

# 第4章 过程控制语句

# 目录

- ▶ 4.1 while语句
- ▶ 4.2 do...while语句
- ▶ 4.3 for语句
- ▶ 4.4 switch语句
- ▶ 4.5 转向语句
- ▶ 4.6 过程应用: 求 $\pi$
- ▶ 4.7 过程应用: 判明素数
- ▶ 4.8 过程应用: 求积分

# 本章目标

- ▶ 学会使用 *while* 循环语句
- ▶ 学会使用 *do...while* 循环语句
- ▶ 学会使用 *for* 循环语句
- ▶ 学会使用 *switch* 多重选择语句
- ▶ 学会使用 *break* 和 *continue* 语句

# 4.1 while 语句

- ▶ 循环语句使程序员能够指定当某个条件一直为真时重复执行某个动作。

- ▶ 例如伪代码：

While （购物单上还有其他商品）

{

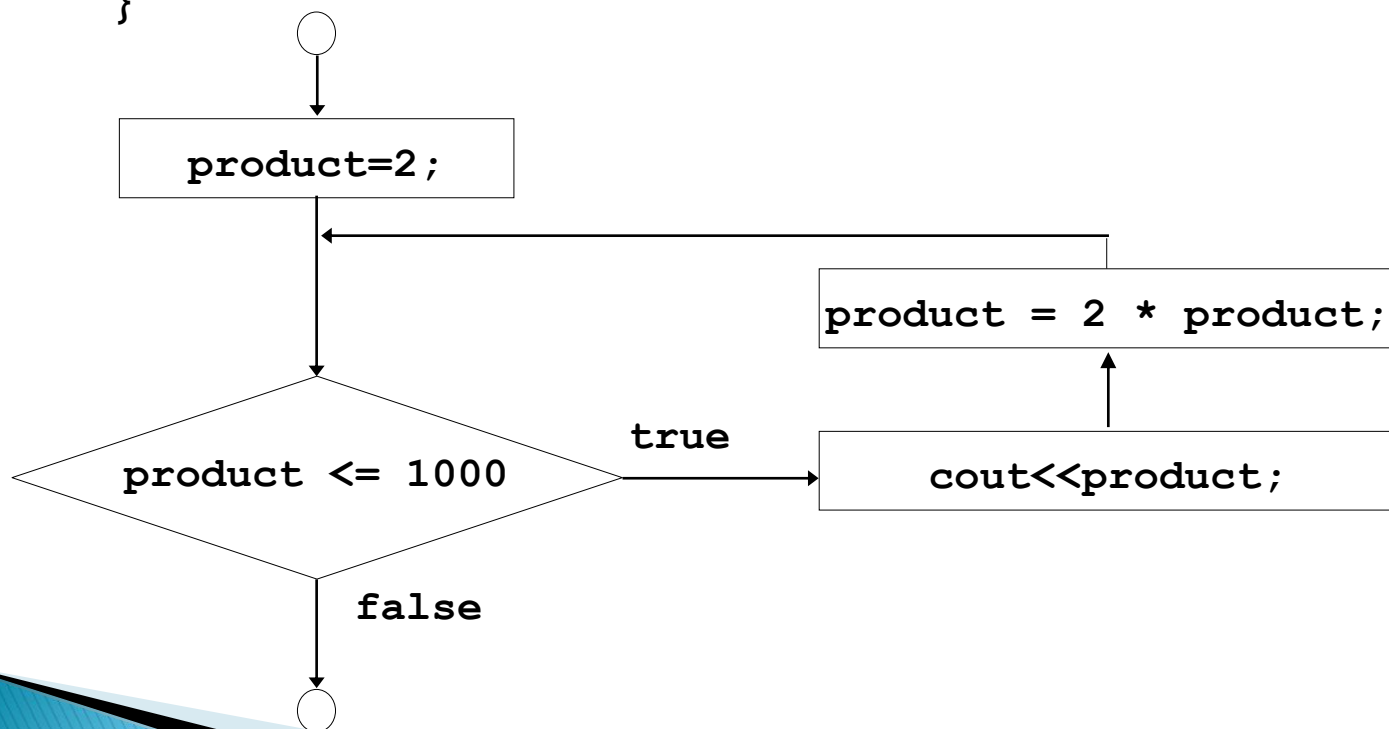
    买下一个商品

    把它从购物单上划掉

}

# 例如

```
int product = 2;  
while ( product <= 1000 )  // product累积到超1000停止  
{  
    cout<<product;  
    product = 2 * product;  // product值不断翻倍  
}
```



