

C++大学基础教程

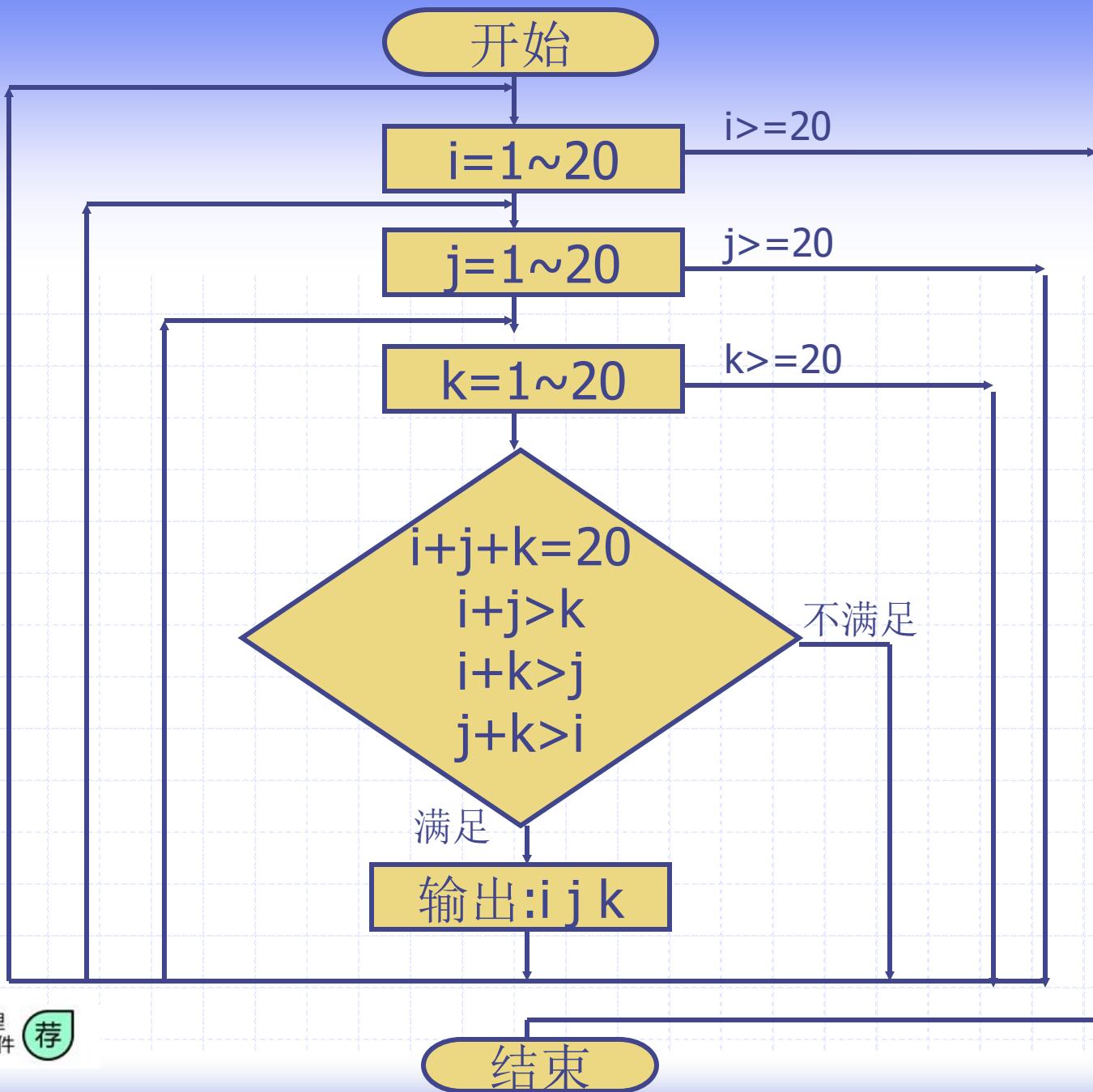
第3章 C++控制语句

北京科技大学
信息基础科学系

◆用计算机解决实际问题，首先要分析问题，设计合适的算法，然后编程实现算法。

已知三角形的周长为20，求出所有满足条件的三角形（求边长）。

```
#include <iostream>
using namespace std;
void main()
{
    for(int i=1;i<20;i++)
        for(int j=1;j<20;j++)
            for(int k=1;k<20;k++)
                if(i+j+k==20&&i+j>k&&i+k>j&&j+k>i)
                    cout<<"i=""<<i<<"j=""<<j<<"k=""<<k<<endl;
}
```



