



# 第4章 分支控制

## ——条件语句



哈尔滨工业大学

苏小红

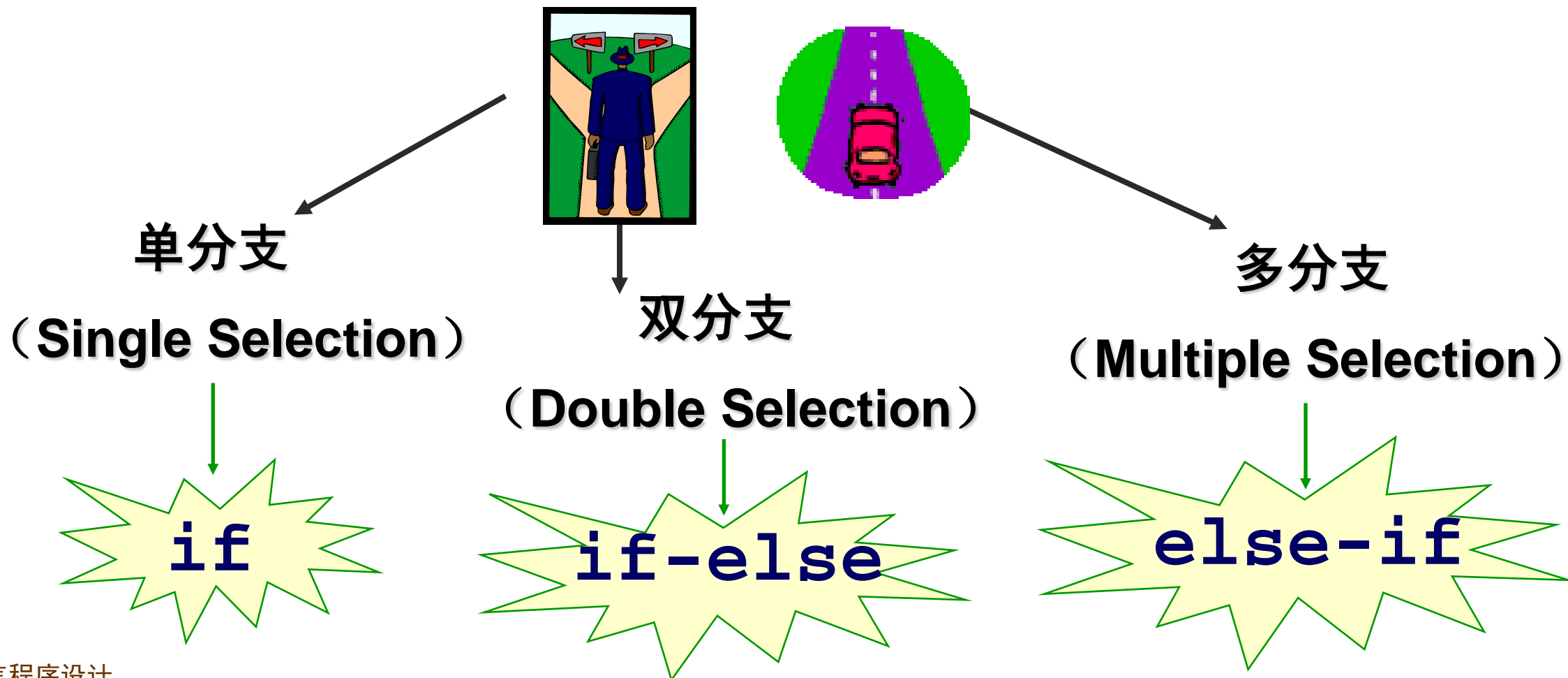
sxh@hit.edu.cn

# 本节要讨论的主要问题

- C语言提供了哪几种控制语句支持分支控制？
- 条件语句有哪几种形式？
- 何为条件运算符？
- 在分支中使用复合语句有什么好处？



# 选择结构

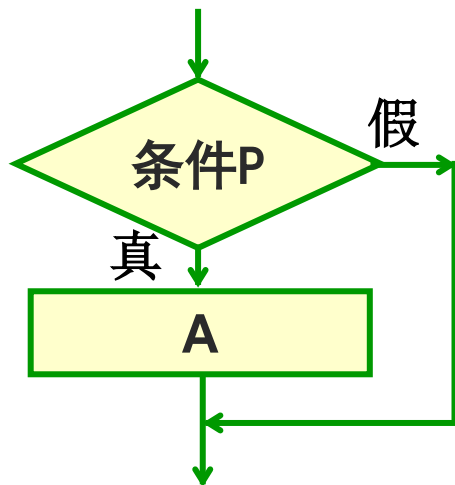


# 选择结构

## ■ 单分支选择结构（Single Selection）

\* 面临的选择是：要么执行一个操作，要么跳过它

## ■ if 语句



**if** ( 表达式P )  
语句A

