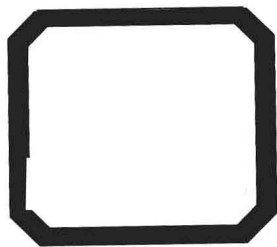


21世纪高校计算机规划教材

C语言程序设计上机指导与习题选解

于晓强 张 菁 主编

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE



计算机规划教材

C 语言程序设计 上机指导与习题选解

主 编 于晓强 张 菁
参 编 贺晓阳 康 丽 邵 利
王宏伟 赵秀岩
主 审 阎丕涛

内 容 简 介

本书是“计算机程序设计(C语言)”课程的实践教材,内容包括上机指导和习题选解两部分内容,并在附录中汇总了C语言基本语法规则。

上机指导使用环境为 Visual C++ 6.0 的集成开发环境,该部分包括 11 个相对独立的实验,每个实验根据内容的不同,分别设置了调试示例、调试练习、编程示例、编程练习等环节;习题选解部分根据作者多年的教学经验,从编者学校历年期末考试和全国计算机等级考试试题中选取难度适中的试题,以供学生学习、练习。实验和习题内容丰富,前后联系紧密,具有一定的启发性、综合性。

本书是非计算机专业学生学习 C 语言和上机实践的必备参考书。

图书在版编目(CIP)数据

C 语言程序设计上机指导与习题选解 / 于晓强, 张菁
主编. —北京: 中国铁道出版社, 2009. 12
21 世纪高校计算机规划教材
ISBN 978-7-113-10822-9

I. ①C… II. ①于…②张… III. ①
C 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 (2009) 第 224025 号

书 名: C 语言程序设计上机指导与习题选解
作 者: 于晓强 张 菁 主编

策划编辑: 严晓舟 李志国

责任编辑: 侯 颖

封面设计: 付 巍

责任印制: 李 佳

编辑部电话: (010) 63583215

封面制作: 白 雪

出版发行: 中国铁道出版社 (北京市宣武区右安门西街 8 号 邮政编码: 100054)

印 刷: 河北省遵化市胶印厂

版 次: 2010 年 1 月第 1 版 2010 年 1 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 13 字数: 318 千

书 号: ISBN 978-7-113-10822-9/TP·3677

定 价: 23.00 元

版权所有 侵权必究

凡购买铁道版图书, 如有印制质量问题, 请与本社计算机图书批销部联系调换

前 言

计算机是一门实践性非常强的学科,学习过程中的上机实践特别重要,程序设计类课程尤为突出,只有通过具体的上机操作、调试、运行,才能真正理解计算机程序的编写、调试方法与过程。实践环节是巩固理论知识,增强动手能力,提高计算机应用能力至关重要的一个教学环节。为配合“计算机程序设计(C语言)”课程的需要,我们特地组织具有丰富教学经验的一线教师编写了本书。

中国高等院校计算机基础教育改革课题研究组于2008年发表课题报告CFC 2008,提出了复合型人才对于计算机应用能力和知识结构要全面,明确了课程体系与能力培养的关系。本书就是依据应用能力培养的要求,按照应用能力培养这一核心,本着先易后难、逐步深入的原则,精心编写实验项目,并设置了调试示例、调试练习、编程示例、编程练习等环节,以方便学生在学习过程中,逐步理解、掌握相关知识,学习、理解并初步掌握基本的程序设计方法。

本书内容包括上机指导和习题选解两大部分,并在附录中汇总了C语言基本语法规则。其中上机指导分为C语言基本运行环境,数据类型、运算符和表达式,输入输出函数的使用,逻辑运算和选择结构,循环控制,数组,函数,编译预处理,指针,结构体和共用体,位运算和文件共11个实验项目;习题选解按题型分为选择题、填空题、程序改错题、程序填空题和程序设计5种。

本书由大连工业大学、大连水产学院和大连医科大学共同编写,由于晓强和张菁负责组织编写并统稿,阎丕涛主审。其中,上机指导的实验一、二、三、四由邵利编写,实验五、六、七由康丽编写,实验八、九、十、十一由贺晓阳编写,习题选解的选择题、程序改错题由于晓强负责整理,填空题由王宏伟负责整理,程序填空题由赵秀岩负责整理,程序设计题由张菁负责整理。

