第5章 循环控制

——计数控制的循环

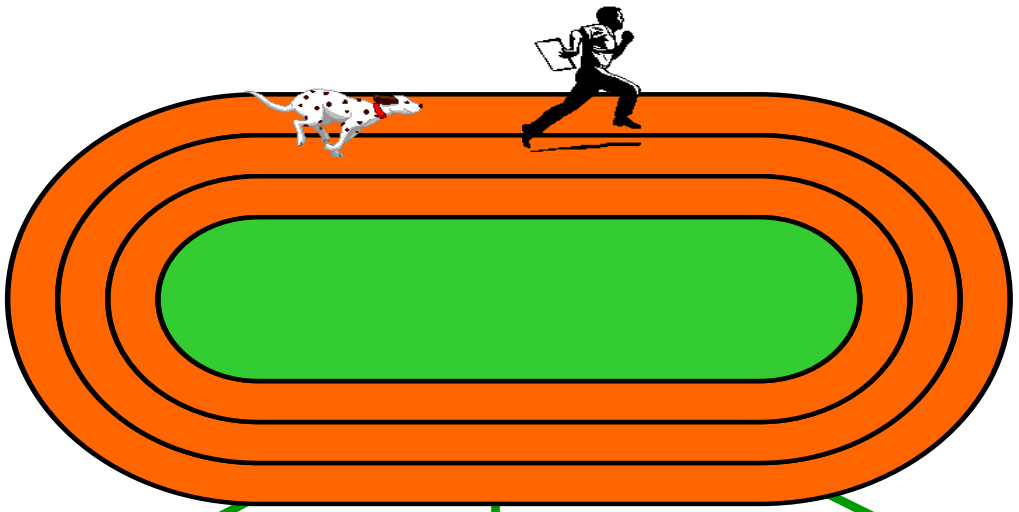
循环结构有什么用？

国外某男子攻打自己的女友，并导致女友受伤，法官除判处他监禁和提供金钱补偿外，还处罚他抄写5000遍道歉辞：

“Boys do not hit girls.”

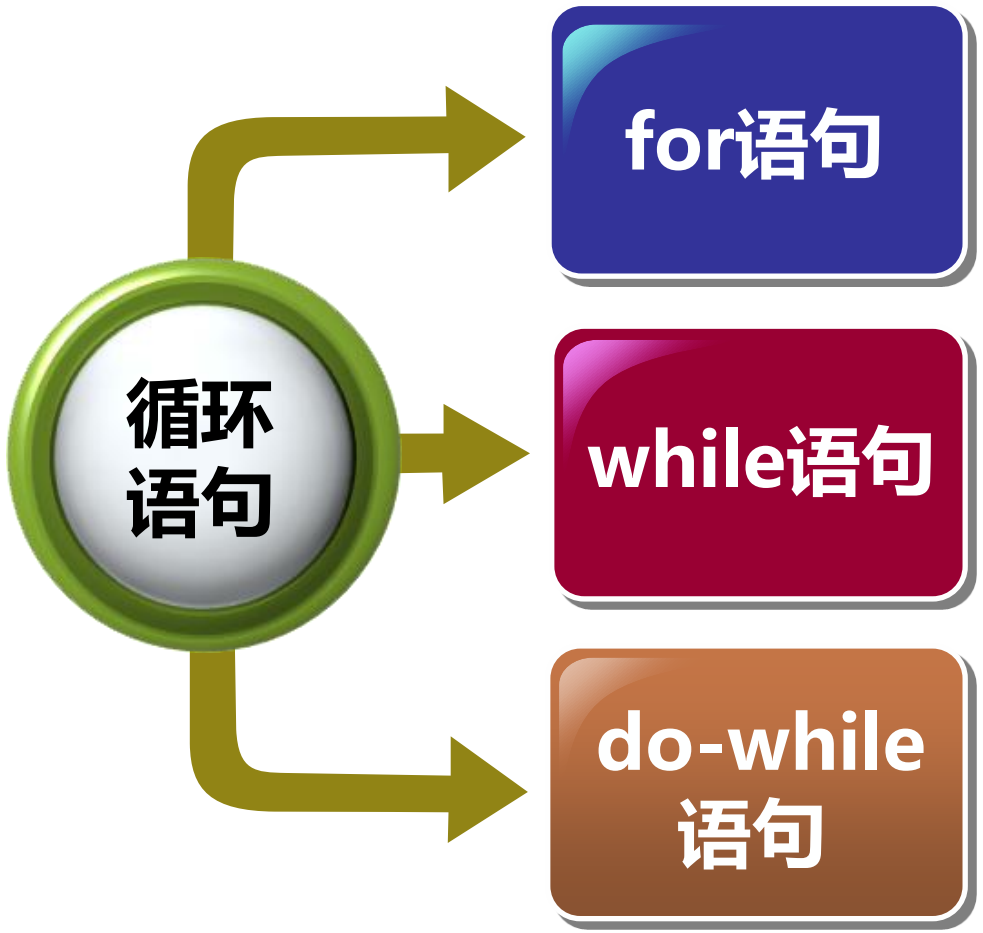
```
for (i=0; i<5000; i++)
{
    printf("Boys do not hit girls.");
}
```

