第5章

C程序结构

C语句

- 控制语句
- 函数调用语句
- 表达式语句
- 空语句
- 复合语句

控制语句

- if()～else～ (条件语句)
- for()～ (循环语句)
- while()～ (循环语句)
- do～ while() (循环语句)
- continue (结束本次循环语句)
- break (中止执行switch或循环语句)
- switch (多分支选择语句)
- goto (转向语句)
- return (从函数返回语句)

函数调用语句

```c
printf("This is a C statement.");
```

```c
double x = sin(y);
```

表达式语句

```
a = 4;
```

```
x = 3 * a + 5;
```

空语句

؛

复合语句

```
{
  z = x + y;
  t = z / 100;
  printf("%f", t);
}
```

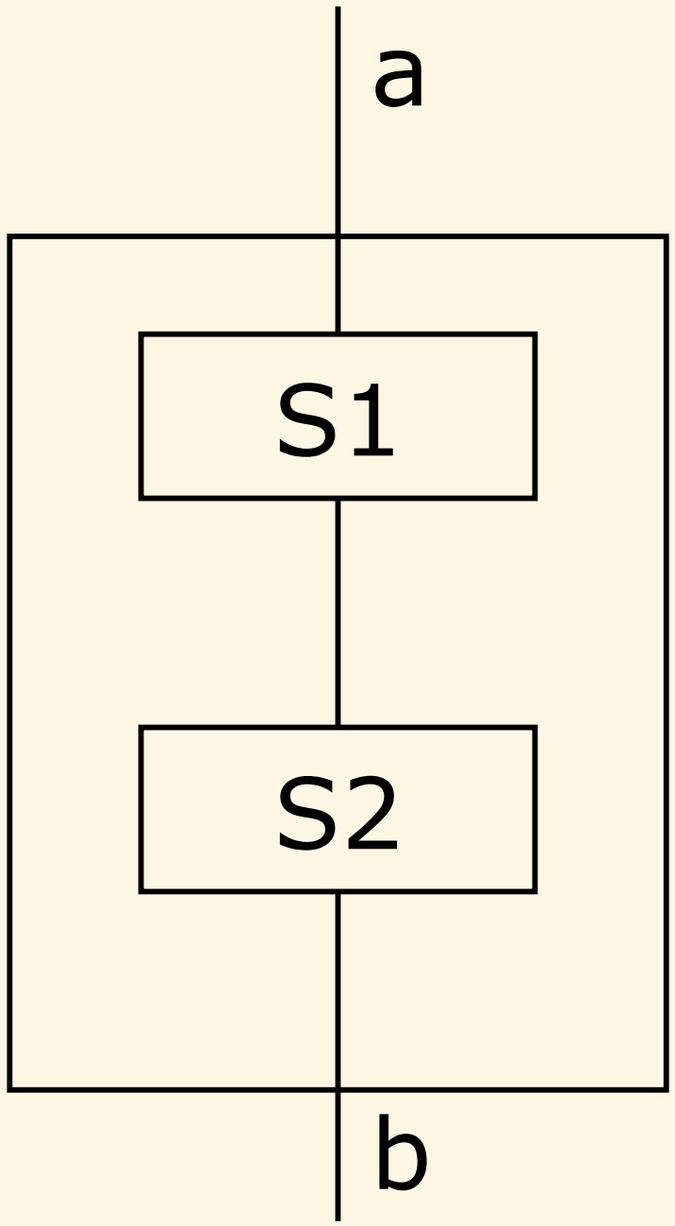
程序设计基础

一般步骤

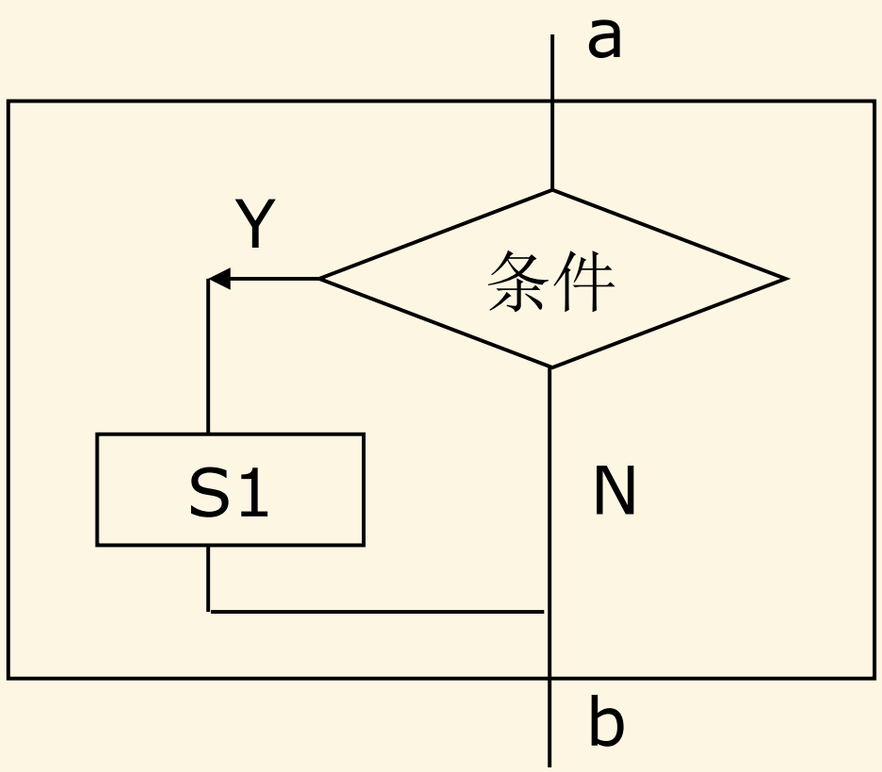
1. 明确问题
2. 分析问题，建立数学模型
3. 确定处理方案，即进行算法设计
4. 绘制流程图
5. 编写程序
6. 调试和测试程序
7. 编写文档资料
8. 程序的运行和维护