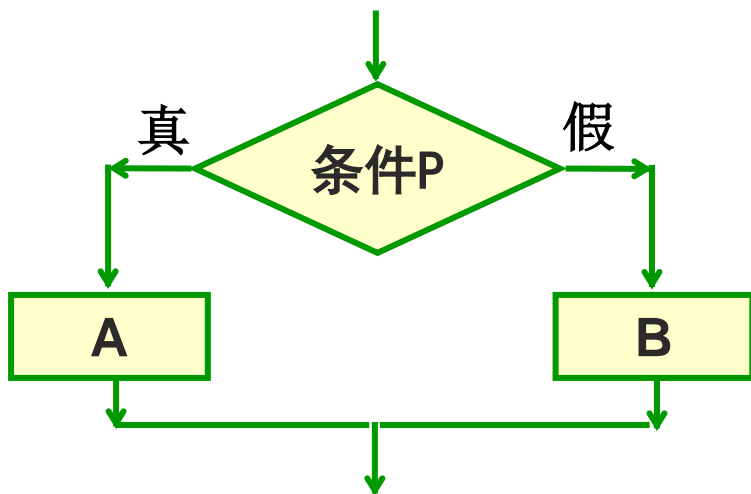
不局限于关系表达式，  
也可为数值表达式，表  
达式的值非0时，为真

# 选择结构

## ■ 双分支选择结构（Double Selection）

\* 面临的选择是：在两个不同的操作中选择其中的一个来执行

## ■ **if-else**语句



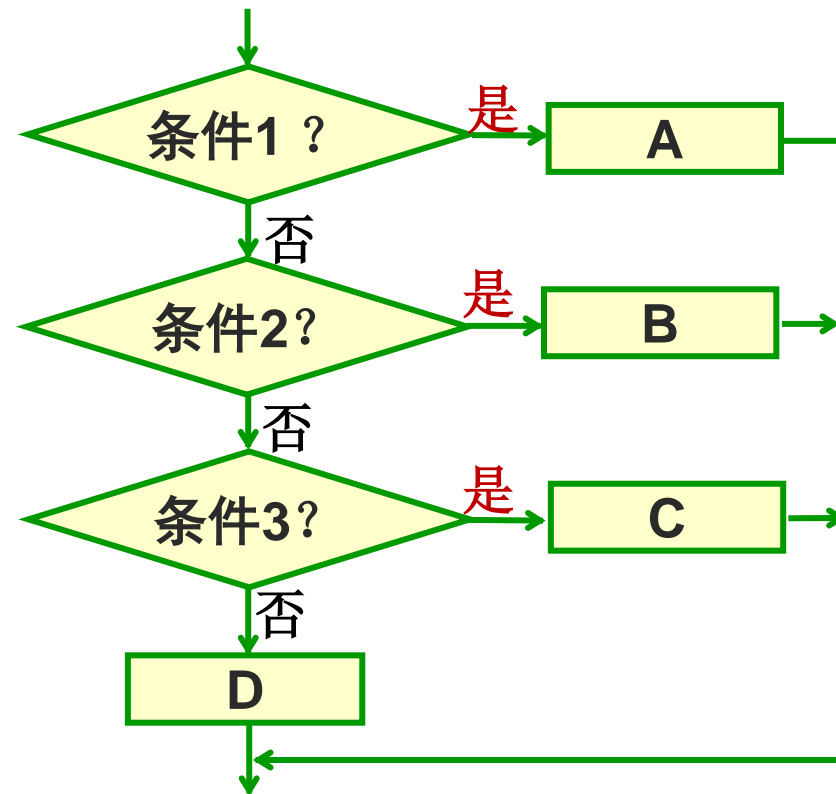
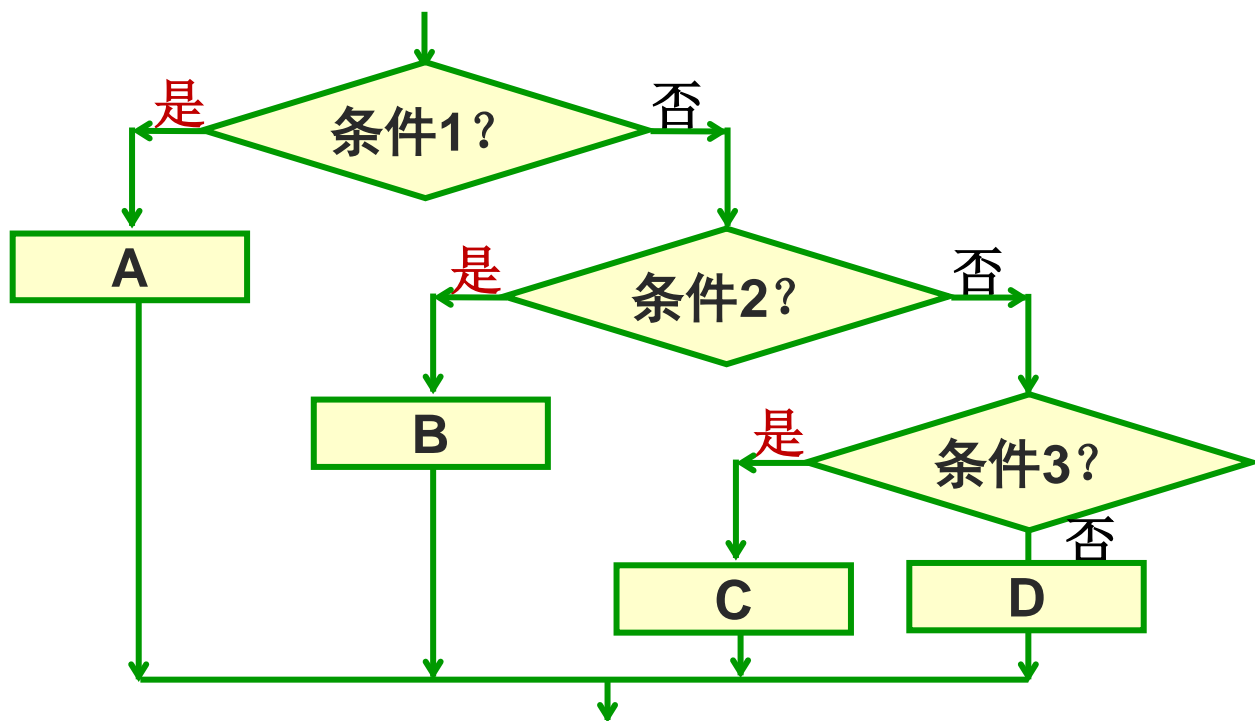
```
if ( 表达式P )  
    语句A  
else  
    语句B
```



