



高等院校计算机基础精品系列规划教材

C语言程序设计 实验指导与习题解答

孙淑霞 主编



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

高等院校计算机基础精品系列规划教材

C 语言程序设计 实验指导与习题解答

主 编：孙淑霞

参 编：（按姓氏拼音排序）

陈晓丹 樊富有 柳 军

罗兴贤 罗 奕 王星捷

肖阳春 银 星 袁继敏

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书作为《C 语言程序设计》(孙淑霞等编著)一书的配套用书,其主要内容由 3 部分组成:实验指导、习题解答和模拟试题。实验指导部分由 11 个实验组成,实验内容的难度和实验量适中,并给出了相应的指导;习题解答部分是针对《C 语言程序设计》一书的课后习题给出的解答;模拟试题部分是为了便于学生自我测试对本课程的掌握情况,同时又熟悉计算机等级考试的题型。全书内容根据普通院校教学的实际情况编写,实用性较强。

本书适合作为高等院校非计算机专业本科生、研究生学习 C 语言程序设计课程的教学辅导书,也可作为 C 语言程序设计自学者的参考用书。

图书在版编目(CIP)数据

C 语言程序设计实验指导与习题解答/孙淑霞等编著.
北京:中国铁道出版社,2009.3
(高等院校计算机基础精品系列规划教材)
ISBN 978-7-113-09730-1

I. C… II. 孙… 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(2009)第 027502 号

书 名: C 语言程序设计实验指导与习题解答
作 者: 孙淑霞 主编

策划编辑: 秦绪好 杨 勇

责任编辑: 王占清

编辑助理: 李庆祥

封面设计: 付 巍

编辑部电话: (010) 63583215

封面制作: 白 雪

责任印制: 李 佳

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: 2009 年 6 月第 1 版 2009 年 6 月第 1 次印刷

开 本: 787mm×1092mm 1/16 印张: 9.5 字数: 219 千

印 数: 5 000 册

书 号: ISBN 978-7-113-09730-1/TP·3166

定 价: 16.00 元

版权所有 侵权必究

本书封面贴有中国铁道出版社激光防伪标签,无标签者不得销售

凡购买铁道版的图书,如有缺页、倒页、脱页者,请与本社计算机图书批销部调换。

高等院校计算机基础精品系列规划教材

编审委员会

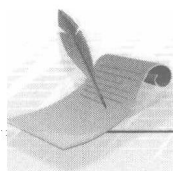
顾问：黄迪明 黎明

主任：匡松

副主任：孙淑霞 胡念青 刘玉萍 何振林 王锦 王超
杜小丹 刘容 王绍清 邢跃 袁继敏 关忠仁
管群 王晓 吕峻闽 蒋明礼 刘科 刘家彬
陈宗荣

委员：(排名不分先后)

管群	四川大学	唐应辉	四川师范大学
关忠仁	成都信息工程学院	匡松	西南财经大学
丁照宇	成都理工大学	黎明	西南石油大学
何振林	成都中医药大学	樊富友	宜宾学院
王晓	四川理工学院	黄敏	西南科技大学
邢跃	绵阳师范学院	卓学锋	内江师范学院
刘帮涛	泸州医学院	李进	四川警察学院
袁继敏	攀枝花学院	刘亚平	西昌学院
王锦	西华师范大学	杨清平	四川文理学院
王超	四川农业大学	蒋明礼	西华大学
刘玉萍	西南民族大学	胡念青	四川师范大学文理学院
刘毅	乐山师范学院	吕峻闽	西南大学天府学院
张志敏	四川大学锦城学院	伍良富	四川大学锦江学院
刘科	四川外语学院成都学院	许泽明	西南石油大学
柳军	成都理工大学工程技术学院	肖阳春	成都理工大学



