



C程序设计案例教程

- ▶ 单元5： 分支 / 选择结构程序设计





本章主要内容有

- 关系运算符和关系表达式
- 逻辑运算符和逻辑表达式
- if语句
- switch语句



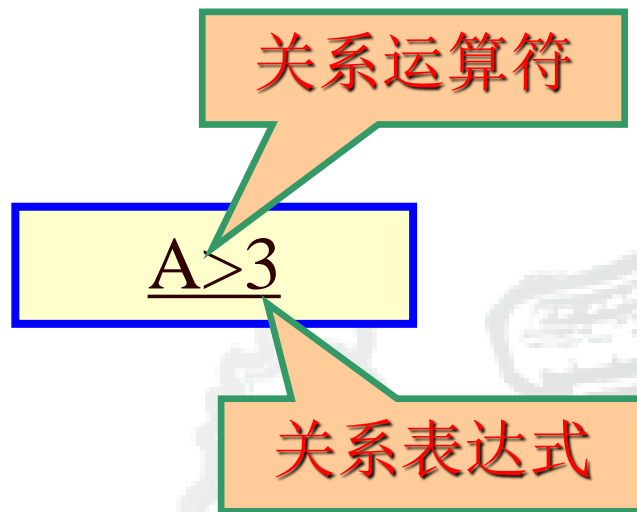


- 在C语言中，表示条件一般用关系表达式或逻辑表达式
- 实现选择结构用if语句或switch语句：
 - 用if语句实现双分支选择
 - 用switch语句实现多分支选择



5.1 关系运算

- “关系运算”即“比较运算，是对两个值进行比较，比较的结果是得到真假两种值。





★关系运算符及其优先次序

❖C语言提供6种关系运算符

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



优先级	关系运算符
高	$<$ (小于), $<=$ (小于等于), $>$ (大于), $>=$ (大于等于)
低	$=$ (等于), $!=$ (不等于)

```

c>a+b // c>(a+b)
a>b!=c // (a>b)!=c
a==b<c // a==(b<c)
a=b>c // a=(b>c)

```



5.2 关系表达式

❖ 定义：由关系运算符将两个可作比较运算的表达式联结起来所得到的式子称为关系表达式

$a > b$, $(a + b) > (b + c)$, $5 == 3$

❖ 关系表达式的值：是逻辑值“真”或“假”，用1和0表示

```
int a=3,b=2,c=1,d,f;  
a>b  
(a>b)==c  
b+c<a  
d=a>b  
f=a>b>c
```



❖ 关系运算几点注意:

例 $5 > 2 > 7 > 8$ 在C中是允许的,
值为 **0**

例 `int i=1, j=7, a;`
`a=i+(j%4!=0);`
则 `a=` **2**

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

> 结合方向
自左至右

用ASCII值
比较



❖ 关系运算注意优先级:

例 $0 > 2 > 7 > -8$ 值为 1

表达式中都为关系运算符 $>$ ，结合方向自左至右，首先判断 $0 > 2$ 值为0，得到 $0 > 7 > -8$ ，再判断 $0 > 7$ 值为0，依此类推

例 `int a=0,b=2,c=-2,d;`
`d=a==b<c;`
则d= 1

表达式`d=a==b<c`中，第一个`=`是优先级最低的赋值运算符，`==`和`<`是优先级较高的关系运算符，且`<`优先级高于`==`，首先判断`b<c`值为0，再判断`a==0`值为1，最后将1赋值给d



❖ 关系运算中应该注意

应避免对**实数**作相等或不等于**0**的判断

如 $1.0/3.0*3.0==1.0$ 可改写为:

```
fabs(1.0/3.0*3.0-1.0)<1e-6
```

注意区分 “**=**”与 “**==**”

```
int a = 0,b =1;
```

```
if(a = b)
```

```
    printf("a equal to b");
```

```
else
```

```
    printf("a not equal to b");
```



5.3 逻辑运算符

逻辑运算符用于对逻辑对象进行运算。

C 语言有三种逻辑运算符

“&&”和“||”是双目运算符

“!”是单目运算符

运算符	名称	示例	含义
&&	逻辑与	a && b	若a、b同时为真，则a && b为真
	逻辑或	a b	若a、b之一为真，则a b为真
!	逻辑非	! a	若a为真，则! a为假；a为假，则! a为真

