



选择结构实践专题

如果从键盘输入 “Hello World”，程序的输出是什么

```
#include<stdio.h>
int main()
{
    char a[100]={0};
    scanf("%s",a);
    printf("%s",a);
    return 0;
}
```



选择结构实践专题

高级版有奖竞猜

你心里默想一个小于50的数，我问你：在以下的6张卡片中，哪几张卡片里有你想的那个数？之后，我将不费吹灰之力就能猜出这个数！

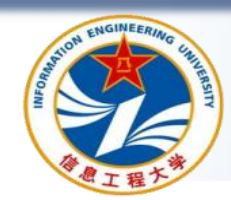


选择结构实践专题

高级版有奖竞猜

0#卡片

1	3	5	7	9
11	13	15	17	19
21	23	25	27	29
31	33	35	37	39
41	43	45	47	49



选择结构实践专题

高级版有奖竞猜

1#卡片

2	3	6	7	10
11	14	15	18	19
22	23	26	27	30
31	34	35	38	39
42	43	46	47	50



选择结构实践专题

高级版有奖竞猜

2#卡片

4	5	6	7	12
13	14	15	20	21
22	23	28	29	30
31	36	37	38	39
44	45	46	47	



选择结构实践专题

高级版有奖竞猜

3#卡片

8	9	10	11	12
13	14	15	24	25
26	27	28	29	30
31	40	41	42	43
44	45	46	47	



选择结构实践专题

高级版有奖竞猜

4#卡片

16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	48	49	50	



选择结构实践专题

高级版有奖竞猜

5#卡片

32	33	34	35	36
37	38	39	40	41
42	43	44	45	46
47	48	49	50	



选择结构实践专题

0#卡片

1	3	5	7	9
11	13	15	17	19
21	23	25	27	29
31	33	35	37	39
41	43	45	47	49

1#卡片

2	3	6	7	10
11	14	15	18	19
22	23	26	27	30
31	34	35	38	39
42	43	46	47	50

2#卡片

4	5	6	7	12
13	14	15	20	21
22	23	28	29	30
31	36	37	38	39
44	45	46	47	

37

3#卡片

8	9	10	11	12
13	14	15	24	25
26	27	28	29	30
31	40	41	42	43
44	45	46	47	

4#卡片

16	17	18	19	20
21	22	23	24	25
26	27	28	29	30
31	48	49	50	

5#卡片

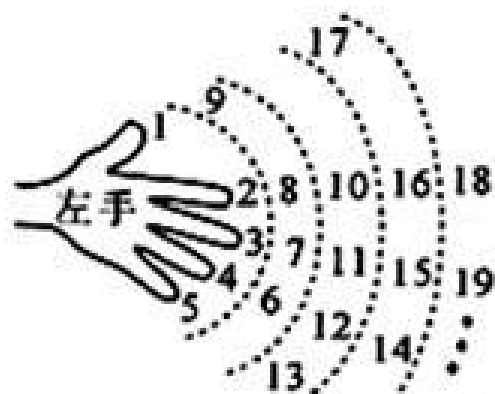
32	33	34	35	36
37	38	39	40	41
42	43	44	45	46
47	48	49	50	

数数的手指（编程竞赛题）

一个小女孩正在用左手手指数数，从1到1000，她从拇指算作1开始数起，然后食指为2，中指为3，无名指为4，小指为5。接下来调转方向，无名指算作6，中指7，食指为8，大拇指为9，如此反复。