序

21 世纪的高校计算机基础教育进入了一个新的时期。为了适应日新月异、快速发展的信息化社会对大学生的实际需要,使大学生拥有更丰富的计算机知识和更强的计算机应用技能,计算机基础课程的教学内容必须紧跟当前计算机技术的发展和水平,教学模式、教学方法和教学手段需要深入改革和突破,更加注重计算机综合能力、实践动手能力与创新精神的全面培养,使大学生能够在今后的学习和工作中,将计算机技术与本专业紧密结合,并有效地应用于各专业领域,大力提升学生的社会适应能力和竞争力。

教材作为教学指导思想、培养目标、教学要求、教学内容的载体和具体体现,可以帮助教师全面、具体地理解教学改革要求与教学内容,并以此为依据进行讲授并组织教学活动。学生通过教材进行学习,掌握知识和能力。教材的好坏,关乎教学质量能否得以保障。

为了更好地推动四川省本科院校教师的计算机基础教育的最新研究成果在一线教学中得以实践,中国铁道出版社精心组织四川省计算机教育专家、教授、一线教师队伍编写和出版了“高等院校计算机基础精品系列教材”。

本系列教材根据教育部对高等学校计算机基础教学提出的指导意见和基本要求,以社会需求为导向,以拓宽知识面、提高计算机应用能力、培养创新精神为目标编写而成,同时认真贯彻和体现中国高等院校计算机基础教育改革课题研究组的最新研究成果——《中国高等院校计算机基础教育课程体系》的思路和课程要求。

本系列教材的主编和作者都是多年深入教学第一线、教学经验丰富的专家、教授,是一大批国家级与省级教学改革研究项目、国家“十一五”规划教材、精品课程的负责人,他们对计算机基础教育改革的方向和思路有深切的体会和清醒的认识。因而可以说,本系列教材是他们的最新研究成果、教学经验全面总结的具体化。

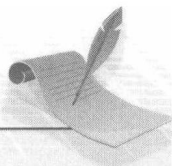
本系列教材的出版和推广,对进一步推动计算机基础教学的深入改革,提高计算机基础课程的教学质量,将发挥积极作用和深远影响。

教授

全国高等学校计算机教育研究会理事
全国高等院校计算机基础教育研究会理事
四川省高等学校计算机应用知识与能力等级考试委员会委员

匡 松

前言



程序设计课程是高等院校理工科各专业计算机应用能力训练的必修课之一。C 语言既具有高级语言的优点,又具有低级语言的许多特点,是目前国内外广泛使用的一种程序设计语言,也是很多高校程序设计课程的首选语言。

本书是《C 语言程序设计》课程配套的实验辅导教材,其内容是按照教育部非计算机专业计算机基础课程教学指导分委会提出的 C 语言程序设计课程教学基本要求中的较高要求编写的。全书由 3 部分组成:实验指导、习题解答和模拟试题。

实验指导部分由 11 个实验组成:C 语言简单程序的编写和调试、C 语言程序设计基础、选择结构程序设计、循环结构程序设计、数组、指针、函数、编译预处理与变量的存储类型、结构体与共用体,文件和 C++ 程序设计基础;实验内容的难度和实验量适中,并给出了相应的指导。习题解答部分是针对《C 语言程序设计》一书的课后习题给出的解答。考虑到一些学校学生有计算机等级考试的需求,给出了模拟试题将便于学生进行自我测试,同时熟悉计算机等级考试的题型。

本书主要具有以下特点:

(1) 面向低年级大学生,因此实验内容充分考虑了低年级学生所具有的数学基础知识,在掌握语言的学习基础上,注重学生的编程训练。

(2) 习题解答部分针对主教材的课后习题编写,为了读者更好地理解习题中的知识点,对中等难度以上的习题给出了较为详细的解答。

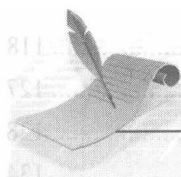
(3) 模拟试题的题型兼顾了全国计算机等级考试和省计算机等级考试,便于学生考前复习测试。