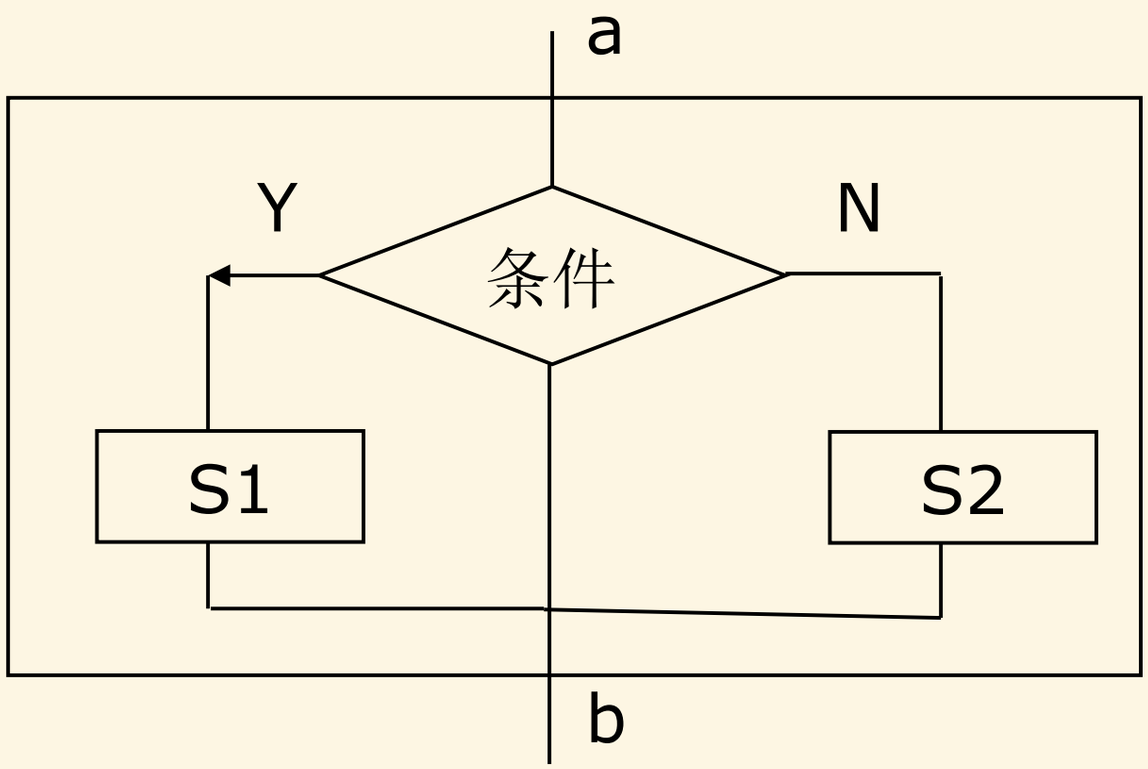
三种基本结构

顺序

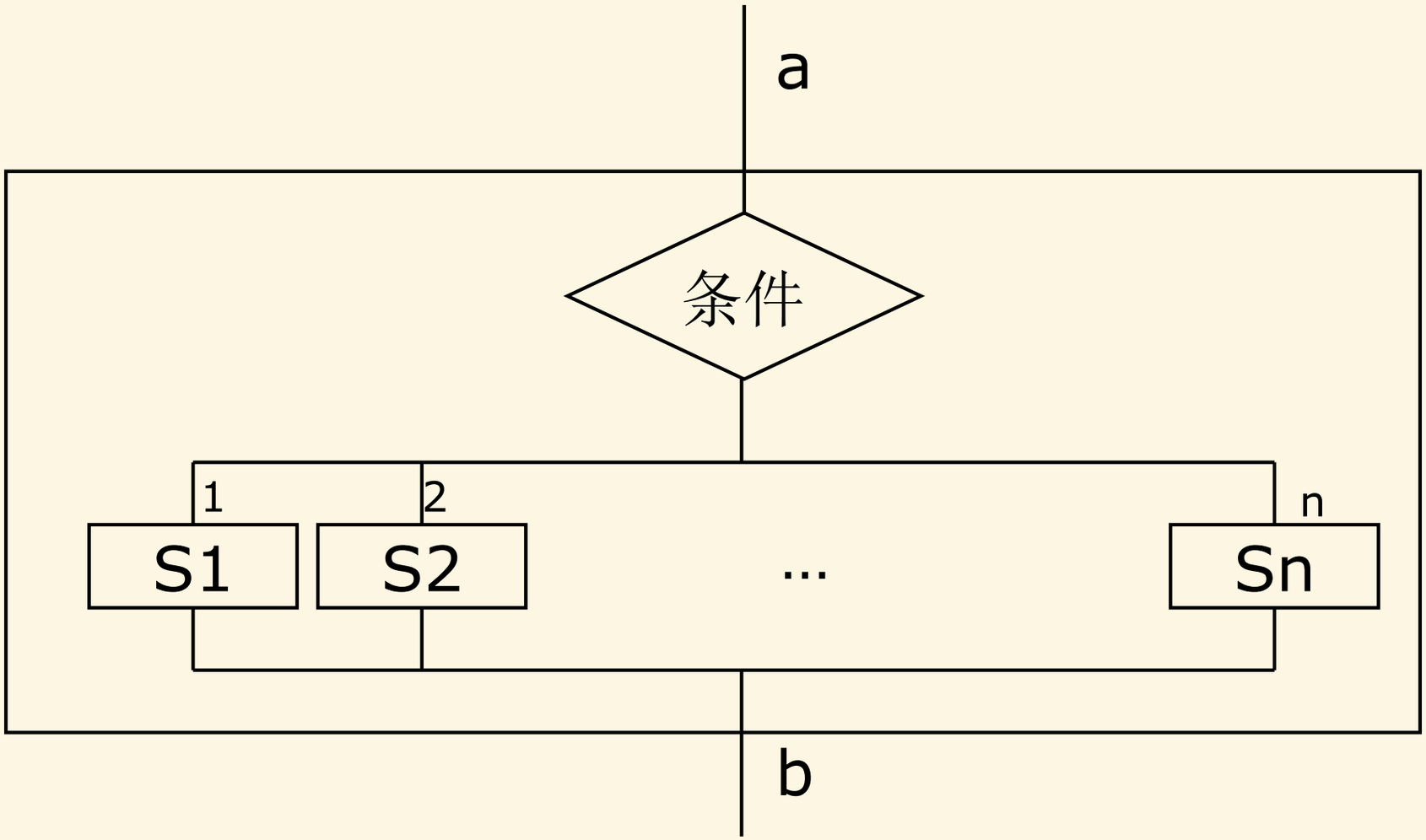


选择/分支

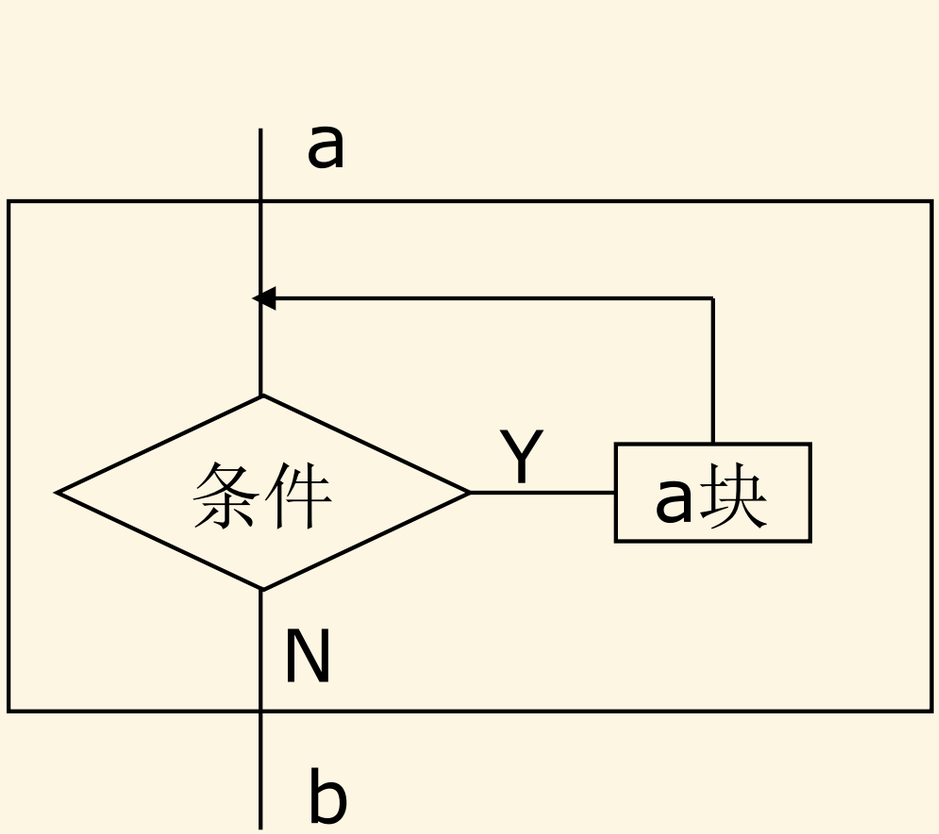


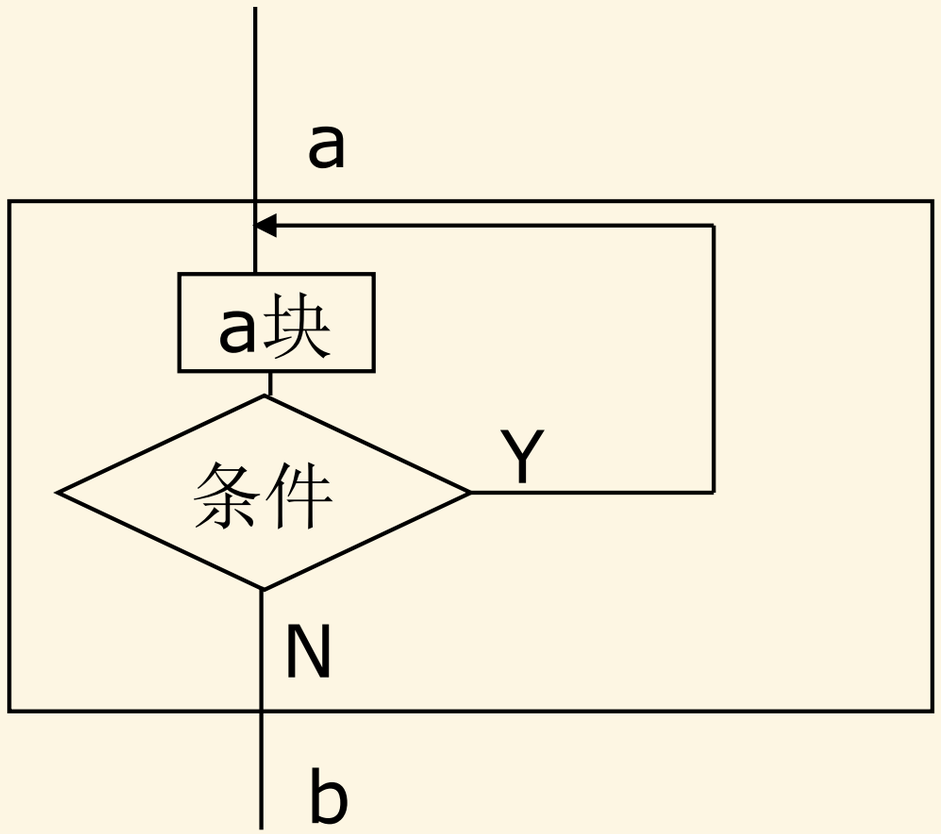


选择/分支



循环

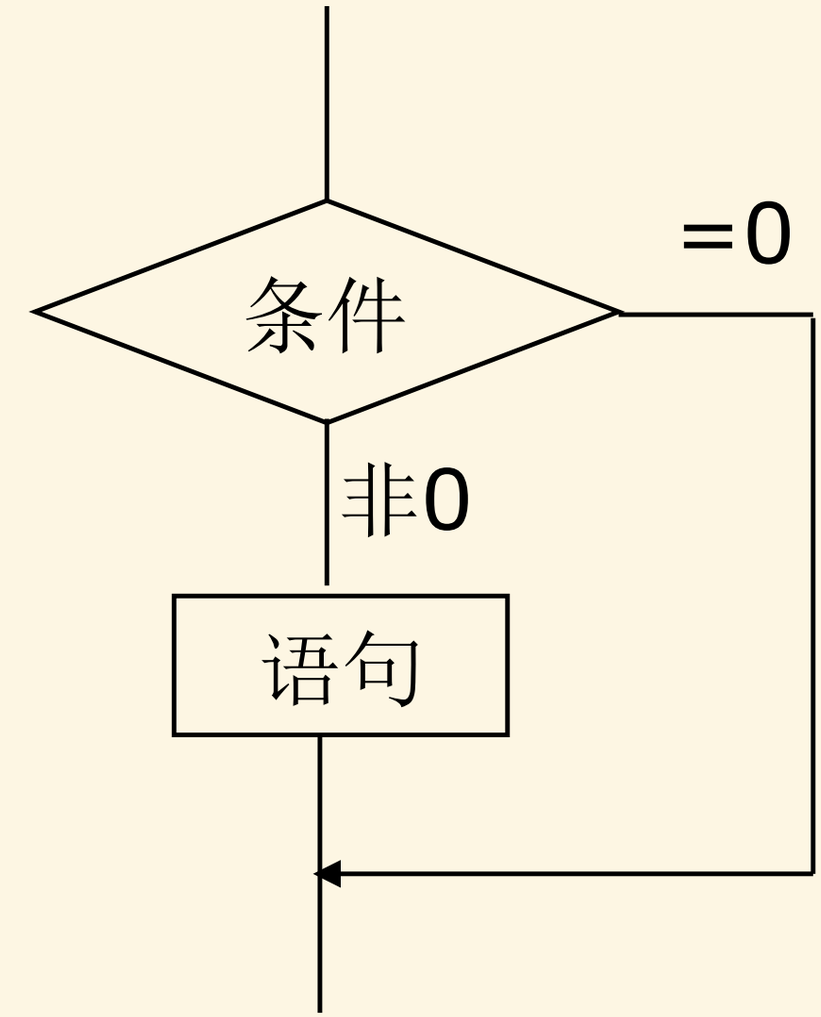