问：最后结束时停在哪儿手指上？

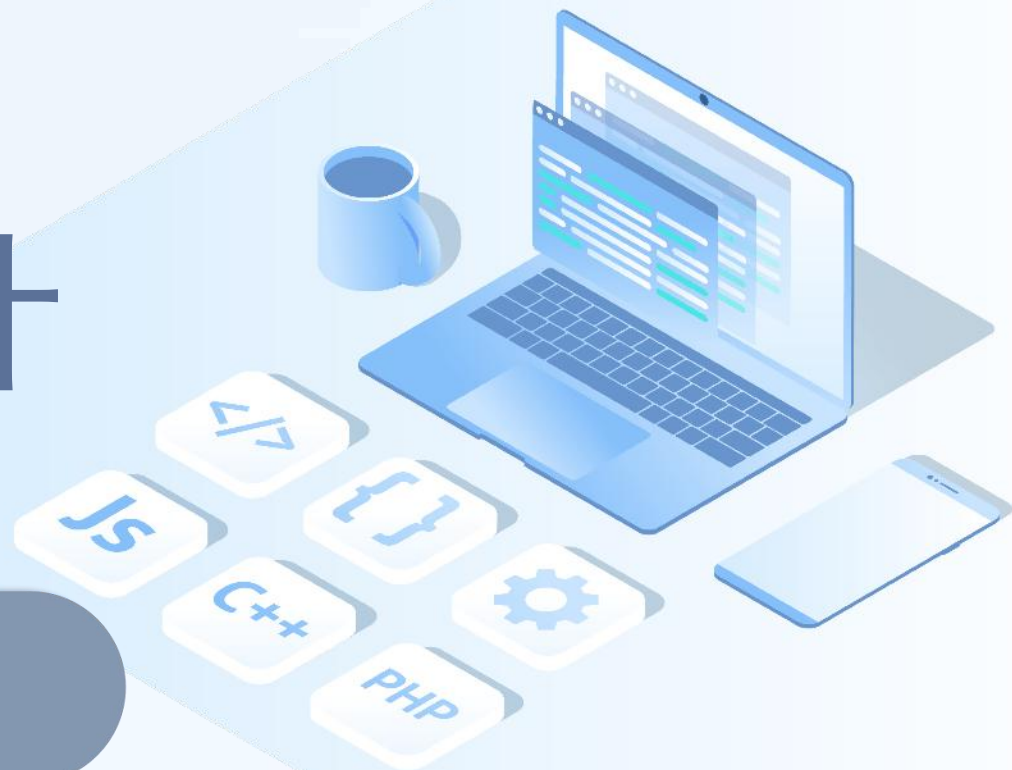
请编程，从键盘输入 n ，从1数到 n ，输出最后停在哪儿手指上。



第四章

选择结构程序设计

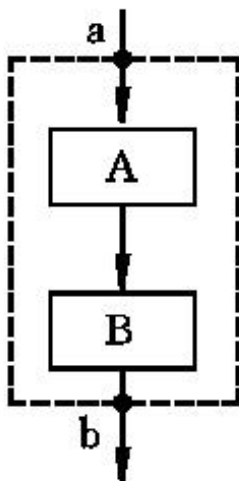
五院三教 许岩



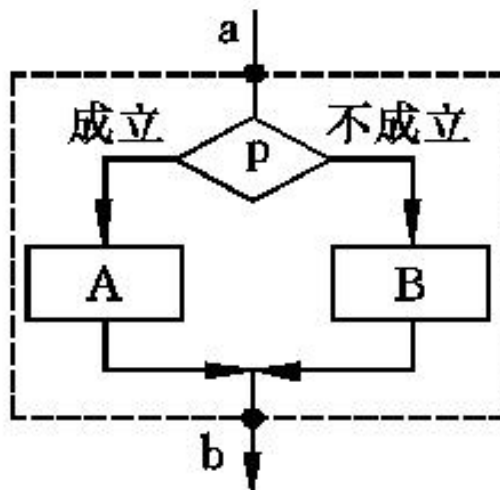


任何一个算法都可以用三种基本结构表示，这三种基本结构为：

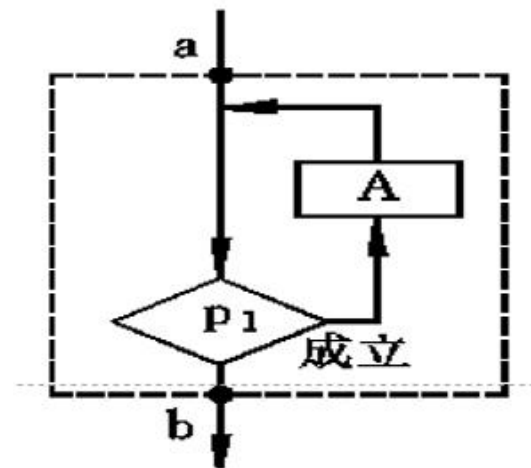
顺序结构、选择结构和循环结构。



(1) 顺序结构



(2) 选择结构



(3) 循环结构



C运算符

算术运算符: (+ - * / % ++ --)

关系运算符: (< <= == > >= !=)

逻辑运算符: (! && ||)

位运算符 : (<< >> ~ | ^ &)

赋值运算符: (= 及其扩展)

条件运算符: (?:)

逗号运算符: (,)

指针运算符: (* &)

求字节数 : (sizeof)

强制类型转换: ((类型))

分量运算符: (. ->)

下标运算符: ([])

其它 : (() -)



4.1 关系运算符与关系表达式

4.2 逻辑运算符与逻辑表达式

4.3 条件语句（if语句）

4.4 开关语句（switch语句）

4.1 关系运算符和关系表达式

➤ 关系运算符

- 种类: $<$ $<=$ $==$ $>=$ $>$ $!=$
- 结合方向: 自左向右
- 优先级别:

$<$	}	优先级 (高)
$<=$		
$>$		
$>=$		
$==$	}	优先级 (低)
$!=$		

例 `int a=3,b=2,c=1,d,f;`

`a>b == c; /*(a>b)==c*/ /*表达式值1*/`

`b+c<a; /*(b+c)<a*/ /*表达式值0*/`

`d=a>b; /*d=(a>b)*/ /*d=1, 表达式的值为1*/`

`'a'>'A'; /*表达式值为1 */`

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

`8>7>6; /*表达式值为0*/`

示

4.2 逻辑运算符和逻辑表达式



➤ 逻辑运算符

- 种类： ! (非) && (与) || (或)
“&&” 和 “||” 是双目运算符，“!” 是单目运算符

- 逻辑运算真值表

a	b	!a	!b	a&& b	a b
真	真	假	假	真	真
真	假	假	真	假	真
假	真	真	假	假	真
假	假	真	真	假	假

- C语言中, 操作数:
0表示“假”
非0表示“真”
运算结果:
0表示“假”
1表示“真”

4.2 逻辑运算符和逻辑表达式



● 优先级：

! (非)	↑ 高 ↓ 低
算术运算符	
关系运算符	
&& (与)	
(或)	
赋值运算符	

● 结合方向：

!	:从右向左
&&	:从左向右
	:从左向右

例	<code>a<=x && x<=b</code>	<code>/* (a<=x) && (x<=b) */</code>
	<code>a = b&& x>y</code>	<code>/* a = (b&& (x>y)) */</code>
	<code>a + b x==y</code>	<code>/* (a + b) (x==y) */</code>
	<code>!a a>b</code>	<code>/* (!a) (a>b) */</code>

4.2 逻辑运算符和逻辑表达式



➤ 逻辑运算符

● 优先级 ·

● 结合方向 ·

例

a=4; b=5;

a&&b

/*值为1*/

!a || b

/*值为1*/

4&&0 || -2 /*值为1*/

5>3&&8<4-!0

/* (5>3) && (8<(4-(!0))) 值为0*/

'c'&&'d' /*值为1*/

4.2 逻辑运算符和逻辑表达式



➤ 逻辑运算符

- **短路特性**：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*/*

4.2 逻辑运算符和逻辑表达式



例：判别某年`year`是否为闰年

闰年的条件是：

- 1、能被4整除，但是不能被一百整除
- 2、能被400整除

条件1

条件2

逻辑表达式来判断：

$(\text{year} \% 4 == 0 \ \&\& \ \text{year} \% 100 != 0) \ || (\text{year} \% 400 == 0)$

非闰年的判断条件：

! ((year%4==0 && year%100!=0) || year%400==0)

$(\text{Year} \% 4 != 0 || \text{year} \% 100 == 0) \ \&\& \ \text{year} \% 400 != 0$

简化： $(\text{year} \% 4 || !(year \% 100)) \ \&\& \ \text{year} \% 400$



3.7.2 数据的输入

☆ 格式输入函数

格式：scanf(“格式控制串”，地址列表)

功能：按指定格式从键盘读入数据，存入地址表指定的存储单元中，并按回车键结束

- 地址列表：变量的地址，常用取地址运算符&
- 格式字符：d, i, o, x, u, c, s, f, e (课本p69)

例

```
scanf(“%d”, &a);  
    输入：10 ↵  
    则      a=10
```

例

```
scanf(“%x”, &a);  
    输入：11 ↵  
    则      a=17
```

4.2 逻辑运算符和逻辑表达式



例：判别某年year是否为闰年

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

```
int main() {
```

```
    int year;
```

```
    printf("Enter year:");
```

```
    scanf("%d", &year);
```

```
    if( ( year%4==0 && year%100!=0 ) || (year%400==0) )
```

```
        printf("year%d is a leap year\n",year);
```

```
    else
```

```
        printf ("year%d is not a leap year\n",year);
```

```
    return 0;
```

```
}
```

