

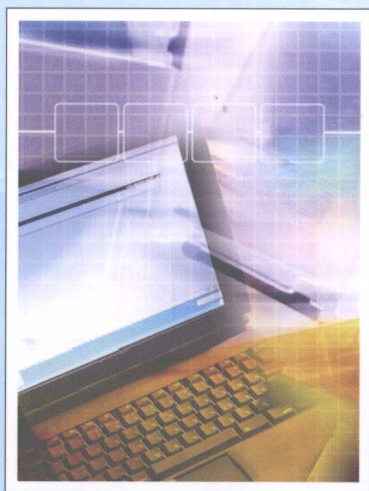


21世纪高等学校系列教材

21 Shiji Gaodeng Xuexiao Xilie Jiaocai

# C语言程序设计 实验指导与习题解答

蒋清明 主编 黄晓宇 向德生 何宏 编著



人民邮电出版社  
POSTS & TELECOM PRESS



清华大学出版社  
TSINGHUA UNIVERSITY PRESS

# C语言程序设计 实验指导与习题解答

张松海 主编 张松海 张松海 张松海 张松海



清华大学出版社  
TSINGHUA UNIVERSITY PRESS



21世纪高等学校系列教材

21 Shiji Gaodeng Xuexiao Xilie Jiaocai

图例索引 (CIP) 数据

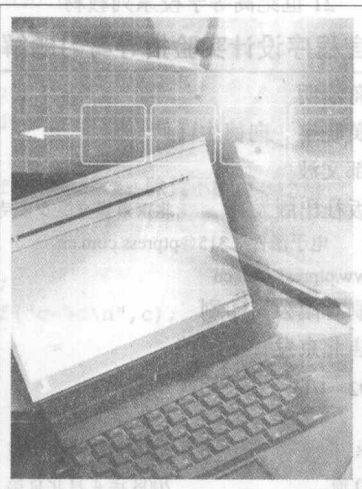
C语言程序设计习题解答与实验指导  
黄晓宇、向德生、何宏、蒋清明主编  
北京：人民邮电出版社，2008.4

(21世纪高等学校系列教材)

ISBN 978-7-112-17503-8

# C语言程序设计 实验指导与习题解答

蒋清明 主编 黄晓宇 向德生 何宏 编著



定价：19.00元

印刷：2008年11月 北京人民邮电出版社

人民邮电出版社

北京



## 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 / 蒋清明主编; 黄晓宇, 向德生, 何宏编著. —北京: 人民邮电出版社, 2008.4

(21 世纪高等学校系列教材)

ISBN 978-7-115-17503-8

I. C… II. ①蒋…②黄…③向…④何… 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 005274 号

## 内 容 提 要

本书是《C 语言程序设计》的配套实验指导和习题解答, 主要包括 3 部分: C 语言程序设计实验指导、C 语言程序设计习题与解答、C 语言程序设计等级考试二级模拟试卷。

本书的实验目的、实验任务和实验方法明确, 可大大改善实验效果, 减轻教师指导学生实验的工作量; 同时, 书中给出的题解有助于学生巩固知识、拓宽思路、提高程序设计水平。

本书可以作为各类高等院校、高职院校计算机专业及理工科非计算机专业学生学习“计算机程序设计”课程的教材, 也可作为广大计算机爱好者学习 C 程序设计语言的参考书。

21 世纪高等学校系列教材

## C 语言程序设计实验指导与习题解答

- ◆ 主 编 蒋清明
- 编 著 黄晓宇 向德生 何 宏
- 责任编辑 邹文波
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京华正印刷有限公司印刷  
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16  
印张: 11  
字数: 267 千字 2008 年 4 月第 1 版  
印数: 1—6 500 册 2008 年 4 月北京第 1 次印刷

