

第4讲 关系、逻辑运算符和if语句

深圳大学计算机系



本次内容讲解课本第3章



顺序结构

- 顺序结构：之前所写程序为顺序结构，从主函数的第1个语句开始逐个语句执行，直至主函数结束。
- 结构化程序设计：程序由顺序、分支、循环三种基本结构组成。
- 下面学习分支结构—**if**语句。先介绍流程图概念。



一个程序应包括两个方面的内容:

- 对数据的描述: 数据结构(data structure)
- 对操作的描述: 算法(algorithm)

著名计算机科学家沃思提出一个公式:

数据结构 + 算法 = 程序

完整的程序设计应该是:

数据结构 + 算法 + 程序设计方法 + 语言工



算法的概念

广义地说，为解决一个问题而采取的方法和步骤，就称为“算法”。

对同一个问题，可有不同的解题方法和步骤

例：求 $\sum_{n=1}^{100} n$

- 方法1：1+2，+3，+4，一直加到100 **加99次**
- 方法2：100+(1+99)+(2+98)+...+(49+51)+50
= 100 + 49 × 100 + 50 **加51次**



算法的概念

为了有效地进行解题，不仅需要保证算法正确，还要考虑算法的质量，选择合适的算法。希望方法简单，运算步骤少。

计算机算法可分为两大类别：

- **数值运算算法**：求数值解，例如求方程的根、求函数的定积分等。
- **非数值运算**：包括的面十分广泛，最常见的是用于事务管理领域，例如图书检索、人事管理、行车调度管理等。



简单算法举例

例2.1: 求 $1 \times 2 \times 3 \times 4 \times 5$

太繁琐

步骤1: 先求 1×2 , 得到结果2

步骤2: 将步骤1得到的乘积2再乘以3, 得到结果6

步骤3: 将6再乘以4, 得24

步骤4: 将24再乘以5, 得120

如果要求 $1 \times 2 \times \dots \times 1000$, 则要写999个步骤



可以设两个变量：一个变量代表被乘数，一个变量代表乘数。不另设变量存放乘积结果，而直接将每一步骤的乘积放在被乘数变量中。设p为被乘数，i为乘数。用循环算法来求结果，算法可改写：

S1: 使 $p=1$

S2: 使 $i=2$

S3: 使 $p \times i$ ，乘积仍放在变量p中，可表示为：
 $p \times i \Rightarrow p$

S4: 使i的值加1，即 $i+1 \Rightarrow i$ 。

S5: 如果i不大于5，返回重新执行步骤S3以及其后的步骤S4和S5；否则，算法结束。最后得到p的值就是5!的值。



如果题目改为：求 $1 \times 3 \times 5 \times \dots \times 1000$
算法只需作很少的改动：

算法简练

S1: $p \Rightarrow 1$

S2: $i \Rightarrow 3$

S3: $p \times i \Rightarrow p$

S4: $i + 2 \Rightarrow i$

S5: 若 $i \leq 1000$ ，返回S3。否则，结束。



流程图

- 程序 = 算法 + 数据结构

- 对算法的表示:

自然语言描述

伪代码

流程图

N-S图

PDS图



流程图

- 是一种传统的算法表示法，利用几何图形的框代表各种不同性质的操作，用流程线表示算法的执行方向。



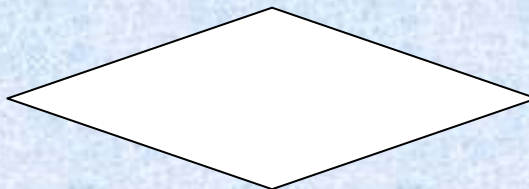
流程图符号

美国国家标准化协会ANSI (American National Standard Institute)

