

C语言程序设计

第五章 循环结构程序设计



问题提出

```
int result1,result2,result3;  
int result4,result5;  
result1 = 1 * 10;  
printf("1 × 10 = %d \n",result1);  
result2 = 2 * 10;  
printf("2 × 10 = %d \n",result2);  
result3 = 3 * 10;  
printf("3 × 10 = %d \n",result3);  
result4 = 4 * 10;  
printf("4 × 10 = %d \n",result4);  
result5 = 5 * 10;  
printf("5 × 10 = %d \n",result5);
```

输出结果

```
1 × 10 = 10  
2 × 10 = 20  
3 × 10 = 30  
4 × 10 = 40  
5 × 10 = 50
```

重复语句



$0 + 1 \longrightarrow \textcircled{1} \times 10 = 10$
 $1 + 1 \longrightarrow \textcircled{2} \times 10 = 20$
 $2 + 1 \longrightarrow \textcircled{3} \times 10 = 30$

上个数字 + 1

 $\longrightarrow \textcircled{4} \times 10 = 40$
... $5 \times 10 = 50$

有好的办法吗？

循环



01 while语句及应用

02 do-while语句及应用

03 for语句及应用

04 循环结构程序设计



01

while语句及其应用



while语句

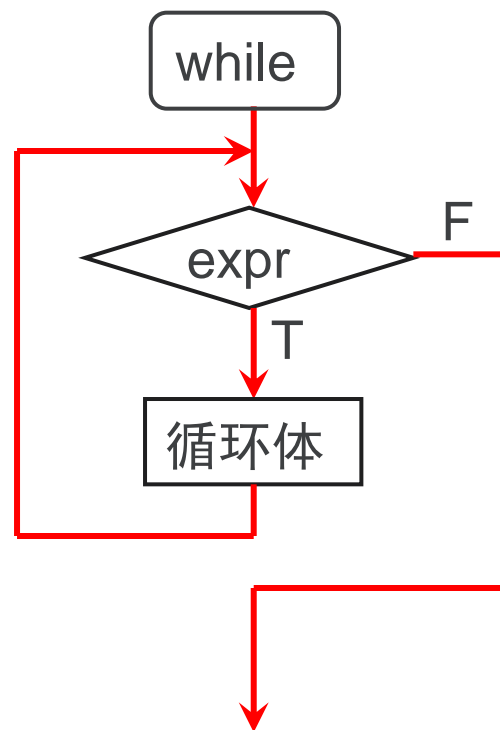
➤ 语法规则

```
while (条件)  
{ 循环体语句;  
  .....; }
```

➤ 说明

条件结果：非零时，循环条件成立，执行循环体的所有语句；
否则，循环结束，执行循环体之后的语句。

➤ 执行流程



while应用

➤ 例

输出公元1000-2000年间的所有闰年

分析：怎么判断闰年？

循环条件的初始值和终止值

方法

有两种情况都是闰年

- 能被4整除，但不能被100整除；
- 能被400整除；

条件用C语言描述为：

$(year \% 4 == 0 \&\& year \% 100 != 0)$
 $\parallel (year \% 400 == 0)$

➤ 实现

```
#include<stdio.h>
void main ()
{
    int y=1000; 循环初值
    while(y<=2000) 循环终值
    { if ( (y%400==0)|| (y%4==0&& y%100!=0))
        printf("%4d",y);
        y++; 循环变量更新
    }
}
```

