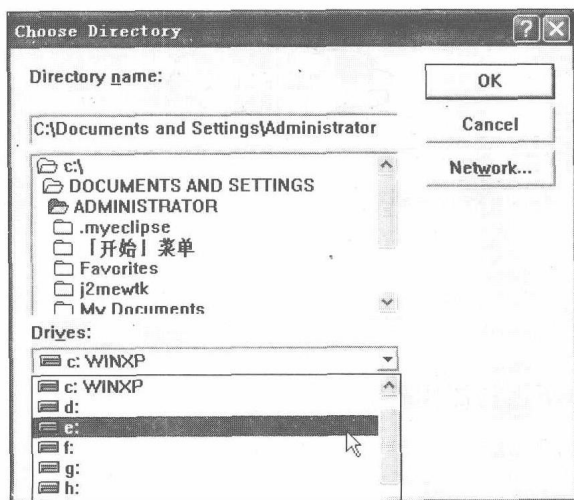


图 1-3 Choose Directory 对话框

择目录)对话框,在下部的 Drives(驱动器)选项框中先选择驱动盘(例如 e:),然后在上部的 Directory name(目录名)选项框中选择文件夹(例如 VCLIST),然后两次单击 OK 按钮,可以得到如图 1-4 所示的选择结果。此时单击 OK 按钮,进入到如图 1-5 所示的对话框。

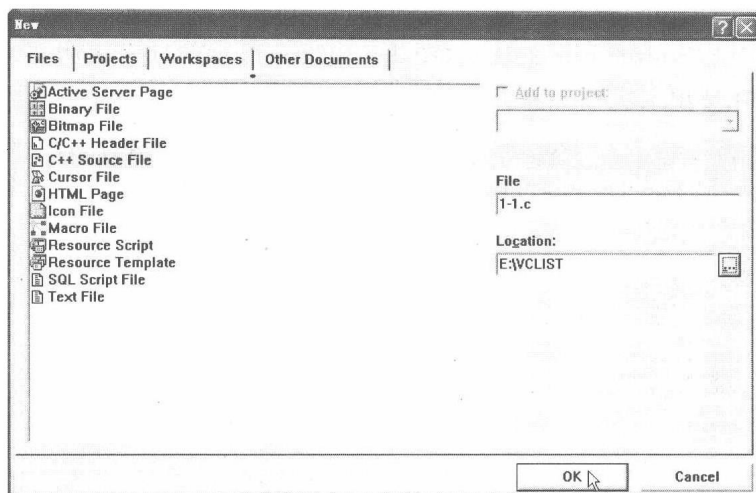


图 1-4 全部设置完成的 New 对话框

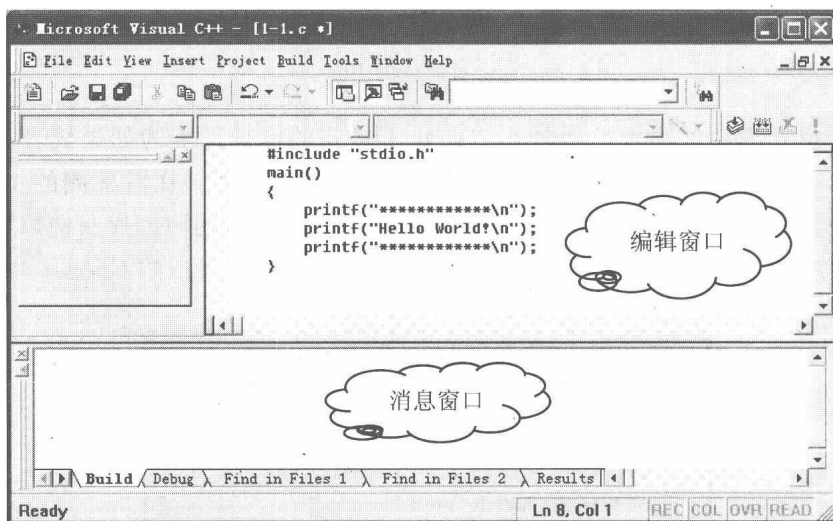


图 1-5 在 VC 中编辑 C 程序

2. 编辑源程序

在编辑窗口中输入下面的源代码,如图 1-5 所示。

```
#include "stdio.h"
main()
{
```

```

printf("*****\n");
printf("Hello World!\n");
printf("*****\n");
}