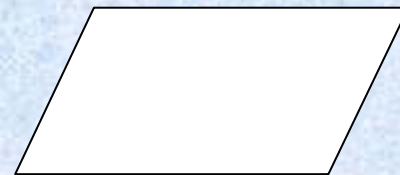
定了一些常用的流程图符号：



起止框



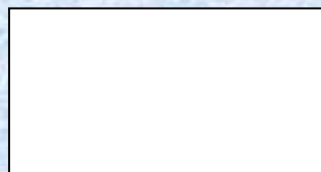
判断框



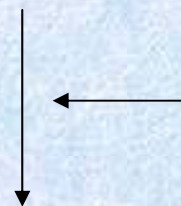
输入输出框



连接点



处理框



流程线

流程图例

对下列编程题，画出流程图描述求解算法。

求三角形面积：

$$A = \sqrt{s(s-a)(s-b)(s-c)}$$

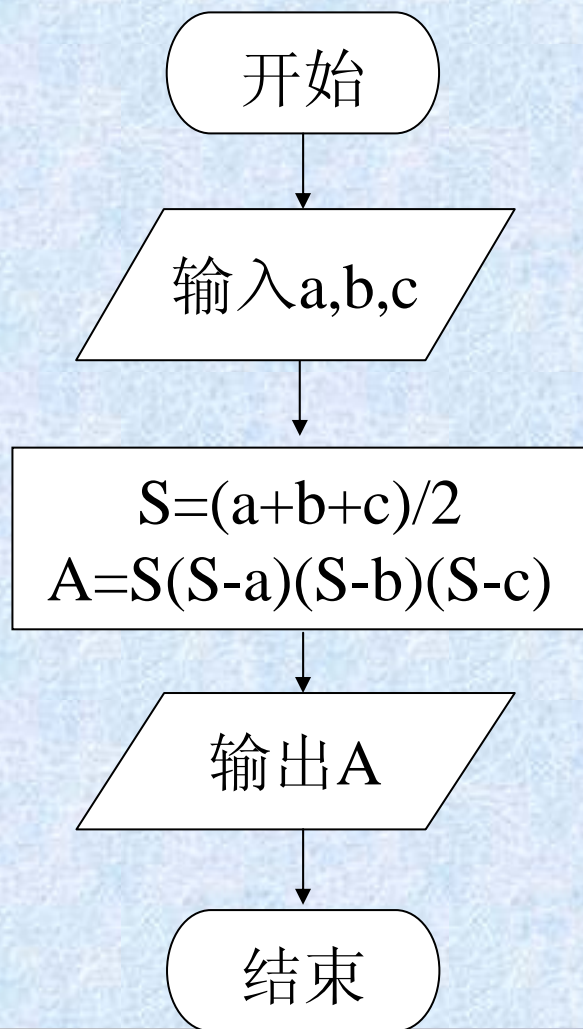
$$s = (a + b + c) / 2$$

边长a, b, c从屏幕读入。



流程图举例

1、



关系运算符、逻辑运算符

1. 关系运算符和关系表达式

➤ 关系运算符

关系运算符	含义	优先级	结合性
>	大于	这些关系运算符等优先级，但比下面的优先级高	
>= (>和=之间没有空格)	大于或等于		
<	小于		

注意：

- C语言用0表示假，非0表示真。
- 一个关系表达式的值不是0就是1，0表示假，1表示真。

➤ 关系表达式

用关系运算符连接起来的式子称为**关系表达式**。

关系表达式的一般形式为：**表达式 关系运算符 表达式**

例： $a + b > c - d$ $x > 3 / 2$ $'a' + 1 < c$ $-i - 5 * j == k + 1$



➤ 关系运算符的优先级

算术运算符	↑ 高 ↓ 低
移位运算符	
<u>关系运算符</u>	
&、 、^	
赋值运算符	

例如:

$c > a + b$ 等价于: $c > (a + b)$

例: `int a = 3, b = 2, c = 1, d, f;`

`a == b < c` 等价于: `(a == b) < c`
`a > b` //表达式值1

`a > 2 < (a > b) == c` 等价于: `(a > 2) < ((a > b) == c)` //表达式值1

`a & 4 > b | c` 等价于: `(a & 4) > (b | c)`
`b + c < a` //表达式值0

`d = a > b` //d = 1

`f = a > b > c` //f = 0



➤ 关系运算注意:

例 若 $a = 0$; $b = 0.5$; $x = 0.3$;
则 $a \leq x \leq b$ 的值为 **0**

例 `int i = 1, j = 7, a;`
`a = i + (j % 4 != 0);`
则 $a = \mathbf{2}$

例 $5 > 2 > 7 > 8$ 在 C 中是
允许的, 值为 **0**

例 $'a' > 0$ 结果为 **1**
 $'A' > 100$ 结果为 **0**

例 应避免对 **实数** 作相等或不等的判断
如 $1.0/3.0*3.0 == 1.0$ 结果为 **0**
可改写为: `fabs(1.0/3.0*3.0-1.0) < 1e-6`



2. 逻辑运算符和逻辑表达式

➤ 逻辑表达式

用逻辑运算符连接起来的式子称为**逻辑表达式**。

逻辑表达式的一般形式为：**表达式 逻辑运算符 表达式**

例： a < b

例：

a && b < c、x > 10 || x < -10、!x && !y

➤ 逻辑运算符的优先级

!、~、++、--、sizeof
算术运算符
移位运算符
关系运算符
&、|、^
&&、||
赋值运算符

a = 4; b = 5;