## 4.2 do...while 语句

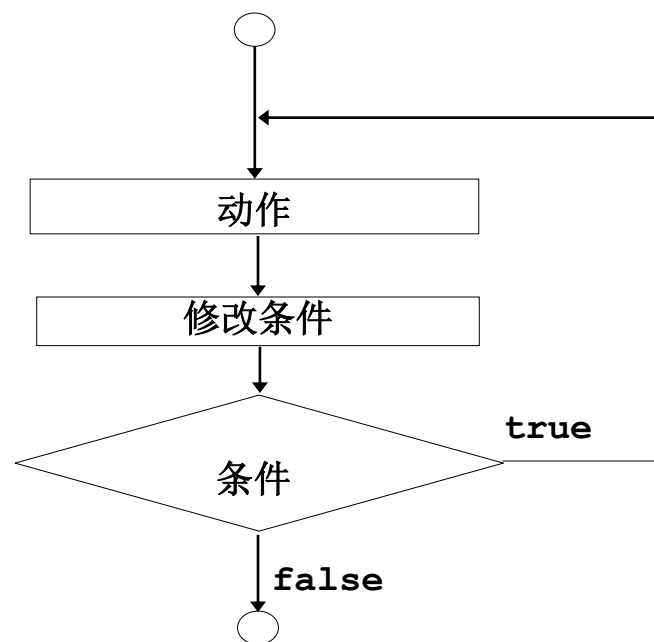
### ▶ 格式

```
do {  
    statement;  
    modify condition;  
} while ( condition );
```