if

```
if (<条件表达式>)
    <语句>;
```

```
if (x > y)
    printf("%d", x);
```



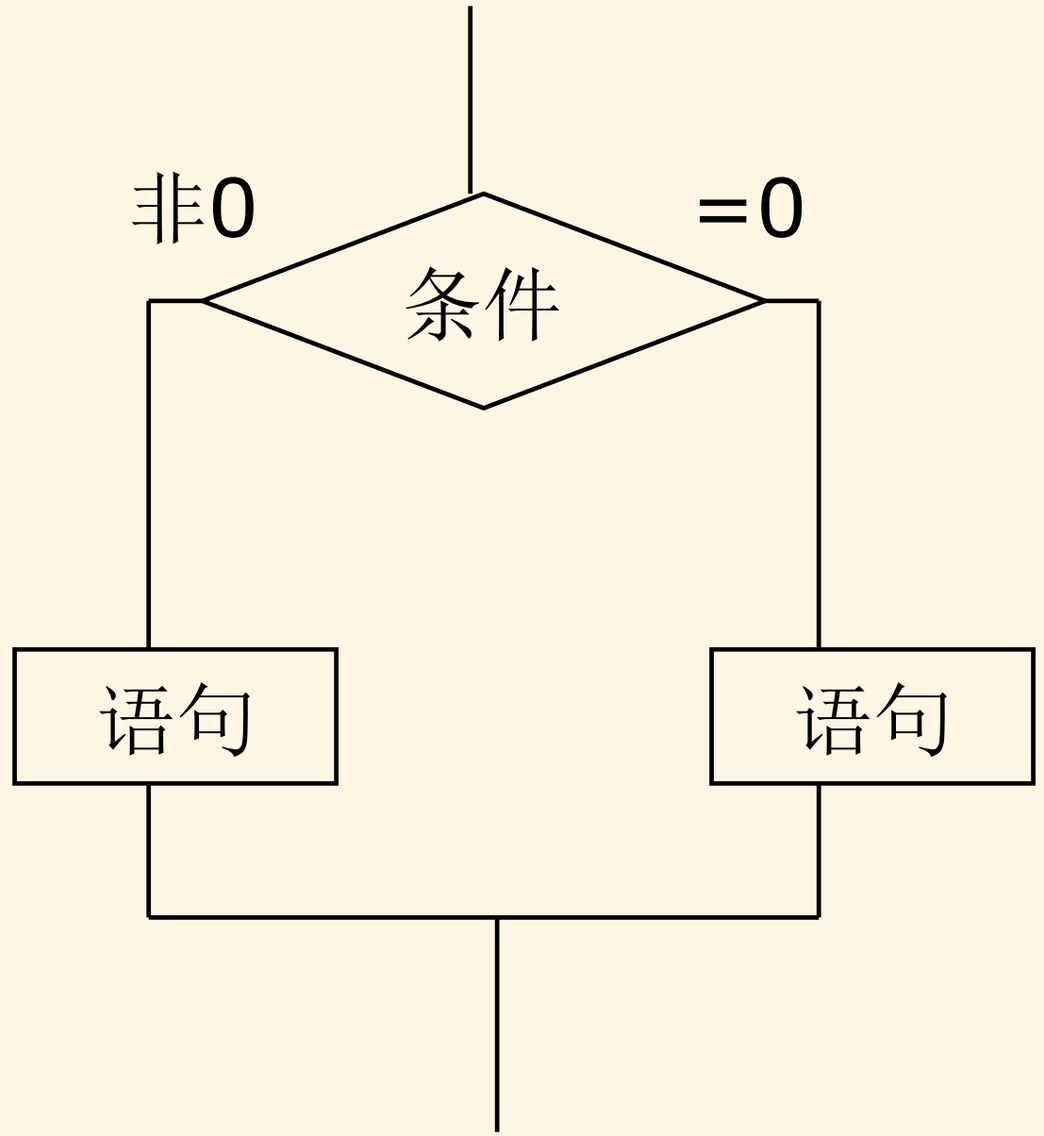
输入三个数，按从小到大顺序输出

```c
#include <stdio.h>

int main()
{
    float a, b, c, t;
    scanf("%f,%f,%f", &a, &b, &c);
    if(a > b) {
        t = a;
        a = b;
        b = t;
    }
    if(a > c) {
        t = a;
        a = c;
        c = t;
    }
    if(b > c) {
        t = b;
        b = c;
        c = t;
    }
    printf("%5.2f,%5.2f,%5.2f\n", a, b, c);

    return 0;
}
```