测试用例：

2016 2015 1900 2000



4.1 关系运算符与关系表达式

4.2 逻辑运算符与逻辑表达式

4.3 条件语句 (if语句)

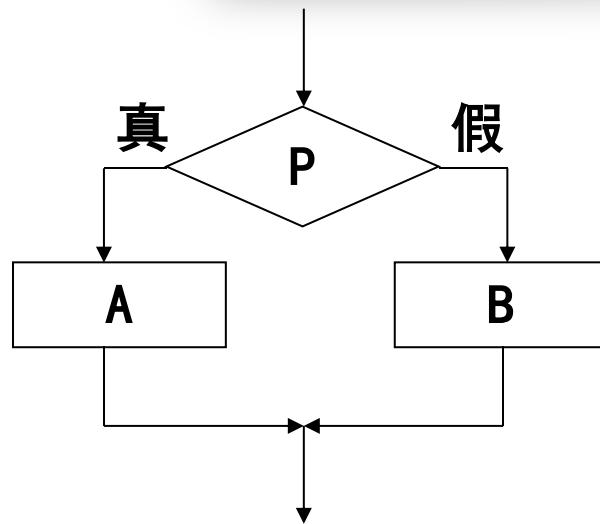
4.4 开关语句 (switch语句)

4.3 条件语句（if语句）

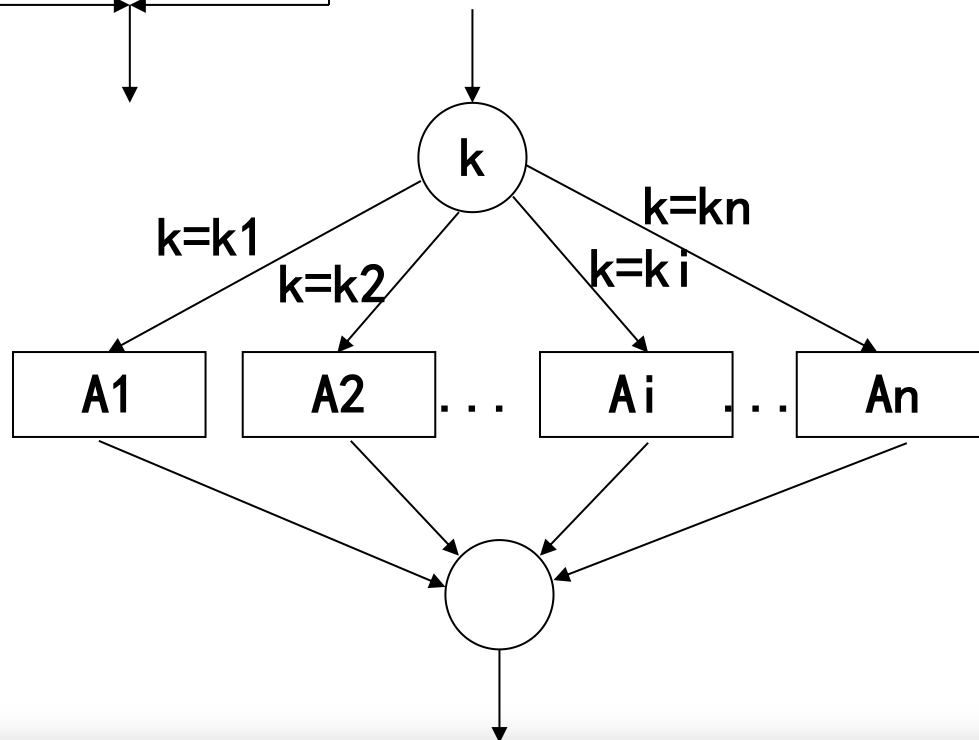


➤ 选择结构

- 二分支选择结构



- 多分支选择结构



4.3 条件语句（if语句）

➤ if语句（条件选择语句）

● if语句的三种形式

□形式一：条件执行

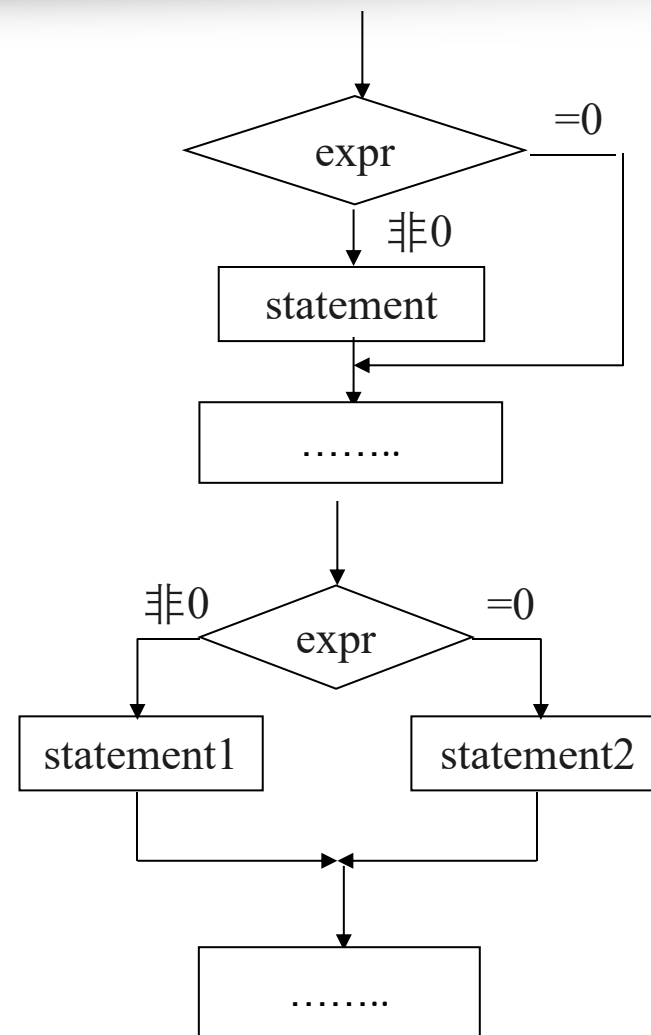
- 格式：if (expression)

例：if (x>y)
printf(“%d”,x);

□形式二：分支选择

- 格式：if (expression)
statement1

例：if (x>y) max=x;
else max=y;



4.3 条件语句 (if 语句)



```
if (a>5)
    max=100;
    printf("%d",max);
```

```
if (a>5){
    max=100;
    printf("%d",max);} 
```

```
if (a>5);
    max=a;
    printf("%d",max);
```