本书实验部分的实验 1、实验 2 由孙淑霞编写,实验 3、实验 4 由罗奕编写,实验 5、实验 6 由罗兴贤编写,实验 7、实验 8 由肖阳春编写,实验 9 由陈晓丹编写,实验 10 由袁继敏和银星编写,实验 11 由樊富有编写;习题解答部分的第 1~3 章由孙淑霞编写,第 4、5 章由罗奕编写,第 6、7 章由罗兴贤编写,第 8 章由柳军编写,第 9 章由王星捷编写,第 10 章由陈晓丹编写,第 11 章由袁继敏和银星编写,第 12 章由樊富有编写;模拟试题由孙淑霞编写;全书由孙淑霞统稿。

由于编者水平有限,书中不妥之处在所难免,恳请广大读者批评指正。

编著者

2008 年 12 月



目 录

第一部分 实验指导

实验 1	C 语言简单程序的编写和调试	1
实验 2	C 语言程序设计基础	5
实验 3	选择结构程序设计	9
实验 4	循环结构程序设计	14
实验 5	数组	17
实验 6	指针	22
实验 7	函数	26
实验 8	编译预处理与变量的存储类型	30
实验 9	结构体与共用体	32
实验 10	文件	39
实验 11	C++程序设计基础	44

第二部分 习题解答

第 1 章	C 语言简单程序的编写和调试	47
第 2 章	算法与结构化程序设计基础	48
第 3 章	C 语言程序设计基础	49
第 4 章	选择结构程序设计	53
第 5 章	循环结构程序设计	58
第 6 章	数组	63
第 7 章	指针	75
第 8 章	函数	85
第 9 章	编译预处理与变量的存储类型	92
第 10 章	结构体与共用体	97
第 11 章	文件	103
第 12 章	C++程序设计基础	111

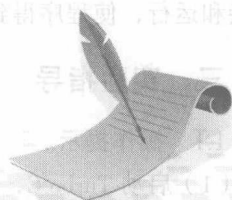


第三部分 模拟试题

第 1 套笔试题	118
第 1 套上机题	127
第 1 套笔试题解答	128
第 1 套上级题解答	134
第 2 套笔试题	134
第 2 套上机题	142
第 2 套笔试题答案	143
第 2 套上机题解答	143

第一部分

实验指导



实验 1 C 语言简单程序的编写和调试

一、实验目的和要求

1. 熟悉 Turbo C 集成环境的使用方法，掌握在 Turbo C 集成环境下输入、编译、连接、运行和调试 C 程序的基本过程和方法。
2. 熟悉 Visual C++ 6.0 编译系统的窗口界面，学会在该系统下编辑、编译、连接、运行和调试 C++ 程序的基本方法。
3. 通过编写简单程序，掌握 C 程序的基本组成和结构，以及用 C 程序解决实际问题的步骤。

二、实验内容

- 1.1 在 Turbo C 集成环境中输入程序 1-1.C，然后对其进行编译、连接和运行。

```
/* 1-1.C */
#include <stdio.h>
void main()
{
    printf("Hello, C!\n");
}
```

- 1.2 练习程序 1-2.C 的调试，直到 1-2.C 调试正确。

```
/* 1-2.C */
#include <stdio.h>
void mian()
{
    int a,b,sum;
    a=40;b=50;
    sum=a+b;
    printf("Hello, C!\n")
}
```



```
printf("Sum is %d\n",sum);  
}
```

1.3 在 Visual C++ 6.0 中练习单文件程序 1-3.CPP 的编译、连接和运行,使程序得到正确结果。

1.4 在 Visual C++ 6.0 中练习多文件程序 CPP1.CPP 和 CPP2.CPP(参见教材第 1 章)的编译、连接和运行,使程序得到正确结果。

三、实验指导

1.1 操作提示。

(1) 启动 Turbo C, 进入 Turbo C 集成环境。

(2) 输入程序 1-1.C。

(3) 执行 Compile 菜单中的 Compile to OBJ 命令, 编译程序 1-1.C, 生成目标程序 1-1.OBJ。

(4) 执行 Compile 菜单中的 Link EXE file 命令, 连接目标程序 1-1.OBJ 和函数库 stdio.h, 生成可执行程序 1-1.EXE。