逻辑运算与其它运算的关系：

- 逻辑运算的运算对象必须具有真假值（逻辑值），而关系运算的运算对象都需要相同类型。
- ! 运算具有最高级，|| 运算具有最低级，通过括号可改变优先级。|| 和&& 低于关系运算级别，!高于算术运算级



❖ 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

❖ 逻辑运算符的优先次序

!(非) , &&(与), ||(或)

优先次序: 高 (2) (11) (12) 低

结合方向: 从右向左 从左向右 从左向右

例: 优先次序

$(a > b) \&\&(x > y)$ //写成 $a > b \&\&x > y$

$(a == b) || (x == y)$ //写成 $a == b || x == y$

$(!a) || (a > b)$ //写成 $!a || a > b$



5.4逻辑表达式

用逻辑运算符将关系表达式或逻辑量连接起来的式子就是逻辑表达式。

C语言中，运算对象： 0表示“假”， 运算结果： 0表示“假”，
非0表示“真”， 1表示“真”，

例 a=4;b=5;

!a

a&&b

a||b

!a||b

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

'c'&&'d'



❖ 逻辑运算中的注意点:

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。

<code>a&&b&&c</code>	//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
<code>a b c</code>	//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

若 `a=1;b=2;c=3;d=4;m=1;n=1;`
则 `(m=a>b)&&(n=c>d)`



❖ 复杂逻辑条件的表述

判断某一年year是否为闰年

判别闰年的条件(int year):

能被4整除:

$year \% 4 == 0$

能被4整除但不能被100整除:

$(year \% 4 == 0) \&\& (year \% 100 != 0)$

能被400整除:

$year \% 400 == 0$

综合起来:

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

优化语句:

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

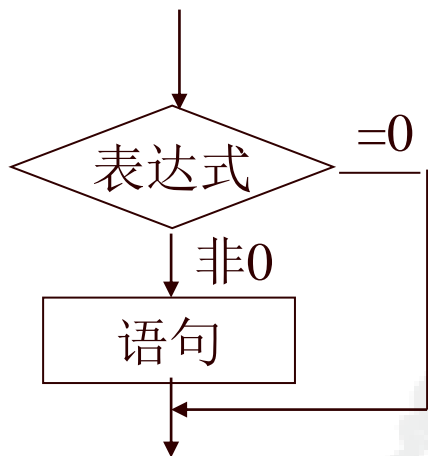


5.4 if 语句（条件选择语句）

★ If语句的三种形式

❖ 形式一：

- 格式：if (表达式) 语句
- 执行过程：



说明：

“表达式”：除常见的关系表达式或逻辑表达式外，也允许是其它类型的数据，如整型、实型、字符型等。

```
main()
{
    int x , y ;
    x=20 ;
    y=10 ;
    if (x>y)
        printf("%d",x);
}
```

没有“;”



IF语句格式一举例

- 计算 x 和 y 之间的最大值，存入 z ；
 - ▶ $z=x$;
 - ▶ $\text{if} (x<y) z=y$;
 - ▶ $\text{printf}("max(\%d,\%d)=\%d",x,y,z)$;

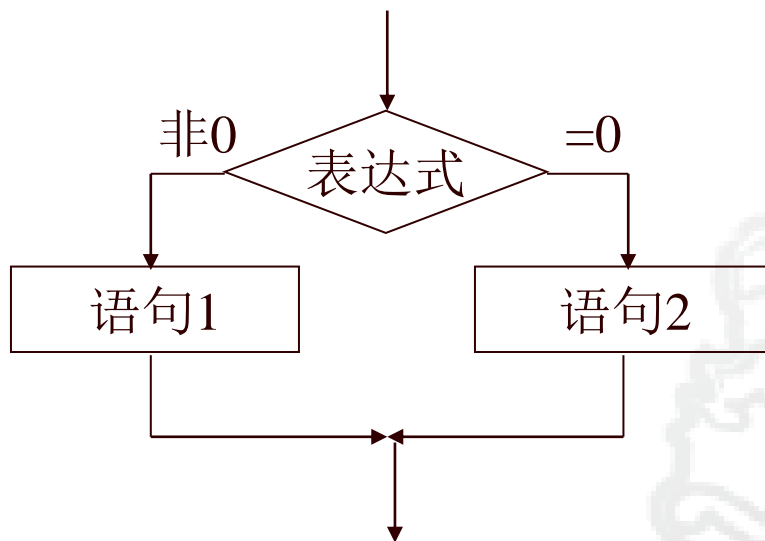




❖ 形式二:

- 格式: `if (表达式)`
 语句1
 else
 语句2

- 执行过程:



```
main()
{
    int x , y ;
    x=20 ;
    y=10 ;
    if (x>y)
        printf("%d",x);
    else
        printf("%d",y);
}
```



If语句格式二举例

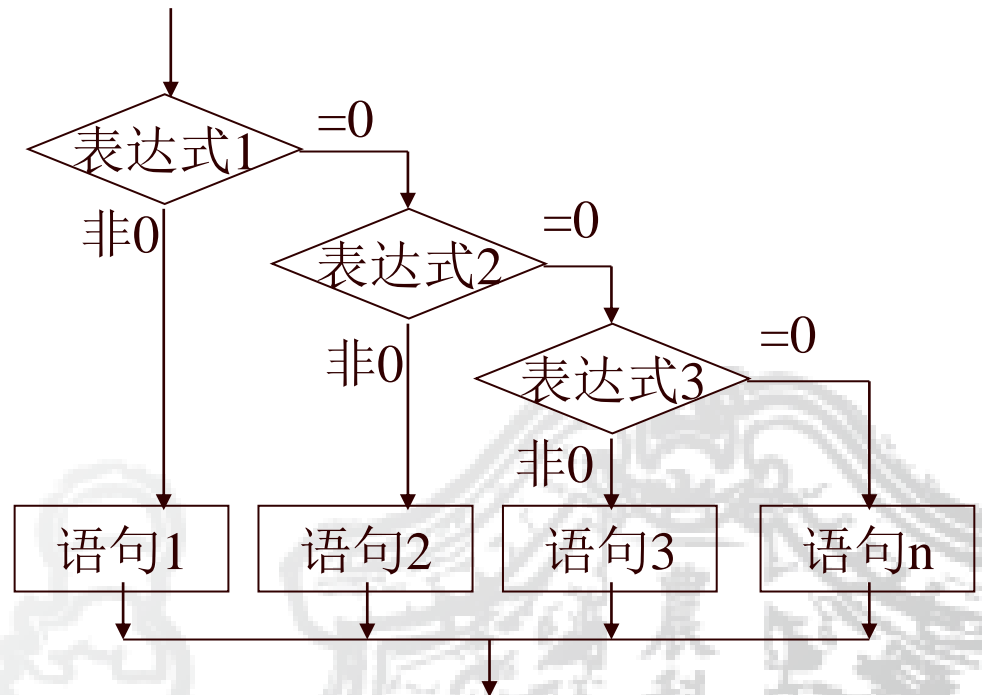
- 计算 x 和 y 之间的最大值，存入 z ；
 - ▶ `if (x<y) z=y;`
 - ▶ `else z=x;`
 - ▶ `printf("max(%d,%d)=%d",x,y,z);`





❖ 形式三:

- 格式: `if (表达式1)`
 语句1
- `else if (表达式2)`
 语句2
- .
- .
- `else if (表达式n)`
 语句n
- `else`
 语句n+1



- 执行过程:



必须有
“ ; ”

```
例： if (number>500)      cost=0.15 ;  
      else if (number>300)  cost=0.1 ;  
      else if (number>100)  cost=0.075 ;  
      else if (number>50)   cost=0.05 ;  
      else                   cost=0 ;  
      printf("%lf\n", cost);
```

```
例： if (number>500)      cost=0.15 ;  
      if (number>300)     cost=0.1 ;  
      if (number>100)     cost=0.075 ;  
      if (number>50)      cost=0.05 ;  
      printf("%lf\n", cost);
```



```
if(a ==b&& x==y) printf("a=b,x=y");  
if(3) printf("OK");  
if('a') printf("%d",a);
```

- if 后面的语句可以是复合语句，必须要加{ }

考虑下面程序的输出结果：

```
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);  
}
```

{ }后没有
“ ; ”

Compile Error!