循环的控制方法



- 条件控制的循环 Condition Controlled
- 计数控制的循环 Counter Controlled
- 标记控制的循环 Sentinel Controlled

循环语句



for语句

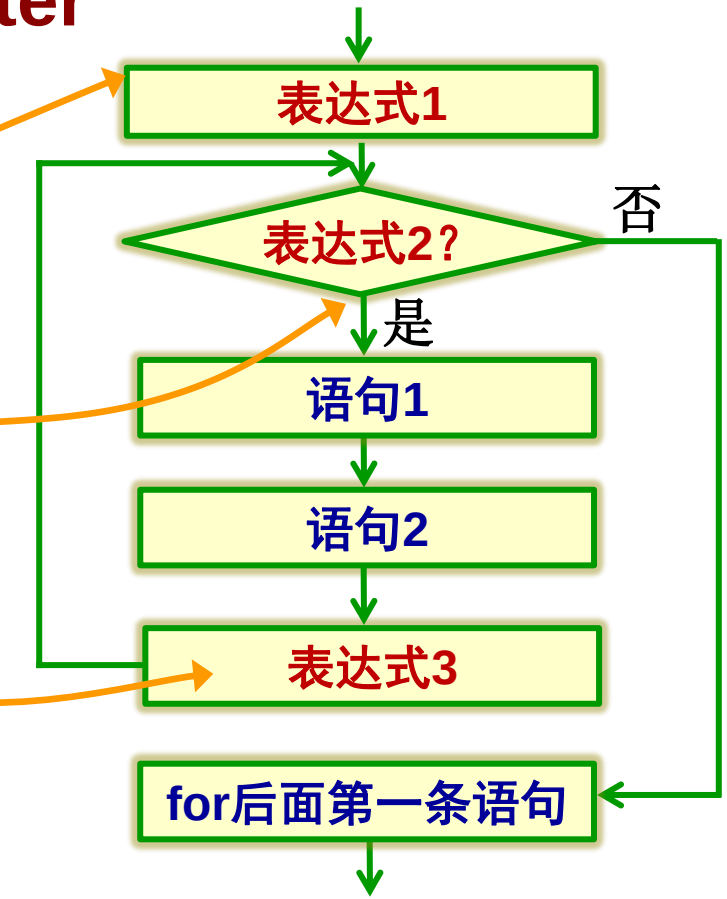
- 计数控制——Loop is controlled by a counter

循环初始条件　循环控制条件　循环转化条件

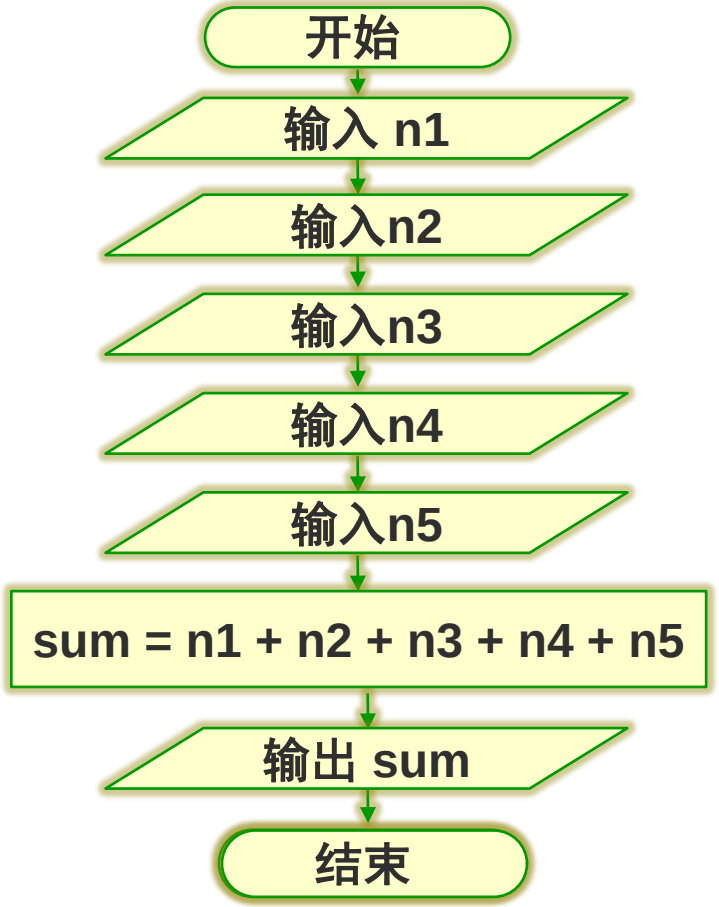
```
for (表达式1； 表达式2； 表达式3)
{
    语句1
    语句2
}
```