(5) 执行 Run 菜单中的 Run 命令, 按组合键【Alt+F5】, 观察屏幕上输出的程序运行结果:

```
Hello, C!
```

1.2 操作提示。

(1) 执行 Compile 菜单中的 Compile to OBJ 命令, 编译程序 1-2.C, 这时 Turbo C 集成环境的 Message 窗口中出现如下信息:

```
Compiling E:\C\1-2.C:  
Error E:\C\1-2.C 9: Statement missing; in function main  
Warning E:\C\1-2.C 10: 'sum' is assigned a value while is never used in function  
main
```

其中:

第 1 条信息表示当前正在编译的程序是 E: 盘文件夹 C 中的 1-2.C。

第 2 条信息指出在程序 1-2.C 的第 9 行有语法错误, 错误是在 main() 函数的第 9 行缺少分号“;”。注意, 这种错误通常是因为上一行的末尾少了分号“;”, 而不是当前行少分号。

第 3 条信息指出在程序 1-2.C 的第 10 行有警告错误, 错误是在 main() 函数中, 变量 sum 被赋予了一个未使用过的值。这个错误是由于上一个错误造成的, 如果在上一行的末尾加上分号, 该错误就不会再出现。

提示:在调试程序的过程中, 如果出现编译错误, 要由上至下一个一个地修改, 每改一处, 就要重新编译一次, 这是因为有时一个错误会引起下面程序段中与之有关的行也出现错误, 改正了这一个错误, 其他与之有关行的错误也就随之消失了。

(2) 执行 Compile 菜单中的 Compile to OBJ 命令, 重新编译程序 1-2.C, 没有错时生成文件 1-2.EXE。

这时不再有错。可见, 上面的警告错误并没有被修改就自动消失了, 这是因为改正了第 2 条信息中的错误。因此, 修改 C 程序编译错误时, 每修改一处, 就要编译一次程序, 后面的一些错误可能是前面的某一处错误引起的。

(3) 执行 Compile 菜单中的 Make EXE file 命令, 连接 1-2.OBJ 生成文件 1-2.EXE。

这时出现一处错误, Message 窗口中显示如下错误信息:

```
Linking E:\C\1-2.EXE:
```

```
Linking Error: Undefined symbol '_main' in module COS
```

其中:

第1条信息表示当前正在连接的程序是E: 盘文件夹C中的1-2.EXE。

第2条信息指出连接错误是: 没有定义main()函数, 造成该错误的原因是因为函数名main写错了。

提示:

有些错误会出现在连接阶段。例如, 在程序1-2.C中, 把main误写成了mian, 编译程序把mian当成是用户自定义函数进行编译, 没有语法错误, 也就没有报错。但由于C程序必须要有一个且只能有一个main()函数, 连接程序没有发现main()函数, 因此在连接阶段报错。

与之类似的错误, 如将printf误写为print, 连接程序也会因为找不到相应的库, 在连接阶段给出连接错误的提示信息:

```
Undefined sybner '_print'
```

说明 print 是未定义符号。

1.3 操作提示。

(1) 在 Visual C++ 6.0 中打开程序 1-3.CPP。

启动 Visual C++ 6.0, 打开 Visual C++ 6.0 窗口。执行 File 菜单中的 Open 命令, 或单击“标准”工具栏上的“打开”按钮, 弹出“打开”对话框。在该对话框中的“查找范围”下拉列表框中指定待打开文件被存放的位置, 在文件列表框中找到要打开的文件 1-3.CPP, 双击该文件, 或单击该文件再单击“打开”按钮。

(2) 编译。

选择下面方法之一进行编译:

- ① 执行 Build 菜单中的 Compile 1-3.cpp 命令;
- ② 单击“编译微型条”工具栏中的“编译”按钮;
- ③ 按【Ctrl+F7】组合键编译程序 1-3.CPP。

如果没有建立工程工作区, 将弹出如图 1-1 所示的对话框。询问是否要建立一个默认的工程工作区。

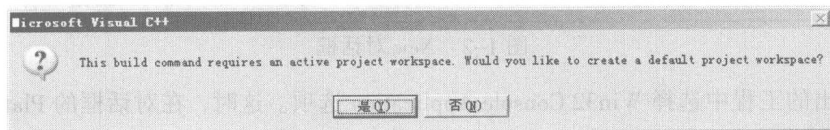


图 1-1 信息提示对话框

单击“是”按钮, 开始编译该文件。编译过程中, 如果有错, 则在主窗口下方的编译输出窗口中显示错误信息。错误信息将指出错误的性质、出错位置(比如行数)以及错误原因; 如果没有错, 编译输出窗口中就会显示生成的.OBJ 文件名。例如:

```
1-3.obj - 0 error(s), 0 warning(s)
```

修改编译错误时, 在编译输出窗口中双击某一错误信息行时, 在该错误信息行前就会出现一个提示箭头。用户可以根据错误的性质修改程序, 修改后再重新编译, 直到没有任何错误为止。



(3) 连接。

执行 Build 菜单中的 Build 1-3.exe 命令或按【F7】键对被编译后的目标文件进行连接。如果连接过程中发现错误，则会发出连接错误的信息，修改程序直到连接没有错误为止。编译连接成功时，在编译输出窗口中就会显示如下信息：

```
1-3.exe-0 error(s) 0warning(s)
```

表明这时生成了可执行文件 1-3.exe。

(4) 执行。

执行 Build 菜单中的 Execute 1-3.exe 命令，或按【Ctrl+F5】组合键，1-3.exe 文件被执行，并在另一个输出结果的窗口中显示结果：

```
area is 200
```

```
Press any key to continue
```

按任意键后，关闭输出结果窗口，回到源程序窗口。

1.4 操作提示。

(1) 创建工程 (project)。

在 Visual C++ 6.0 主窗口中，执行 File 菜单中的 New 命令，弹出 New 对话框，选择 Projects 选项卡，如图 1-2 所示，然后按以下步骤执行：

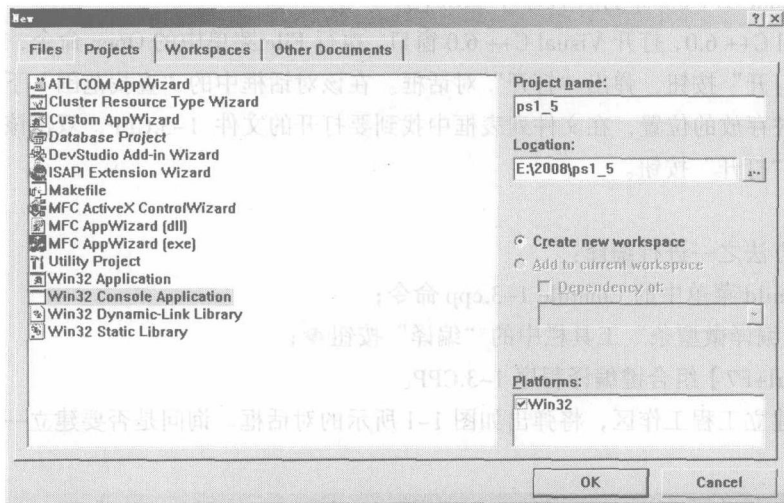


图 1-2 New 对话框

① 在列出的工程中选择 Win32 Console Application 选项。这时，在对话框的 Platforms 列表框中出现 Win32 选项；

② 在右边的 Projects Name 文本框中输入要建立的工程名（如 ps1_5）；

③ 在 Location 框中选择工程所在的路径，单击 OK 按钮。

④ 弹出 Win32 Console Application-Step 1 of 1 对话框，选择 An empty project 单选按钮，单击 Finish 按钮，弹出 New Project Information 对话框，该对话框中将显示所创建的控制台应用程序新框架的项目特性，单击 OK 按钮，返回系统主窗口。