由于编者水平有限,疏漏和不妥之处在所难免,敬请读者批评指正。

编 者

2009年10月

目 录

第一部分 《C 语言程序设计》上机指导

上机实验指导思想和要求	1
实验一 C 语言基本运行环境.....	2
实验二 数据类型、运算符和表达式	10
实验三 输入输出函数的使用	17
实验四 逻辑运算和选择结构	22
实验五 循环控制	28
5.1 基本循环.....	28
5.2 嵌套循环.....	33
实验六 数组	37
6.1 一维数组	37
6.2 二维数组及字符数组	42
实验七 函数	45
实验八 编译预处理	51
实验九 指针	54
9.1 指针的基本使用.....	54
9.2 指针与数组	58
9.3 指针与字符串	61
实验十 结构体和共用体	65
10.1 结构体	65
10.2 共用体	68
10.3 链表.....	70
实验十一 位运算和文件	76
11.1 位运算	76
11.2 文件.....	78

第二部分 《C 语言程序设计》习题选解

题型一 单项选择题	81
1.1 C 语言概述.....	81
1.2 数据类型及其运算.....	82
1.3 C 语言输入输出	86

1.4 C 语言控制结构	88
1.5 数组	93
1.6 函数与编译预处理	95
1.7 指针	98
1.8 结构体和共用体	101
1.9 文件	103
单项选择题答案	105
题型二 填空题	107
2.1 数据类型及输入输出	107
2.2 C 语言控制结构	108
2.3 数组	110
2.4 函数与编译预处理	111
2.5 指针	112
2.6 结构体、共用体和文件	113
填空题答案	115
题型三 程序改错题	116
3.1 基本控制结构	116
3.2 数组与指针	126
3.3 字符串与指针	135
3.4 指针简单应用与递归	139
3.5 结构体、共用体、文件	141
程序改错题答案	143
题型四 程序填空题	148
4.1 基本控制结构	148
4.2 数组与指针	155
4.3 字符串与指针	161
4.4 指针简单应用及递归	164
4.5 结构体、共用体、文件	164
程序填空答案	167
题型五 程序设计	171
5.1 基本控制结构	171
5.2 数组与指针	175
5.3 字符串与指针	179
5.4 指针简单应用与递归	181
程序设计答案	183
附录 A C 语言基本语法规则汇总	187

上机实验指导思想和要求

一、上机实验的目的

学习 C 语言程序设计的目的不仅是能看懂书上的程序，也不仅是单纯地去验证教材和讲课的内容，或者验证自己所编的程序正确与否，而且是要逐步理解并初步掌握一定的程序设计能力，掌握独立分析问题、解决问题、编写程序、上机调试程序、运行程序和分析结果的能力。

所以，上机实验不仅是为了加深对讲授内容的理解，更是为了熟悉 C 语言程序的开发环境，学会上机调试程序（即善于发现程序中的错误，并且能很快排除这些错误，使程序能正确运行）。

二、上机实验前的准备工作及实验步骤

在上机实验前应事先做好准备工作，以提高上机实验的效率，准备工作至少应包括：

1. 复习和掌握与本实验有关的教学内容。
2. 准备好上机所需的程序，这可以说是学好语言课的前提。程序应书写整齐，并经人工检查无误后再上机，以提高上机效率。一定不要抄袭别人的程序或不编程序就上机！
3. 对运行中可能出现的问题应事先做出估计；对程序中有疑问的地方，应事先做好标记，以便在上机时给予注意。
4. 准备好调试和运行程序时所需的数据。
5. 上机时，输入和调试自己编写的程序，要一人一组，独立上机。上机过程中出现的问题，除了系统问题外，一般应自己独立处理。尤其对“出错信息”，更要自己分析判断，这是学习调试程序、获取经验的良好机会（尽可能以最快的速度获得更多的经验值）。程序调试通过后，应用不同的值予以检查、运行，以检查在不同的情况下程序是否正确。
6. 上机结束后，应加以总结。总结内容主要包括本次上机的任务、对运行情况的分析、调试程序所得经验，如程序未能通过，还要找出原因。这一阶段可以说是最为重要的，在这里将理论与实践二者联系起来，才可能将书本知识真正消化理解，成为自己的知识。当然，并不是说一定要将总结以书面形式加以落实，即使是在下机后的路上进行思考总结也是有用的。