!a

//值为0

a && b

//值为1

a || b

//值为1

!a || b

//值为1

4 && 0 || 2

//值为1

5 > 3 && 2 || 8 < 4 - !0

//值为1

'c' && 'd'

//值为1

已知：

问：

答：

a = 4; b = 5;
c = b > 3 && 2 || 8 < b - !a; 的值?

c 的值为1。 等价于：c = (((b > 3) && 2) || (8 < (b - (!a))));

➤ 逻辑运算注意:

逻辑表达式求解时，并非所有的逻辑运算符都被执行，只是在必须执行下一个逻辑运算符才能求出表达式的解时，才执行该运算符。

例 `a && b && c` //只在a为真时，才判别b的值；
 只在a、b都为真时，才判别c的值

例 `a || b || c` //只在a为假时，才判别b的值；
 只在a、b都为假时，才判别c的值

例 `a = 1; b = 2; c = 3; d = 4; m = 1; n = 1;`
`(m = a > b) && (n = c > d)` //结果: `m = 0, n = 1`



运算符优先级

- **!**
- ***, /, %**
- **+, -**
- **<, <=, >, >=**
- **==, !=**
- **&&**
- **||**

详细内容参见教材P31



练习

编程：求解一元二次方程 $ax^2 + bx + c = 0$
 a, b, c 的值由键盘输入。

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

若 $a=0$ 有解吗？

若 $b^2 - 4ac < 0$ 有实根吗？

exec37.cpp



选择结构的程序设计

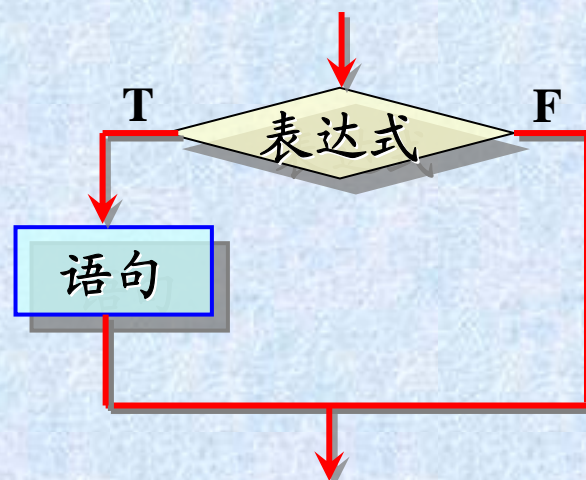
1. if语句

➤ 简单if语句形式

- 格式:

```
if (表达式)  
    语句;
```

- 执行流程:



➤ if-else形式

例如: 下面的程序段是输入两个整数, 输出其中的大数。

```
int a, b, max;
```

```
printf("Input two numbers: ");
```

```
cin >> a >> b;
```

```
max = a;
```

```
if (max < b)
```

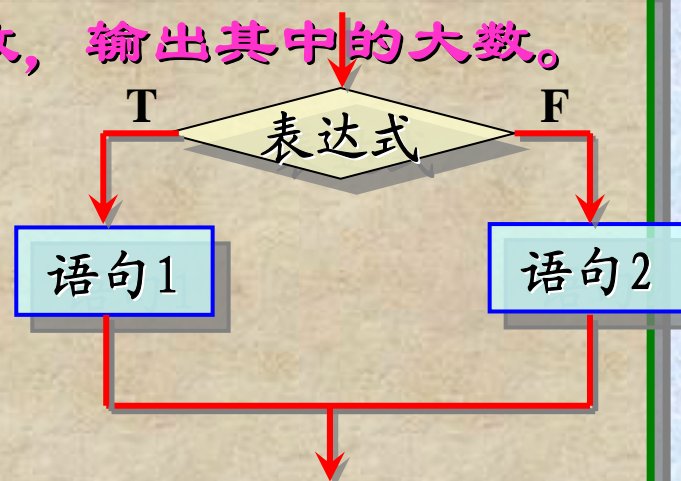
```
    max = b;
```

```
printf("max = %d", max);
```

```
if (表达式)  
    语句1;
```

```
else  
    语句2;
```

- 执行流程:



1. if语句

➤ if-else-if形式

● 格式:

例如：下面的程序段是判断输入字符的种类。

```
char c;  
printf ("Enter a character: ");  
c = getchar ( );  
if (c < 0x20)  
    printf ("The character is a control character\n");  
else if (c >= '0' && c <= '9')  
    printf ("The character is a digit\n");  
else if (c >= 'A' && c <= 'Z')  
    printf ("The character is a capital letter\n");  
else if (c >= 'a' && c <= 'z')  
    printf ("The character is a lower letter\n");  
else  
    printf ("The character is other character\n");
```