02

do-while语句及其应用

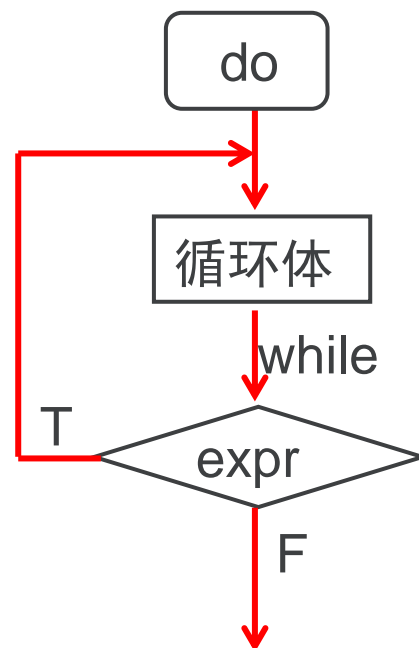


do-while语句

➤ 语法规则

```
do  
{ 循环体语句;  
    .....; }  
while (条件);
```

➤ 执行流程



➤ 说明

while: 先判断, 后执行。初始条件不成立, 循环体不执行

do-while: 先执行, 后判断。初始条件不成立时, 循环体已执行1次



do-while应用

➤ 同例

输出公元1000-2000年间的所有闰年

分析：

方法不变，套用do-while语法

方法

有两种情况都是闰年

- 能被4整除，但不能被100整除；
- 能被400整除；

条件用C语言描述为：

$(year \% 4 == 0 \&\& year \% 100 != 0)$
 $\parallel (year \% 400 == 0)$

➤ 实现

```
#include<stdio.h>
void main ()
{
    int y=1000;
    do
    { if ( (y%400==0)|| (y%4==0&& y%100!=0))
        printf("%4d",y);
        y++;
    } while(y<=2000); 分号不能少
}
```

03

for语句及应用



for语句

➤ 语法规则

```
for(表达式1; 表达式2; 表达式3)  
{ 循环体语句;  
  .....; }
```

➤ 说明

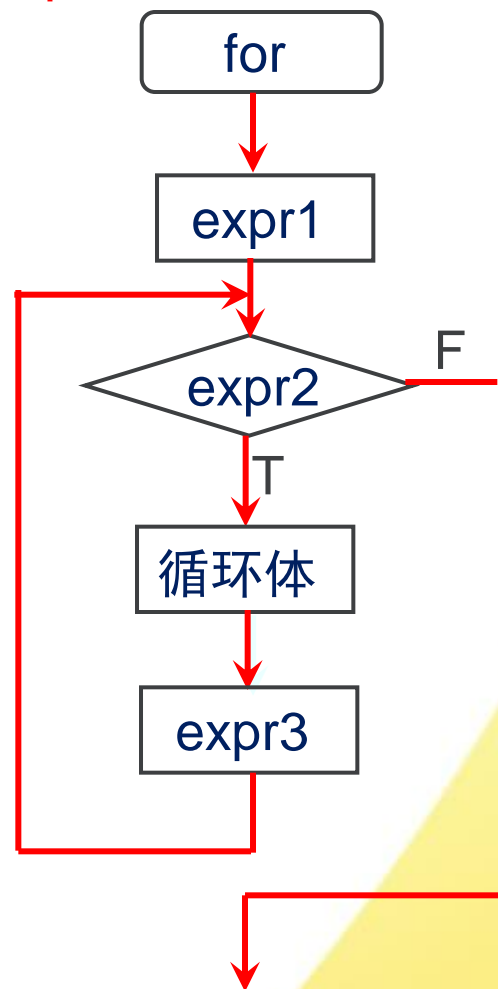
表达式1: 给循环变量赋初值。

表达式2: 循环控制条件。

表达式3: 循环变量增或减。

表达式之间用**分号**分隔, 且不能缺省。

➤ 执行流程



for语句应用

➤ 同例

输出公元1000-2000年间的所有闰年

分析

方法不变，用 for 语法格式

注意

循环三要素

- 循环初值
- 循环条件及终值
- 循环变量更新

➤ 实现

```
#include<stdio.h>
```

```
void main ()
```

```
{
```

```
int y;
```

```
for (y=1000;y<=2000;y++)
```

```
{ if ( (y%400==0)|| (y%4==0&& y%100!=0))
```

```
    printf("%4d",y);
```

```
}
```

```
}
```