第3章 C++控制语句

3. 1 算法的基本控制结构

3. 2 if选择语句

3. 3 switch选择语句

3. 4 循环语句

3. 5 循环嵌套

3. 6 应用举例

3.1 算法的基本控制结构

3. 1 算法的基本控制结构

◆ 算法

- 1、计算机执行的操作
- 2、这些操作的顺序

◆ 伪码

采用类似计算机语言的分句格式，以自然语言描述算法。

3.1 算法的基本控制结构

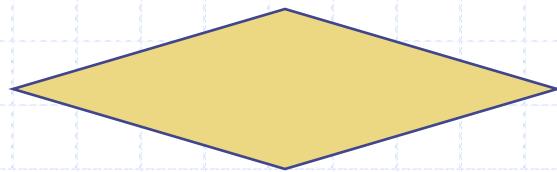
◆ 流程图



开始/结束



执行



判断

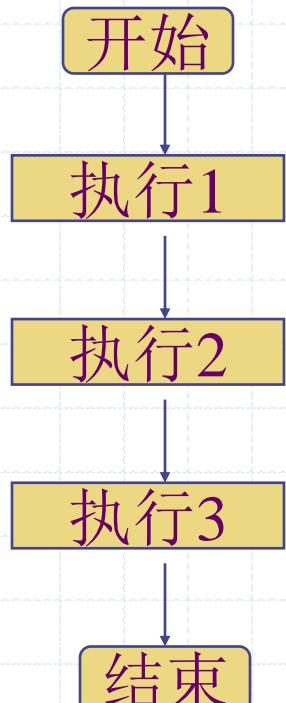


联系

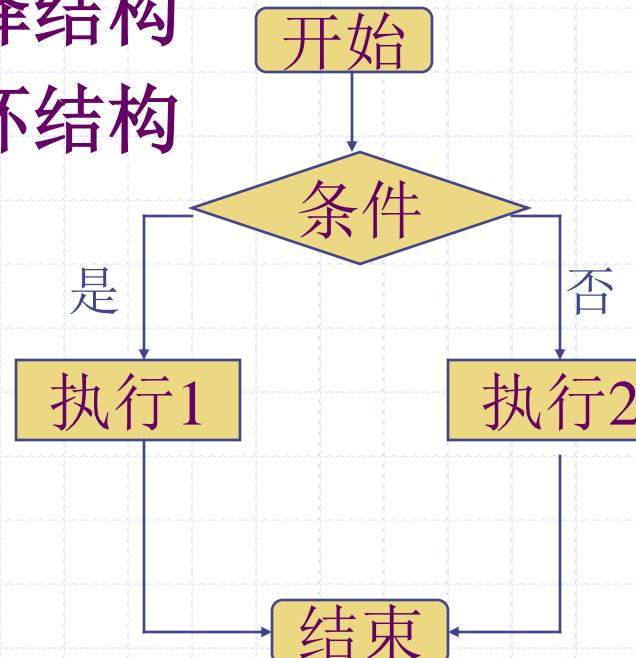
3.1 算法的基本控制结构

◆ 程序的三种基本控制结构

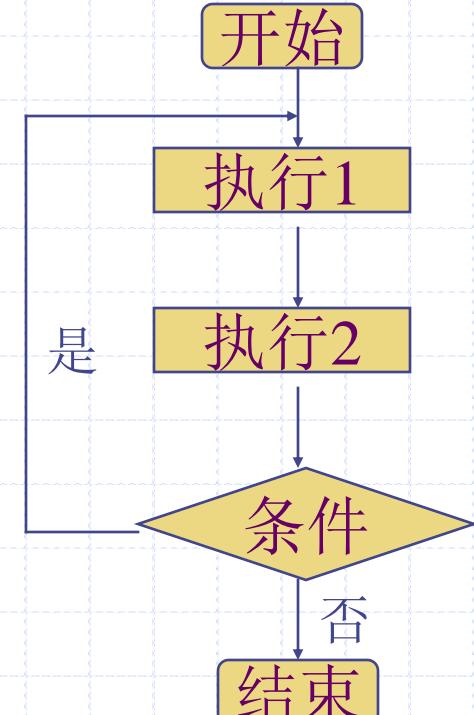
(1) 顺序结构



(2) 选择结构



(3) 循环结构



3. 1 算法的基本控制结构

◆ 流程控制语句

选择语句: if..else、 switch

循环语句: while、 for 、 do..while

转向语句: break、 continue

3. 2 if选择语句

3. 2 if选择语句

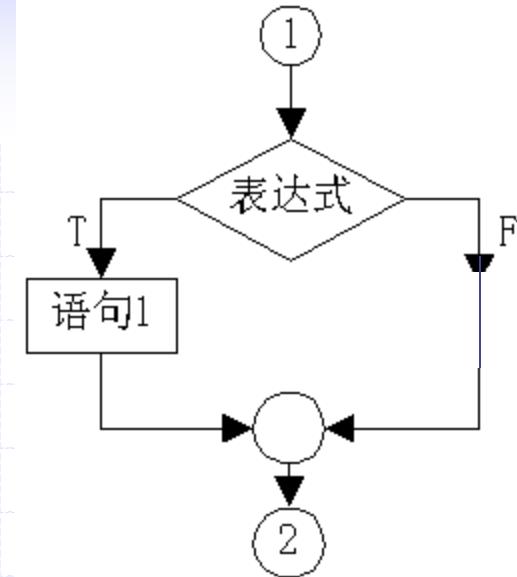
◆ if_else

- 1、没有else分支
- 2、双分支
- 3、多分支
- 4、if 语句的嵌套

3. 2 if选择语句

◆ 1、没有else分支

```
if (表达式)  
    语句;
```



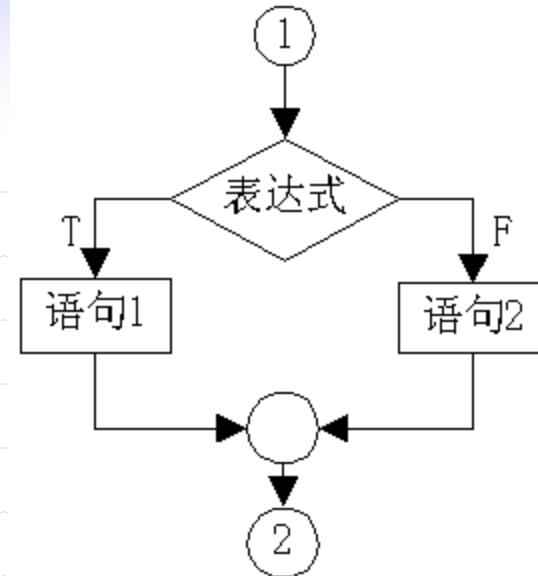
例如：比较x和y的大小，如果x大则输出x的值

```
if ( x>y )  
    cout<<x<<endl;
```