4.3 条件语句（if语句）



➤ if语句（条件选择语句）

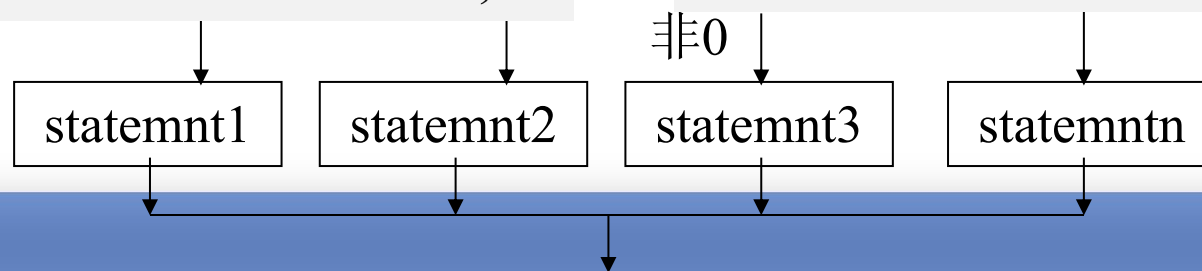
□形式三：阶梯式

- 格式：
- 执行过程：

```
if ( expr1 )      statement1;  
else if (expr2 )  statement2;  
else if (expr3 )  statement3;  
.....  
[ else           statementn ; ]
```

例：if (salary>1000) index=0.4;
else if (salary>800) index=0.3;
else if (salary>600) index=0.2;
else if (salary>400) index=0.1;
else

if (salary>1000) index=0.4;
if (salary>800) index=0.3;
if (salary>600) index=0.2;
if (salary>400) index=0.1;
else index=0;



4.3 条件语句（if语句）



➤ if语句（条件选择语句）

●说明：

◆ if后面的表达式类型任意

◆ 语句可以是复合语句

如： ~~if(a==b&&x==y) printf("a=b,x=y");~~
~~if(x) if(x!=0)~~

~~if(!x) printf("OK");~~
~~if(x) if(x!=0)~~

if(例 考虑下面程序的输出结果：

```
#include <stdio.h>
int main() {
    int x,y;
    scanf("%d,%d",&x,&y);
    if(x>y)
    { x=y; y=x; }
    else
    { x++; y++; }
    printf("%d,%d\n",x,y);
}
```

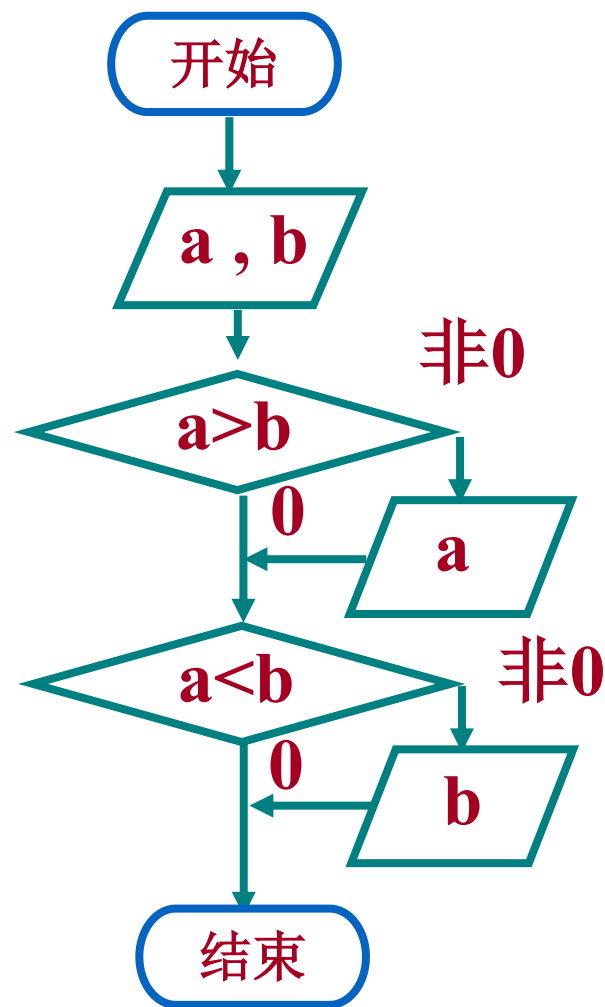
Compile
Error!

4.3 条件语句（if语句）

➤ if语句（条件选择语句）

例 从键盘输入两个整型数，输出最大数。

```
int main() {  
    int a,b ;  
    scanf("%d,%d",&a,&b);  
    if(a>b) printf("max=%d\n",a);  
    if(b>a) printf("max=%d\n",b);  
    return 0;}
```

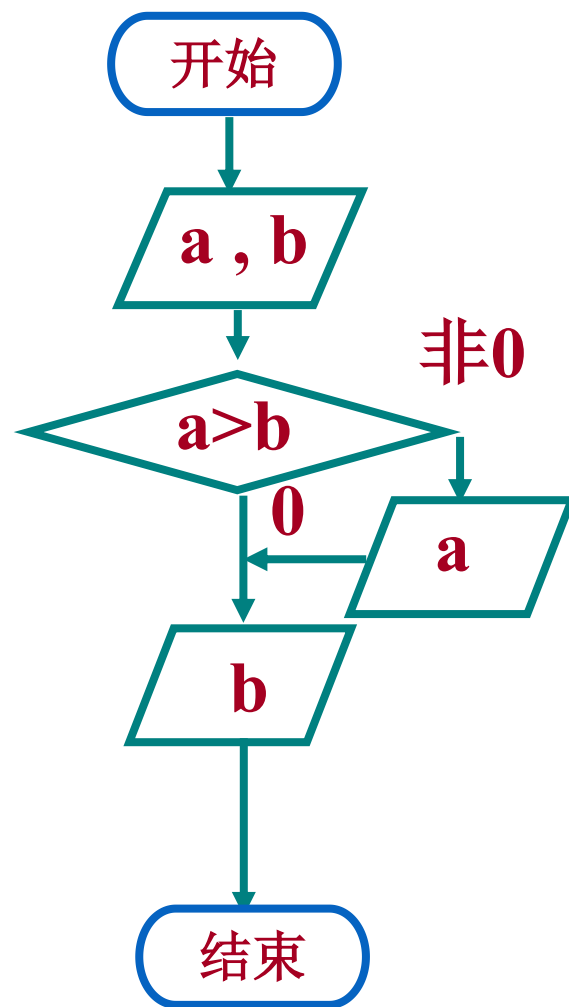


4.3 条件语句（if语句）

➤ if语句（条件选择语句）

例 从键盘输入两个整型数，输出最大数。

```
int main() {  
    int a,b ;  
    scanf("%d,%d",&a,&b);  
    if(a>b) printf("max=%d\n",a);  
    else printf("max=%d\n",b);  
    return 0;}
```