➤ if语句注意事项

(1) if语句后面的表达式必须用括号括起来。

(2) 表达式通常是逻辑表达式或关系表达式，也可以是

可以是一个变量
就是if语句
必须注意的
否相
=。

例: if x > 0
printf ("x > 0");

例: if (x > 0)
printf ("x > 0");

例:

```
#include <stdio.h>
void main ( )
{
    int x = 0;
    if (x == 0)
        printf ("x = 0\n");
    else
        printf ("x != 0\n");
}
```

运行结果: x = 0

例:

```
#include <stdio.h>
void main ( )
{
    int x = 0;
    if (x = 0)
        printf ("x = 0\n");
    else
        printf ("x != 0\n");
}
```

运行结果: x != 0

练习

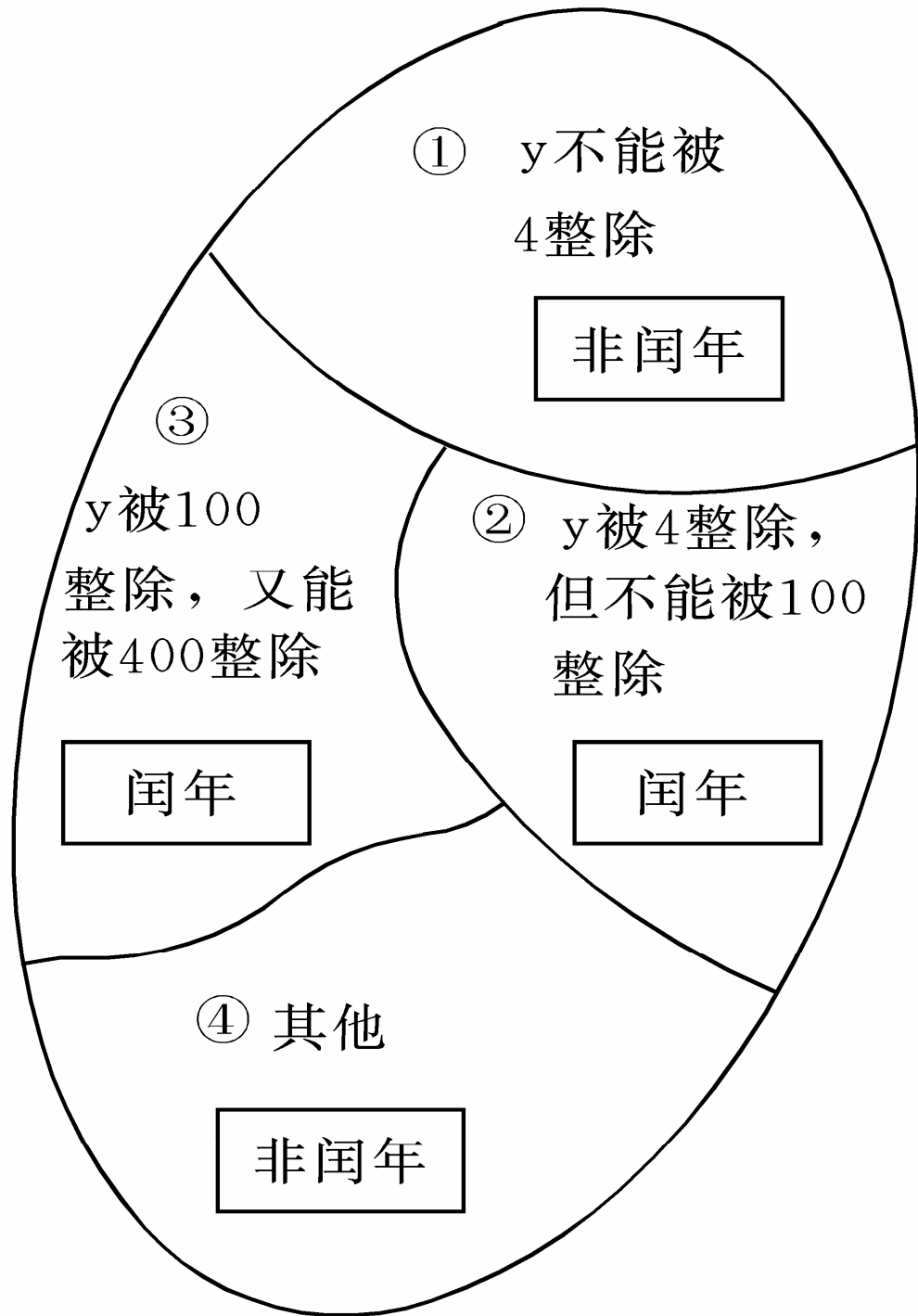
编程：输入年，输出1，0分别代表该年是否闰年。/*exec41.cpp*/