3. 2 if选择语句

◆ 2、双分支

```
if (表达式)  
    语句1;  
else  
    语句2;
```



◆ 例如：比较x，y的大小，将较大的数输出

```
if ( x>y )  
    cout<<x<<endl;  
else  
    cout<<y<<endl;
```

例子

任意输入一个4位整数的年份，判断该年是否是润年？

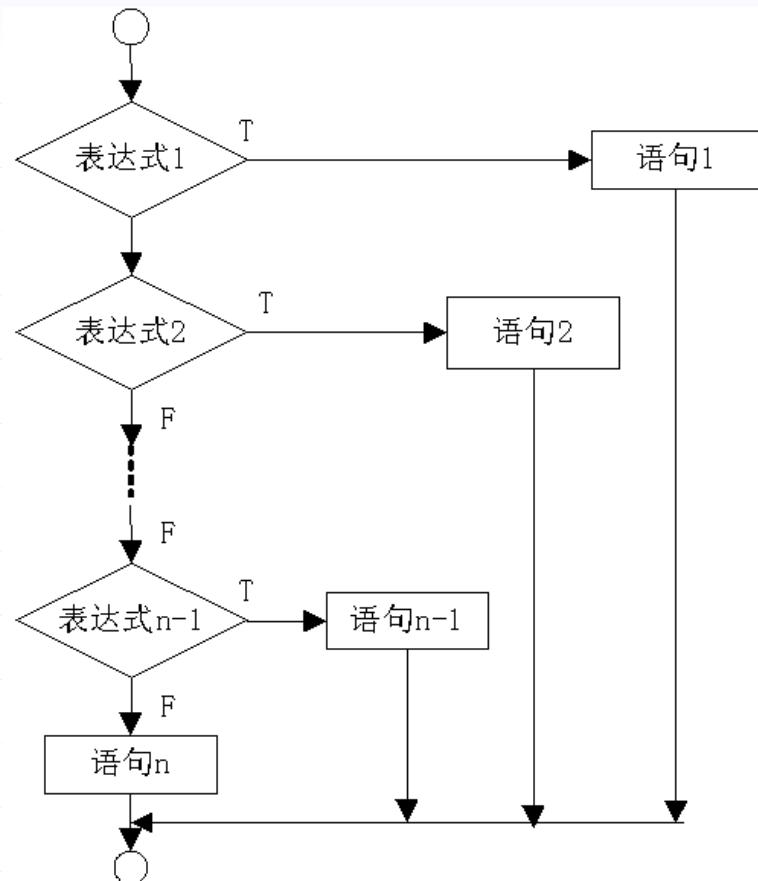
是指年份能被4整除，并且不能被100整除或能被400整除的那些年

```
#include <iostream>
using namespace std;
void main()
{
    int year;
    cin>>year;
    if((year % 4==0) && (year %100 != 0)
        ||(year % 400==0))
        cout<<"润年"<<endl;
    else
        cout<<"不是润年"<<endl;
}
```

3. 2 if选择语句

◆ 3、多分支

```
if (表达式1)
    语句1;
else if (表达式2)
    语句2;
else if (表达式3)
    语句3;
    ...
else
    语句n;
```



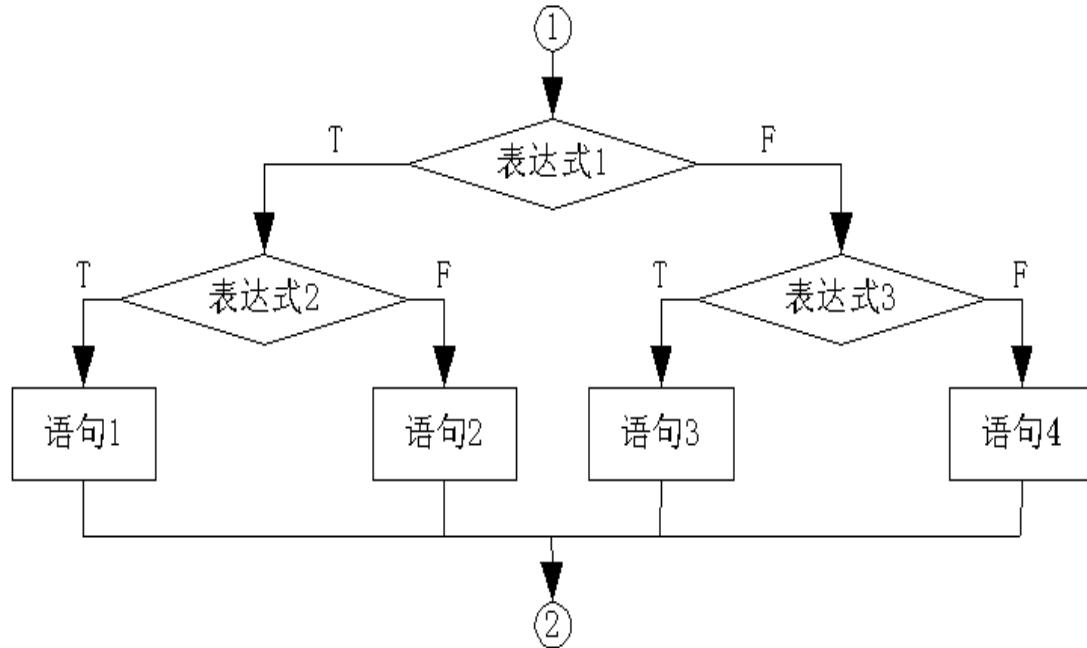
例子：输出学生的成绩

```
#include <iostream>
using namespace std;
void main()
{
    int x;    cin>>x;
    if(x>=90)
        cout<<“优秀” <<endl;
    else if(x>=80)
        cout<<“良好” <<endl;
    else if(x>=60)
        cout<<“合格” <<endl;
    else
        cout<<“不及格” <<endl;
}
```

