

C/C++程序设计案例实战

——古代神奇妙算

华中农业大学信息学院 翟瑞芳

生活中的循环结构

- **While**语句
超市自助收银系统
- **do while**语句
我能成功登陆系统吗？
- **for**语句
神奇妙算
警察抓小偷



神奇妙算：for循环结构

今有物不知其数：
三三数余二，
五五数余三，
七七数余四，
问此物几何？

```
#include<iostream>
using namespace std;
int main()
{

return 0;
}
```

神奇妙算：for循环结构

今有物不知其数：
三三数余二，
五五数余三，
七七数余四，
问此物几何？

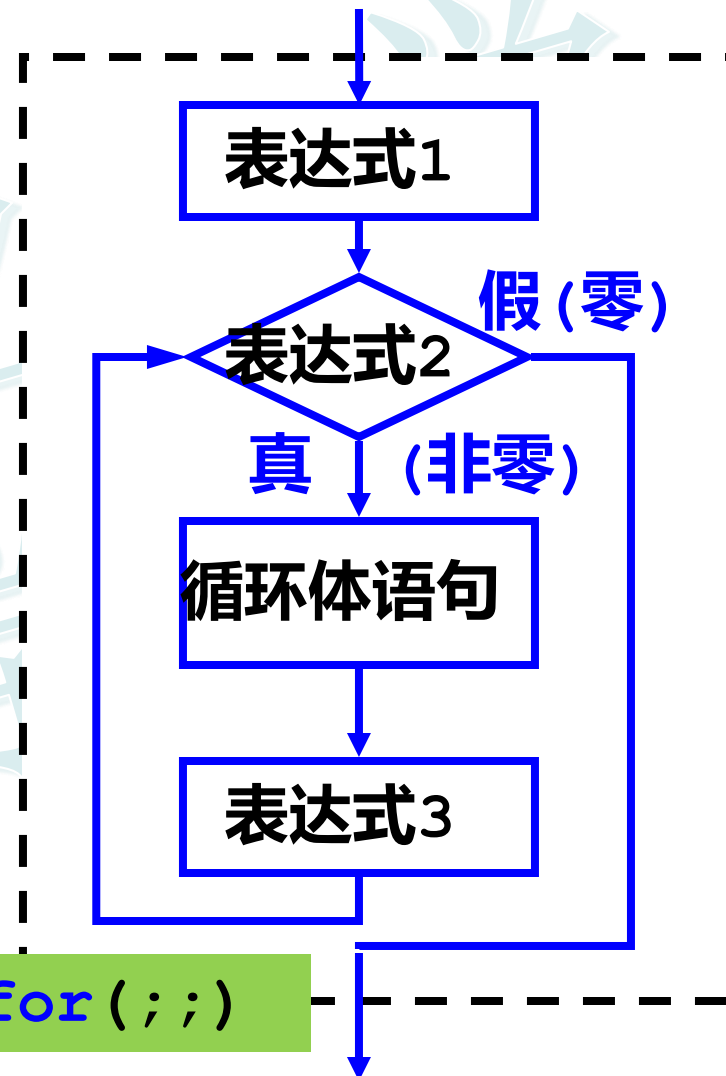
```
#include<iostream>
using namespace std;
int main()
{
    int num;
    for(num = 1; num <= 1000; num++)
    {
        if(num%3==2 && num%5==3 && num%7==4)
        {
            cout<<"num = "<<num<<endl;
        }
    }
    return 0;
}
```