```

3. 编译、调试程序

在 Build(组建)菜单中单击 Compile(编译)命令(如图 1-6 所示),或者单击工具栏上如图 1-6 中箭头指向的编译按钮就可以进行编译了。在此过程中一般系统要提示需要建立活动工作空间以及对源文件进行保存等消息,只要单击“是”按钮即可。编译完成后,在消息窗口会出现相应的编译信息,如图 1-7 所示。

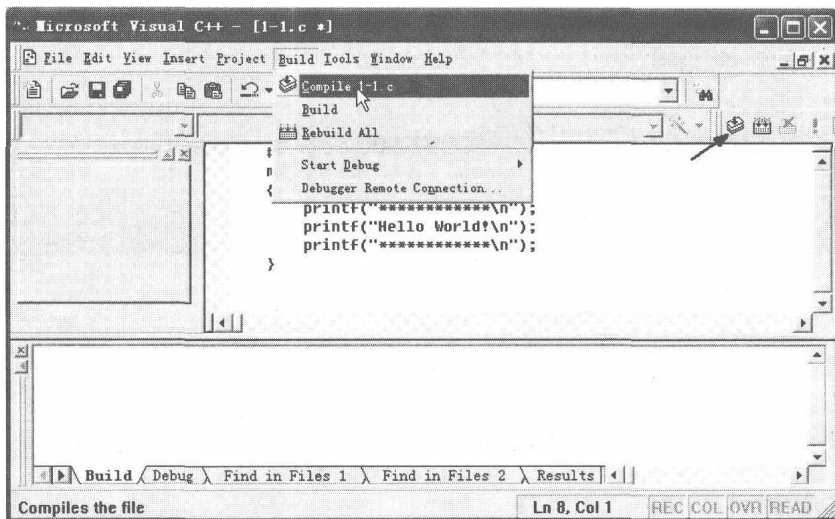


图 1-6 VC 的 Compile(编译)命令

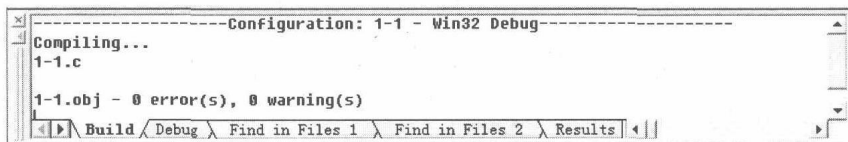


图 1-7 编译信息

从图 1-7 中可以看出,本次编译的是 1-1.c 源文件,编译生成的是 1-1.obj 文件,编译中出现 0 个 error 和 0 个 warning。编译没有 error 才证明编译通过。在编译中出现 warning 信息一般不影响进一步的操作。例如程序中定义了变量但是没有使用,或者为一个浮点型变量赋值整型数据等操作都会出现警告信息。

如果程序中含有语法错误,在编译时会指出错误的位置和错误原因。例如,在编辑的时候,作者故意把第 5 行后面的分号“;”去掉了,然后进行编译,则出现了如图 1-8 所示的编译信息。

从图 1-8 中可见,编译出现了 1 个 error,出现在第 6 行(在第 5 行的语句缺少了分号,

其错误信息一定出现在下一行,这是由于一个 C 语句允许写在多行上,在下一行的合适位置没有发现分号,则错误提示在第 6 行,希望读者掌握这种提示的情况),提示信息是在标识符 printf 之前缺少了分号“;”,双击该错误提示,在编辑区就会出现一个指向错误行的符号 ■,这样可帮助读者快速定位错误行。程序改正了错误后要重新编译。