X>=80 && x<90

3. 2 if选择语句

◆ 4、if语句的嵌套

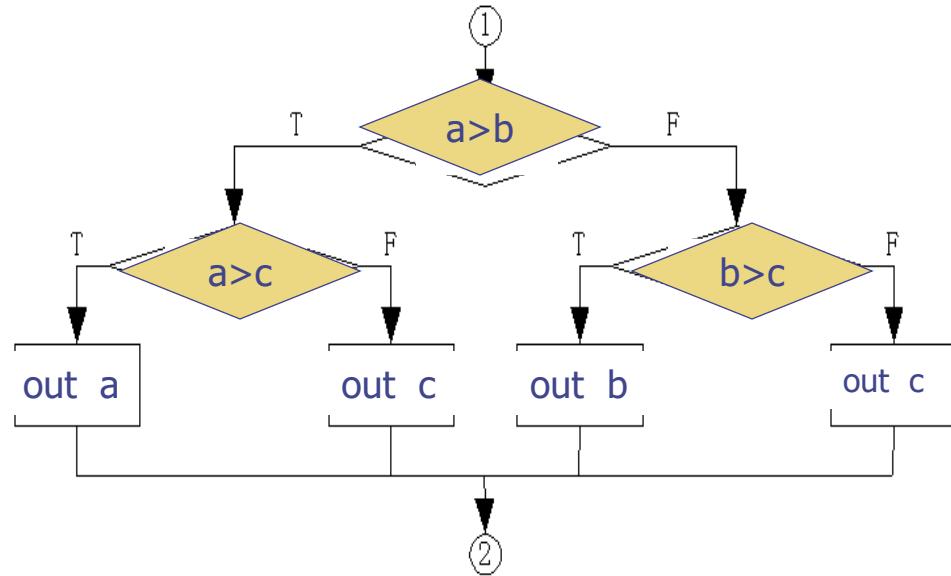


```
if (表达式1)
    if (表达式2)
        语句1
    else
        语句2
else
    if (表达式3)
        语句3
    else
        语句4
```

例子：输出a, b, c中较大的数

```
#include <iostream>
using namespace std;
void main()
{
    int a,b,c;
    cin>>a>>b>>c;
    if (a>b)
    {
        if (a>c) cout<<a<<endl;
        else      cout<<c<<endl;
    }
    else
    {
        if (b>c) cout<<b<<endl;
        else      cout<<c<<endl;
    }
}
```

```
cout<< (a>b?(a>c?a:c):(b>c?b:c) );
```



3. 2 if选择语句

◆ 注意：二义性

```
int x; cin>>x;
```

```
if (x>0)
```

```
{ if (x<50)
```

```
    cout<<"x is <50"<<endl; }
```

```
else
```

```
    cout<<"x is <=0"<<endl;
```

```
if (x>0)
    if (x<50)
        cout<<"x is <50"<<endl;
    else
        cout<<"x is <=0"<<endl;
```

if和else的对应关系，else总是与离它最近的if匹配。

3. 3 switch选择语句

3. 3 switch选择语句

switch (表达式)

{

case 常量表达式1:

 语句1

 break;

case 常量表达式2:

 语句2

 break;

⋮

case 常量表达式n:

 语句n

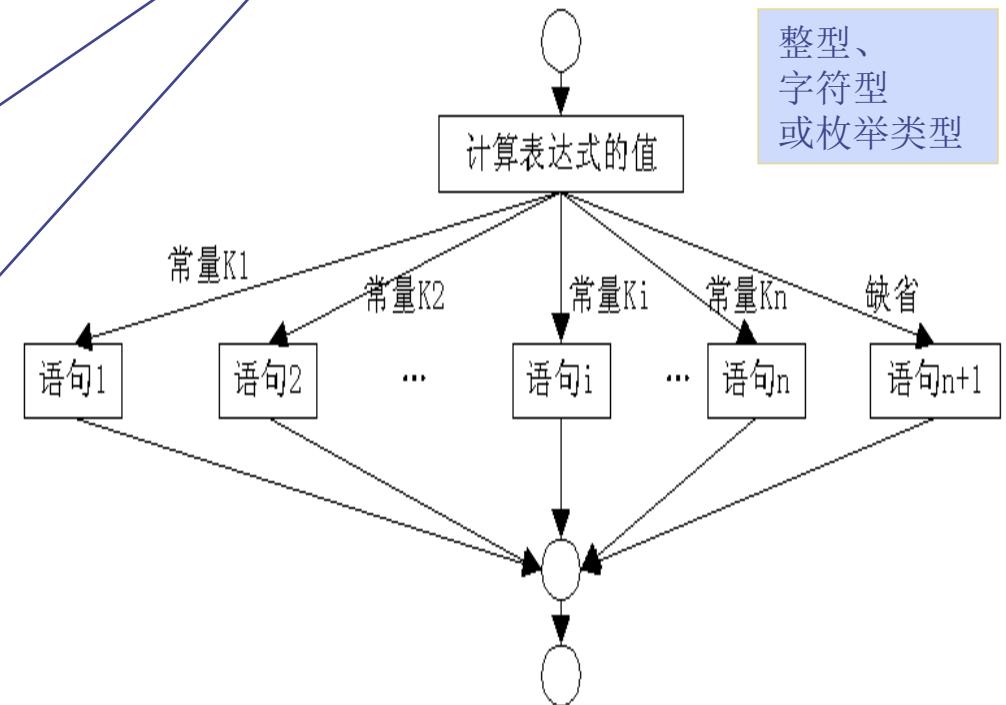
 break;

default: 语句n+1

}

跳出switch语句

整型、
字符型
或枚举类型



例子：输出学生的成绩

```
int x; cin>>x;
switch(x/10)
{
    case 10:
    case 9:cout<<“优秀” <<endl;
              break;
    case 8: cout<<“良好” <<endl;
              break;
    case 7:
    case 6: cout<<“合格” <<endl;
              break;
    default: cout<<“不及格” <<endl;
}
```

3.4 循环语句

3.4 循环语句

三种基本语法

1、 while 语句

2、 do...while 语句

3、 for 语句

```
#include <iostream>
#include <iomanip>
using namespace std;
void main(){
    cout<<setfill('*')
        <<setw(2)<<'\n'
        <<setw(3)<<'\n'
        <<setw(4)<<'\n'
        <<setw(5)<<'\n'
        <<setw(6)<<'\n'
        <<setw(7)<<'\n'
        <<setw(8)<<'\n';
}
```

*
**


```
#include <iostream>
#include <iomanip>
using namespace std;
void main(){
    int i=2;
    cout<<setfill('*');
    while(i<=8)
    {
        cout<<setw(i)<<'\n';
        i++;
    }
}
```

循环控制
条件

*
**

循环变量

循环体

3.4 循环语句

◆ 主要包括3个部分：

1、循环控制条件：

判断循环操作是否进行的条件；

2、循环体：

重复进行的操作；

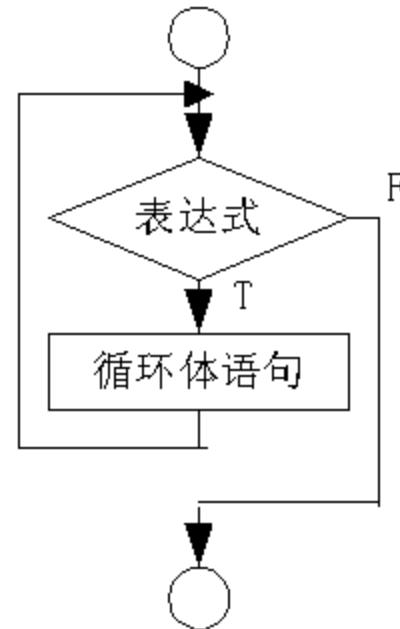
3、循环控制变量：

记录循环体执行的次数，或控制循环的结束条件。

3.4 循环语句

◆ while循环语句

while(表达式)
 循环体语句；



例1：编程循环输出1-9这9个数字？

运行结果：

```
#include <iostream>
using namespace std;
void main()
{
    int i=1;
    while (i<=9)
    {
        cout<<i<<endl;
        i++;
    }
    cout<<“循环结束” <<endl;
}
```

1
2
3
4
5
6
7
8
9

循环结束

例2：编程计算 $1+2+3+\dots+10$ 的值

```
#include <iostream>
using namespace std;
void main()
{
    int sum=0, i=1;
    while(i<=10)
    {
        sum+=i;
        i++;
    }
    cout<<sum<<endl;
}
```

55

例3：打印可视字符

```
#include<iostream>
using namespace std;
void main()
{
    unsigned char ch=32;
    while(ch<127)
    {
        cout<<(int)ch<<'\'t'<<ch<<'\'t';
        ch++;
    }
}
```