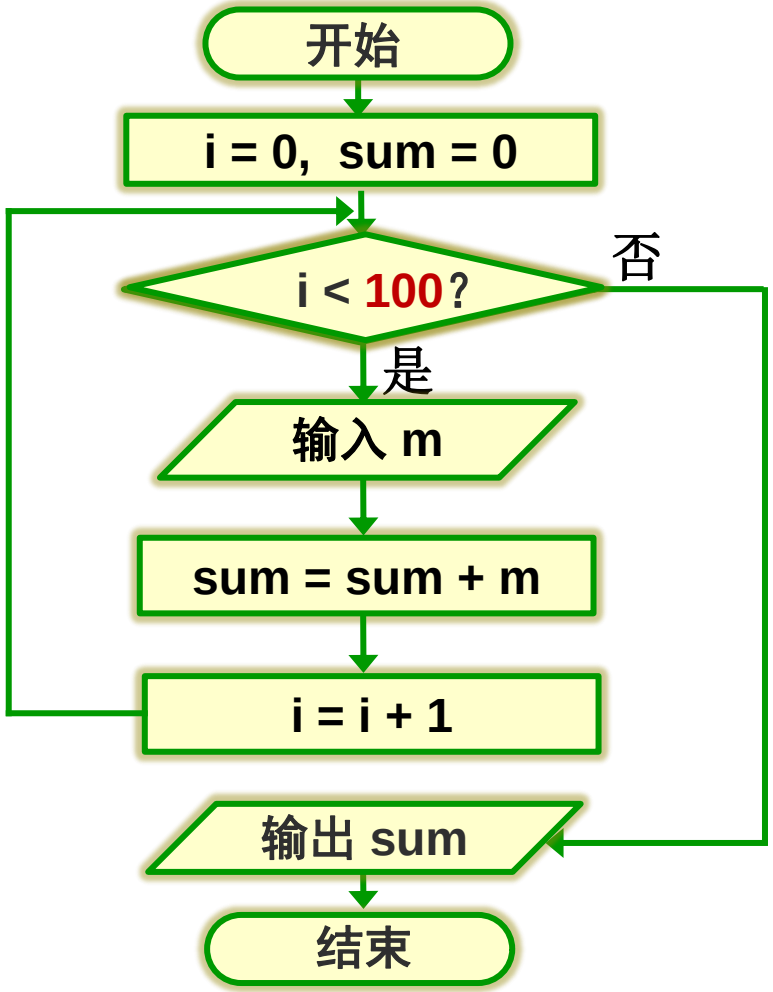
复合语句做循环体
被当作一条语句看待

不要重复在循环体内
改变循环变量的值

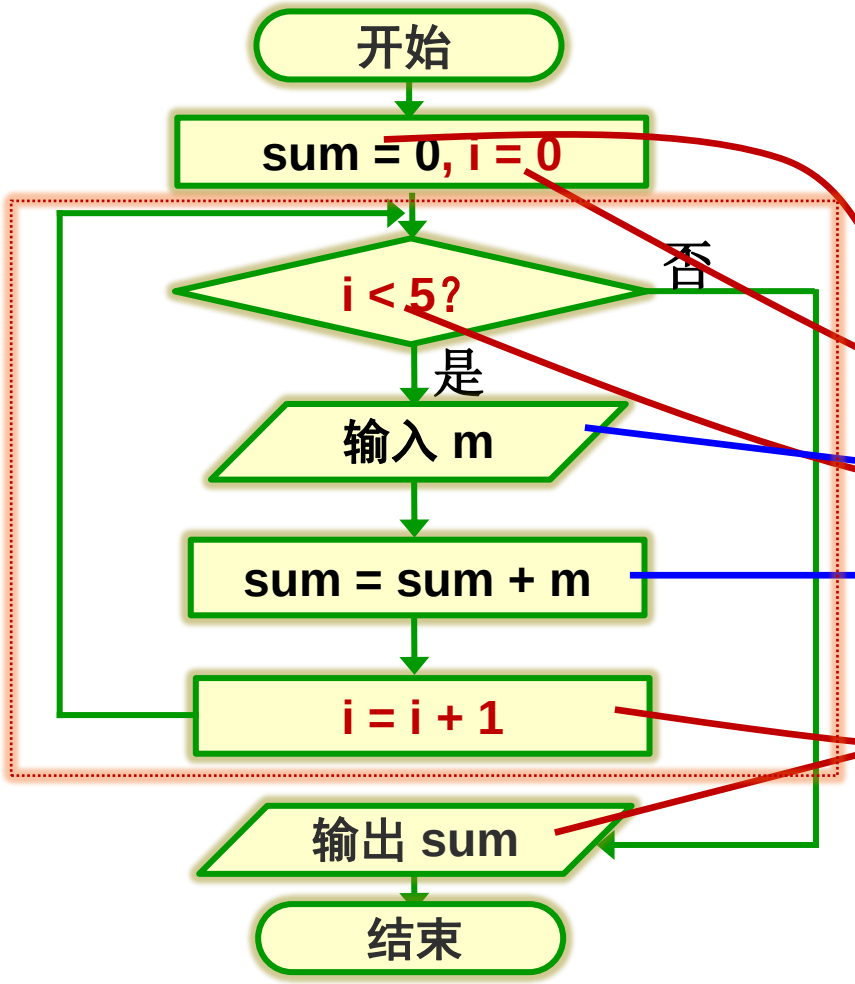


5个输入数据求和→100个输入数据求和





5个输入数据求和的程序实现




```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m;
   for (i=0; i<5; i++)
   {
      scanf("%d", &m);
      sum = sum + m;
   }
   printf("sum = %d", sum);
   return 0;
}
```