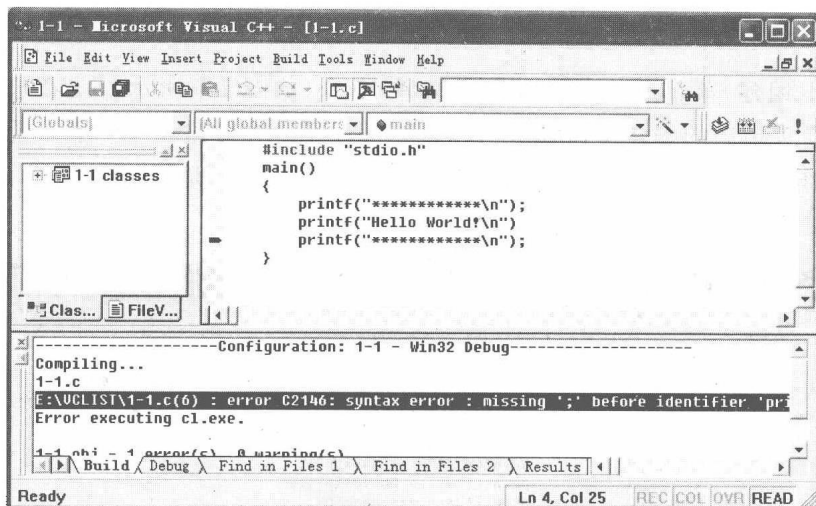


图 1-8 编译出现错误的信息提示

需要读者注意的是,有的时候编译一个程序出现了很多错误提示信息,请不要着急将错误都改正。一般来说从第一条错误改起,然后试着再编译一次。因为有的错误是连带错误(例如类型定义错误或者变量名写错等),前面的改了,后面的可能就不再是错误了。

4. 运行程序

编译成功后,Build 菜单的命令也有了变化,如图 1-9 所示,读者可以与图 1-6 中的 Build 菜单进行比较。对 C 程序一般有编辑、编译、连接和运行 4 个步骤。在 VC 系统中,

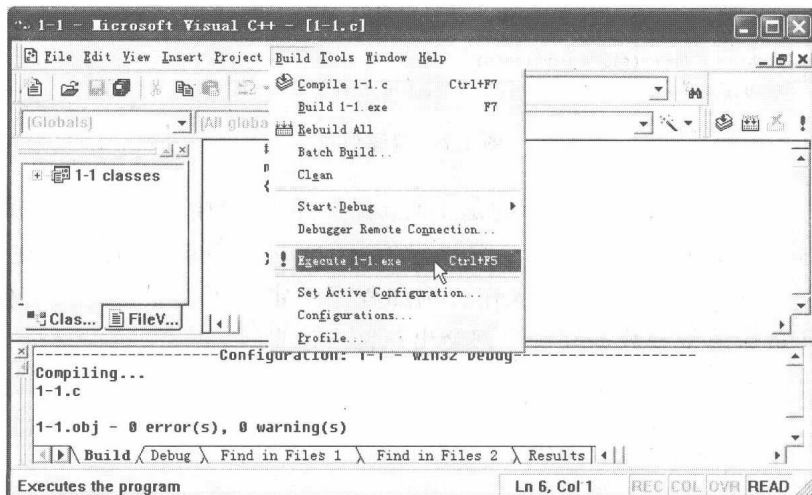


图 1-9 VC 的 Execute(执行)命令

可以通过 Build 菜单的 Build 命令来进行连接,以得到可执行文件。也可以不连接直接单击如图 1-9 中鼠标指向的 Execute(执行)命令来执行,这样系统自动把连接和执行进行了集成处理。当然也可以单击工具栏上的红色的执行程序按钮 ! (在编译通过之前,该按钮是灰色的,不能点击)。

程序运行后的输出结果如图 1-10 所示。

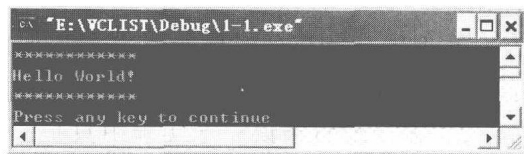


图 1-10 程序的运行结果

5. 编辑新的程序

在运行完一个程序后,如果要编辑新的程序,要特别注意几条:第一,不能直接在本程序的下面再继续写新的程序,因为一个 C 程序中只能有一个 main 函数;第二,不可以在这种条件下直接用 New 命令新建一个源程序,因为在这种情况下虽然能够编译新的程序,但是连接就会出现问题的。

正确的方法如下。

(1) 在 File 菜单中选择 Close Workspace 命令,关闭工作空间。然后再使用 File 菜单的 New 命令,按照第一个程序的方法新建文件即可。下面给出一个源程序代码(假设文件名是 1-2.c):

```
#include "stdio.h"
main()
{
    int i,sum=0;
    for(i=1;i<=100;i++)
        sum=sum+i;
    printf("sum=%d",sum);
}
```