这时，空工程 ps1_5 创建结束。

(2) 向工程中添加文件。

在系统主窗口中, 执行 Projects | Add File into Project | Files 菜单命令, 弹出 Insert Files into Project 对话框, 在该对话框中, 在“查找范围”下拉列表框内选择要添加到当前工程中的文件所在的目录(文件夹), 在列表框内选定要添加的所有文件, 如图 1-3 所示。

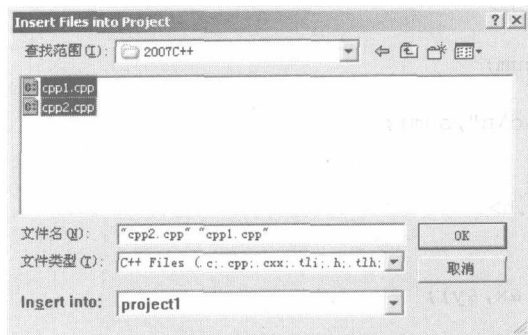


图 1-3 Insert Files into Project 对话框

单击“OK”按钮。

(3) 编译连接和运行工程文件。

执行 Build 菜单中的 ps1_5.exe 命令, 系统将对 ps1_5 中的各个文件逐个进行编译, 然后连接。如果没有错, 则生成一个可执行文件, 并执行该文件, 运行结果显示在 MS-DOS 窗口中。

实验 2 C 语言程序设计基础

一、实验目的和要求

1. 掌握 C 语言的运算符和表达式的正确使用方法。
2. 理解和掌握运算符与运算对象的关系。例如, 单目运算符只对一个运算对象进行操作, 双目运算符需要两个运算对象。
3. 理解和掌握运算符的优先级和结合方向。
4. 通过编写程序, 掌握 C 语言的几种基本数据类型, 如整型 int、字符型 char、实型 float、双精度型 double, 以及由这些基本类型构成的常量和变量的使用方法。
5. 掌握基本输入/输出函数的使用方法。其中包括 printf() 函数、scanf() 函数、getchar() 函数和 putchar() 函数。

二、实验内容

2.1 按下面要求运行程序 2-1.C, 观察运行结果。理解整除和不同类型数据进行混合运算的转化规律, 对结果进行合理的解释。

(1) 在 TC 中直接运行程序 2-1.C。

(2) 对程序 2-1.C 作如下修改再运行:

- ① 将 average 改为 float 类型。
- ② 将 $\text{average}=(x+y+z)/3$ 改为 $\text{average}=(x+y+z)/3.0$ 。
- ③ 将 $\text{printf}(\text{"ave}=\text{d}\backslash\text{n"}, \text{ave})$ 改为 $\text{printf}(\text{"ave}=\text{f}\backslash\text{n"}, \text{ave})$ 。



2.2 依次运行程序 S2-2-1.C、S2-2-2.C、S2-2-3.C，比较 3 个程序，认识如何为用户编写一个好用的程序。

```
/* 2-2-1.C */
#include <stdio.h>
void main()
{ int x=1,y=1,sum;
  sum=x+y;
  printf("sum=%d\n",sum);
}
/* 2-2-2.C */
#include <stdio.h>
void main()
{ int x,y,sum;
  scanf("%d%d",&x,&y);
  sum=x+y;
  printf("sum=%d\n",sum);
}
/* 2-2-3.C */
#include <stdio.h>
void main()
{ int x,y,sum;
  printf("\nEnter x y: ");
  scanf("%d%d",&x,&y);
  sum=x+y;
  printf("sum=%d\n",sum);
}
```