双路分支

```
if (<条件表达式>)
    <语句1>
else
    <语句2>
```

```
if (x > y)
    printf("%d", x);
else
    printf("%d", y);
```

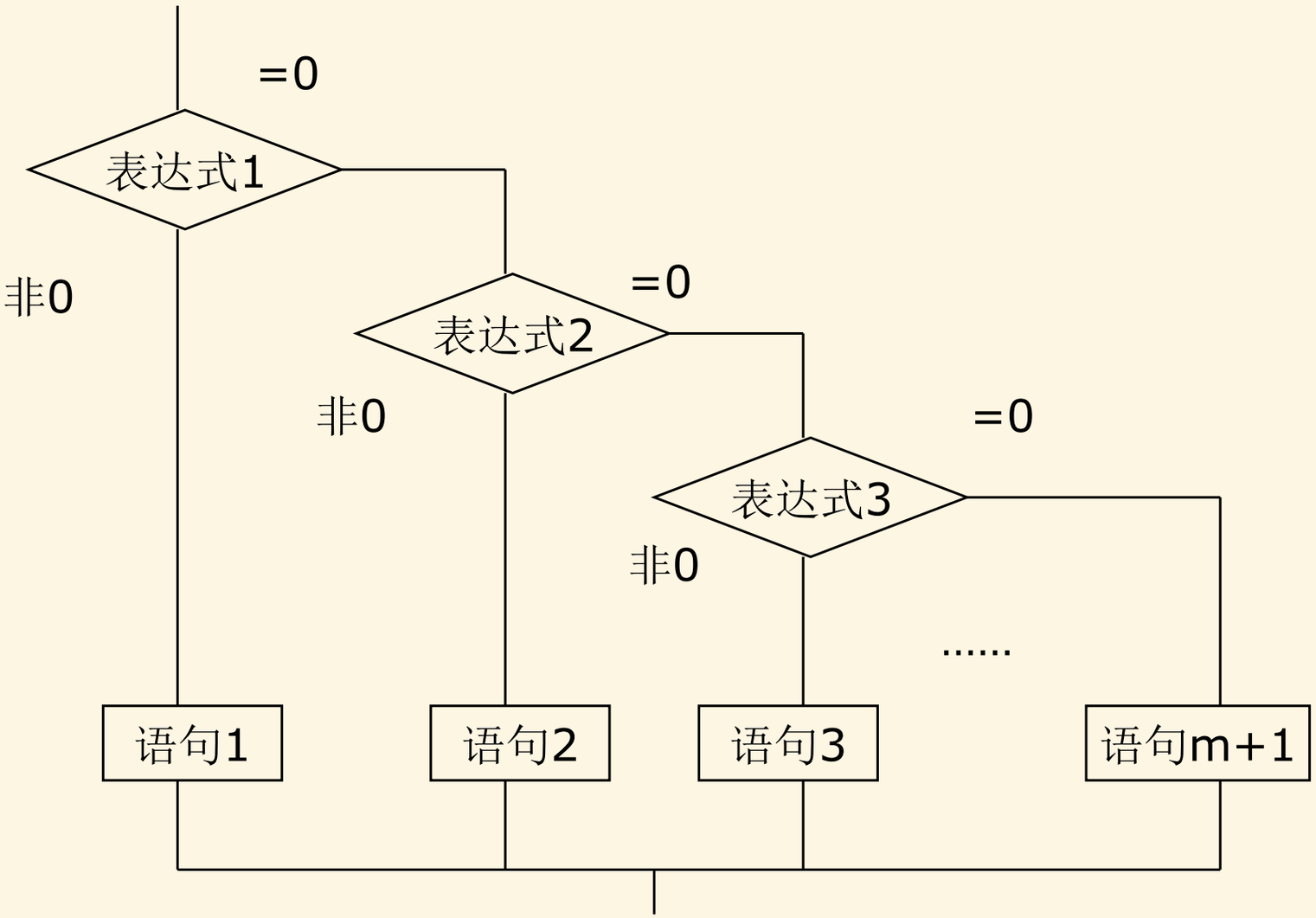


输入三个数，按从小到大顺序输出

```c
#include <stdio.h>

int main()
{
    float a, b, c, t;
    scanf("%f,%f,%f", &a, &b, &c);
    if(a > b) {
        t = a;
        a = b;
        b = t;
    }
    if(a > c) {
        t = a;
        a = c;
        c = t;
    }
    if(b > c)
        printf("%5.2f,%5.2f,%5.2f\n", a, c, b);
    else
        printf("%5.2f,%5.2f,%5.2f\n", a, b, c);

    return 0;
}
```

多路分支

```
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 if (<表达式m>)
    语句m
else
    语句m+1
```