实验一

C 语言基本运行环境

一、实验目的

- ☑ 熟悉 Visual C++ 6.0 环境，掌握运行一个 C 语言程序的基本步骤。
- ☑ 了解 C 语言程序的基本结构，能够编写简单的 C 语言程序。
- ☑ 能够找到错误示例中的错误，并且调试成功。

二、实验内容

1. 准备工作

在磁盘上新建一个文件夹用于存放 C 语言程序，如 D:\班级-姓名-学号。

2. 编程示例

启动 Visual C++ 6.0。执行“开始”→“所有程序”→“Microsoft Visual Studio 6.0”→“Microsoft Visual C++ 6.0”命令，进入 Visual C++ 6.0 编程环境，如图 1-1 所示。

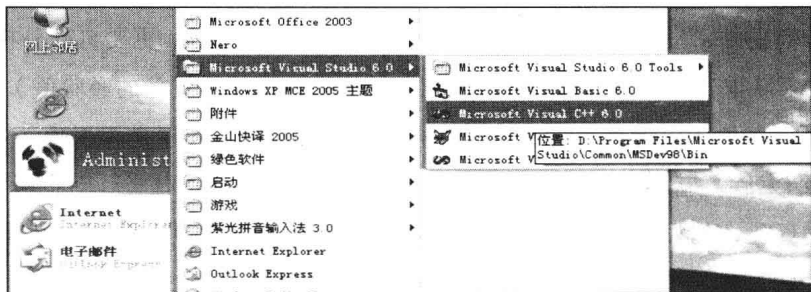


图 1-1 运行 Visual C++ 6.0

(1) 新建工程。执行“File”→“New”命令，在弹出的“New”对话框中单击“Projects”选项卡，建立一个 Win32 控制台工程（在列表框中选择 Win32 Console Application 选项），在“Project name”文本框中输入工程名称，单击“Location”文本框右侧的按钮可以改变工程的保存路径。本实验要求存放在文件夹 D:\班级-姓名-学号中，实验名字为 test1_1、test1_2，依此类推，如图 1-2 所示。

