

实验 5 while、do-while 语句构成的循环

一、实验目的

- 掌握 while、do-while 两种语句的语法规则和执行步骤。
- 掌握 while、do-while 两种语句循环的控制条件。
- 掌握初值、循环条件和循环体语句顺序三者的制约关系。
- 掌握两种语句的异同之处。

二、调试内容

1. 程序改错(一)

用 while 语句求 $\sum_{i=1}^{100} i$ 。

```
#include "stdio.h"
main()
{   int i, sum;

                                     /* 该行需添加 1 条语句 */

    i=1;
        while(i<=100)
    {   sum=sum+i;

                                     /* 该行需添加 1 条语句 */

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

本程序没有语法错误,但是运行后会出现死循环,原因是变量 i 的值始终为 1。死循环的问题解决后,运行结果也是错误的,原因是第一次用到变量 sum 时,程序还没有给它赋值。将这两个问题都解决后才能得到正确结果。

2. 程序改错(二)

用 do-while 语句求 $\sum_{i=1}^{100} i$ 。

```
#include "stdio.h"
main()
{
    int i, sum;
```

```

i=1;
sum=1;                                /* 该行有 1 处错误 */
do
{
    sum=sum+i;
    i=i+1;
}
while(i>100);                          /* 该行有 1 处错误 */
printf("sum=%d\n", sum);
}

```

本程序没有语法错误,但是有 2 处逻辑错误,改正过来,方能得到正确的运行结果。

注意: while 后括号中表达式值为真则继续执行循环,值为假则循环结束。

3. 程序改错(三)

用 do-while 语句求 $n!$ 。

```

#include "stdio.h"
main()
{
    int n,i;
    double t;
    scanf("%d",&n);
    t=1;
    i=2;                                /* 该行有 1 处错误 */
    do
    {   t=t*i;
        i++;
    }
    while(i<=n)                          /* 该行有 1 处错误 */
    printf("%f\n",t);
}

```

本程序倒数第 3 行有 1 处语法错误,改正该错误后,运行时输入 1 或 0,即可发现第 7 行的逻辑错误——变量 i 的初值不能为 2。

分析讨论:

如果将本程序中 do-while 语句构成的循环改为由 while 语句来实现,程序的第 7 行还需要修改吗?为何会出现这样的不同?当型循环和直到型循环的根本区别是什么?

4. 程序改错(四)

下面程序的功能是从键盘输入若干学生的成绩,统计并输出最高成绩和最低成绩,当输入负数时循环结束。

```

#include "stdio.h"

```

```

main()
{
    float x,amax,amin;
    scanf("%f",&x);
    amax=amin=x;
    while(x>0)                                /* 该行有 1 处错误 */
    {
        if(x>amax) amax=x;
        if(x<amin) amin=x;

                                                /* 该行需添加 1 条语句 */
    }
    printf("max=%.2f,min=%.2f\n",amax,amin);
}

```

本程序有 2 处逻辑错误：一是会出现死循环,需要添加 1 条语句来解决此问题;二是当有学生成绩为 0 时运行结果不正确,需要修改第 6 行。

分析讨论:

(1) 如果学生成绩数据为

96 78.5 83 0 76.2 65 77

应如何输入数据?

(2) 程序的第 5 行是为最大值和最小值赋初值,该行是否可以去掉?

5. 程序改错(五)

求 10 个数中的最大值。

```

#include "stdio.h"
main()
{
    int a,max,i;
    scanf("%d",&a);
    max=a;
    for(i=1;i<=10;i++)                        /* 该行有 1 处错误 */
    {
        if(a>max) max=a; scanf("%d",&a);      /* 该行有 1 处错误 */
    }
    printf("最大值为%d\n",max);
}

```

本程序有 2 处逻辑错误：一是输入 11 个数才能出现运行结果,因此需要修改程序的第 6 行;二是当最后一个数是最大值时,结果不正确,必须改变 for 语句的循环体内语句的顺序。

测试数据:

① 9 6 7 8 8 3 10 2 5 1

② 6 7 1 0 2 3 9 8 7 16

6. 程序改错(六)

求以下表达式的近似值,要求误差小于 10^{-4} 。

$$\frac{1}{1 \times 2} + \frac{1}{2 \times 3} + \cdots + \frac{1}{n \times (n+1)} + \cdots$$