ISBN 978-7-115-17503-8/TP

定价: 19.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154

# 目 录

## 第 1 部分 C 语言程序设计

|                                    |    |                     |    |
|------------------------------------|----|---------------------|----|
| 实验指导 .....                         | 1  | 实验 8 存储类型 .....     | 16 |
| 实验 1 Visual C++ 6.0 IDE 上机操作 ..... | 1  | 8.1 实验目的 .....      | 16 |
| 过程 .....                           | 1  | 8.2 实验范例 .....      | 16 |
| 1.1 实验目的 .....                     | 1  | 8.3 实验 .....        | 17 |
| 1.2 实验范例 .....                     | 1  | 实验 9 一维数组 .....     | 19 |
| 1.3 实验步骤 .....                     | 1  | 9.1 实验目的 .....      | 19 |
| 实验 2 输入/输出操作 .....                 | 3  | 9.2 实验范例 .....      | 19 |
| 2.1 实验目的 .....                     | 3  | 9.3 实验 .....        | 20 |
| 2.2 实验范例 .....                     | 3  | 实验 10 二维与多维数组 ..... | 20 |
| 2.3 实验 .....                       | 4  | 10.1 实验目的 .....     | 20 |
| 实验 3 分支结构 .....                    | 4  | 10.2 实验范例 .....     | 21 |
| 3.1 实验目的 .....                     | 4  | 10.3 实验 .....       | 24 |
| 3.2 实验范例 .....                     | 5  | 实验 11 指针 .....      | 24 |
| 3.3 实验 .....                       | 6  | 11.1 实验目的 .....     | 24 |
| 实验 4 循环结构 .....                    | 8  | 11.2 实验范例 .....     | 24 |
| 4.1 实验目的 .....                     | 8  | 11.3 实验 .....       | 25 |
| 4.2 实验范例 .....                     | 8  | 实验 12 命令行参数 .....   | 25 |
| 4.3 实验 .....                       | 9  | 12.1 实验目的 .....     | 25 |
| 实验 5 break/continue 语句 .....       | 11 | 12.2 实验范例 .....     | 25 |
| 5.1 实验目的 .....                     | 11 | 12.3 实验 .....       | 26 |
| 5.2 实验范例 .....                     | 11 | 实验 13 结构体 .....     | 27 |
| 5.3 实验 .....                       | 12 | 13.1 实验目的 .....     | 27 |
| 实验 6 函数 .....                      | 13 | 13.2 实验范例 .....     | 27 |
| 6.1 实验目的 .....                     | 13 | 13.3 实验 .....       | 30 |
| 6.2 实验范例 .....                     | 13 | 实验 14 共用体 .....     | 30 |
| 6.3 实验 .....                       | 14 | 14.1 实验目的 .....     | 30 |
| 实验 7 递归函数 .....                    | 14 | 14.2 实验范例 .....     | 30 |
| 7.1 实验目的 .....                     | 14 | 14.3 实验 .....       | 31 |
| 7.2 实验范例 .....                     | 15 | 实验 15 文件操作 .....    | 32 |
| 7.3 实验 .....                       | 15 | 15.1 实验目的 .....     | 32 |
|                                    |    | 15.2 实验范例 .....     | 32 |
|                                    |    | 15.3 实验 .....       | 32 |

## 第 2 部分 C 语言程序设计习题 与解答

|                 |    |
|-----------------|----|
| 题解 1 绪论         | 34 |
| 1.1 习题          | 34 |
| 1.2 习题解答        | 35 |
| 题解 2 基本数据类型与运算符 | 35 |
| 2.1 习题          | 35 |
| 2.2 习题解答        | 38 |
| 题解 3 控制结构       | 39 |
| 3.1 习题          | 39 |
| 3.2 习题解答        | 43 |
| 题解 4 函数         | 50 |
| 4.1 习题          | 50 |
| 4.2 习题解答        | 54 |
| 题解 5 数组         | 57 |
| 5.1 习题          | 57 |
| 5.2 习题解答        | 62 |
| 题解 6 指针         | 67 |
| 6.1 习题          | 67 |
| 6.2 习题解答        | 72 |
| 题解 7 结构与共用      | 77 |
| 7.1 习题          | 77 |
| 7.2 习题解答        | 85 |
| 题解 8 文件         | 90 |

|          |    |
|----------|----|
| 8.1 习题   | 90 |
| 8.2 习题解答 | 96 |

## 第 3 部分 C 语言程序设计等级 考试二级模拟试卷

|            |     |
|------------|-----|
| 试卷 1       | 102 |
| 1.1 笔试试题   | 102 |
| 1.2 上机操作试题 | 113 |
| 1.3 参考答案   | 115 |
| 试卷 2       | 117 |
| 2.1 笔试试题   | 117 |
| 2.2 上机操作试题 | 127 |
| 2.3 参考答案   | 129 |
| 试卷 3       | 131 |
| 3.1 笔试试题   | 132 |
| 3.2 上机操作试题 | 144 |
| 3.3 参考答案   | 146 |
| 试卷 4       | 147 |
| 4.1 笔试试题   | 148 |
| 4.2 上机操作试题 | 159 |
| 4.3 参考答案   | 161 |