例：输入两个数并判断两数是否相等

```
#include <stdio.h>
int main()
{
    int a,b;
    printf("Enter integer a:");
    scanf("%d",&a);
    printf("Enter integer b:");
    scanf("%d",&b);
    if(a==b)
        printf("a==b\n");
    else
        printf("a!=b\n");
}
```

```
运行： Enter integer a:12↵
        Enter integer b:12↵
        a==b
```

```
运行： Enter integer a:12↵
        Enter integer b:9↵
        a!=b
```



例：判断输入字符种类

```
#include<stdio.h>
int main()
{ char c;
  printf("Enter a character:");
  c=getchar();
  if(c<32) 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");
}
```



例：判断输入

运行： Enter a character: F1 ↵
The character is other character

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

```
int main()
```

```
{ char c;
```

```
printf("Enter a character: ");
```

```
c=getchar();
```

```
if(c<32) printf("The character is a control character\n");
```

```
else if(c>='a' && c<='z') printf("The character is a lower letter\n");
```

```
else if(c>='A' && c<='Z') printf("The character is a capital letter\n");
```

```
else if(c>='0' && c<='9') printf("The character is a digit\n");
```

```
else printf("The character is other character\n");
```

```
}
```

运行： Enter a character: h↵
The character is a lower letter

运行： Enter a character: D↵
The character is a capital letter

运行： Enter a character: 8 ↵
The character is a digit

运行： Enter a character: ↵
The character is a control character



例5.1：输入两个实数，按由小到大的次序输出两数

```
#include <stdio.h>
int main()
{
    float a,b,t ;
    scanf("%f,%f",&a,&b);
    if(a>b)
        {t=a;a=b;b=t;}
    printf("%5.2f,%5.2f",a,b);
}
```

输入： 3.6, -3.2 ↵

输出： -3.20, 3.60



例5.2：输入两个数，输出较大值

```
#include <stdio.h>
int main()
{
    int a,b,c;
    scanf("%d,%d",&a,&b);
    if (a>b)
        { c=a;}
    else
        {c=b;}
    printf("%d\n", c);
    return 0;
}
```

输入：3,7 ↵
输出：7.00



★ If 语句的嵌套

- If语句中又包含一个或多个if语句称为if语句的嵌套。
- 实际上只要将前述if语句的形式1和2中的内嵌语句用一个if语句代替，即成为if语句的嵌套。
- 嵌套的if语句还可以嵌套另一个if语句，形成多重嵌套。

❖ 一般形式：

```
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
```

内嵌if

内嵌if



❖ if 嵌套的几种变形

是这样吗?

```
if (条件1)
    语句1
else
    if (条件2) 语句2
    else      语句3 } 内嵌if
```

```
if (条件1)
    if (条件2) 语句1
else
    if (条件3) 语句2
else      语句3 } 内嵌if
                内嵌if
```

```
if (条件1)
    if (条件2) 语句1
    else      语句2 } 内嵌if
else
    语句3
```

```
if (条件1)
    {if (条件2) 语句1} 内嵌if
else
    if (条件3) 语句2
    else      语句3 } 内嵌if
```



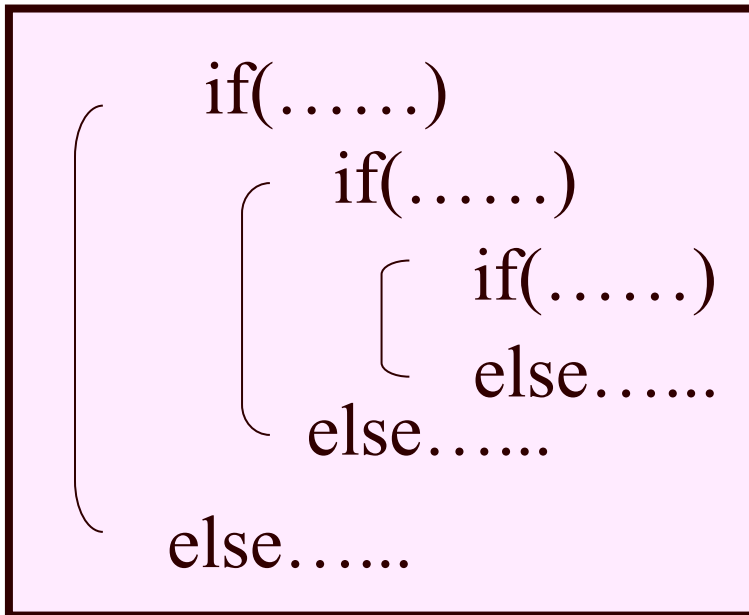
注意：

- 1) 多个嵌套的结果，else存在匹配的问题：else总是与最近的尚未匹配的if匹配。
- 2) 格式化问题：同一级的if-else都向左缩进一定的空格。

必要时用{ }限定
内嵌if范围



❖ if ~ else 配对原则：缺省{ }时，else总是和它上面离它最近的未配对的if 配对。



例 考虑下面程序的输出结果

```
int 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



例：输入两个数并判断其大小关系