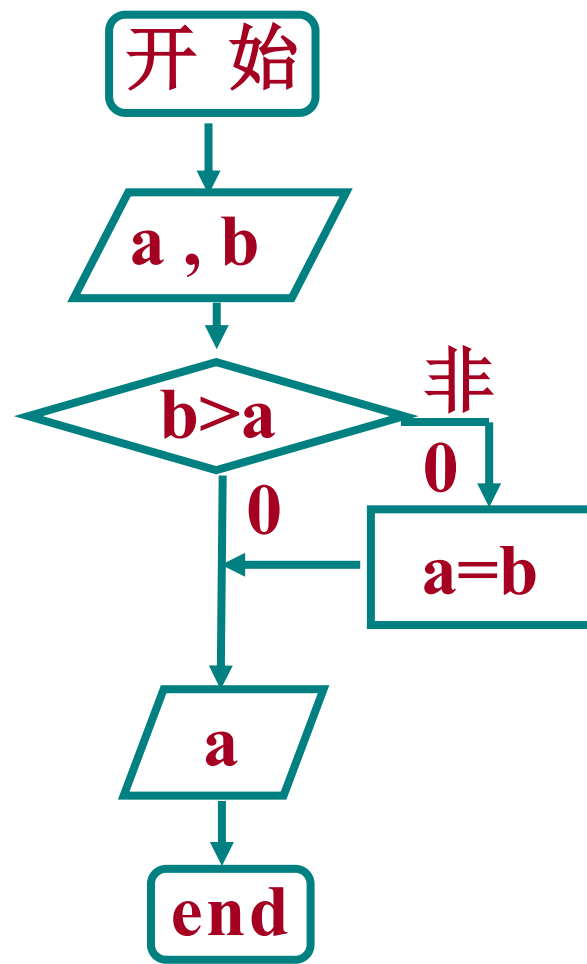
4.3 条件语句（if语句）

➤ if语句（条件选择语句）

例 从键盘输入两个整型数，输出最大数。

```
int main() {  
    int a,b ;  
    scanf("%d,%d",&a,&b);  
    if (b>a)    a=b;  
    printf("max=%d\n",a);  
    return 0;}  

```

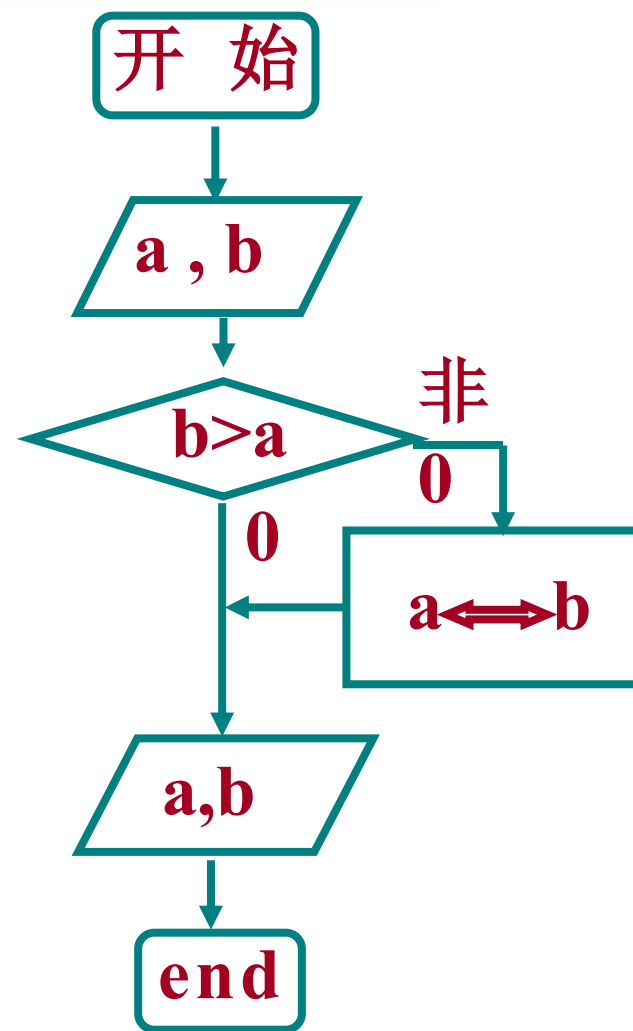


4.3 条件语句（if语句）

➤ if语句（条件选择语句）

例 输入两个整型数，按大小输出。

```
int main() {  
    int a,b,temp ;  
    scanf("%d,%d",&a,&b);  
    if (b>a) {temp=a;a=b;b=temp;}  
    printf("%d,%d\n",a,b);  
    return 0; }
```



4.3 条件语句（if语句）



➤ if语句（条件选择语句）

例 输入互不相等的三个数，按从小到大顺序输出它们。

分析：设已知的三个数为 a ， b ， c 。

（1）可直接通过逻辑表达式和条件语句设计程序。

六种情况：

$a < b < c$

$a < c < b$

$a < b \&\& b < c$

$b < a < c$

$b < c < a$

$c < b < a$

$c < a < b$

4.3 条件语句（if 语句）



➤ if 语句（条件选择语句）

例 输入互不相等的三个数，按从小到大顺序输出它们。

```
int main( ) {
```

方法(1):

```
    int a, b, c;  
    scanf("%d%d %d ", &a, &b, &c);  
    if(a<b && b<c) printf("%d,%d,%d", a, b, c);  
    if(a<c && c<b) printf("%d,%d,%d", a, c, b);  
    if(b<a && a<c) printf("%d,%d,%d", b, a, c);  
    if(b<c && c<a) printf("%d,%d,%d", b, c, a);  
    if(c<b && b<a) printf("%d,%d,%d", c, b, a);  
    if(c<a && a<b) printf("%d,%d,%d", c, a, b);}
```