这个循环是如何执行的？

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m;
   for (i=0; i<5; i++)
   {
       scanf("%d", &m);
       sum = sum + m;
   }
   printf("sum = %d", sum);
   return 0;
}
```

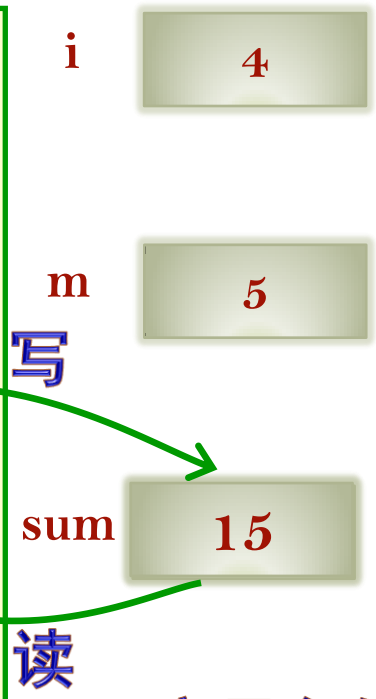
sum=0的作用？

- 计数控制的循环
- 假设输入1，2，3，4，5

计数器 i	sum ←	sum +	m
0	1	0	1
1	3	1	2
2	6	3	3
3	10	6	4
4	15	10	5
5			

为什么它能实现累加运算？

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m;
   for (i=0; i<5; i++)
   {
       scanf("%d", &m);
       sum = sum + m;
   }
   printf("sum = %d", sum);
   return 0;
}
```



如何理解程序设计中的

sum = sum + m

赋值操作

sum新值 = sum旧值 + m

通过变量名访问变量的值

变量名的含义

如何保证循环是可终止的？

i=0,1,2,3,4 → i<5

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m;
   for (i=0; i<5; i++)
   {
       scanf("%d", &m);
       sum = sum + m;
   }
   printf("sum = %d", sum);
   return 0;
}
```

i=5,4,3,2,1 → i>0

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m;
   for (i=5; i>0; i--)
   {
       scanf("%d", &m);
       sum = sum + m;
   }
   printf("sum = %d", sum);
   return 0;
}
```

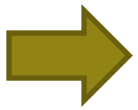
执行循环体时必须改变一个或多个变量的值

保证经有限次重复后，循环控制条件不再满足

如何实现n个键盘输入数据的累加？

- 循环次数n从键盘输入

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m;
   for (i=0; i<5; i++)
   {
       scanf("%d", &m);
       sum = sum + m;
   }
   printf("sum = %d", sum);
   return 0;
}
```



```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m, n;
   printf("Input n:");
   scanf("%d", &n);
   for (i=0; i<n; i++)
   {
       scanf("%d", &m);
       sum = sum + m;
   }
   printf("sum = %d", sum);
   return 0;
}
```

计算并输出1+2+3……+n的值

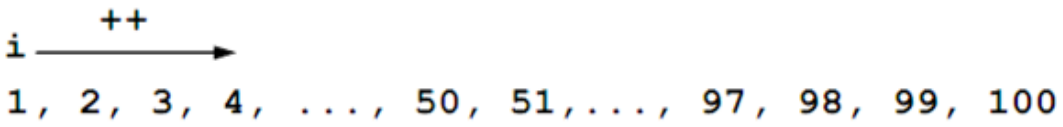
```
#include  <stdio.h>
int main()
{
   int i, sum = 0, m, n;
  printf("Input n:");
  scanf("%d", &n);
  for (i=0; i<n; i++)
  {
      scanf("%d", &m);
      sum = sum + m;
  }
  printf("sum = %d", sum);
  return 0;
}
```

```
Input n: 100 ↙
sum = 5050
```

```
#include  <stdio.h>
int main()
{
   int i, sum = 0,    n;
  printf("Input n:");
  scanf("%d", &n);
  for (i=1; i<=n; i++)
  {

      sum = sum + i;
  }
  printf("sum = %d", sum);
  return 0;
}
```

如何快速计算1+2+3……+100的值？




```
#include  <stdio.h>
int main()
{
   int i, sum = 0    ;
   for (i=1; i<=100; i++)
   {
       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