```
#include <stdio.h>
int 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");
}
```

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



例5.3 有一函数

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

编一程序，输入一个x值，输出y值。

- ▶ #include<stdio.h>
- ▶ int main()
- ▶ {
- ▶ int x,y;
- ▶ scanf("%d",&x);
- ▶ {
- ▶ 程序段
- ▶ }
- ▶ printf("x=%d,y=%d\n",x,y);
- ▶ return 0;
- ▶ }





程序1:

```
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);  
}
```

程序2:

```
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);  
}
```

程序



请考虑:

$y=1$ 或 $y=0$ 时, if语句应
该如何写?

{ }什么情况下可以不要?

程序3:

```
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);  
}
```

输入: -2
得到: x=-2, y=0

程序3改:

```
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);  
}
```

程序



▶ if 语句+复合语句

```
#include <stdio.h>
int main()
{
    int x,y;
    scanf("%d",&x);
    if (x<0 )
    {
        y=-1;
        printf("x=%d,y=%d\n",x,y);
    }
    return 0;
}
```

```
#include <stdio.h>
int main()
{
    int x,y;
    scanf("%d",&x);
    if (x<0 )
        y=-1;
    printf("x=%d,y=%d\n",x,y);
    return 0;
}
```



★ 条件运算符

if语句中，当表达式为“真”和“假”时，都只执行一个赋值语句给同一个变量赋值时，可以用条件运算符处理。

```
if (a>b) max=a;  
else max=b;
```

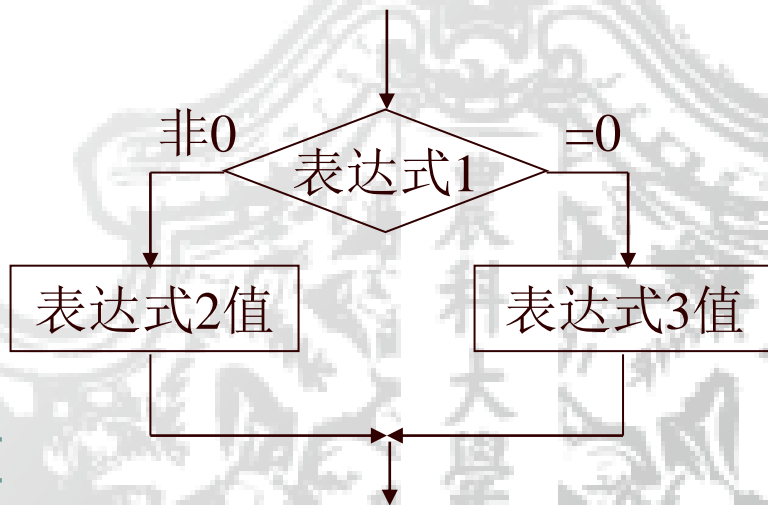


```
max=(a>b)? a:b;
```

表达式1 ? 表达式2 : 表达式3



条件运算符是 C 语言中**唯一**的
三目运算符





❖ 条件运算符练习

例 $(a==b)? 'Y':'N'$
 $(x\%2==1)?1:0$
 $(x>=0)?x:-1*x$
 $(c>='a' \ \&\& \ c<='Z')?c-'a'+'a':c$

例 求 $a+|b|$
 $\text{printf}("a+|b|=%d\n",b>0?a+b:a-b);$

❖ 条件运算符几点说明:

- 条件运算符可嵌套
- 优先级: 13
- 结合方向: 自右向左
- 表达式1? 表达式2: 表达式3 类型可以不同, 表达式值取表达式2和表达式3中较高的类型

嵌套: $x>0?1:(x<0?-1:0)$

$\text{max}=(a>b)? a:b; //\text{max}=a>b? a:b; \quad a>b?a:c>d?c:d \Leftrightarrow a>b?a:(c>d?c:d)$

$x? 'a':'b' \quad //x=0, \text{表达式值为}'b'; x\neq 0, \text{表达式值为}'a'$
 $x>y?1:1.5 \quad //x>y, \text{值为}1.0; x<y, \text{值为}1.5$



例5.4 输入一个字母，大写转小写，然后输出字母