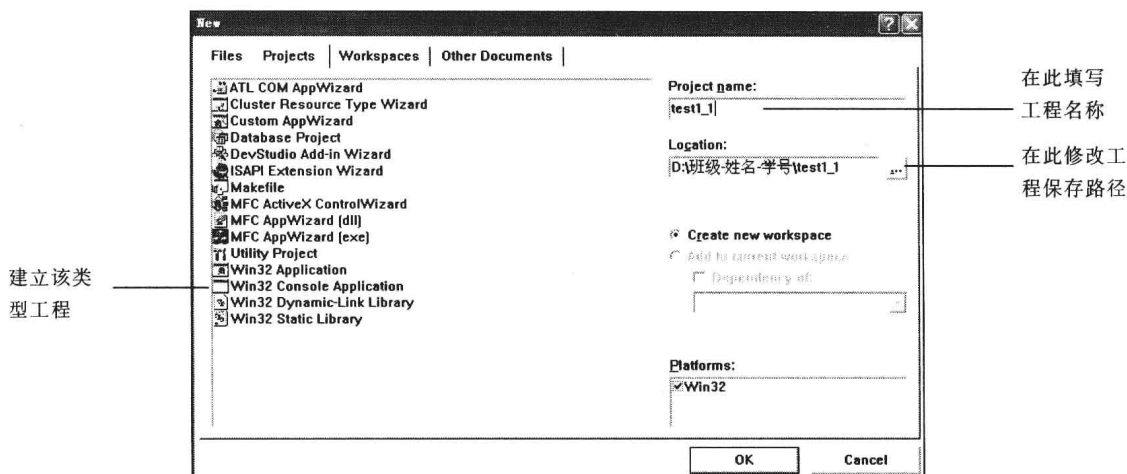


图 1-2 建立工程第一步

(2) 然后单击“确定”按钮，弹出图 1-3 所示的对话框，选择第一个单选按钮“An empty project”，最后单击“完成”按钮。

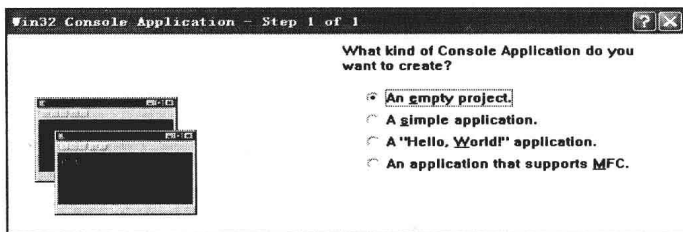


图 1-3 建立工程第二步

(3) 新建文件。执行“File”→“New”命令，在弹出的“New”对话框中单击“Files”选项卡建立一个 C++源文件（在列表框中选择 C++ Source File 选项），在“File”文本框中输入文件名称，通过单击“Location”文本框后面的按钮可以改变源文件的保存路径（一般情况下，为了保持源文件与工程路径一致，不需要做改动），如图 1-4 所示。

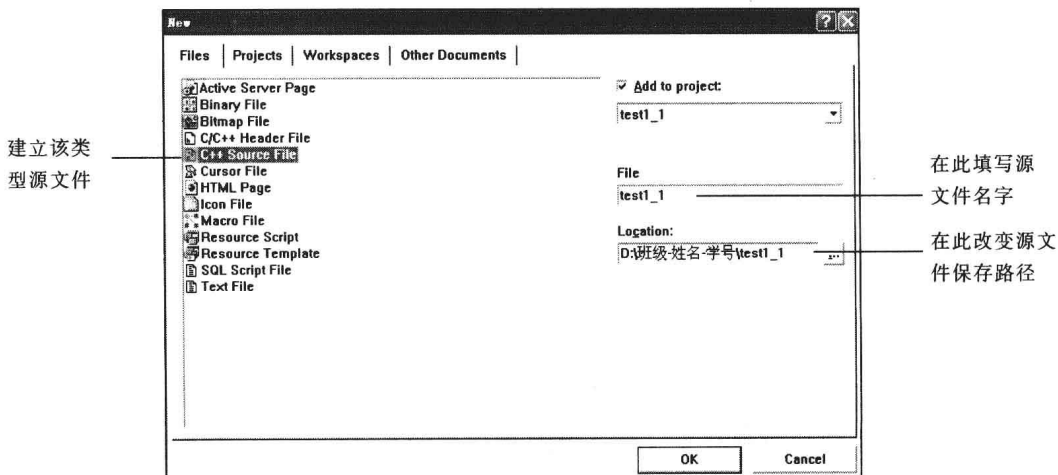


图 1-4 建立源文件

单击“确定”按钮，进入程序编辑界面，如图 1-5 所示。编辑界面包含标题栏、菜单栏、工具栏、编辑窗口、项目工作区窗口和信息窗口 6 部分。

- 标题栏：显示当前编辑窗口中文档的文件名和工程名。
- 菜单栏：包含了开发环境中几乎所有命令。
- 工具栏：列出菜单中的一些常用命令，以方便操作。
- 项目工作区窗口：包含用户项目的一些信息，包括类视图（ClassView）、文件视图（FileView）。其中类视图用来显示当前文件所包含的函数，文件视图用来显示工程所包含的相关文件（正常情况下，只有 Source File 文件夹下面有且只有一个 .cpp 文件，如果 Source File 文件夹下面有多个文件，必须删除多余文件，只能保留一个文件；如果其他文件夹下面有文件，说明工程建立错误，必须重新建立工程）。
- 编辑窗口：用来编辑源文件。
- 信息窗口：显示编译和连接时的进度及错误信息。错误信息可以通过上、下滚动条与左、右滚动条翻动察看。最后一条信息会提示是否编译连接成功，如果成功则显示“*.obj-0 error(s),0 warning(s)”；如果不成功，则会标识错误的个数和警告的个数（如果显示错误，说明程序中存在严重的错误，必须改正；如果显示警告，说明这些错误并未影响目标文件的生成，可以忽略，但通常也应该改正）。