4.3 条件语句（if语句）



➤ if语句（条件选择语句）

例 输入互不相等的三个数，按从小到大顺序输出它们。

分析：设已知的三个数为 a ， b ， c 。

（2）判断交换法：

可通过条件语句和互换变量值设计程序，即经过处理后使：

$$a < b < c$$

- ①若 $a > b$ ，则交换 a 、 b 的值
- ②若 $a > c$ ，则再交换 a 、 c 的值
- ③若 $b > c$ ，则交换 b 、 c 的值

4.3 条件语句（if 语句）



➤ if 语句（条件选择语句）

例 输入互不相等的三个数，按从小到大顺序输出它们。

方法(2):

```
int main( ) {  
    int a, b, c, t;  
    scanf("%d %d %d",&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("%5d, %5d, %5d ", a, b, c);}
```

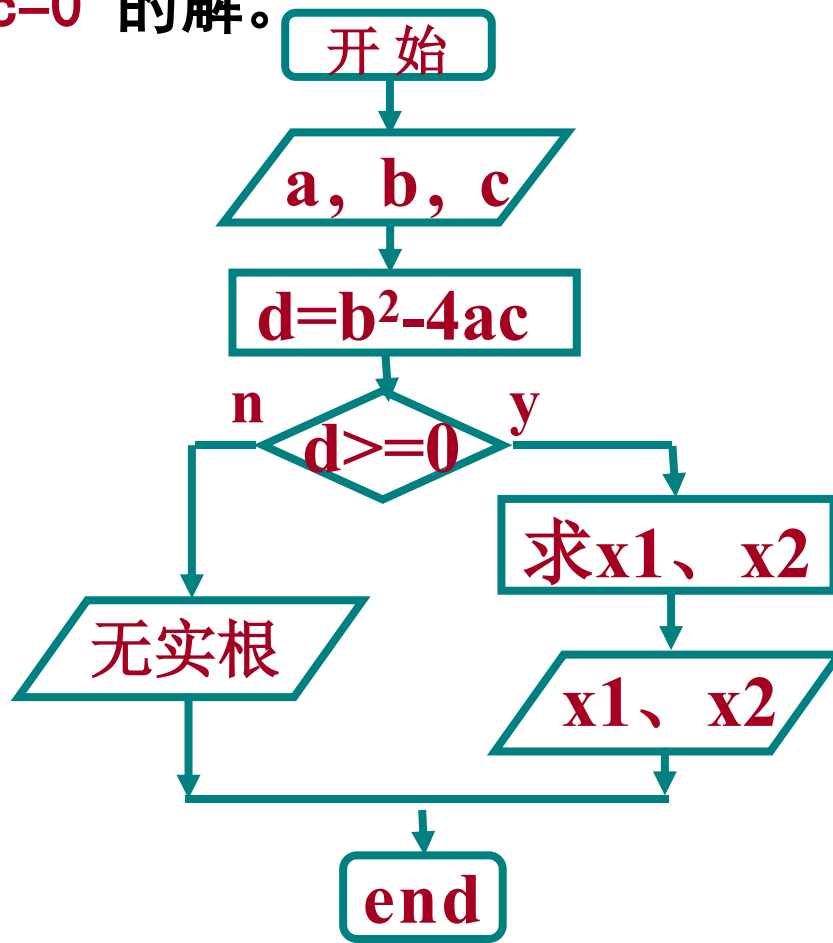
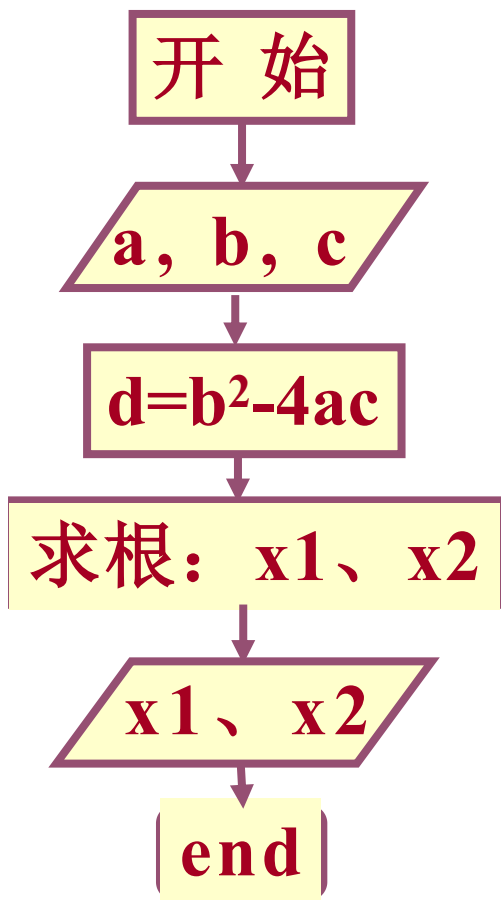
复合语句



4.3 条件语句（if语句）

➤ if语句（条件选择语句）

例 求一元二次方程 $ax^2+bx+c=0$ 的解。





4. 3条件语句（if语句）

➤ if语句（条件选择语句）

例 求一元二次方程 $ax^2+bx+c=0$ 的解。

```
#include<math.h>
void main( ) {
    float a, b, c, d, x1, x2;
    scanf("%f %f %f",&a,&b,&c);
    d=b*b-4*a*c;
    if (d>=0) {
        x1=(-b+sqrt(d))/(2*a);
        x2=(-b-sqrt(d))/(2*a);
        printf("x1=%f,x2=%f\n", x1,x2);}
    else printf("It is no real roots.");}
```



4.3 条件语句（if语句）

➤ if语句（条件选择语句）

例 运输公司吨公里基本运费为 p ，根据距离 s 的远近计费标准如下：

$s < 250\text{km}$ 没有折扣

$250 \leq s < 500$ 2% 折扣

$500 \leq s < 1000$ 5% 折扣

$1000 \leq s < 2000$ 8% 折扣

$2000 \leq s < 3000$ 10%折扣

$3000 \leq s$ 15%折扣

根据货物重量 w 和运距 s ，求运费

$$f = p * w * s (1 - d)。$$



4. 3条件语句（if语句）

➤ if语句（条件选择语句）

例 运输公司吨公里基本运费为 p ，根据距离 s 的远近计费标准如下：

```
void main( ) {  
    float p,s,w,d,f;  
    scanf("%f %f %f",&p,&s,&w);  
    if(s>=3000) d=0.15;  
    else if(s>=2000) d=0.10;  
    else if(s>=1000) d=0.08;  
    else if(s>=500) d=0.05;  
    else if(s>=250) d=0.02;  
    else d=0.0;  
    f=p*w*s*(1-d);  
    printf("f=%15.2f", f); }
```