```
#include "stdio.h"
main()
{
    float sum=0,t;
    int i=1;                                /* 该行有 1 处错误 */
    do
    {
        i++;
        t=1/(i*(i+1));                    /* 该行有 1 处错误 */
        sum=sum+t;
    }while(t<=1e-4);                      /* 该行有 1 处错误 */
    printf("sum=%f\n",sum);
}
```

本程序没有语法错误,但是有 3 处逻辑错误:一是 int 型变量 i 的初值需要修改(不要改变 i 的类型);二是整型量相除的问题;三是循环的条件有误。

修改正确后运行结果为

```
sum=0.990099
```

分析讨论:

本程序修改正确后,被加到 sum 中的最后一项 t 的值一定小于 10^{-4} 。如果要求加到 sum 中的每一项 t 的值都需大于或等于 10^{-4} ,则应如何修改程序?

7. 程序改错(七)

根据公式 $\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots$,求 π 的近似值。求公式等号右侧表达式的近似值时,精度要求为 10^{-6} 。

```
#include "stdio.h"
#include "math.h"
main()
{
    int k,i;
    double s,t;
    s=0;
    k=1;
```

```

i=1;
t=1;
while(t>=1e-6)                /* 该行有 1 处错误 */
{
    s=s+t;
    k=-k;
    i=i+2;
    t=k/i;                      /* 该行有 1 处错误 */
}
printf("%lf\n", 4 * s);
}

```

本程序没有语法错误,但是有 2 处逻辑错误:一是循环的条件有误,据此条件,当 t 值为第 2 项 $-1/3$ 时,条件就不满足,循环结束,此时 s 值为 1;二是整型量相除的问题,修改方法有多种,可以改为 $t=1.0 * k/i$;或 $t=(0.0+k)/i$;或 $t=(float)k/i$ 。总之,参与除法运算的两个量中,至少有一个为浮点型即可。

修改正确后运行结果为

3.141591

8. 程序改错(八)

用辗转相除法求两个整数的最大公约数。

```

#include "stdio.h"
main()
{
    int m,n,r;
    scanf("%d%d", &m, &n);
    do
    {
        r=m/n;                /* 该行有 1 处错误 */
        m=n;
        n=r;
    }
    while(r=0);               /* 该行有 1 处错误 */
    printf("gcd=%d\n", n);    /* 该行有 1 处错误 */
}

```

本程序没有语法错误,但是有 3 处逻辑错误,请调试改正过来。

9. 运行程序

```

#include "stdio.h"
main()
{
    int n=12345,d;

```

```

while(n!=0)
{
    d=n%10;
    printf("%d\n",d);
    n=n/10;
}
}

```

分析讨论:

- (1) 该程序完成的功能是什么?
- (2) 增加适当的变量和语句,求任意一个正整数各位数字之和。
- (3) 如何求任意一个正整数的位数?

三、编程题

- (1) 求出 1~100 中的奇数之和、偶数之积。请注意所用变量的类型。

答案: 2500, 3.424322e+079。

- (2) 计算以下表达式的值。

$$\frac{1}{1 \times 2} + \frac{1}{2 \times 3} + \frac{1}{3 \times 4} + \cdots + \frac{1}{20 \times 20}$$

答案: 0.952381。

- (3) 分别求出一批非零整数中奇数、偶数的平均值,用 0 作为终止标记。
- (4) 输入一批正整数,用 -1 作终止标记。求其中能被 3 或 7 整除的数之和。
- (5) 计算 $S_n = a + aa + aaa + \cdots + \underbrace{aa \cdots a}_{n \uparrow}$ 的值,其中 a 是一个数字, n 表示个数。例如,

当 a 是 3, n 是 4 时,表达式是 $3 + 33 + 333 + 3333$ 。

- (6) 根据以下表达式计算 π 的近似值。求公式右侧表达式的近似值时,要求误差不大于 10^{-5} 。

$$\frac{\pi}{2} \approx 1 + \frac{1}{3} + \frac{1}{3} \times \frac{2}{5} + \frac{1}{3} \times \frac{2}{5} \times \frac{3}{7} + \cdots + \frac{1}{3} \times \frac{2}{5} \times \frac{3}{7} \times \cdots \times \frac{i}{2i+1} + \cdots$$

答案: 3.141566。

- (7) 某单位排队形,开始排成 3 路纵队,末尾多出了 2 人;后来改成 5 路纵队,末尾又多出了 3 人;最后改成 7 路纵队,正好没有余数。编程序求出该单位至少有多少人。

答案: 98。

- (8) 计算 $n!$, 求出小于 32000 的最大的阶乘值,并求出此时 n 的值。

答案: 5040, 7。