说明

- if后面表达式，一般为逻辑表达式或关系表达式，但也可以是算术表达式和赋值表达式等。
- 第二、三种形式的if语句中，在每个else前面有一分号，整个语句结束处有一分号。
- 在if和else后面只可含一个操作语句。有多个操作语句时，用花括号“{ }”将几个语句括起来成为一个复合语句。

```
if (number > 500)
     cost = 0.15;
else if (number > 300)
     cost = 0.10;
else if (number > 100)
     cost = 0.075;
else if (number > 50)
     cost = 0.05;
else
     cost = 0;
```

```c
#include <stdio.h>

int main()
{
    float a, b, c, t;
    scanf("%f,%f,%f", &a, &b, &c);
    if(a > b) {
        t = a;
        a = b;
        b = t;
    }
    if(a > c)
        printf("%5.2f,%5.2f,%5.2f\n", c, a, b);
    else if(b > c)
        printf("%5.2f,%5.2f,%5.2f\n", a, c, b);
    else
        printf("%5.2f,%5.2f,%5.2f\n", a, b, c);

    return 0;
}
```

嵌套的if语句

```
if (条件1)
    语句1
else
    语句2
```

```
if (条件1)
     if (条件2)
          语句1
     else
          语句2
else
     if (条件3)
          语句3
     else
          语句4
```

就近配对原则

分段函数

$$y = \begin{cases} -1 & x < 0 \\ 0 & x = 0 \\ 1 & x > 0 \end{cases}$$

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    if(x < 0)
        y = -1;
    else if(x == 0)
        y = 0;
    else
        y = 1;
    printf("x=%d,y=%d\n", x, y);
    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    if(x < 0)
        y = -1;
    else if(x == 0)
        y = 0;
    else
        y = 1;
    printf("x=%d,y=%d\n", x, y);
    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    if(x >= 0)
        if(x > 0)
            y = 1;
        else
            y = 0;
    else
        y = -1;
    printf("x=%d,y=%d\n", x, y);

    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    if(x >= 0)
        if(x > 0)
            y = 1;
        else
            y = 0;
    else
        y = -1;
    printf("x=%d,y=%d\n", x, y);

    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    y = -1;
    if(x != 0)
        if(x > 0)
            y = 1;
    else
        y = 0;
    printf("x=%d,y=%d\n", x, y);
    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    y = -1;
    if(x != 0)
        if(x > 0)
            y = 1;
    else
        y = 0;
    printf("x=%d,y=%d\n", x, y);
    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    y = 0;
    if(x >= 0)
        if(x > 0)
            y = 1;
    else
        y = -1;
    printf("x=%d,y=%d\n", x, y);
    return 0;
}
```

```c
#include <stdio.h>

int main()
{
    int x, y;
    scanf("%d", &x);
    y = 0;
    if(x >= 0)
        if(x > 0)
            y = 1;
    else
        y = -1;
    printf("x=%d,y=%d\n", x, y);
    return 0;
}
```

判断闰年

闰年条件

- 能被4整除，但是不能被100整除
- 能被400整除

```c
#include <stdio.h>

int main()
{
    int year, leap;
    scanf("%d", &year);
    if(year % 4 == 0) {
        if(year % 100 == 0) {
            if(year % 400 == 0)
                leap = 1;
            else
                leap = 0;
        } else
            leap = 0;
    } else
        leap = 0;
    if(leap)
        printf("%d is ", year);
    else
        printf("%d is not ", year);
    printf("a leap year\n");
    return 0;
}
```

如何精简上述程序?

switch

```
switch (<表达式>) {
case <常量表达式1>: <语句1>
case <常量表达式2>: <语句2>
……
case <常量表达式n>: <语句n>
default: <语句n+1>;
}
```

说明

- 常量只能是整型、字符型
- 每个case的值不能相同
- default可选
- 比较流程
- 多个case可共用一组执行语句

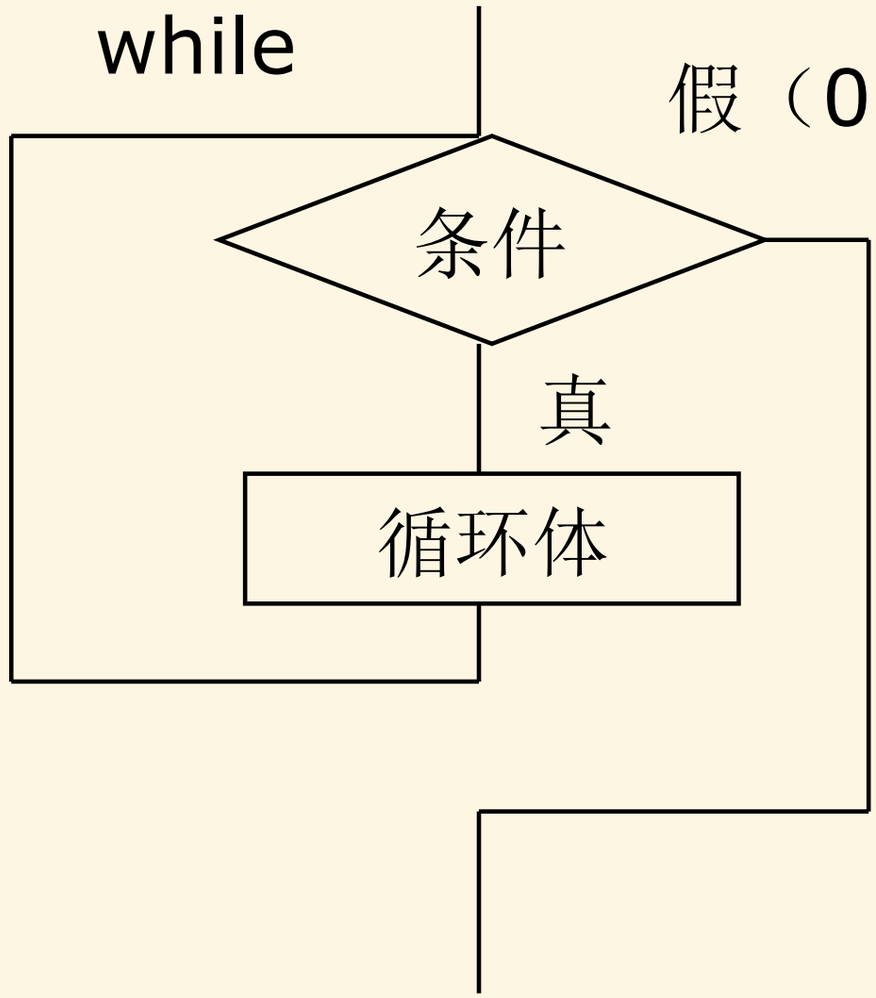
```
switch(score) {
case 'A':
    grade[0]++;
    break;
case 'B':
    grade[1]++;
    break;
case 'C':
    grade[2]++;
    break;
case 'D':
    grade[3]++;
    break;
case 'E':
    grade[4]++;
    break;
default:
    break;
}
```

if v.s. switch

循环

while