### ▶ 例子 (令counter=1)

```
do {  
    cout<<counter;  
    ++counter;  
} while (counter<=10);
```

### ▶ 流程图

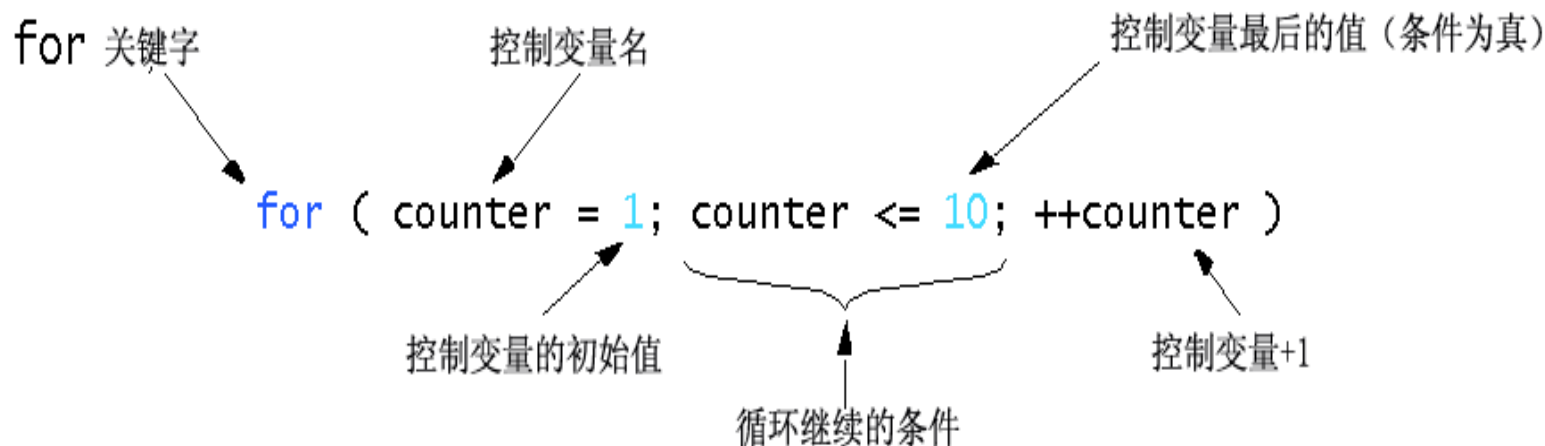


## 4.3 for语句

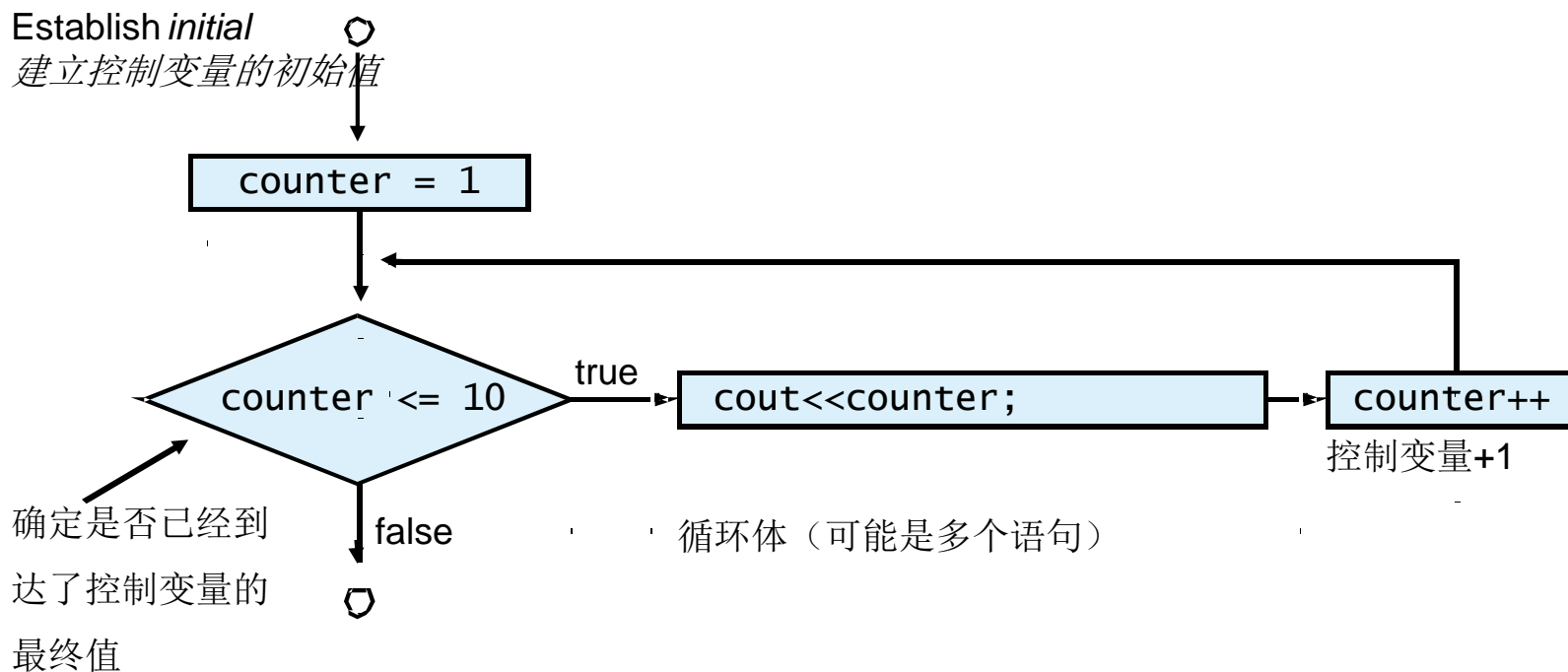
### ▶ for语句语法

for ( 初始化; 循环控制; 增量 )  
循环语句块;