循环三要素
全部在for语句中体现

多重循环

```
while(  
{ .....  
  do  
  { .....  
    }while( );  
  .....  
}
```

```
for( ;;) 外循环  
{ .....  
  do  
  { .....  
    }while(); 内循环  
  .....  
  while()  
  { .....  
    } 内循环  
  .....  
}
```

注意

- 三种循环可互相嵌套,层数不限

break与continue

```
#include <stdio.h>
void main ( )
{
    int i, a, sum = 0;
    for (i = 1; i <= 5; i++)
    {
        scanf ("%d", &a);
        if (a <= 0) break;
        sum += a ;
    }
    printf ("和是: %d\n", sum);
}
```

运行结果：和是： 1

循环输入5个数
1 -2 3 4 5

```
#include <stdio.h>
void main ( )
{
    int i, a, sum = 0;
    for (i = 1; i <= 5; i++)
    {
        scanf ("%d", &a);
        if (a <= 0) continue;
        sum += a ;
    }
    printf ("和是: %d\n", sum);
}
```

运行结果：和是： 13

break：能终止并跳出最近一层循环；
continue：结束当次循环，跳过循环体中尚未执行的语句，进行下一次是否执行循环体的判断。

04

循环结构程序设计



循环结构程序设计

➤ 例

编写程序，输出所有水仙花数

➤ 分析

1 水仙花数：

该数每位数字的立方和等于该数本身

如： $153=1^3+5^3+3^3$

2 循环初值 num=100

3 循环条件 num<1000

➤ 实现

```
#include<stdio.h>
void main()
{ int num=100,a,b,c;
  while( num<1000)
  { a=num/100;
    b=num/10%10;
    c=num%10;
    if ( num==a*a*a+b*b*b+c*c*c)
      printf("%6d",num);
      num++;
    }
}
```



循环结构程序设计

➤ 例 输入任意两个正整数，求他们的最大公约数和最小公倍数

➤ 分析

1 最大公约数 算法

- (1) 输入两个正整数m和n。
- (2) 用m除以n，余数为r，如果r等于0，则n是最大公因子，算法结束，否则 执行(3)。
- (3) 把n赋给m，把r赋给n，转(2)

2 最小公倍数

=两数乘积 ÷ 最大公约数

➤ 实现

```
#include <stdio.h>
void main ( )
{ int m, n, r, p;
  scanf ("%d%d", &m, &n);
  p=m*n;
  while (n != 0)
  { r = m % n; //求余数
    m = n;
    n = r; }
  printf ("最大公约数%d\n", m);
  printf ("最小公倍数%d\n", p/m);
}
```



循环结构程序设计

➤ 例 输出如图所示：九九乘法表

```
1×1=1
1×2=2  2×2=4
1×3=3  2×3=6  3×3=9
1×4=4  2×4=8  3×4=12  4×4=16
1×5=5  2×5=10  3×5=15  4×5=20  5×5=25
1×6=6  2×6=12  3×6=18  4×6=24  5×6=30  6×6=36
1×7=7  2×7=14  3×7=21  4×7=28  5×7=35  6×7=42  7×7=49
1×8=8  2×8=16  3×8=24  4×8=32  5×8=40  6×8=48  7×8=56  8×8=64
1×9=9  2×9=18  3×9=27  4×9=36  5×9=45  6×9=54  7×9=63  8×9=72  9×9=81
```

➤ 难点

需要双重循环，分清内外循环

➤ 分析

1 外循环

是每一行，i 取值1~9，共循环9次

2 内循环

是每一个算式，j 总是 < i

且每个算式的表述为 j*i

➤ 实现

```
#include<stdio.h>
void main()
{int i,j;
for(i=1;i<=9;i++)
{
for(j=1;j<=i;j++)
printf("%d*%d=%d\t",j,i,i*j);
printf("\n");
}
}
```



THANKYOU