```
while (<表达式>) 语句
```



输入一个正整数

```c
#include <stdio.h>

int main()
{
    int n;

    while (scanf("%d", &n), n <= 0)
        ;

    printf("n = %d\n", n);

    return 0;
}
```

輸入一个偶数

如何结合输入结束？

输入

计算1+2+…+n

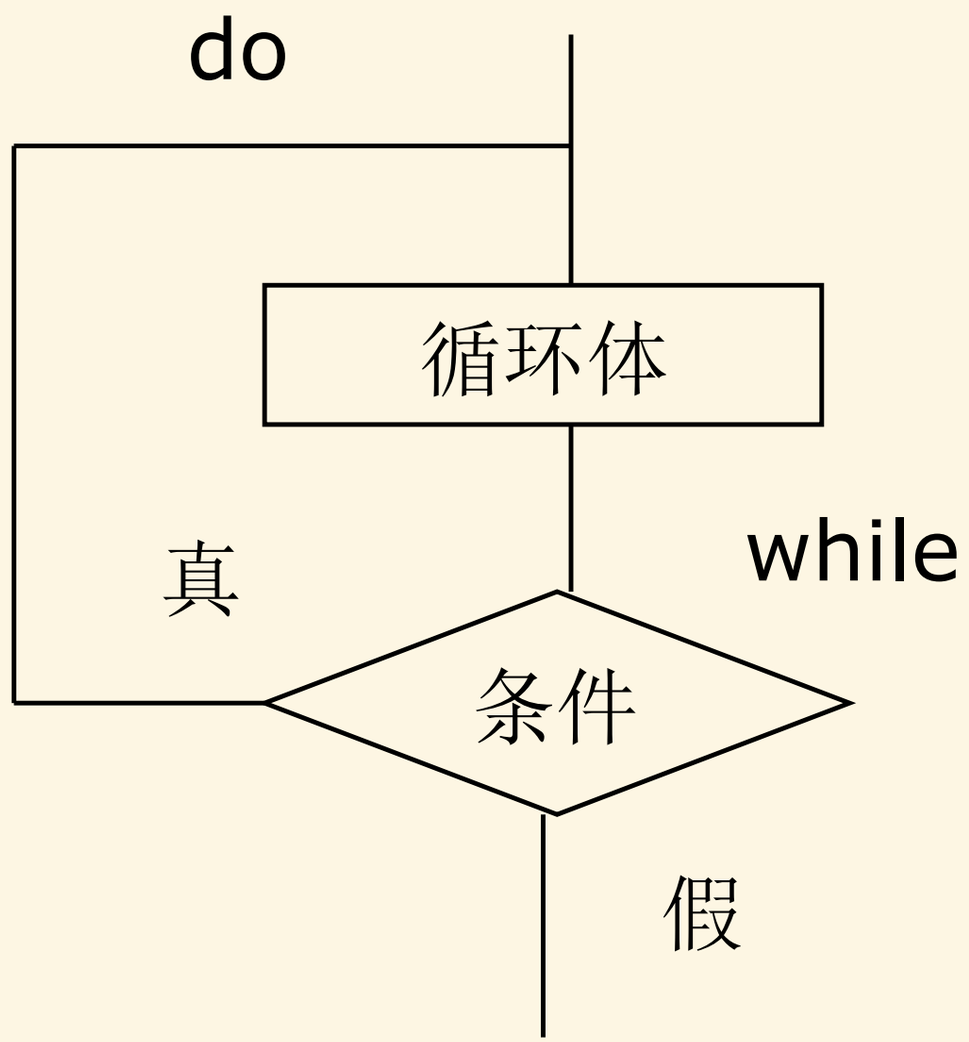
```c
#include <stdio.h>

int main()
{
    int  sum = 0, i = 1, n;

    scanf("%d", &n);

    while(i <= n) {
        sum = sum + i;
        i++;
    }

    printf("%d\n", sum);

    return 0;
}
```

$$1^2 + 2^2 + \ldots + n^2$$

$$1! + 2! + \cdots + n!$$

do-while