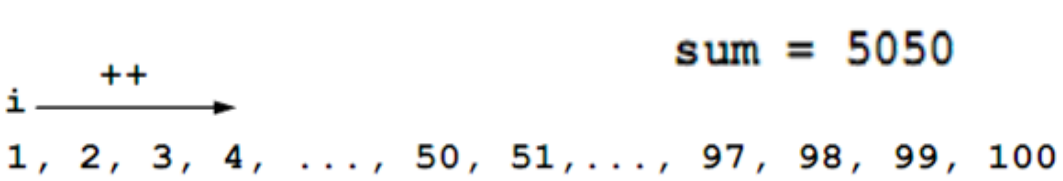
```
Input n: 100 ↙
sum = 5050
```

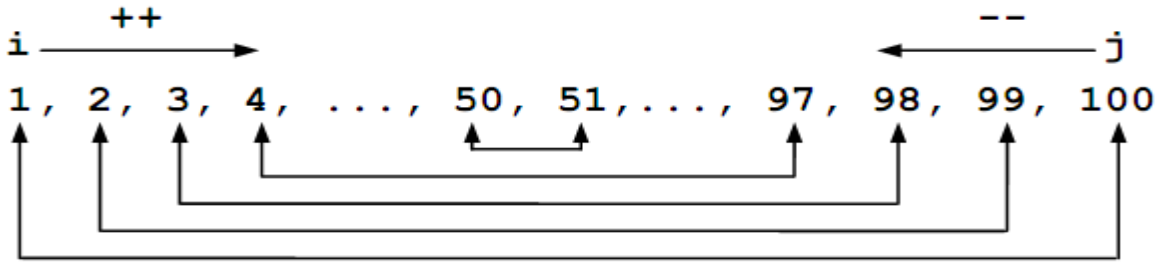
```
sum = 5050
```

```
#include  <stdio.h>
int main()
{
   int i, sum = 0,     n;
   printf("Input n:");
   scanf("%d", &n);
   for (i=1; i<=n; i++)
   {

       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

如何快速计算1+2+3……+100的值？





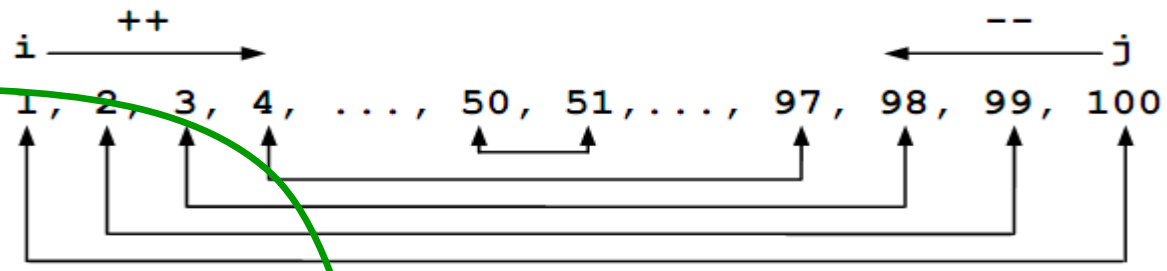

```
#include  <stdio.h>
int main()
{
   int i, sum = 0;
   for (i=1; i<=100; i++)
   {
       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

```
#include  <stdio.h>
int main()
{
   int i, j, sum = 0;
   for (i=1,j=100; i<=j; i++,j--)
   {
       sum = sum + i + j;
   }
   printf("sum = %d", sum);
   return 0;
}
```