## 附录 常见出错信息及排错示例

|      |     |
|------|-----|
| 参考文献 | 172 |
|------|-----|

# 第 1 部分

## C 语言程序设计实验指导

### 实验 1 Visual C++ 6.0 IDE 上机操作过程

#### 1.1 实验目的

1. 学会用 Visual C++ 6.0 IDE 来编辑、编译、执行一个 C 语言程序，为学习复杂 C 语言程序的编写、调试和执行打好基础。
2. 通过运行简单的 C 语言程序，初步了解 C 程序的特点及在 Visual C++ 6.0 IDE 环境下的调试方法。

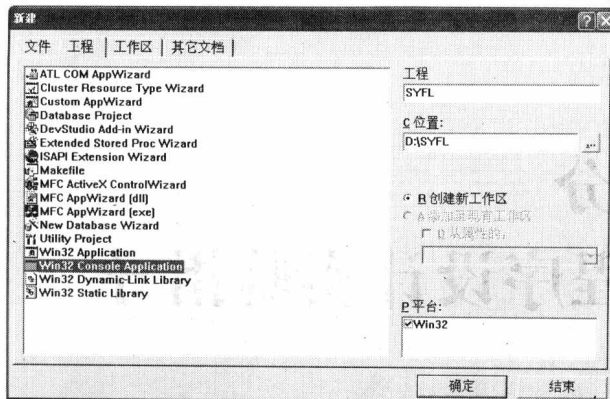
#### 1.2 实验范例

**[范例]** 下面程序实现在 Visual C++ 6.0 IDE 环境下输出：This is my first C program!

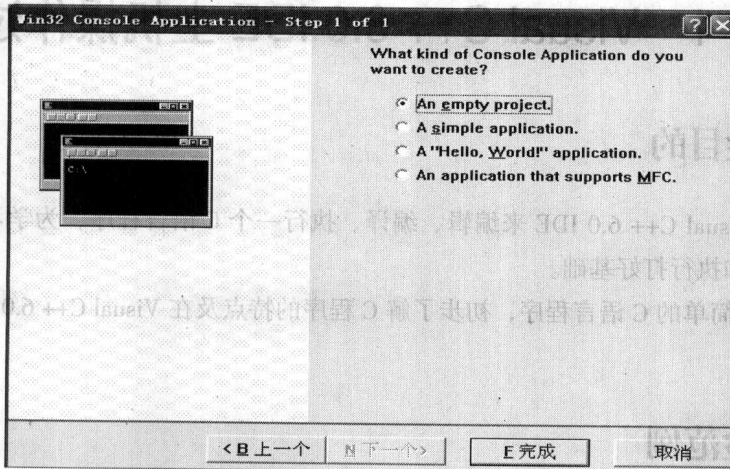
```
/* syfl1-1.c */
#include "stdio.h"
void main( )
{
    printf("This is my first C program!");
}
```

#### 1.3 实验步骤

1. 启动 Visual C++ 6.0，单击“文件(F)”菜单中的“新建(N)...”子菜单项，系统将弹出“新建”对话框。
2. 选择新建对话框中的“工程”选项卡，再选中“Win32 Console Application”项，在“工程名(N)”文本框中输入欲建工程名，如 SYFL；然后在“位置(C)”文本框中输入欲保存该工程的路径(Visual C++ 6.0 IDE 自动将用户输入的工程名作为文件夹名)，或是通过单击其右边的...按钮，在弹出的“选择目录”对话框中选择保存源文件的路径。