神奇妙算：for循环结构

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

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

- 表达式1被且仅被执行一次;
- 3个表达式均可缺省, 缺省表达式2表示true, 但两个分号不能缺省; `for(;;)`

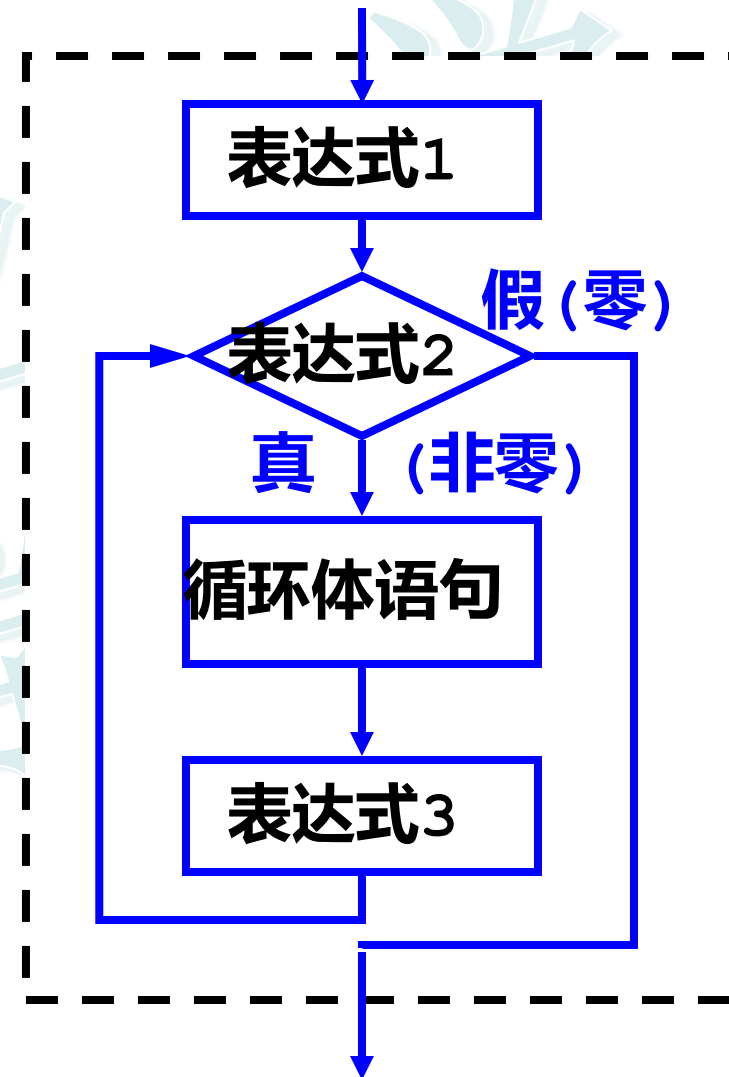


神奇妙算：for循环结构

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

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

- 表达式1被且仅被执行一次;
- 3个表达式均可缺省, 缺省表达式2表示true, 但两个分号不能省略;
- 可以将循环体语句与表达式3合并。



```
for (i=1; i<=1000; num+=i, i++);
```

循环结构中的break和continue语句

今有物不知其数：
三三数余二，
五五数余三，
七七数余四，
问：
此物至少几何？

```
#include<iostream>
using namespace std;
int main()
{
    int num;
    for(num = 1; num <= 1000; num++)
    {
        if(num%3==2 && num%5==3 && num%7==4)
        {
            cout<<"num = "<<num<<endl;
            break;
        }
    }
    return 0;
}
```

循环结构中的break和continue语句

今有物不知其数：
三三数余二，
五五数余三，
七七数余四，
问：
此物至少几何？

```
#include<iostream>
using namespace std;
int main()
{
    int num;
    for(num = 1; num <= 1000; num++)
    {
        if(num%3==2 && num%5==3 && num%7==4)
        {
            cout<<"num = "<<num<<endl;
            break;
        }
    }
    return 0;
}
```

break语句：仅结束其所在的一层循环语句。

循环结构中的break和continue语句

今有物不知其数：
三三数余二，
五五数余三，
七七数余四，
问：
此物至少几何？

```
#include<iostream>
using namespace std;
int main()
{
    int num;
    for(num = 1; num <= 1000; num++)
    {
        if(num%3==2 && num%5==3 && num%7==4)
        {
            cout<<"num = "<<num<<endl;
            break;
        }
    }
    return 0;
}
```

break语句：只用于循环结构或switch语句中。常和if语句一起使用，表示当条件满足时，跳出循环结构。

循环结构中的break和continue语句

今有物不知其数：

三三数余二，

五五数余三，

七七数余四，

问：
此物几何？

```
#include<iostream>
using namespace std;
int main()
{
    int num, iCount = 0;
    for(num = 1; num <= 1000; num++)
    {
        if(num%3==2 && num%5==3 && num%7==4)
        {
            iCount ++;
        }
    }
    cout<<"iCount= "<<iCount<<endl;
    return 0; }
```

循环结构中的break和continue语句

今有物不知其数：

三三数余二，

五五数余三，

七七数余四，

问：
此物几何？

```
#include<iostream>
using namespace std;
int main()
{
    int num, iCount = 0;
    for(num = 1; num <= 1000; num++)
    {
        if(num%3==2 && num%5==3 && num%7==4)
        {
            continue;
            iCount ++;
        }
    }
    cout<<"iCount= "<<iCount<<endl;
    return 0; }
```