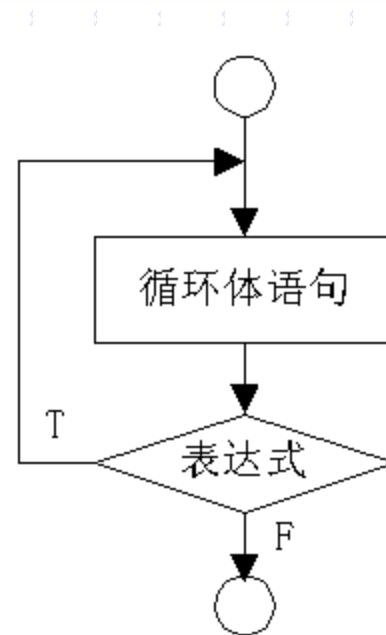
例4：编程计算 $\frac{1}{n} \sum_{k=1}^n k^2$ 的值

```
#include <iostream>
using namespace std;
void main()
{
    int k=1,sum=0,n;
    cout<<"input n(n>0):";
    cin>>n;
    while(k<=n)
    {
        sum=sum+k*k;
        k++;
    }
    if(n>0) cout<<sum/n<<endl;
    else cout<<"n不能<=0"<<endl;
}
```

3.4 循环语句

◆ do..while循环语句

```
do  
{  
    循环体语句;  
} while(表达式);
```



```
#include <iostream>
#include <iomanip>
using namespace std;
void main()
{
    int i=2;
    cout<<setfill('*');

    do
    {
        cout<<setw(i)<<'\n';
        i++;
    } while(i<=8);
```

*
**

例子：编程计算 $\frac{1}{n} \sum_{k=1}^n k^2$ 的值

```
#include <iostream>
using namespace std;
void main()
{
    int k=1,sum=0,n;
    cout<<"input n(n>0):";
    cin>>n;
    do
    {
        sum=sum+k*k;
        k++;
    }while(k<=n);
    if(n>0)    cout<<sum/n<<endl;
    else        cout<<"n不能<=0"<<endl;
}
```

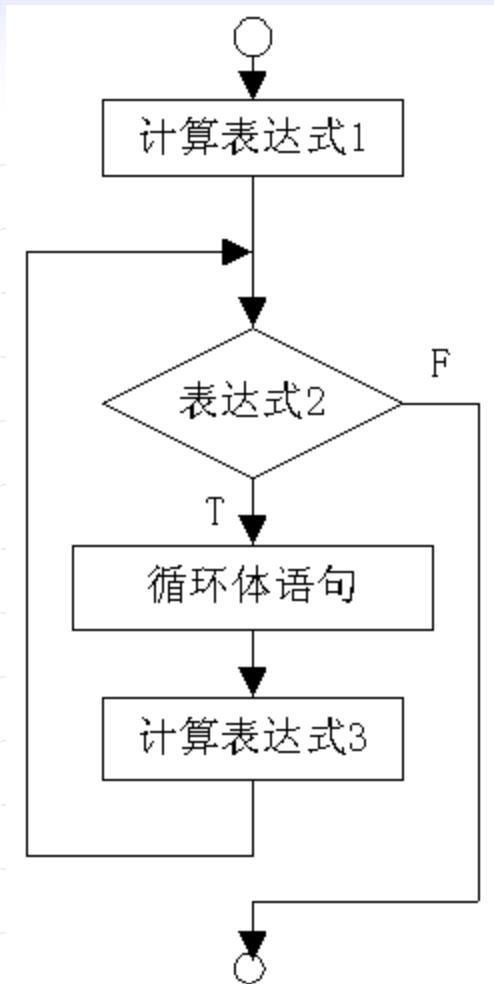
3.4 循环语句

◆ for循环语句

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

◆ 循环结构

- 1、表达式1 → 循环变量初始化；
- 2、表达式2 → 循环控制条件；
- 3、表达式3 → 循环变量的变化；



例1：编程循环输出1-9这9个数字？

运行结果：

```
#include <iostream>
using namespace std;
void main()
{
    for (int i=1;i<10;i++)
        cout<<i<<endl;
    cout<<“循环结束” <<endl;
}
```

1
2
3
4
5
6
7
8
9

循环结束 -39-

例2:编程计算 $1+2+3+\dots+10$ 的值

```
#include <iostream>
using namespace std;
void main()
{
    int sum=0;
    for (int i=1;i<=10;i++)
        sum+=i;
    cout<<sum<<endl;
}
```

运行结果： 55

例3

```
#include <iostream>
#include <iomanip>
using namespace std;
void main(){
    cout<<setfill('*');
    for(int i=2;i<=8;i++)
        cout<<setw(i)<<'\n';
}
```

*
**

3.4 循环语句——for语句

- ◆ 1、<表达式1>可以省略

```
for(int i=1 ; i<=9 ; i++)
```

循环体；



```
int i=1;
```

```
for( ; i<=9 ; i++)
```

循环体；

3.4 循环语句——for语句

◆ 2、<表达式2>可以省略

```
for(int i=1; i<=9; i++)
```

 循环体；



```
for(int i=1; ; i++)
```

```
{
```

```
    if (i<=9)
```

 循环体；

```
    else
```

```
        break; //跳转语句，用来跳出循环
```

```
}
```

3.4 循环语句——for语句

◆ 3、<表达式3>可以省略

```
for(int i=1; i<=9; i++)
```

循环体；



```
for(int i=1; i<=9; )
```

```
{
```

循环体；

```
i++;
```

```
}
```

3.4 循环语句——for语句

◆ 3、<表达式1><表达式3>同时省略

for(int i=1; i<=9; i++)

 循环体;



 int i=1;

 for(; i<=9;)

 {

 循环体;

 i++;