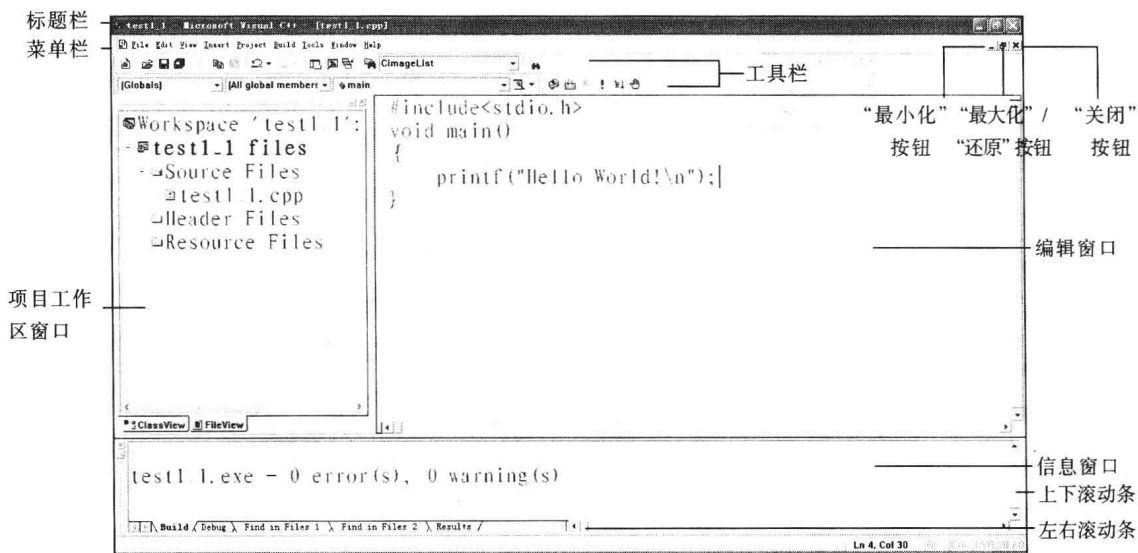


图 1-5 开发环境

注意

右上角第二排、“最小化”、“最大化” / “还原”、“关闭”按钮（见图 1-5）用来操作文档窗口的大小。

（4）编辑和保存文件。在编辑窗口中输入源程序，然后执行“File”→“Save”命令，或者按【Ctrl+S】快捷键保存源文件（没有被保存的源文件，源文件的标题栏会有一个星号）。

(5) 编译。有 3 种编译方式：

- 执行“Build”→“Compile”命令。
- 按【Ctrl+F7】快捷键。
- 按工具栏中的编译按钮，如图 1-6 所示。

编译完成在信息窗口显示编译结果，并且生成.obj 文件，如图 1-7 所示。

(6) 连接。有 3 种连接方式：

- 执行“Build”→“Build”命令。
- 按【F7】快捷键。
- 按工具栏中的连接按钮，如图 1-6 所示。

连接完成在信息窗口显示连接结果，并且生成.exe 文件，如图 1-8 所示。

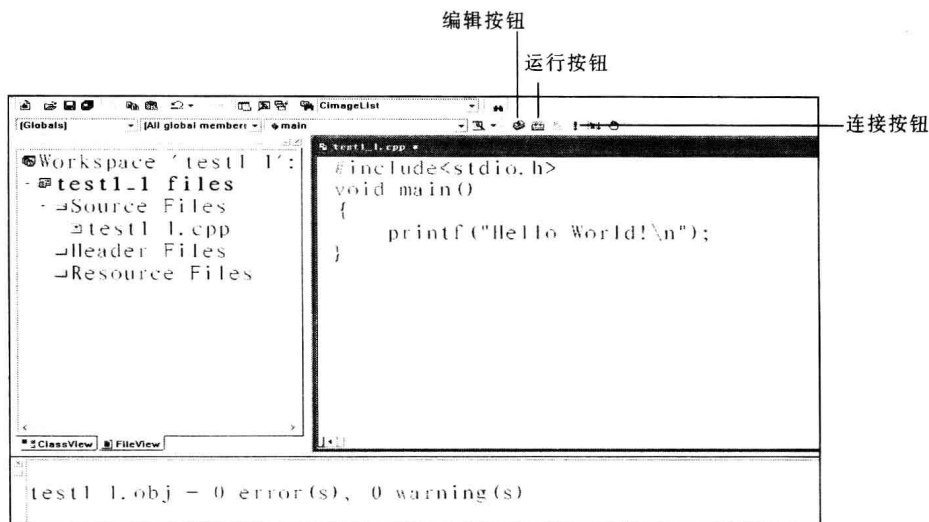


图 1-6 编辑保存文件

```
Compiling...
test1_1.cpp
test1_1.obj - 0 error(s), 0 warning(s)
```