### ▶ 例子



# 循环语句-for流程





# 循环语句-for语句头部描述示例

- ▶ `for(int i=1; i<=100; i++)` //1~100,1为间隔,100次
- ▶ `for(int i=100; i>=1; i--)` //100~1,-1为间隔,100次
- ▶ `for(int i=7; i<=77; i+=7)` //7~77,7为间隔,11次
- ▶ `for(int i=20; i>=2; i-=2)` //20~2,-2为间隔,10次
- ▶ `for(int i=99; i>=0; i-=3)` //99~0,-3为间隔,34次

## for语句的说明

```
int a=1;
for(; a<10; a++)
    cout<<setw(5)<<a;
```

```
int a=1;
for(; a<10; )
{
    cout<<setw(5)<<a;
    a++;
}
```

```
int a=1;
for(; ; )
{
    cout<<setw(5)<<a;
    a++;
    if (a>10)
        break;
}
```

# 循环的说明

- ▶ 一个正确的循环语句结构
  - 初始化:只在循环开始时执行一次
  - 循环条件:条件成立，则继续执行；否则将退出循环
  - 循环体:包含循环执行的动作和**循环控制变量的修改**

```
#include <iostream>
using namespace std;
int main()
{
    short a = -2, b = 1;
    while (b > 0)
    {
        b++;
        a--;
    }
    cout << "a=" << a
    << "\nb=" << b << endl;
    return 0;
}
```