 }

3.4 循环语句——for语句

◆ 5、极端情况，可实现无限循环

```
for( ; ; )
```

```
{
```

 循环体；

```
}
```

例4：判断输出结果？

分析：

```
#include <iostream>
using namespace std;
void main(){
    for(int i=0,j=10,k=0; i<j; i++,j--)
        k+=i*j;
    cout<<k<<endl;
}
```

i	j	k
0	10	0
1	9	9
2	8	25
3	7	46
4	6	70
5	5	跳出

3.4 循环语句

◆ 1、for:

最适合循环次数固定，循环规律非常明显程序。

◆ 2、while:

使用最多最频繁的循环结构，具有普适性。

◆ 3、do_while:

最适合至少循环一次的循环结构。

3.4.4 转向语句

1、break语句

2、continue语句

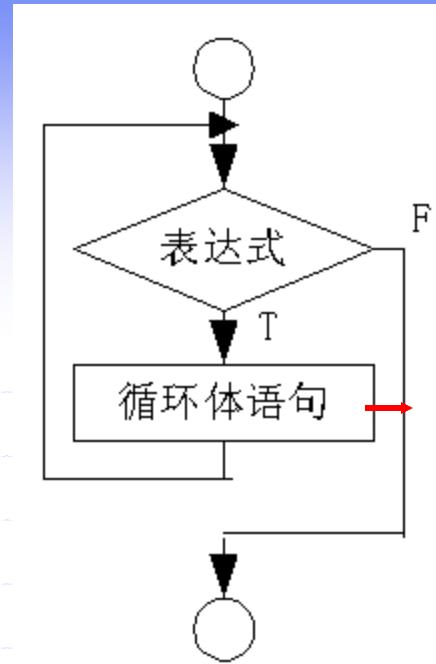
3.4.4 转向语句——break

用途：

常用在while、do-while、for和switch语句中，用来跳出循环或switch语句。

例1：判断输出结果？

```
#include <iostream>
using namespace std;
void main(){
    int sum=0;
    for (int i=1;i<10;i++)
    {
        if (i % 3==0) break;
        sum+=i;
    }
    cout<<sum<<endl;
}
```



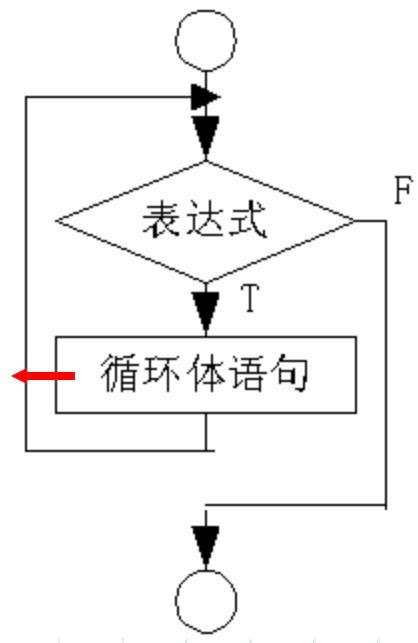
分析

i	sum
1	1
2	3
3	跳出

3.4.4 转向语句——continue

用途

用于循环语句中，结束本次循环，进行下一次循环；即跳过循环体中尚未执行的部分。



例1：判断输出结果？

```
#include <iostream>
using namespace std;
void main(){
    int sum=0;
    for (int i=1;i<10;i++)
    {
        if (i % 3==0) continue;
        sum+=i;
    }
    cout<<sum<<endl;
```

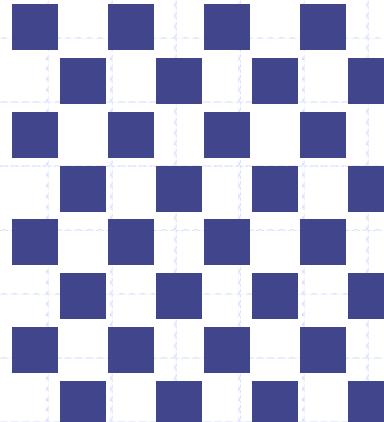
分析	
i	sum
1	1
2	3
3	3
4	7
5	12
6	12
7	19
8	27
9	27
10	跳出

3.5 循环嵌套

3.5 循环嵌套

◆ 编程输出国际象棋棋盘

```
#include <iostream>
using namespace std;
void main()
{
    for (int i=0;i<8;i++)
    {
        if (i%2==0)
            cout<<"■ ■ ■ ■ ";
        else
            cout<<" ■ ■ ■ ■ ";
        cout<<"\n";
    }
}
```



思考

◆ 题目：使用For循环输出如下图形：

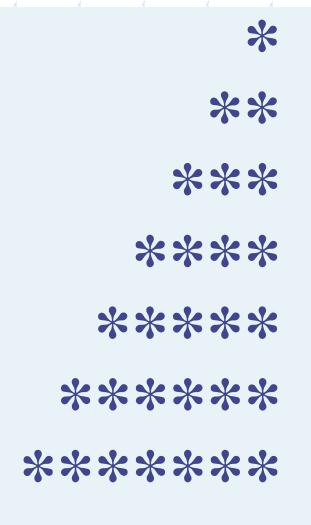
```
*  
**  
***  
****  
*****  
*****  
*****
```

也是循环嵌套的问题：

循环7次，输出7行

每行也要循环：先输出几个空格
再输出几个 “*”

```
#include <iostream>
using namespace std;
void main()
{
    for (int i=1;i<=7;i++)
    {
        for (int j=7; j>i; j--)
            cout<<' ';
        for (j=1; j<=i; j++)
            cout<<'*';
        cout<<endl;
    }
}
```



```
*
**
***
*****
*****
*****
*****
```