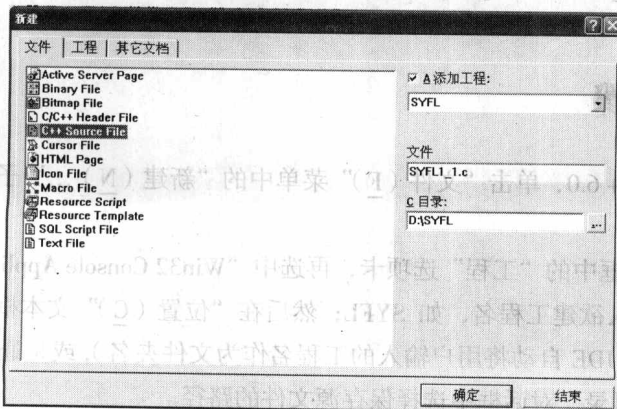


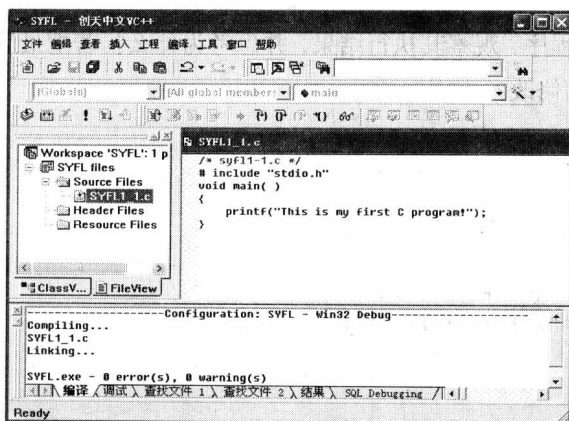
3. 单击“确定”按钮，系统将弹出一个对话框让用户选择建立何种工程。选中“一个空工程 (An empty project)”的单选项后并单击“完成”按钮。



4. 向工程中添加源文件并编辑保存源文件。

在下拉菜单——项目 (Project) 加到项目 (Add to project) 中选择新 (new) 标签，再选择文件 (File) 标签，选择文件类型为 C++ Source File，输入源文件名如 SYFL1\_1.c，选择保存源文件的位置，按确定按钮后将生成一个新的空文件 SYFL1\_1.c，并出现源文件编辑窗口，在编辑窗口中输入与修改程序代码，完成后可保存源文件。





### 5. 运行项目程序。

选择下拉菜单——编译 (Build) | 开始调试 (Start Debug) | 运行 (Go), 对应的快捷方式为 F5, 将运行项目程序。

### 6. 写出实验报告, 实验报告要求如下。

- (1) 记录简单 C 程序在上机调试运行时出现的各种问题及其解决方法。
- (2) 简明扼要地写出在调试运行一个 C 程序时的完整步骤。
- (3) 总结本次实验的经验与教训。

## 实验 2 输入/输出操作

### 2.1 实验目的

1. 掌握 C 语言的基本数据类型及其定义方法。掌握 C 运算符的种类、运算优先级和结合性。
2. 掌握不同类型数据间的转换与运算, 掌握 C 表达式类型 (赋值表达、算术表达式、关系表达式、逻辑表达式、条件表达式和逗号表达式) 和求值规则。
3. 掌握基本的输入/输出函数 scanf(), getchar(), printf() 和 putchar() 函数。
4. 进一步熟悉 C 程序的编辑、编译、连接和运行的过程。

### 2.2 实验范例

**[范例 1]** 运行下列程序, 观察其执行结果, 并思考为什么?

```
/*syfl2_1.c*/
#include <stdio.h>
main( )
{
    char c1='a',c2='b',c3='c',c4='\102',c5='\x61';
    printf("a%c%c\t%c\tabc\n",c1,c2,c3);
    printf ("\t\tb%c %c\n",c4,c5);
    printf ("\\\t\\t\\t\\t");
    printf ("\n%c\t%d",c1,c1);
}
```

**[范例 2]** 运行下列程序, 观察其执行结果, 并思考为什么? 若把最后一个语句(++x,y++)外的括号去掉, 程序还要做何修改?

```
/*syfl2_2.c*/
#include <stdio.h>
main( )
{
    int y=4,x=6,z=2;
    printf("%d %d %d\n", ++y, --x, z++);
    printf("%d %d\n", (++x, y++), z+2);
}
```