闰年：能被**400**整除或者能被**4**整除但不能被**100**整除。

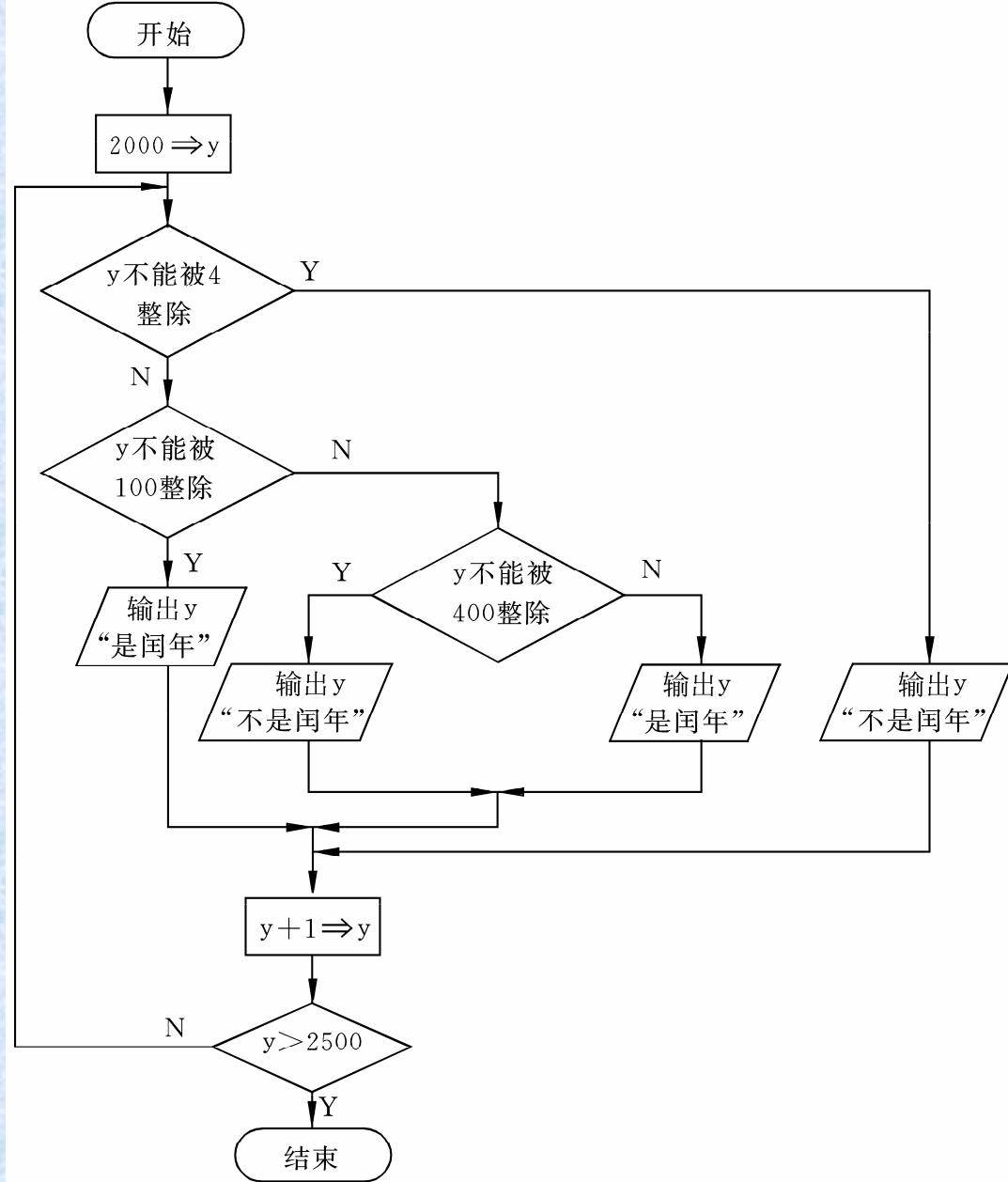
exec41.cpp



闰年：能被
400整除或者
能被4整除但
不能被100整
除。



判定2000~2500年
中的每一年是否闰
年，将结果输出。



条件运算符与表达式

- 一般形式: **expr1 ? expr2 : expr3**

- 执行过程

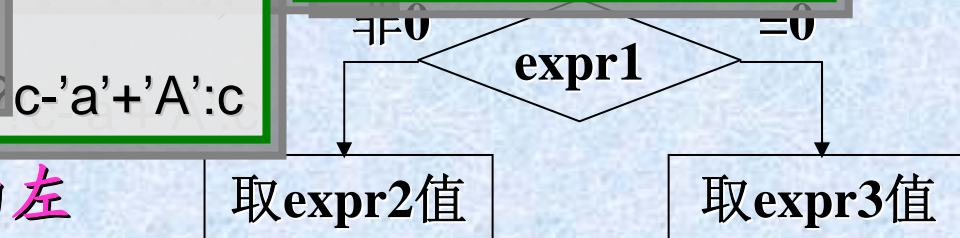
- 功能: 相当于条件语句

例 if (a>b)
 printf("%d",a);
else
 printf("%d",b);

输出结果 (假设输入为w) :

ch1 = v, ch2 = x

printf("%d",a>b?a:b);



- 结合方向: **自右向左**

如 **a > b ? a : c > d ? c : d** \Leftrightarrow **a > b ? a : (c > d ? c : d)**

- expr1、expr2、expr3类型可不同, 表达式值取较高的类型

例 x?'a':'b' //x=0,表达式值为'b'; x≠0,表达式值为'a'
x>y?1:1.5 //x>y ,值为1.0; x<y ,值为1.5



条件运算符与表达式

例如：下面的程序段是输入两个整数，输出其中的大数。

```
int a, b, max;  
printf("input two numbers: ");  
cin>>a>>b;  
a>b? max = a: max = b;  
printf("max = %d", max);
```



复合语句

- 语法:

{

语句1;

...

语句n;

}

语义: {...}中的语句作为整体一起执行。



if中的复合语句

```
if ( 表达式 )  
    语句;
```

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

上图中的语句，也可以是复合语句。即若分支中有多个语句执行，注意加{ ... }。



练习

编程：求解一元二次方程 $ax^2 + bx + c = 0$
a, b, c的值由键盘输入。

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

若 **a=0** ,一元一次方程.

若 $b^2 - 4ac < 0$ 无实根
不输出运算结果.

