

1.4.3控制语句_循环语句for

循环语句可以相互转换。

- for：循环次数明确
 - 循环条件也明确
- while：循环条件明确
 - 循环次数不明确，或比较难计算
- do...while：至少循环一次
 - 特殊的while

```
1  for (<表达式1>; <表达式2>; <表达式3>)  
2      <语句>
```

- 说明：
 - <表达式1>用于设置进入for循环时的初始状态；
 - <表达式2>是循环条件表达式，其值为真时执行循环，为假时终止循环；
 - <表达式3>用于改变某些变量的值，以便使表达式2的值为假，最终使for循环结束。
- 执行过程：
 - 先对<表达式1>求值，然后对<表达式2>求值，如果<表达式2>的值为真，执行循环体中的语句；
 - 执行完循环体后，对<表达式3>求值，此时完成一次循环每执行一次循环后再对<表达式2>求值，以决定是否进行下次循环。
- 在for循环后面的圆括号内可以声明变量，还可使用逗号表达式，如：

```
1  for (int i = 0, j = 0; i<100; i++, j+=5)  
2      cout<<i<<j;
```

- 组成for循环的表达式可以部分或全部省略掉。
- //程序不断读入整数，直到遇到一个大于零的整数为止。

```
1  int n = -1;  
2  for (; n<=0; )  
3      cin>>n
```

例子

```
1  #include <bits/stdc++.h>  
2  using namespace std;  
3  
4  void eg2_14()  
5  {  
6      const double PI = atan(1.0) * 4;  
7      int rMax = 10;  
8      // printf("%.10lf",PI);
```

```

9      cout << "r\t\tS\t\tV" << endl;
10     for (int r = 1; r <= rMax; r++)
11     {
12         cout << r << "\t\t" << PI * r * r << "\t\t"
13             << PI * r * r * r * r * 4 / 3 << endl;
14     }
15 }
16 int main()
17 {
18     eg2_14();
19     return 0;
20 }

```

改进

```

1 void eg2_14()
2 {
3     const double PI = atan(1.0) * 4, c1 = 4.0/3;
4     int rMax = 10;
5     cout << "r\t\tS\t\tV" << endl;
6     for (int r = 1, rr = 1; r <= rMax; r++, rr = r*r)
7     {
8         cout << r << "\t\t" << PI * rr << "\t\t"
9             << PI * rr * r * c1 << endl;
10    }
11 }
12 int

```