- 阅读左边程序代码，判断该程序是否是死循环，根据自己的判断投票

- ☐ A 死循环
- ☐ B 不是循环

提交

## 4.4 switch语句

### ▶ switch

有时，一个算法将包含一系列判断，在这些判断过程中，一个变量或表达式会分别针对其可能会取得每个**常数整型**进行测试，并采取不同的动作。

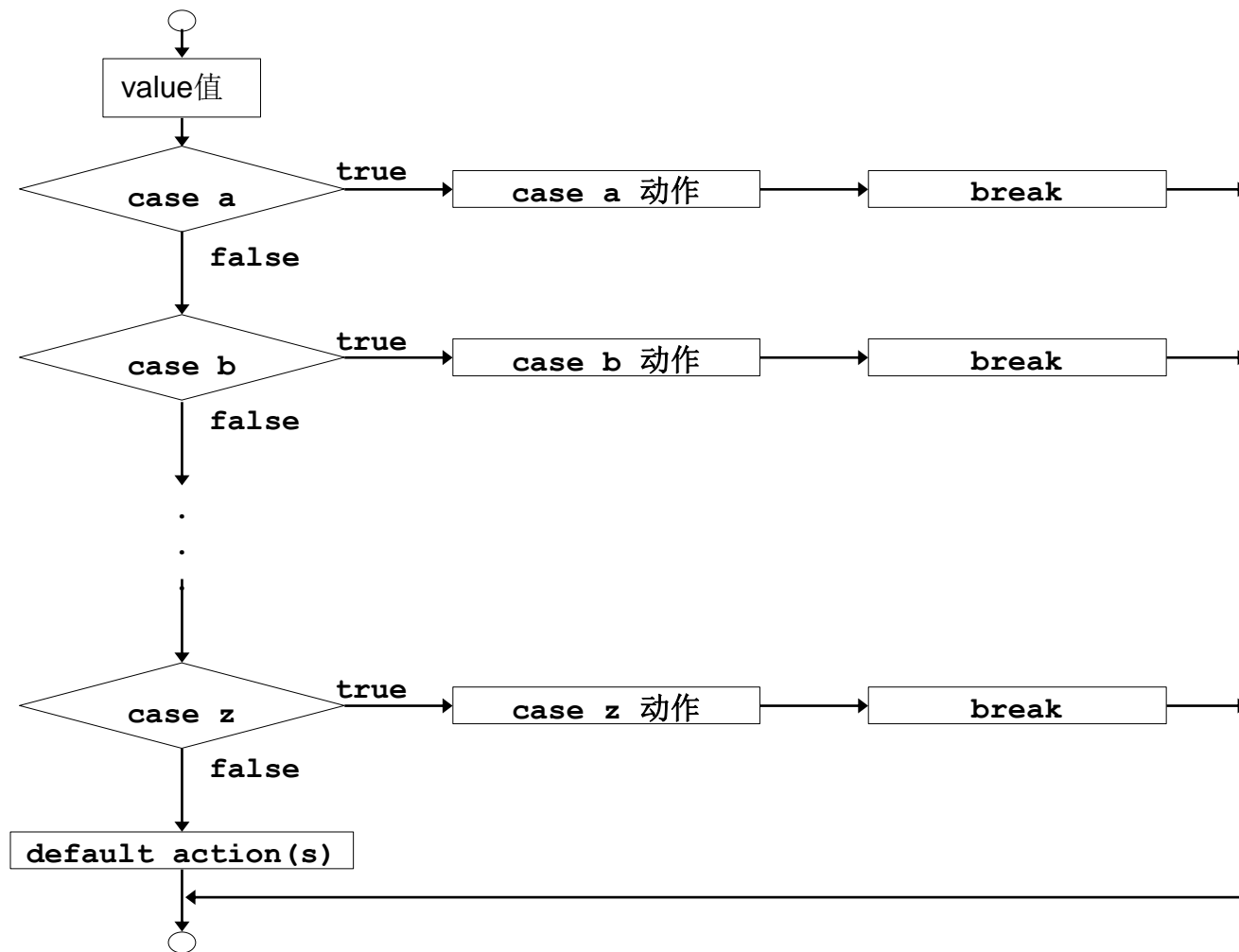
### ▶ 格式

- 一系列case标签，default可选

```
switch ( value ){  
    case '1':  
        actions  
    case '2':  
        actions  
    default:  
        actions  
}
```

- break; 从语句中退出

## Switch语句流程图



# switch语句的说明

## 说明:

- E1,E2,...En是常量表达式,且值必须互不相同
- 每个case层必须用break跳出, 否则就接着执行下一个case层
- case后可包含多个可执行语句, 且不必加{ }
- switch可嵌套
- 多个case可共用一组执行语句

```
如: .....  
    case 'A' :  
    case 'B' :  
        cout<<"score>70\n";  
    case 'C' :  
        cout<<"score>60\n";  
        break;  
    .....
```

## 4.5 转向语句

```
#include<iostream>
using namespace std;
int main()
{
    int x;
    for(x=1; x<=10; x++)
    {
        if(x==5)
        {
            break;    // 跳出for循环
        }
        cout<<x<<" ";
    }
    cout<<"\nBroke out of loop at x = "<<x<<"\n";
}
```

运行结果:

1 2 3 4

Broke out of loop at x = 5

# break和continue语句

```
#include<iostream>
using namespace std;
int main()
{
    int x;
    for(x=1; x<=10; x++)
    {
        if(x==5)
            continue;
        cout<<x<<" ";
    }
    cout<<"\nUsed continue to skip printing the value 5\n";
}
```

运行结果:

1 2 3 4 6 7 8 9 10

Used continue to skip printing the value 5



## 4.6 过程应用——求 $\pi$

- ▶ 利用公式： $\frac{\pi}{4} \approx 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots$ 求 $\pi$ 的近似值，直到最后一项的绝对值小于 $10^{-8}$ 为止
- ▶ 分析
  - 公式的第 $n$ 项表示方法： $(-1)^{n-1} * \frac{1}{2n-1}$
  - 循环结束条件的表示方法： $S_n < 1e-8$

## 4.7 过程应用——判明素数

- ▶ 从键盘输入一个整数 $m$ ，判断其是否为素数
- ▶ 判断素数的条件
  - $m$ 能被 $2, 3, \dots, m-1$ 整除
- ▶ 判断素数的优化条件
  - $m$ 不能被 $2, 3, \dots, \sqrt{m}$ 整除
- ▶ 扩展
  - 从键盘输入一个整数 $m$ ，找出 $[0, m]$ 区间的所有素数，并输出到屏幕上，要求每行输出5个整数，每个整数占5列，右对齐。

## 4.7 过程应用——求积分

### ▶ 积分公式

- $I = \int_a^b \frac{\exp(x)}{1+x^2} dx$
- 从键盘输入a,b的值，计算积分值。
- 课后完成。