正常执行后,该程序的运行结果如图 1-11 所示。

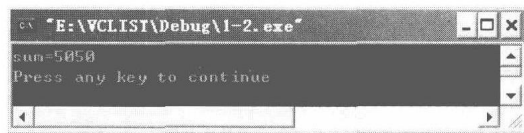


图 1-11 程序运行结果

(2) 如果仅仅是用来练习,对于以前编辑的程序并不需要保存的时候,也可以在原来的编辑状态下删除原来的文件内容,然后重新编辑新的程序。这时候文件名还是原来的文件名。

6. 打开已经存在的文件

如果源文件已经存在,需要进行修改或者再运行的时候,可以使用 File 菜单的 Open (打开)命令来打开以前保存的文件。

在打开一个新的文件的时候,同样要关闭刚刚运行程序的工作空间,否则一样会出现可以编译但是不能运行的情况。

VC 的功能是很强大的,在编辑 C 程序的时候能给读者提供很大的方便,例如系统对于关键字或者特殊的预定义标识符采用蓝色字体显示、在回车的时候一般能够采取自动缩格等处理,这些设置都能大大减少编辑错误。本实验中只是非常简单地介绍了编辑 C 程序中用到的最初级的功能,希望读者在应用中体会其他的功能。

实验 2 简单的 C 程序

一、实验目的

- 掌握程序调试的方法。
- 掌握赋值语句的应用。
- 理解整型和实型数据的应用方法。
- 掌握各种表达式的应用。

二、调试内容

1. 程序改错。

```
#include "stdio.h"
main()
{
    int a;b;c;
    a=24;
    b=13;
    c=a+b;
    printf("c=%d\n",c);
}
```

运行上面的程序,指出程序出错的原因(只有一处错误)。改正错误,运行程序并得到正确结果,进而避免在以后的程序设计中出现同类的错误。

分析讨论: 以下的讨论是在已经改正了程序错误的基础上进行的。

(1) 把程序中的倒数第 2 行改为

```
printf("c=%f\n",c);
```

程序的运行结果是否正确? 进而掌握关于输出格式符与数据类型的匹配问题。

(2) 如果把程序的倒数第 3 行去掉最后的分号“;”,然后运行程序,查看错误信息是什么? 进而掌握对漏写语句结束符错误的调试。

(3) 删除代码的第 1 行

```
#include "stdio.h"
```

再编译程序,查看出错信息。看看是否影响程序的运行结果?

提示: 在 VC 环境下编辑 C 程序最好在第一行都写上该命令行,养成一个好的习惯。

2. 整型和实型数据输出格式的练习。

```
#include "stdio.h"
```

```

main()
{
    int a,b;
    float x,y,z;
    a=12;b=12345;
    x=123.4567;
    y=11.1111111;
    z=x+y;
    printf("%d,%d,%2d,%2d\n",a,b,a,b);
    printf("%f,%f,%f\n",x,y,z);
    printf("%7.2f,%.2f,%3.2f\n",x,y,z);
}

```

分析讨论:

(1) 对于整型数据,采用“%d”和“%md”的区别。特别是采用“%md”的方式,当指定数据输出的位数 m 比数据实际的位数少的时候,结果是怎样处理的?

(2) 对于实型数据采用“%f”格式输出的时候保留几位小数? 如果实际小数位数不够或者小数多的时候是怎么处理的?

(3) 当使用“%m.nf”格式输出实型数据的时候,如果实际小数位数多于格式约定的小数位数,n 是如何处理的? 是否进行了四舍五入处理? 如果格式约定的输出宽度 m 不够的时候是如何处理的? 实际输出的位数是怎么确定的?

(4) 如果把程序的倒数 2、3 行合并为:

```
printf("%f,%f,%f\n%7.2f,%.2f,%3.2f\n",x,y,z,x,y,z);
```

这样的一行,结果会有不同吗?

(5) 采用 VC 环境编译该程序的时候,对于第 7 和第 8 行出现了什么警告错误提示信息? 如果把代码第 5 行改为

```
double x,y,z;
```

还有那样的提示信息吗?

3. 无格式符 printf 函数的输出练习。

```

#include "stdio.h"
main()
{
    int i;
    printf("*****\n");
    printf("Hello World!\n");
    for(i=1;i<=12;i++) printf(" * ");
    printf("\n");
}

```