循环结构中的break和continue语句

今有物不知其数：
三三数余二，
五五数余三，
七七数余四，
问：
此物几何？

```
#include<iostream>
using namespace std;
int main()
{
    int num, iCount = 0;
    for(num = 1; num <= 1000; num++)
    {
        if(num%3==2 && num%5==3 && num%7==4)
        {
            continue;
            iCount ++;
        }
    }
    cout<<"iCount= "<<iCount<<endl;
    return 0; }
```

continue语句：结束本次循环，即跳过本层循环体中余下尚未执行的语句，接着再一次进行循环条件的判定。

循环结构中的break和continue语句

- **break**
 - 仅结束其所在的一层循环语句或者开关语句
- **continue**
 - 提前结束本次循环，准备进入下一次循环

警察抓小偷：多层循环

警察局抓住了A、B、C、D四名盗窃嫌疑犯，其中只有一人是小偷。A说：“我不是小偷”；B说：“C是小偷”；C说：“小偷肯定是D”；D说：“C在冤枉好人”。现在已知这四人中有三人说的是真话，一人说的是假话。请问到底谁是小偷？

```
#include<iostream>
using namespace std;
int main(){
```

```
return 0;}
```

警察抓小偷：多层循环

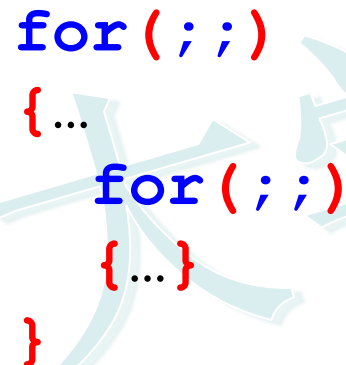
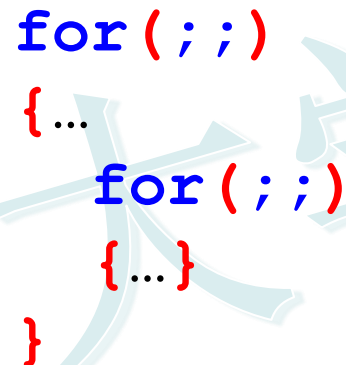
警察局抓住了A、B、C、D四名盗窃嫌疑犯，其中只有一人是小偷。A说：“我不是小偷”；B说：“C是小偷”；C说：“小偷肯定是D”；D说：“C在冤枉好人”。现在已知这四人中有三人说的是真话，一人说的是假话。请问到底谁是小偷？

```
#include<iostream>
using namespace std;
int main(){
    int a,b,c,d;
    for(a=1;a>=0;a--) //1:Yes 0:No
        for(b=1;b>=0;b--) //1:Yes 0:No
            for(c=1;c>=0;c--) //1:Yes 0:No
                for(d=1;d>=0;d--) //1:Yes 0:No
                    if((a==0)+(c==1)+(d==1)+(d==0)==3&& a+b+c+d==1)
                    {
                        cout<<"A "<<(a?"Yes":"No")<<endl;
                        cout<<"B "<<(b?"Yes":"No")<<endl;
                        cout<<"C "<<(c?"Yes":"No")<<endl;
                        cout<<"D "<<(d?"Yes":"No")<<endl;
                    }
    return 0;}
```

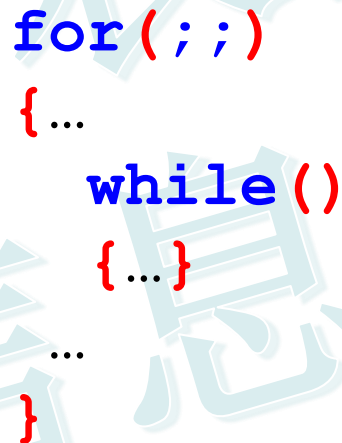
警察抓小偷：for循环结构

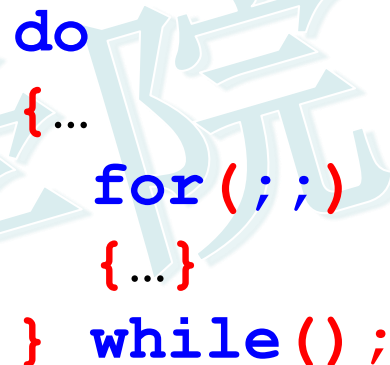
```
while ()  
{ ...  
    while ()  
    { ... }  
}
```

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

```
for (;;)   
{ ...  
    for (;;)   
    { ... }  
}
```

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

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

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

3种循环语句可以互相嵌套。

小结

- `for`语句
- 循环语句中的`break`和`continue`语句
- 循环的嵌套

延伸

- 用`while`语句和`do...while`改写神奇妙算程序，计算此物至少几何？
- 将用`for`语句实现的神奇妙算程序中表达式2设为缺省，改写程序，计算此物至少几何？体会形式上死循环，实际上不“死”的技巧。