4.3 条件语句（if语句）

➤ 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

4.3 条件语句（if语句）



➤ if语句嵌套

例 输入两数并判断其大小关系。

```
/*ch4_4.c*/
#include <stdio.h>
void main() {
    int x,y;
    printf("Enter integer x,y:");
    scanf("%d,%d",&x,&y);
    if(x!=y)
        if(x>y) printf("X>Y\n");
        else    printf("X<Y\n");
    else
        printf("X==Y\n");}
```

运行： Enter integer x,y:12,23↵
X<Y
Enter integer x,y:12,6↵
X>Y
Enter integer x,y:12,12↵
X==Y

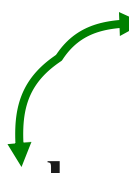
4.3 条件语句（if语句）



➤ if语句嵌套

例：

```
if (a==b)
    if(b==c)
        printf("a==b==c");
    else
        printf("a!=c");
```



修改：

```
if (a==b) {
    if(b==c)
        printf("a==b==c");
}
else
    printf("a!=c");
```

实现if ~ else
正确配对方法：
加{ }



4. 3条件语句（if语句）

➤ if语句嵌套

例 考虑下面程序输出结果：

```
void main()
{  int x=100,a=10,b=20;
    int v1=5,v2=0;
    if(a<b)
        if(b!=15)
            if(!v1)
                x=1;
            else
                if(!v2) x=10;
        x=-1;
    printf("%d",x);
}
```

结果： -1



4. 3条件语句（if语句）

➤ if语句嵌套

例 求一元二次方程 $ax^2+bx+c=0$ 的解。

分析：	$a=0$	不是二次方程。
	$b^2-4ac=0$	有两个相等实根。
	$b^2-4ac>0$	有两不等实根。
	$b^2-4ac<0$	有两共轭复根。

编程：

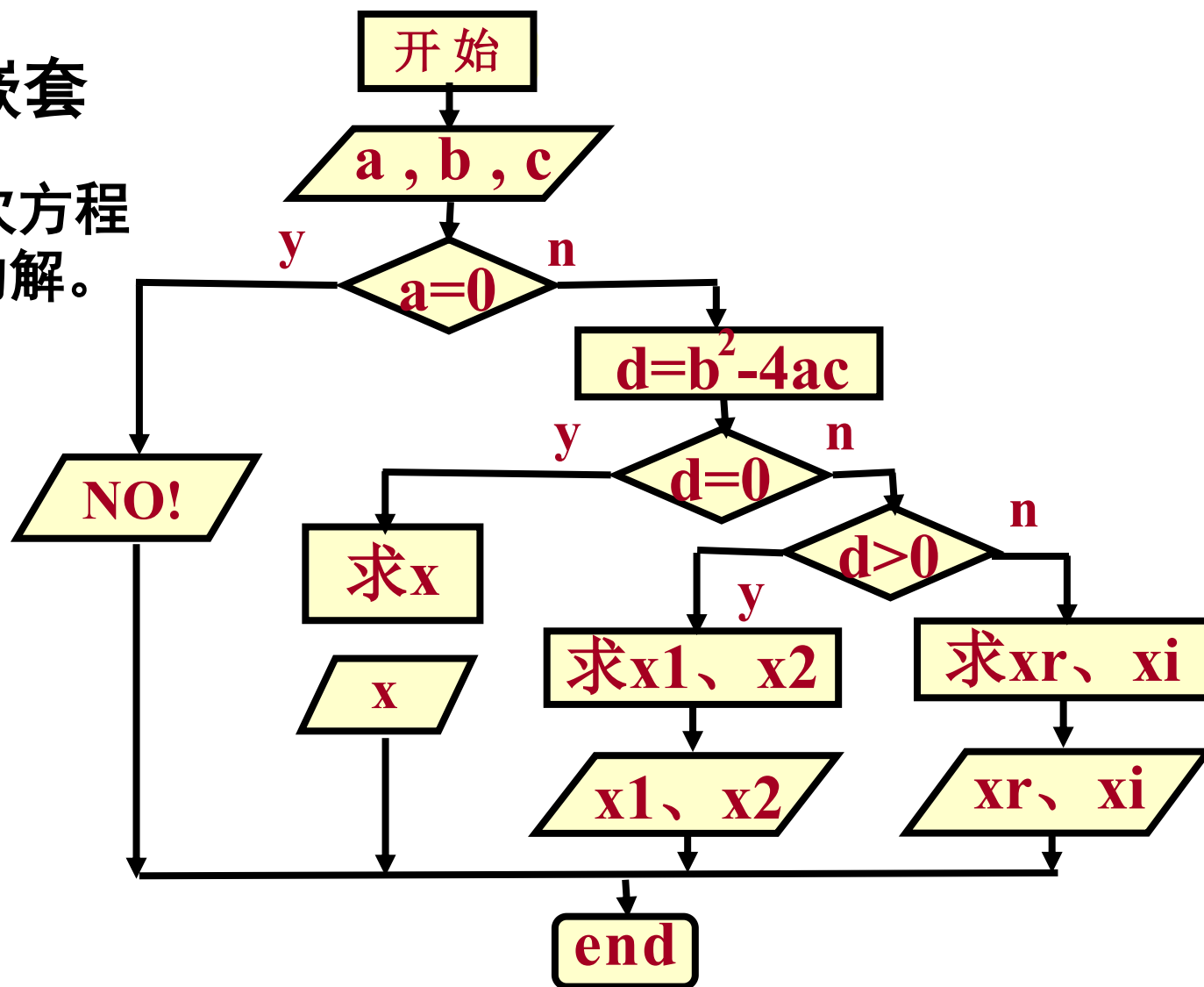
```
#include <math.h>
void main( ) {
    float a,b,c,d,x,x1,x2,xr,xi;
    scanf("%f %f %f",&a,&b,&c);}
```



4. 3条件语句（if语句）

➤ if语句嵌套

例 求一元二次方程
 $ax^2+bx+c=0$ 的解。





4. 3条件语句（if语句）

例 求一元二次方程 $ax^2+bx+c=0$ 的解。

```
if ( fabs(a-0)<1e-6) printf("No!");  
else {  
    d=b*b-4*a*c;  
    if ( fabs(d-0)<1e-6 ) {  
        x= -b/(2*a);  
        printf("Two equal roots:%f ", x);  
    }  
    else if ( d>0 ) {  
        x1=(-b+sqrt(d))/(2*a); x2=(-b -sqrt(d))/(2*a);  
        printf("Two real roots:%f,%f ",x1,x2);  
    }  
    else {  
        xr= -b/(2*a); xi=sqrt(-d)/(2*a);  
        printf("Complex roots:%f,%f ", xr,xi); }  
}
```