## 2.3 实验

1. 调试上面的实验范例。

2. 根据分析提示, 试着完成下面的程序, 并上机调试成功。

编程: 编写一个程序从键盘输入圆柱体的半径  $r$  和高度  $h$ , 计算其底面积和体积。

分析: 已知半径  $r$  和高度  $h$ , 依据圆面积的计算公式  $S = \pi * r * r$  和圆柱体体积计算公式  $V = \pi * r * r * h$ , 可计算其底面积  $S$  和体积  $V$ 。

不完整程序如下, 应先在下划线位置填写正确的参数或表达式, 再运行该程序:

```
/*sy2_1.c*/
#include<stdio.h>
main( )
{
    float pi=3.1415926;
    float r,h,S,V;
    printf("Please input r,h:");
    scanf("%f, _____", &r, _____); /*键盘输入圆半径 r 和高度 h*/
    S=_____ /*计算圆面积 S 的值*/
    V=_____ /*计算圆柱体体积 V 的值*/
    printf("底面积= _____ \t 圆柱体积= _____ \n", S, V);
}
```

3. 写出实验报告, 实验报告要求如下。

- (1) 将上面不完整程序补充完整, 并要保证其正确性。
- (2) 记录源程序在上机调试时出现的各种问题及其解决办法。
- (3) 总结本次实验的经验与教训。

# 实验 3 分支结构

## 3.1 实验目的

1. 掌握 C 语言逻辑量的表示方法 (以 0 代表“假”, 1 代表“真”)。学会正确地使用关系表

达式和逻辑表达式。

2. 掌握用 if 语句实现选择结构。
3. 掌握用 switch 语句实现多分支选择结构。
4. 掌握选择结构的嵌套。

## 3.2 实验范例

**【范例 1】** 从键盘输入一年份，判断年份是否为闰年。

```
/*syfl3_1.c*/
#include <stdio.h>
main( )
{
    int year;
    scanf("%d", &year);           /*键盘输入年份值*/
    if (year%4==0&&year%100!=0 || year%400==0)
        printf("This year is a leap year!\n");    /*如果是则输出是闰年*/
    else
        printf("This year is not a leap year!"); /*否则输出不是闰年*/
}
```

**【范例 2】** 猜数游戏。假如设定一个整数  $m=123$ ，然后让其他人从键盘上猜该数字，如果猜对，输出"RIGHT",如果猜错，则输出"WRONG",并指出设定的数比输入的数大还是小。

```
/*syfl3_2.c*/
#include <stdio.h>
main( )
{
    int data;
    printf("Input a data : ");      /*显示输入提示信息*/
    scanf("%d", &data);             /*键盘输入一个整数*/
    if(data==123)                   /*输入数据与 123 比较*/
        printf("RIGHT\n");          /*输入数据等于 123 则输出"RIGHT"*/
    else
    {
        printf("WRONG\n");          /*输入数据不等于 123 则输出"WRONG"*/
        if(data>123)
            printf("It is LARGE\n"); /*输入数据大于 123 则输出"It is LARGE"*/
        else
            printf("It is SMALL\n"); /*输入数据小于 123 则输出"It is SMALL"*/
    }
}
```

**【范例 3】** 编写程序，给出一个不多于 4 位的正整数，要求：

- (1) 求出它是几位数；
- (2) 分别输出每一位数字；
- (3) 按逆序输出每一位数字。

```
/*syfl3_3.c*/
#include <stdio.h>
main( )
{
    int num, indiv, ten, hundred, thousand, digit;
    printf("Input a integer number(0~9999): ");
    scanf("%d", &num);
    thousand=num/1000;
    hundred=num/100%10;
    ten=num%100/10;
    indiv=num%10;
    if(num>999)
    {
        digit=4;
        printf("Digit=%d\n", digit);
        printf("Each digit is: ");
        printf("%d,%d,%d,%d\n", thousand, hundred, ten, indiv);
        printf("Inversed nunmber is: ");
        printf("%d,%d,%d,%d\n", indiv, ten, hundred, thousand);
    }
    else
        if(num>99)
        {
            digit=3;
            printf("Digit=%d\n", digit);
            printf("Each digit is : %d,%d,%d\n", hundred, ten, indiv);
            printf("Inversed nunmber is: ");
            printf("%d,%d,%d\n", indiv, ten, hundred);
        }
        else
            if(num>9)
            {
                digit=2;
                printf("Digit=%d\n", digit);
                printf("Each digit is: %d,%d\n", ten, indiv);
                printf("Inversed nunmber is: ");
                printf("%d,%d\n", indiv, ten);
            }
            else
            {
                digit=1;
                printf("Digit=%d\n", digit);
                printf("Each digit is: %d\n", indiv);
                printf("Inversed nunmber is: %d\n", indiv);
            }
    }
```