图 1-7 编译结果

```
Linking...
test1_1.exe - 0 error(s), 0 warning(s)
```

图 1-8 连接结果

(7) 运行。有 3 种运行方式：

- 执行“Build”→“Execute”命令。
- 按【Ctrl+F5】快捷键。
- 按工具栏中的运行按钮，如图 1-6 所示。

自动弹出运行窗口，如图 1-9 所示，其中“Press any key to continue”提示用户按任意键退出运行窗口，返回编辑窗口。



图 1-9 运行结果

(8) 关闭程序工作区。执行“File”→“Close Workspace”命令。

(9) 打开文件。执行“File”→“Open Workspace”命令。

注意

(1) 编写一个程序要建立一个工程(即执行上述步骤)。

(2) 如果建立工程的时候没有改变路径,则文件存放的默认路径为“C:\Program Files\Microsoft Visual Studio\MyProjects\建立工程的名字”。

(3) 用 Visual C++ 6.0 编译的程序,所有文件均保存在工程文件夹下面(即建立工程是所填写的路径,如图 1-10 与图 1-11 所示)。其中,“.cpp”文件为源文件,“.h”文件为头文件,“.obj”文件为目标文件,“.exe”文件为可执行文件。

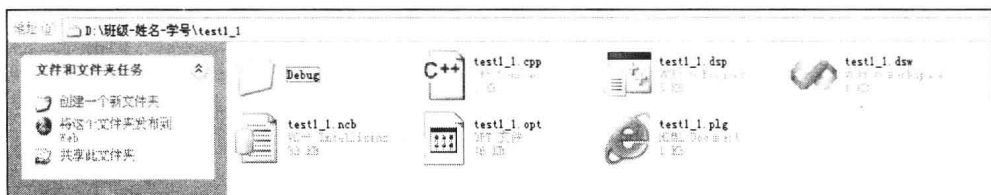


图 1-10 文件存放位置 1

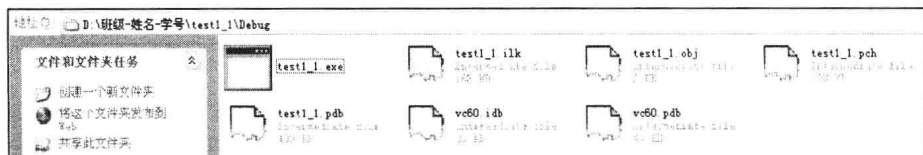


图 1-11 文件存放位置 2

3. 调试改错示例

改正下列程序中的错误,在屏幕上显示短句“Welcome to You!”。

```
#include <stdio.h>
int mian()
{
    printf(Welcome to You!\n")
    return 0;
}
```

修改后程序运行结果为:

Welcome to You!

(1) 按照编程示例中介绍的内容建立工程和文件(名称为 error1_1),并且编写源文件。

(2) 编译,结果如图 1-12 所示。

(3) 查找错误。在信息窗口中双击第一条错误信息，编辑窗口就会出现一个箭头指向程序出错的位置，如图 1-12 所示。一般在箭头的当前行、上一行或者下一行可以找到错误语句。

(4) 改正错误。图 1-12 中所示箭头指向第 4 行，错误信息指出“Welcome”是一个未定义的变量，但“Welcome”并不是一个变量，出错的原因是“Welcome”前少了个双引号，因此在“Welcome”前加上双引号即可。