exec42.cpp



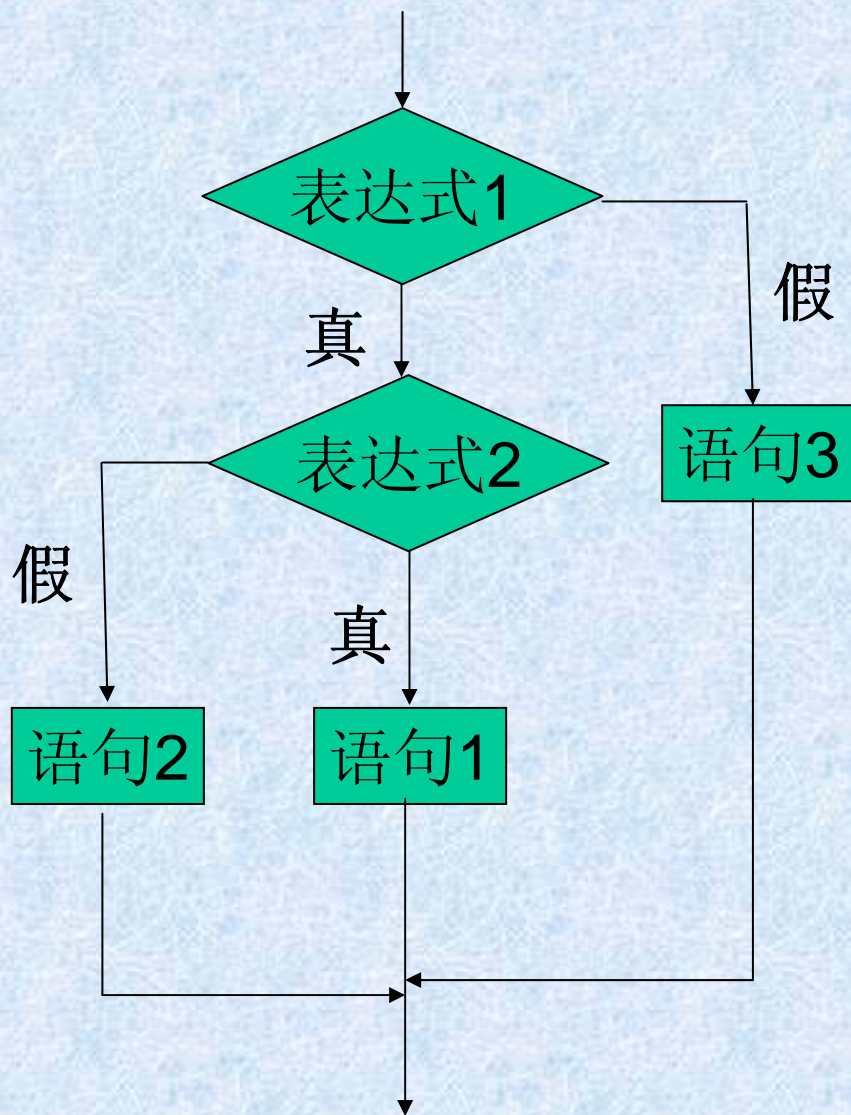
if 语句嵌套

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

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

语句也可以是if、**if...else**语句，形成if语句的嵌套。





➤ if语句嵌套:

● 一般形式:

```
if (expr1)
    if (expr2)
        statement1
    else
        statement2
```

内嵌if

```
if (expr1)
    statement1
else
    if (expr3)
        statement3
    else
        statement4
```

内嵌if

```
if (expr1)
    if (expr2)
        statement1
else
    statement3
```

内嵌if

```
if (expr1)
    if (expr2) statement1
    else      statement2
else
    if (expr3) statement3
    else      statement4
```

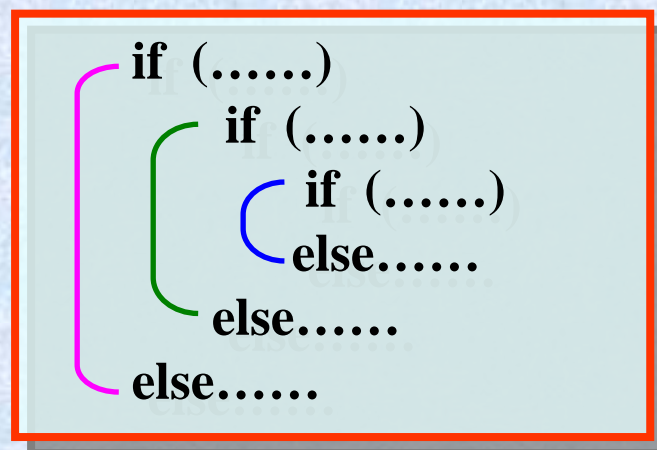
内嵌if

内嵌if



● if_else 配对原则

C语言规定，在缺省{ }时，else总是和它上面离它最近的未配对的if配对



练习

判断输入数据的符号属性。

输入 x ，打印出-1、0、1，分别对应负数、零、正数。

exec43.cpp



if 语句嵌套练习

有一个函数 $y = \begin{cases} -1 & (x < 0) \\ 0 & (x = 0) \\ 1 & (x > 0) \end{cases}$, 编一程序, 输入一个 x 值, 输出 y 值。

算法1:

输入 x
若 $x < 0$, 则 $y = -1$
若 $x = 0$, 则 $y = 0$
若 $x > 0$, 则 $y = 1$
输出 y

算法2:

输入 x
若 $x < 0$, 则 $y = -1$
否则:
若 $x = 0$, 则 $y = 0$
若 $x > 0$, 则 $y = 1$
输出 y



练习

程序段有四个，请判断哪个是正确的？

程序1:

```
If(x<0)
    Y=-1;
else
    If(x==0) y=0;
    Else y=1;
```

正
确

程序2:

```
if(x>=0)
    if(x>0) y=1;
    else y=0;
else y=-1;
```

正
确

程序3:

```
y=-1;
If(x!=0)
    If(x>0) y=1;
else y=0;
```

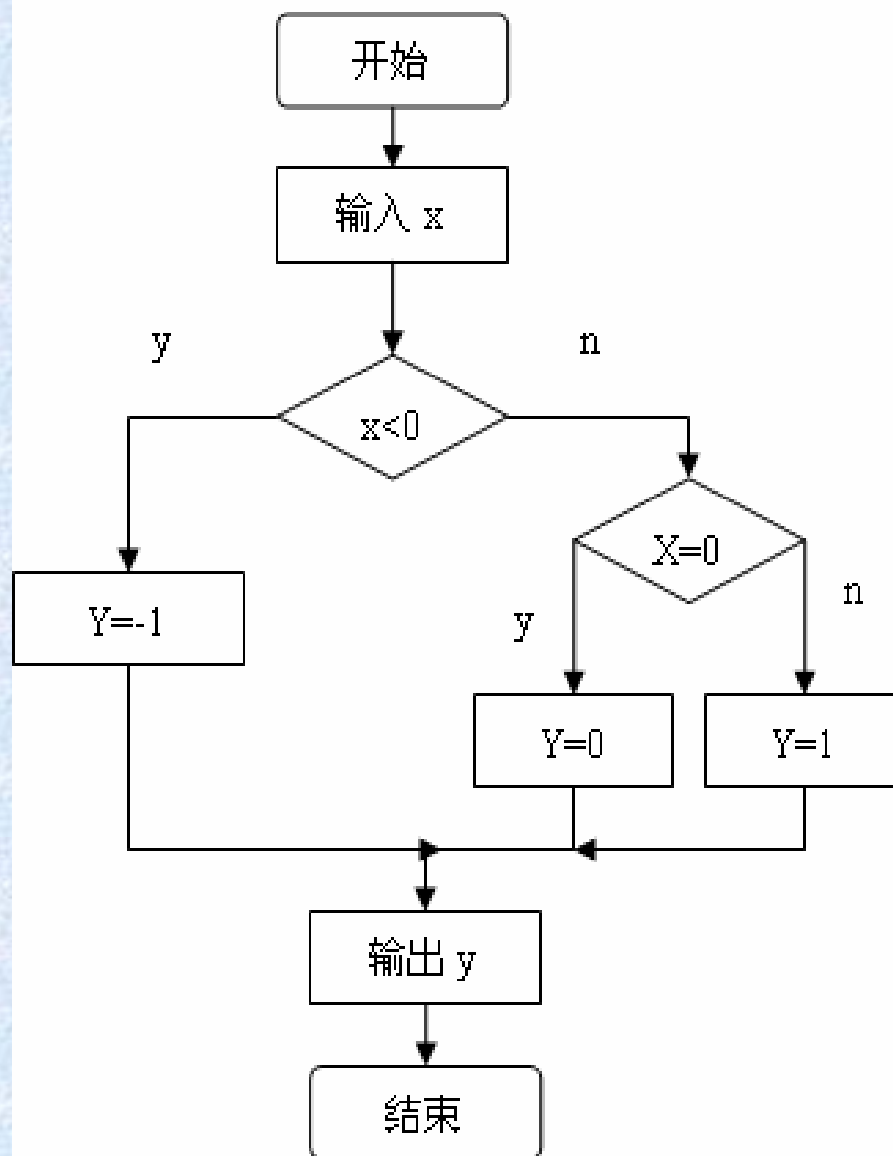
程序4:

```
y=0;
if(x>=0)
    if(x>0) y=1;
else y=-1;
```



判断输入数据的符号属性。
输入 x ，打印出-1、0、1，分别对应负数、零、正数。

算法2 流程图



exec43-a.cpp



练习

分数转换：输入百分制分数，按下列规则转换为对应字母。

90 ~ 100 \Rightarrow A

80 ~ 89 \Rightarrow B

70 ~ 79 \Rightarrow C

60 ~ 69 \Rightarrow D

0 ~ 59 \Rightarrow F

exec44.cpp



练习

- 编程：求解一元二次方程 $a x^2 + b x + c = 0$
- a, b, c 的值由键盘输入。

$$x_1 = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad x_2 = \frac{-b - \sqrt{b^2 - 4ac}}{2a}$$

基本的算法：

- ① $a = 0$ ，不是二次方程。
- ② $b^2 - 4ac = 0$ ，有两个相等实根。
- ③ $b^2 - 4ac > 0$ ，有两个不等实根。
- ④ $b^2 - 4ac < 0$ ，有两个共轭复根。

exec45.cpp



部分预定义的字符函数

- `#include <stdio.h>`
- `getchar()`, `putchar(c)`
- `#include <ctype.h>`
- `islower(c)`, `isupper(c)`, `isalpha(c)`,
`isdigit(c)`, `isalnum(c)`,
`isspace(c)`,.....
`toupper(c)`, `tolower(c)`



练习 p53 --3.5

编程:根据以下函数关系,对输入的每个x, 求y。

$$y=x*(x+2) \quad 2<x\leq 10$$

$$y=2*x \quad -1<x\leq 2$$

$$y=x-1 \quad x\leq -1$$

exec46.cpp



本节知识回顾

- 掌握关系运算符、逻辑运算符；
- 掌握C中真、假的表示；
- 掌握if的单分支、多路分支语法结构；
- 理解复合语句；
- 掌握if的嵌套结构；





希望大家能学出好成绩,我们一起努力!

谢谢大家!