```
do <语句> while (<条件>);
```



什么时候用do-while?

求Fibonacci数列

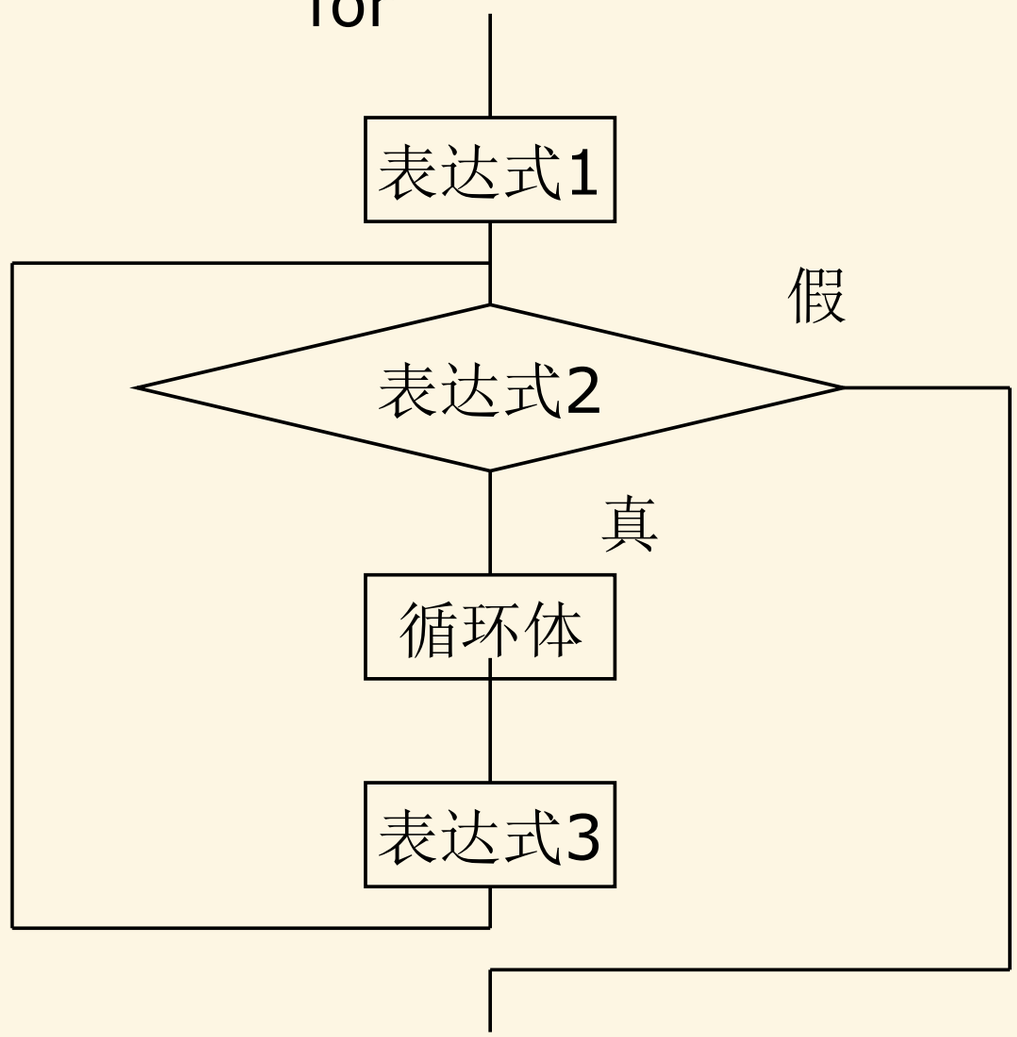
$$F_n = \begin{cases} 1 & n = 1 \\ 2 & n = 2 \\ F_{n-1} + F_{n-2} & n \geq 3 \end{cases}$$

```c
#include <stdio.h>

int main()
{
    int f1, f2;
    int i = 0;
    f1 = 1;
    f2 = 1;
    i = 1;

    do {
        printf("%12d %12d", f1, f2);
        if (i % 2 == 0)
            printf("\n");
        f1 = f1 + f2;
        f2 = f2 + f1;
        i++;
    } while (i <= 20);

    return 0;
}
```

for

for循环

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



执行顺序

(1)先求解表达式1

(2)再求解表达式2，若其值为真(非0)，则执行for语句指定的内嵌语句（循环体)，然后执行下面第(3)步。若表达式2为假(0)，则转到第（5）步

(3)若表达式2为真，在执行完循环体后，求解表达式3

(4)转向上面第（2）步骤继续执行

(5)结束循环，执行for语句下面的一个语句

计算1+2+…+n

```c
#include <stdio.h>

int main()
{
    int i, n, sum = 0;

    scanf("%d", &n);

    for (i = 1; i <= n; i++)
        sum += i;

    printf("sum = %d\n", sum);

    return 0;
}
```

$$1+3+\cdots+(2n-1)$$

for/while/do-while

break

- 跳出switch
- 跳出循环

面积>100的最小圆

```c
#include <stdio.h>

int main()
{
    int r;
    float pi = 3.14159, area;
    for(r = 1; r <= 10; r++) {
        area = pi * r * r;
        if(area > 100)  break;
    }
    printf("%f\n", area);

    return 0;
}
```

continue

- 提前结束本次循环

输入10个数，求正数个数和平均值

```c
#include <stdio.h>

int main()
{
    int i, num = 0, a;
    float sum = 0;
    for(i = 0; i < 10; i++) {
        scanf("%d", &a);
        if(a <= 0)
            continue;
        num++;
        sum += a;
    }
    printf("%d plus integer's sum :%6.0f\n", num, sum);
    printf("Mean value:%6.2f\n", sum / num);

    return 0;
}
```

多重循环

```c
#include <stdio.h>

int main()
{
    int i,j;
    for(i = 1;i <= 3;i++) {
        for(j = 1;j <= 5;j++)
            printf(" %d--%d",i,j);
            printf("\n");
        }

    return 0;
}
```

九九乘法表

求圆周率

$$\frac{\pi}{4} \approx 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \cdots$$

```c
#include <stdio.h>
#include <math.h>

int main()
{
    int s;
    float n, t, pi;
    t = 1;
    pi = 0;
    n = 1.0;
    s = 1;
    while((fabs(t)) >= 1e-6) {
        pi = pi + t;
        n = n + 2;
        s = -s;
        t = s / n;
    }
    pi = pi * 4;
    printf("pi = %.6f\n", pi);

    return 0;
}
```

判断m是否素数

```c
#include <stdio.h>
#include <math.h>

int main()
{
    int m, i, k;
    scanf("%d", &m);
    k = sqrt(m);
    for(i = 2; i <= k; i++)
        if(m % i == 0)
            break;
    if(i >= k + 1)
        printf("%d is a prime number\n", m);
    else
        printf("%d is not a prime number\n", m);

    return 0;
}
```

打印图案

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

```c
#include <stdio.h>

int main()
{
    int i, j, k;
    for(i = 0; i <= 3; i++) {
        for(j = 0; j <= 2 - i; j++)
            printf(" ");
        for(k = 0; k <= 2 * i; k++)
            printf("*");
        printf("\n");
    }
    for(i = 0; i <= 2; i++) {
        for(j = 0; j <= i; j++)
            printf(" ");
        for(k = 0; k <= 4 - 2 * i; k++)
            printf("*");
        printf("\n");
    }
    return 0;
}
```