3.6 应用举例

3.6 应用举例1

- ◆ 任意输入一个整数，判断是否是素数？

问题：什么是素数？

```
#include <iostream>
using namespace std;
void main(){
    int n,i;
    cin>>n;
    for (i=2;i<n;i++)
        if (n%i==0) break;
    if (i<n)
        cout<<n<<"不是素数！\n";
    else
        cout<<n<<"是素数\n";
}
```

改进：输出2-100之间的素数

```
#include <iostream>
#include <cmath>
using namespace std;
void main()
{
    for(int k=2;k<100;k++)
    {
        int n=sqrt(k);
        for (int i=2;i<=n;i++)
            if (k%i==0) break;
        if (i>n)
            cout<<k<<'\t';
    }
}
```

3.6 应用举例2

◆ 题目

任意输入一个整数，按位翻转输出。例如
输入124，输出421。

```
#include <iostream>
using namespace std;
void main(){
    int i, n, result=0;
    cout<<"请输入任意一个整数: ";
    cin>>n;
    cout<<"按位翻转的结果为: ";
    while (n!=0){
        i = n % 10;
        result=result*10+i;
        n /= 10;
    }
    cout<<result<<endl;
}
```

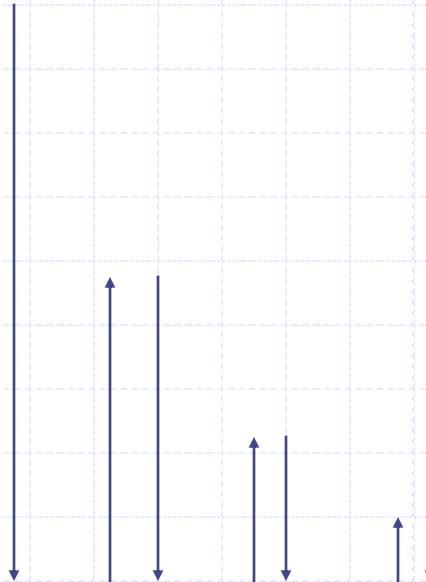
分析:

n	i	result
124	4	4
12	2	42
1	1	421
0		跳出

3.6 应用举例3

◆ 题目

一球从100米高度落下，每次落地后反弹回原高度的一半，再落下。编程求解它在第n次落地时，共经过多少米？



```
#include <iostream>
using namespace std;
void main()
{
    double height=100;
    double sum=height;
    for (int i=1;i<10;i++)
    {
        sum+=height;
        cout<<height<<"\t"<<sum<<"\t" << i<<endl;
        height/=2;
    }
    cout<<sum<<endl;
}
```

3.6 应用举例4

◆ 题目

若一头小母牛，从出生起第四个年头开始每年开始生一头小母牛，按此规律，第n年时有多少头母牛？

$$f(1)=1;$$

$$f(2)=1;$$

$$f(3)=1;$$

$$f(n)=f(n-1)+f(n-3);$$

分析：

$$n=1 \quad 1$$

$$n=2 \quad 1$$

$$n=3 \quad 1+1$$

$$n=4 \quad 1+1+1$$

.....

```
#include <iostream>
using namespace std;
void main()
{
    int n;
    long a=1, b=1, c=1, temp;
    cout <<"please input a value:\n";
    cin >>n;
    for(int i=4; i<=n; i++)
    {
        temp=a+c; a=b; b=c; c=temp;
    }
    cout << c <<endl;
}
```

总结

- ◆ 三种基本结构：顺序、选择和循环
- ◆ 流程图和伪码，帮助我们设计算法、完善算法，增强算法的可读性

本章教学要求：

1. 掌握算法实现的基本结构

- ◆ 顺序结构
- ◆ 选择分支（if, switch）及选择分支的嵌套
- ◆ 循环结构（while, do...while, for）及循环结构的嵌套
- ◆ **break**, **continue**语句

2. 掌握简单的数学算法的C++实现

- ◆ 交换两个变量的数据
- ◆ 判断某个数是否是质数、闰年
- ◆ 简单排序
- ◆ 求连续的N个数的连加和，连乘积等

作业

◆ 第3章习题：3, 4, 9, 10

9. 编程求 $1!+2!+3!+4!+\dots+15!$

```
#include <iostream>
using namespace std;
void main()
{
    unsigned long factorial(1);
    double sum=0;
    for(int i=1; i<=15; i++)
    {
        factorial=1;
        for(int j=1; j<=i; j++)
            factorial= factorial* j;
        cout<<j-1<<"!="<<factorial<<endl;
        sum=sum+factorial;
    }
    cout <<"1!+2!+3!+4!+\dots+15!"<<sum<<endl;
}
```

10. 编程求解问题。若一只小母羊，从出生起第三个年头开始每年生一只小母羊，按此规律，第n年时有多少只母羊？

$$f(1)=1;$$

$$f(2)=1;$$

$$f(n)=f(n-1)+f(n-2);$$

```
#include <iostream>
using namespace std;
void main()
{
    int n;
    long a=1, b=1, temp;
    cout <<"please input a value:\n";
    cin >>n;
    for(int i=3; i<=n; i++)
    {
        temp=a+b; a=b; b=temp;
    }
    cout << b <<endl;
}
```