4.3 条件语句 (if语句)

例 判别某年year是否为闰年。

闰年的条件是：

1、能被4整除，但是不能被一百整除

2、能被400整除

```
void main() { /*if-else嵌套实现*/  
void main() { /*阶梯式if-else实现*/  
{ int year, leap;  
  printf("Enter year:");  
  scanf("%d", &year);  
  if(year%4!=0) leap =0;  
  else if(year%100!=0)leap=1;  
  else if(year%400!=0)leap=0;  
  else leap = 1;  
  if(leap == 1)  
  printf("year%d is a leap year\n",year);  
  else  
  printf ("year%d is not a leap year\n",year);  
}  
}
```

除*/

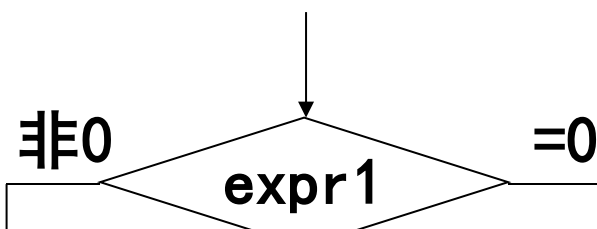
4.3 条件语句（if语句）



➤ 条件运算符与表达式

- 一般形式: $\text{expr1} \ ? \ \text{expr2} \ : \ \text{expr3}$
- 执行过程
- 功能: 相当于条件语句, 但不能取代一般if语句
- 条件运算符可嵌套

如 $x > 0 ? 1 : (x < 0 ? -1 : 0)$



例

```
if (a>b)
```

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*/



4.1 关系运算符与关系表达式

4.2 逻辑运算符与逻辑表达式

4.3 条件语句（if语句）

4.4 开关语句（switch语句）

4.4 开关语句（switch语句）



一般形式：

```
switch( 表达式) {  
    case E1:  
        语句组 1;  
        [break;]  
    case E2:  
        语句组 2;  
        [break;]  
    .....  
    case En:  
        语句组 n;  
        [break;]  
    [default:  
        语句组 ;  
        break; ]  
}
```

执行过程：

- 1、计算表达式的值
- 2、若与**常量表达式E_i**值相等，
则从语句i开始执行；直到遇到
break语句或switch语句的“}”
- 3、若与任何常量表达式的值
都不一致时，则执行default
语句，或执行后续语句

4.4 开关语句（switch语句）



➤ switch语句（开关分支语句）

❖说明：

- E1, E2, ...En是**常量表达式**，且值必须互不相同
- 常量表达式的值E_i仅仅起到语句标号作用，不做求值判断
- 常量表达式的值是唯一的，没有大小顺序
- 语句标号作用，如：
- case后可包含多条语句
- switch可嵌套
- 多个case可共用一条语句

```
case 'A':  
case 'B':  
case 'C':  
    printf("score>60\n");  
    break;  
.....
```

4.4 开关语句（switch语句）



```
例  switch(score) {  
        case 5:  printf("Very good!");  
        case 4:  printf("Good!");  
        case 3:  printf("Pass!");  
        case 2:  printf("Fail!");  
        default : printf("data error!");  
    }
```

运行结果：score为5时，输出：

Very good! Good! Pass! Fail! data error!

4.4 开关语句（switch语句）



```
例 int main() {  
    int x=1,y=0,a=0,b=0;  
    switch(x) {  
        case 1:  
            switch(y) {  
                case 0: a++; break;  
                case 1: b++; break;  
            }  
        case 2: a++;b++; break;  
        case 3: a++;b++;  
    }  
    printf("\na=%d,b=%d",a,b);  
}
```

运行结果： a=2,b=1



4.4 switch语句（开关分支语句）

例 根据输入字母输出字符串



4.4 switch语句（开关分支语句）

例 用switch语句设计“前例运费程序”。



4.4 switch语句（开关分支语句）

例 某运输公司吨公里基本运费为 p ，根据距离 s 的远近
计费标准如下：

```
switch(s) {  
    case s<250: d=0;  
    case s>=250&& s<500: d=0.02;  
    case s>=500&& s<100: d=0.05;  
    .....  
    .....  
    default : d=0.15;  
}
```

$$f = p * w * s (1 - d)。$$



4.4 switch语句（开关分支语句）

例 某运输公司吨公里基本运费为 p ，根据距离 s 的远近
计费标准如下：

$s < 250\text{km}$

没有折扣

$250 \leq s < 500$

2% 折扣

$500 \leq s < 1000$

5% 折扣

$1000 \leq s < 2000$

8% 折扣

$2000 \leq s < 3000$

10%折扣

$3000 \leq s$

15%折扣

根据货物重量 w 和运距 s ，求运费

$$f = p * w * s (1 - d)。$$

$s/250$

0: $d=0$

1: $d=0.02$

2, 3: $d=0.05$

4-7: $d=0.08$

8-11: $d=0.1$

其它: $d=0.15$



4.4 switch语句（开关分支语句）

例 某运输公司吨公里基本运费为 p ，根据距离 s 的远近
计费标准如下：

```
void main( ) {  
    float p, w, s, d, f; int c;  
    scanf("%f", &s);  
    „scanf(“%f %f  
    „, &p, &w);  
    switch((int)s/250) {  
        case 0 : d=0; break;  
        case 1 : d=0.02; break;  
        case 2 :  
        case 3 : d=0.05; break;  
        case 4 :  
        case 5 :
```

```
        case 6 :  
        case 7 : d=0.08; break;  
        case 8 :  
        case 9 :  
        case 10:  
        case 11: d=0.1; break;  
        default: d=0.15; break;  
    }  
    f=p*w*s*(1-d);  
    printf(“f=%15.2f ”, f);  
}
```



一元二次方程求根

对于一元二次方程 $ax^2+bx+c=0$ ，方程中的系数由用户在运行时输入。试编程根据用户输入的系数a，b，c求解方程的根。

要求：如果没有实数根，则输出“无实数根”；如果有实数根且不相等，则先输出大的，再输出小的，中间用一个空格隔开；如果实数根相等，则输出一个根的值。



- 选择结构有几种分支控制方式？分别适用什么语句实现？
- 如何判定一个C表达式的“真”和“假”？用什么值表示？
- `break`和`default`在`switch`语句中的作用是什么？
- 何为逻辑运算符的“短路”特性？

下课！