2.3 运行程序 2-3.C，观察输出的结果，并对输出结果作出合理的解释。

```
/* 2-3.C */
#include <stdio.h>
void main()
{ float a1,a2;
  double b1,b2;
  a1=3141.59; a2=0.000001;
  b1=3141.59; b2=0.000001;
  printf("%f, %lf\n",a1+a2,b1+b2);
}
```

2.4 程序 2-4.C 的功能是从键盘上输入 $x=15$, $y=20.7$, $c=M$ ，然后将输入的内容输出到屏幕上。调试程序 2-4.C，修改有错误的语句行，并输出正确的结果。

```
/* 2-4.C */
#include <stdio.h>
void main()
{ int x;
  float y;
  char c;
  scanf("x=%d,y=%d,c=%c",x,y,c);
  printf("\nx=%d,y=%d,c=%c",x,y,c);
}
```

2.5 编写程序 2-5.C。已知 $a=b=c=1$ ，计算 $++a||++b\&\&++c$ 表达式和 $++a\&\&++b||++c$ 表达式的值，通过计算表达式后， a 、 b 、 c 的值分别等于多少？

2.6 改错题。程序 2-6.C 的功能是计算下列公式的值。

$t=1+1/2+1/3+1/4+1/5$

改正程序中的错误，不得增行或删行，也不得更改程序的结构，使程序能得到正确结果 2.283333。

```
/* 2-6.C */
#include <stdio.h>
void main()
{
    double t=1.0;
    t+=1/2+1/3+1/4+1/5;
    printf("\nThe result is %d\n",t);
}
```

三、实验指导

2.1 编程及操作提示。

(1) 程序运行结果为：

average=1

在程序 2-1.C 中，x、y、z 被定义为整型变量， $(x+y+z)/3$ 是整除运算，其结果是舍去小数部分，保留整数部分，不采用四舍五入。

注：在 C 程序中做整除运算要特别小心，否则会得到不是预期的结果。例如： $2/3=0$ 。

(2) 程序运行结果为：

average=1.666667

语句 $\text{average}=(x+y+z)/3.0$ ；将原来的整除改为了整型与实型数据的混合运算，并按 %f 格式输出，因此结果不再是整数。

2.2 操作提示。

(1) 程序 2-2-1.C 运行结果为：

sum=2

在程序 2-2-1.C 中，变量 x、y 的值是定义的同时赋予的，使程序只能求 $x=1$ 与 $y=1$ 的和，如果要求其他数据的和，必须修改程序中的初始化值，然后重新编译、连接并运行。这种编程方法不灵活，只供初学者编程使用。

(2) 程序 2-2-2.C 运行实例如下：

```
1 1✓
sum=2
```

在程序 2-2-2.C 中，变量 x、y 的值是在程序运行过程中通过 scanf() 函数输入的，这种方法不需要对程序进行任何修改，就可以计算其他数据之和。但在输入数据前，没有任何提示信息，初学者很难知道程序在等待输入。

(3) 程序 2-2-3.C 运行实例如下：

```
Enter x y: 1 1✓
sum=2
```

在程序 2-2-3.C 中，变量 x、y 的值也是在程序运行过程中通过 scanf() 函数输入的，与程序 2-2-2.C 不同的是，在输入数据前，增加了屏幕提示信息：



Enter x y:

用户可按屏幕提示信息进行数据的输入。这种方法比前两种更易于理解,更加灵活和人性化。

提示:从键盘输入数据时,最好先给出提示信息,提示要输入的数据(例如,2-2-3.C 中在输入 x 和 y 前用 printf() 函数输出的屏幕提示信息)。

2.3 操作提示。

程序 2-3.C 运行结果如下:

```
3141.590089, 3141.590001
```

程序中的 a1 和 b1、a2 和 b2 的值分别相同,但是程序中 a1+a2 和 b1+b2 的输出结果却不会相同。这是因为 float 型数据和 double 型数据的精度不同,实型数在计算机中是非精确表示的,用 float 型变量作两个位数很大的数的加法运算或除法运算时,将得不到预期的结果,而用 double 型变量进行运算则可以得到较准确的结果 3141.590001。原因是 float 型数据和 double 型数据的精度不同, float 型数据的有效位数只有 6~7 位。编程时应该根据要求定义变量的类型。