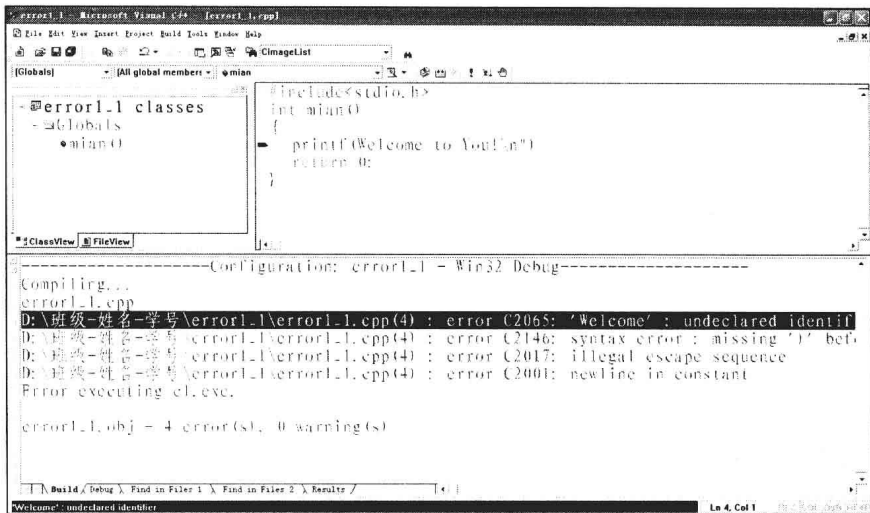


图 1-12 编译产生的错误信息

(5) 重新编译。信息窗口再次显示本次编译的错误信息，如图 1-13 所示。双击该错误信息，指向出错位置，错误信息指出在“return”前缺少分号。改正错误。

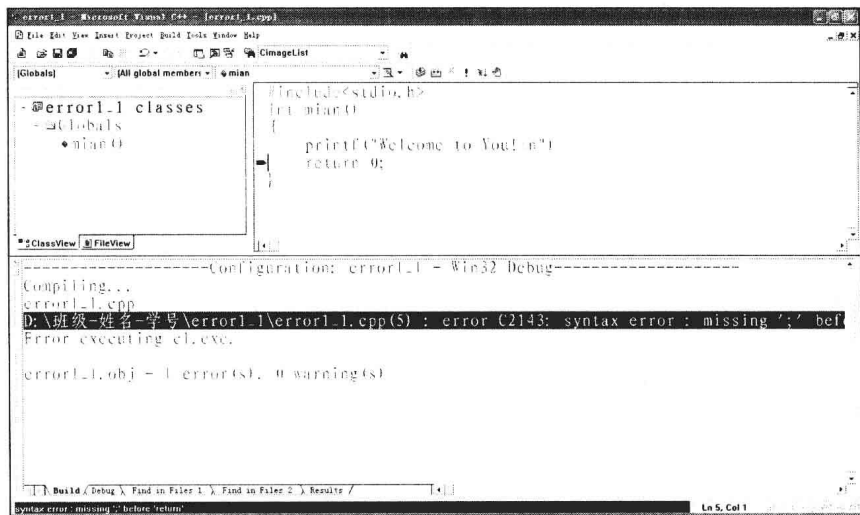


图 1-13 重新编译后产生的错误信息

(6) 再次编译，信息窗口显示编译正确。

(7) 连接。信息窗口显示连接错误信息，如图 1-14 所示。仔细检查，发现主函数名“mian”拼写错误，改正为“main”。

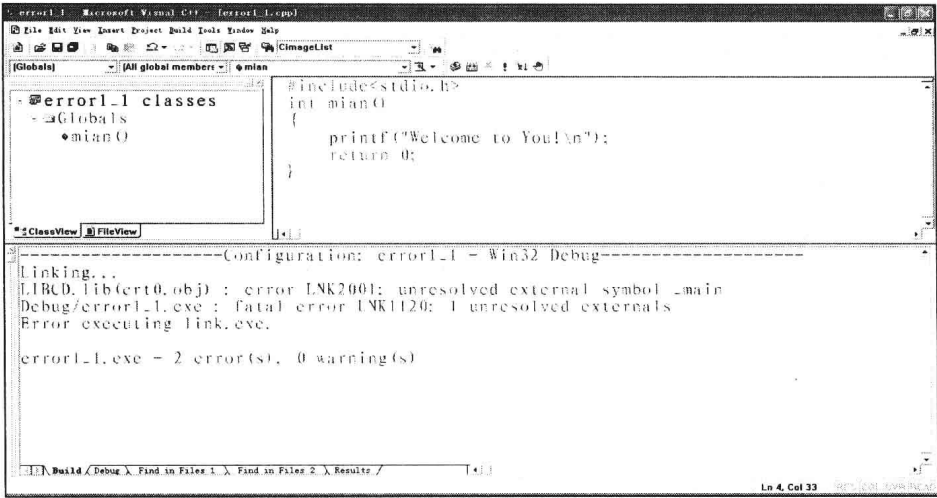


图 1-14 连接后产生的错误信息

(8) 重新编译连接，信息窗口中显示连接正确。

(9) 运行。自动弹出运行窗口，如果运行结果与题目要求结果一致，按任意键返回。程序修改成功。

4. 编程题

(1) 编写程序，在屏幕上显示你自己的班级、姓名和学号。

(2) 编写程序，在屏幕上显示图 1-15 所示的图形。

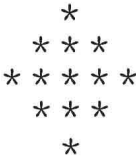


图 1-15 编辑显示结果

5. 调试练习

改正下列程序中的错误，在屏幕上显示图 1-16 所示的信息，每行 15 个星号。

```
#include <stdio.h>
int mian()
{
    printf("*****\n");
    printf("Welcome\n");
    printf("*****\n");
    return 0;
}
```



图 1-16 调试练习显示结果

实验总结

教师
检查

预习情况：	☐ 很好	☐ 较好	☐ 一般	☐ 急需改进
总结情况：	☐ 很好	☐ 较好	☐ 一般	☐ 急需改进

实验二

数据类型、运算符和表达式

一、实验目的

- ☑ 掌握 C 语言的数据类型。
- ☑ 会用 C 语言中各种运算符及其优先级。
- ☑ 掌握 C 语言中各种表达式的运算规则。

二、实验内容

1. 程序示例

给每一条语句加上注释，并且写出程序的运行结果，然后上机验证，分析得到该结果的原因。注意，“□”表示空格（书中下同）。

```
#include <stdio.h>           //添加头文件
void main()                  //主函数又称之为入口函数
{ char c1,c2;                //定义两个字符型变量
  int d1=32;                  //定义两个整型变量