```
#include <stdio.h>
int main()
{ char ch;
  scanf("%c",&ch);
  ch=(ch>='A' && ch<='Z')?(ch+32) : ch;
  printf("%c",ch);
}
```

输入： A
输出： a



§ 5.4 switch语句（多分支选择语句）

❖ 一般形式：

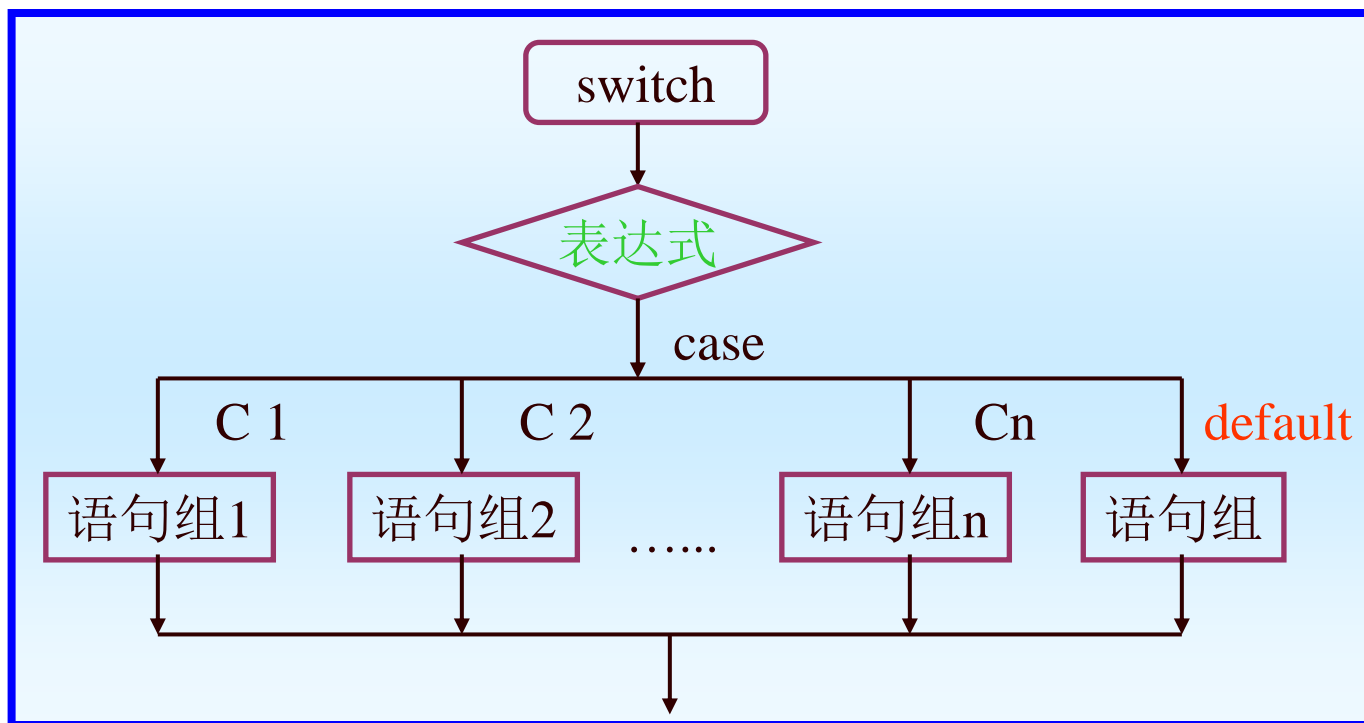
```
switch(表达式e)
{ case C1:
    语句1; break;
  case C2:
    语句2; break;
    .....
  case Cn:
    语句n; break;
  default:语句n+1; break; /*该句可省略*/
}
```

需要跳出
switch语句时
使用



§ 5.4 switch语句（多分支选择语句）

❖ 执行过程





❖ switch几点说明

- C1,C2,...Cn是**常量表达式**,且值必须互不相同
- SWITCH 内的表达式可以为任意类型,但是CASE 后的表达式的类型与SWITCH 内的表达式类型一致
- CASE 和 DEFAULT 的顺序是无关紧要的
- switch与if不同,只能测试相等,不能像if作关系比较,如大于小于之类;



❖ switch 几点说明

- 在每种 case 语句后最好有 break 终止，否则，将执行下面 case 对应的语句。
- case 后可包含多个可执行语句，且不必加 { }
- 多个 case 可共用一组执行语句
- switch 可嵌套