2.4 操作提示。

程序 2-4.C 在编译时有警告错误,如果不改正这些警告错误,程序将得不到正确的运行结果。

调试程序 2-4.C 时要注意的问题有:

(1) scanf() 函数的输入表列要求是变量地址的表列,否则在编译时会给出警告错误:

```
Possible use of 'x' before definition in function main
Possible use of 'y' before definition in function main
Possible use of 'c' before definition in function main
```

(2) scanf() 函数的格式描述符要与输入表列中变量的类型一致。如果类型不一致,将得不到正确的输入值。如果 float 类型变量用 %d 格式输入,得到的是 0; int 型变量用 %f 格式输入,运行时将给出错误信息:

```
scanf: floating point formats not linked
Abnormal program termination
```

(3) 如果 scanf() 函数的格式描述符中有按原样输入的字符,例如 “x=y,c=” 这些字符。输入数据时一定要按格式输入这些字符。运行该程序时应该输入:

```
x=25,y=36.7,c=C
```

如果输入:

```
25,36.7,C
```

则得不到正确结果。

(4) printf() 函数的格式描述符要与输出表列中变量的类型一致。否则得不到正确的输出结果。如果 float 型变量用 %d 格式输出,得到的是 0; int 型变量用 %f 格式输出,运行时会给出错误信息:

```
printf: floating point formats not linked
Abnormal program termination
```

2.5 编程提示。

(1) 在 ++a||++b&&++c 和 ++a&&++b||++c 表达式前都要对 a、b、c 变量赋初值 1。参考程序如下:

```
/* 2-5.C */
#include <stdio.h>
void main()
```

```

{ int a,b,c;
  a=b=c=1;
  ++a||++b&&++c;
  printf("\na=%d,b=%d,c=%d",a,b,c);
  a=b=c=1;
  ++a&&++b||++c;
  printf("\na=%d,b=%d,c=%d",a,b,c);
}

```

程序 2-5.C 运行结果如下:

```

a=2,b=1,c=1
a=2,b=2,c=1

```

(2) 在进行逻辑表达式的计算中要注意:

- ① 对于 $++a||++b&&++c$ 运算, 若 $++a$ 为真, 后面的运算就不再进行。
- ② 对于 $++a&&++b||++c$ 运算, 若 $++a$ 为真, 继续计算 $++b$, 若 $++b$ 为真, 就不再计算 $++c$; 若 $++b$ 为假, 则继续计算 $++c$ 。
- ③ 逻辑运算或关系运算的结果是一个逻辑值, 即 0 或 1。

2.6 改错提示。

(1) 程序中的 $1/2+1/3+1/4+1/5$ 是整除运算, 得不到正确的结果。应该将分子或分母改为实数。例如, $1/2.0+1/3.0+1/4.0+1/5.0$ 。

(2) t 为 double 型的, 输出不应该用 $\%d$ 。

实验 3 选择结构程序设计

一、实验目的和要求

1. 学会正确使用逻辑运算符和逻辑表达式。
2. 熟练掌握 if 语句和 switch 语句。

二、实验内容

3.1 程序改错。下面程序的功能是输入一个整数, 判断它是奇数还是偶数, 改正其中的错误, 使其得到正确结果。

```

/* 3-1.C */
#include <stdio.h>
void main()
{ int n;
  printf("Enter n: ");
  scanf("%d",n);
  if(n%2=0)
    printf("%d 是偶数. \n",n);
  else
    printf("%d 是奇数. \n",n);
}

```

3.2 程序改错。下面程序的功能是判断输入的整数是否大于等于 7, 且小于 10, 并输出结果。