如何快速计算1+2+3……+100的值？

逗号运算符（Comma Operator）

表达式1， 表达式2， …， 表达式n



- 多数情况下，并不使用整个逗号表达式的值
- 更常见的是分别得到各表达式的值——顺序求值运算符
- 主要用在循环语句中，同时对多个变量赋初值等

```
#include <stdio.h>
int main()
{
   int i, j, sum = 0;
   for (i=1,j=100; i<=j; i++,j--)
   {
       sum = sum + i + j;
   }
   printf("sum = %d", sum);
   return 0;
}
```

$$\sum_{i=1}^{n} i = 1+2+3+\ldots+n$$

$$\sum_{i=1}^{n} i^2 = 1^2+2^2+3^2\ldots+n^2$$

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, n;
  printf("Input n:");
  scanf("%d", &n);
   for (i=1; i<=n; i++)
   {
       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

sum = sum + i

sum = sum + i * i

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, n;
  printf("Input n:");
  scanf("%d", &n);
   for (i=1; i<=n; i++)
   {
       sum = sum + i*i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

$$\sum_{i=1}^{n} i = 1+2+3+\ldots+n$$

$$1+3+5\ldots+(2n-1)$$

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, n;
   printf("Input n:");
   scanf("%d", &n);
   for (i=1; i<=n; i++)
   {
       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

sum = sum + i

sum = sum + 2*i-1

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, n;
   printf("Input n:");
   scanf("%d", &n);
   for (i=1; i<=n; i++)
   {
       sum = sum + 2*i-1;
   }
   printf("sum = %d", sum);
   return 0;
}
```

$$\sum_{i=1}^{n} i = 1+2+3+\ldots+n$$

$$1+3+5\ldots+(2n-1)$$

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, n;
   printf("Input n:");
   scanf("%d", &n);
   for (i=1; i<=n; i++)
   {
       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

i = i + 1

i++

i = i + 2

i += 2

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, n;
   printf("Input n:");
   scanf("%d", &n);
   for (i=1; i<=2*n-1; i+=2)
   {
       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

$$\sum_{i=1}^{n} i = 1+2+3+\ldots+n$$

$$n! = 1\times 2\times 3\times \ldots \times n$$

```
#include  <stdio.h>
int main()
{
   int i, sum = 0, n;
  printf("Input n:");
  scanf("%d", &n);
   for (i=1; i<=n; i++)
   {
       sum = sum + i;
   }
   printf("sum = %d", sum);
   return 0;
}
```

sum = sum + i

p = p * i

```
#include  <stdio.h>
int main()
{
   int i, p = 1, n;
  printf("Input n:");
  scanf("%d", &n);
   for (i=1; i<=n; i++)
   {
       p = p * i;
   }
   printf("p = %d", p);
   return 0;
}
```

小结

- 累加求和
  - 初始化为0或第一项
  - 关键是寻找累加项的构成规律 （通项）
- 累乘求积（求阶乘）
  - 初始化为1
- 计算器变量
  - 记录循环执行的次数，在计数控制的循环中控制循环的结束

小结

- 累加项的前后项之间**无关**
  - 1*2*3 + 3*4*5 +… + 99*100*101
    - sum = sum + i*(i+1)*(i+2)
    - i = i + 2　　(i=1,3,…,99)
- 累加项的前后项之间**有关**
  - $x^0 + x^1 + x^2 + \ldots + x^n$
    - sum = sum + pow(x, i);　i = i + 1; (i=0,1,2,…,n)
    - term = term * x;　　term初值为x^0 即1

讨论题

- 修改这个程序使其快速计算1+2+3……+n的值，n从键盘输入。

```
#include  <stdio.h>
int main()
{
   int i, j, sum = 0, n;
   for (i=1,j=100; i<=j; i++,j--)
   {
       sum = sum + i + j;
   }
   printf("sum = %d", sum);
   return 0;
}
```

计算并输出1+2+3……+n的值

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

```
Input n: 10 ↙
i=1,sum=4199399
i=2,sum=4199401
i=3,sum=4199404
i=4,sum=4199408
i=5,sum=4199413
i=6,sum=4199419
i=7,sum=4199426
i=8,sum=4199434
i=9,sum=4199443
i=10,sum=4199453
sum = 4199453
```

计算并输出1+2+3……+n的值

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

```
Input n: 10 ↙
i=1,sum=1
i=2,sum=3
i=3,sum=6
i=4,sum=10
i=5,sum=15
i=6,sum=21
i=7,sum=28
i=8,sum=36
i=9,sum=45
i=10,sum=55
sum = 55
```