### 3.3 实验

编写程序并上机调试通过，然后写出实验报告。

1. 从键盘输入一个字符，判断它是字母、数字还是其他字符。请先分析下面的程序，运行该

程序，分析运行结果。

```
/*sy3_1.c*/
#include <stdio.h>
main( )
{
    char c;
    printf("Enter a character: ");
    scanf("%c",&c);
    if((c>='a'&&c<='z')||(c>='A'&&c<='Z'))
        printf("It's an alphabetic character.\n");
    else
        if(c>=48&&c<=57)
            printf("It's a digit\n");
        else
            printf("It's an other character\n");
}
```

2. 写出以下程序运行的结果。

```
/*sy3_2.c*/
#include <stdio.h>
main( )
{
    int a=-1, b=1;
    if((++a<0)&&!(b--<=0))
        printf("a=%d,b=%d\n",a,b);
    else
        printf("b=%d,a=%d\n",b,a);
}
```

3. 写出以下程序运行的结果。

```
/*sy3_3.c*/
#include <stdio.h>
main( )
{
    int a=0,b=1;
    switch(a)
    {
        case 0: switch(b)
                {
                    case 0: a++; b++; break;
                    case 1: a++; b++;
                    default: a++;
                }
        case 1: a++; b++;
    }
    printf("a=%d,b=%d\n",a,b);
}
```

4. 编写程序。实现根据用户输入的三角形的三条边长判定是何种三角形。
5. 从键盘输入两个操作数和运算符，用 switch 语句实现两个数的加、减、乘、除运算。
6. 写出实验报告，实验报告要求如下。

- (1) 问题分析：写出解决问题的算法思路；画出程序流程图。
- (2) 源程序：根据算法思想或程序流程图编写源程序。
- (3) 调试记录：记录源程序在上机调试时出现的各种问题及其解决办法。
- (4) 总结：总结本次实验的经验与教训。

## 实验 4 循环结构

### 4.1 实验目的

1. 掌握 for 循环结构的灵活运用。
2. 掌握 while 和 do-while 循环结构的灵活运用。

### 4.2 实验范例

**[范例 1]** 试编程序计算  $s = 1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4} + \cdots + \frac{1}{99} - \frac{1}{100}$ 。

```
/*syfl4_1.c*/
#include <stdio.h>
main( )
{
    int n, flag=1;
    float s=0;
    for(n=1; n<=100; n++)
    {
        s=s+1.0/n*flag;
        flag=-flag;
    }
    printf("%.2f\n", s);
}
```

**[范例 2]** 把 411 分成两个数的和，并使其中一个加数能被 13 整除，而另一能被 17 整除，试编程序求这两个加数。

```
/*syfl4_2.c*/
#include <stdio.h>
main( )
{
    int a, b;
    for(a=13; a<411/2; a=a+13)
    {
        b=411-a;
        if(b%17==0) printf("%d=%d+%d\n", 411, a, b);
    }
}
```

**[范例 3]** 已知方程  $x + 3\cos x - 1 = 0$  在  $[-2, 5]$  中有一根，精度要求  $10^{-5}$ ，试用二分法求之。  
算法提示：

- (1) 输入有根区间两端点  $x_0$ 、 $x_1$  和精度。
- (2) 计算  $x=(x_1+x_0)/2$ 。
- (3) 若  $f(x_1)*f(x)<0$ ，则  $x_0=x$ ，转②，否则  $x_1=x$ ，转②。
- (4) 若  $|x_1-x_0|<$ 精度，则输出根  $x$ ，结束。否则转②。

程序如下：

```
/*syfl4_3.c*/
#include <stdio.h>
#include <math.h>
main( )
{
    float x0,x1,x,f1,f;
    x0=-2;
    x1=5;
    do
    {
        x=(x0+x1)/2;
        f1=x1+3*cos(x1)-1;
        f=x+3*cos(x)-1;
        if(f1*f<0)
            x0=x;
        else
            x1=x;
    }while(fabs(x0-x1)<1e-5);
    printf("The equation's root is %f\n", x);
}
```