```
#include <stdio.h>
main()
{ char grade;
  grade = getchar();
  switch(grade)
  { case 'A' : printf("85~100 \n");
    case 'B' : printf("70~84  \n");
    case 'C' : printf("60~69  \n");
    case 'D' : printf("<60   \n");
    default  : printf("Error \n");
  }
}
```

A ↴
85~100
70~84
60~69
<60
Error

缺少 break;



❖ switch 几点说明

- 在每种 case 语句后最好有 break 终止，否则，将执行下面 case 对应的语句。
- case 后可包含多个可执行语句，且不必加 { }
- 多个 case 可共用一组执行语句
- switch 可嵌套

```
#include <stdio.h>
main()
{ char grade;
  grade = getchar();
  switch(grade)
  { case 'A' : printf("85~100 \n"); break;
    case 'B' : printf("70~84  \n"); break;
    case 'C' : printf("60~69  \n"); break;
    case 'D' : printf("<60    \n"); break;
    default  : printf("Error \n");
  }
}
```

A ↵
85~1
00



例 switch嵌套

```
#include <stdio.h>
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);
}
```

a=2 , b=1



§ 5.5 程序举例

例5.5 判断某一年是否闰年

```
#include <stdio.h> ①
int main()
{ int year, leap;
  scanf("Enter year:%d",&year);
  if(year%4==0)
    { if(year%100==0)
      { if(year%400==0) leap=1;
        else leap=0;}
      else leap=1;}
    else leap=0;
  if(leap) printf("%d is",year);
  else printf("%d is not",year);
  printf("a leap year.\n");
}
```

{ }能省略吗?



§ 5.5 程序举例

例5.5 判断某一年是否闰年

```
#include <stdio.h> ②
int main()
{ int year,leap;
  scanf("Enter year:%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) printf("%d is",year);
  else printf("%d is not",year);
  printf("a leap year.\n");
}
```





§ 5.5 程序举例

例5.5 判断某一年是否闰年

③使用逻辑表达式

```
#include <stdio.h>
int main()
{ int year, leap;
  scanf("Enter year:%d",&year);
  if((year%4==0&&year%100!=0)||year%400==0)
    leap=1;
  else leap=0;
  if(leap) printf("%d is",year);
  else printf("%d is not",year);
  printf("a leap year.\n");
}
```




例 根据输入的字母输出相应的字符串

```
#include <stdio.h>
int main()
{ char c;
  printf("Enter m or n or h or other:");
  c=getchar();
  switch(c)
  { case 'm': printf(" Good morning!\n");break;
    case 'n': printf(" Good night!\n");   break;
    case 'h': printf(" Hello!\n");       break;
    default : printf(" ????????\n");
  }
}
```

m↵

Good morning!

例5.6 求 $ax^2+bx+c=0$ 方程的解

```
#include <stdio.h>
#include <math.h>
int main()
{ float a,b,c,d,disc,x1,x2,realpart,imagpart;
  scanf("%f,%f,%f",&a,&b,&c);
  printf("The equation");
  if (fabs(a)<=1e-6) printf("is not a quadratic");
  else { disc=b*b-4*a*c;
        if (fabs(disc)<=1e-6)
          printf("has two equal roots:%8.4f\n",-1.0*b/(2*a));
        else if (disc>1e-6)
          { x1=(-1.0*b+sqrt(disc))/(2*a);
            x2=(-1.0*b-sqrt(disc))/(2*a);
            printf("has distinct real roots:%8.4f and %8.4f\n",x1,x2);}
        else { realpart=-1.0*b/(2*a); imagpart=sqrt(-1.0*disc)/(2*a);
              printf("has complex roots:\n");
              printf("%8.4f + %8.4fi\n",realpart,imagpart);
              printf("%8.4f - %8.4fi\n",realpart,imagpart);
            }
        }
  } }
```