  c1='a';   c2='b';           //为字符变量赋值
  c1=c1-d1;   c2=c2-d1;        //字符变量与整型变量进行算术运算
  printf("%c□□%c\n",c1,c2);   //在屏幕上以字符形式显示 c1, c2 的值
  printf("%d□□%d\n",c1,c2);   //在屏幕上以整型的形式显示 c1, c2 的值
}
```

运行结果： A B
 65 66

【分析】

- ① 字符型变量赋值必须用单引号，并且单引号中间只能有一个字符。
- ② 字符型可以与整型进行混合运算。因为字符型在计算机中是以 ASCII 码值进行处理的。小写字母 a 的 ASCII 值减去 32 之后得到的 ASCII 值为 65，正好是大写字母 A 的 ASCII 值；同理小写字母 b 的 ASCII 值减去 32 之后得到的 ASCII 值为 66，正好是大写字母 B 的 ASCII 值。

③ printf 语句中“%c”用来输出字符变量的值，“%d”用来输出整型变量的值。

2. 阅读程序写结果

给每一条语句加上注释，并且写出题目运行结果，然后上机验证，分析得到该结果的原因。

(1) 算术运算符——加减法运算符和表达式。

```
#include <stdio.h>
void main()
{
    int a=4,b=7;
    printf("%d□%d□%d\n",a=a-1,b+a,b+1);
}
```

【知识点】赋值语句与表达式的区别。例如：

① 赋值语句“a=a+b;”的作用是将 a 与 b 的和赋给 a。运行结果：a 的值发生变化，b 的值没有发生变化，整个表达式的值为 a 的值。

② 算术表达式“a+b”的运行结果：a 的值没有发生变化，b 的值没有发生变化，整个表达式的值为 a+b 的值。

【运行结果】

【结果分析】

(2) 算术运算符——乘除取余运算符和表达式。

```
#include <stdio.h>
void main()
{
    int a=5; int b=9;
    int x,y,m,n,k;
    x=a/b; y=b/a;
    m=a%b; n=b%a;
    k=x*y*m*n;
    printf("%d,%d,%d,%d,%d",x,y,m,n,k);
}
```

【知识点】除法与取余的运算规则。两个整型变量进行除法运算，得到的最终结果仍为整型，如有小数，要求取整。两个变量取余，必须均为整型，如果有 a%b，a 小于 b，则结果为 a。

【运行结果】

【结果分析】