## 4.3 实验

分析或编写程序并上机调试通过，然后写出实验报告。

1. 写出下面程序运行的结果。

```
/*sy4_1.c*/
#include <stdio.h>
main( )
{
    int a,n,count=1;
    long int sn=0,tn=0;
    scanf("%d,%d",&a,&n);
    while(count<=n)
    {
        tn+=a;
        sn+=tn;
        a*=10;
        count++;
    }
    printf("sn=%ld\n",sn);
}
```

2. 写出以下程序的输出结果。

```
/*sy4_2.c*/
```

```

#include <stdio.h>
main( )
{
    int i,j,k=0,m=0;
    for(i=0;i<2;i++)
    {
        for(j=0;j<3;j++)    k++;
        k-=j;
    }
    m=i+j;
    printf("k=%d,m=%d",k,m);
}

```

3. 以下程序的功能是：从键盘上输入若干学生的成绩，统计并输出最高成绩和最低成绩，当输入负数时结束输入。请先将正确的语句或表达式填入下划线处，再运行。

```

/*sy4_3.c*/
#include <stdio.h>
main( )
{
    float x,max,min;
    scanf("%f",&x);
    max=x;
    min=x;
    while(_____)
    {
        if(x>max)
            max=x;
        if(x<min)
            _____;
        scanf("%f",&x);
    }
    printf("Max=%f,Min=%f\n",max,min);
}

```

4. 求水仙花数。水仙花数是一个 3 位正整数，其值等于其各个数位的立方之和。
5. 百马百担问题。有 100 匹马，驮 100 担货，大马驮 3 担，中马驮 2 担，两匹小马 1 担，编程计算共有多少种驮法？
6. 求  $w=1+2^1+2^2+2^3+\cdots+2^{10}$ 。
7. 求下列数列的前 20 项： $f(0)=0, f(1)=1, f(n)=f(n-1)+f(n-2) (n>1)$ 。
8. 由 3 位不同数字构成的 3 位十进制整数  $abc$  ( $a$  非 0，且  $a、b、c$  互不相等)，若能被  $(a+b+c)^2$  除尽，则称  $abc$  为三味数，如 405 就是三味数。问：最小的三味数是什么？ $a、b、c$  均为偶数的三味数是什么？
9. 写出实验报告，实验报告要求如下。
  - (1) 问题分析：写出解决问题的算法思路；画出程序流程图。
  - (2) 源程序：根据算法思想或程序流程图编写源程序。
  - (3) 调试记录：记录源程序在上机调试时出现的各种问题及其解决办法。
  - (4) 总结：总结本次实验的经验与教训。

## 实验 5 break/continue 语句

### 5.1 实验目的

1. 进一步掌握 while、do-while 和 for 语句实现循环的方法。
2. 掌握循环的嵌套结构及 continue 和 break 语句的合理运用。

### 5.2 实验范例

**【范例 1】** 将 4~100 中的偶数分解成两素数之和（每个数只需求出一种分解方法）。

```
/*syfl5_1.c*/
#include <stdio.h>
main( )
{
    int x,a,b,n;
    for(x=4;x<=100;x=x+2)
        for(a=2;a<=x/2;a++)
        {
            for(n=2;n<=a-1;n++)
                if(a%n==0) break;
            if(a==n)
            {
                b=x-a;
                for(n=2;n<=b-1;n++)
                    if(b%n==0) break;
                if(b==n)
                {
                    printf("%d=%d+%d\t",x,a,b);
                    break;
                }
            }
        }
}
```

**【范例 2】** 若  $n$  使  $2^n-1$  为素数，则  $n$  称为梅森尼数。求 [1, 21] 范围内：有多少个梅森尼数？最大的梅森尼数？

```
/*syfl5_2.c*/
#include <stdio.h>
main( )
{
    long int i,j,max,sum=0, s=2,p;
    for(i=2;i<=21;i++)
    {
        s=s*2;
        p=s-1;
        for(j=2;j<=p-1;j++)
            if(p%j==0) break;
    }
```