# 选择结构

## ■ 多分支选择结构（Multiple Selection）

\* 面临2种以上选择（需连续执行多个条件判断）时

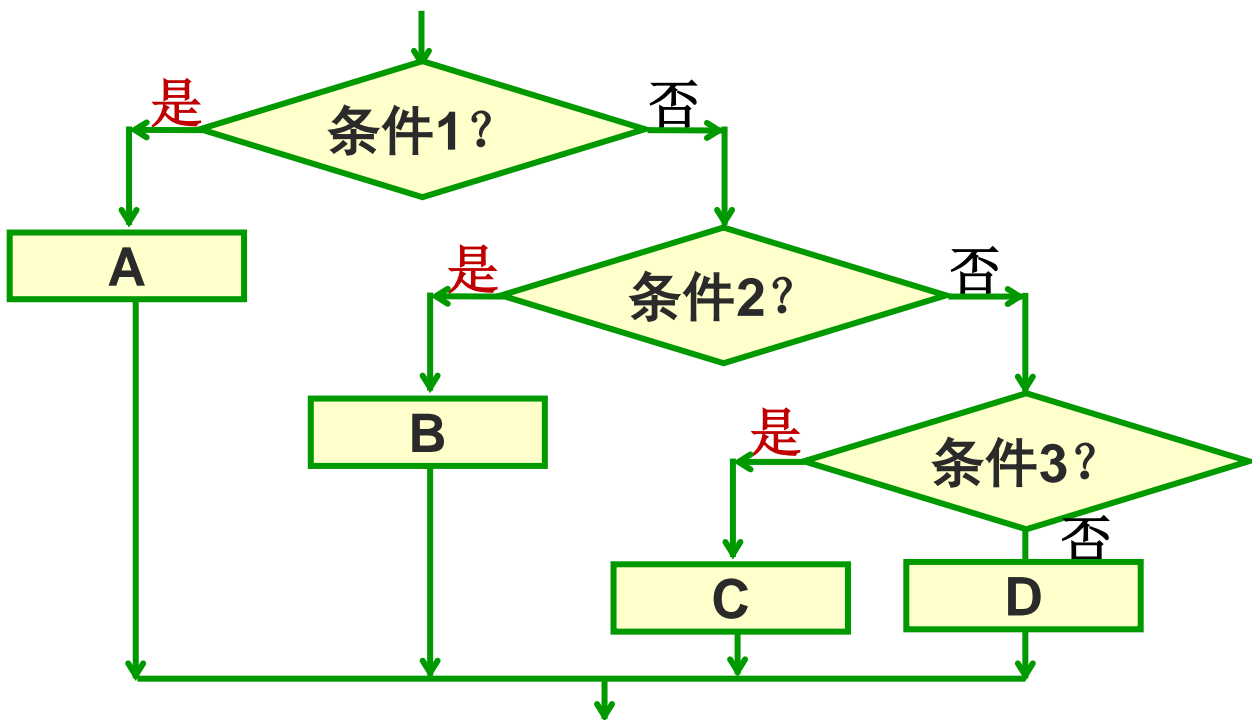




# 选择结构

## ■ 多分支选择结构（Multiple Selection）

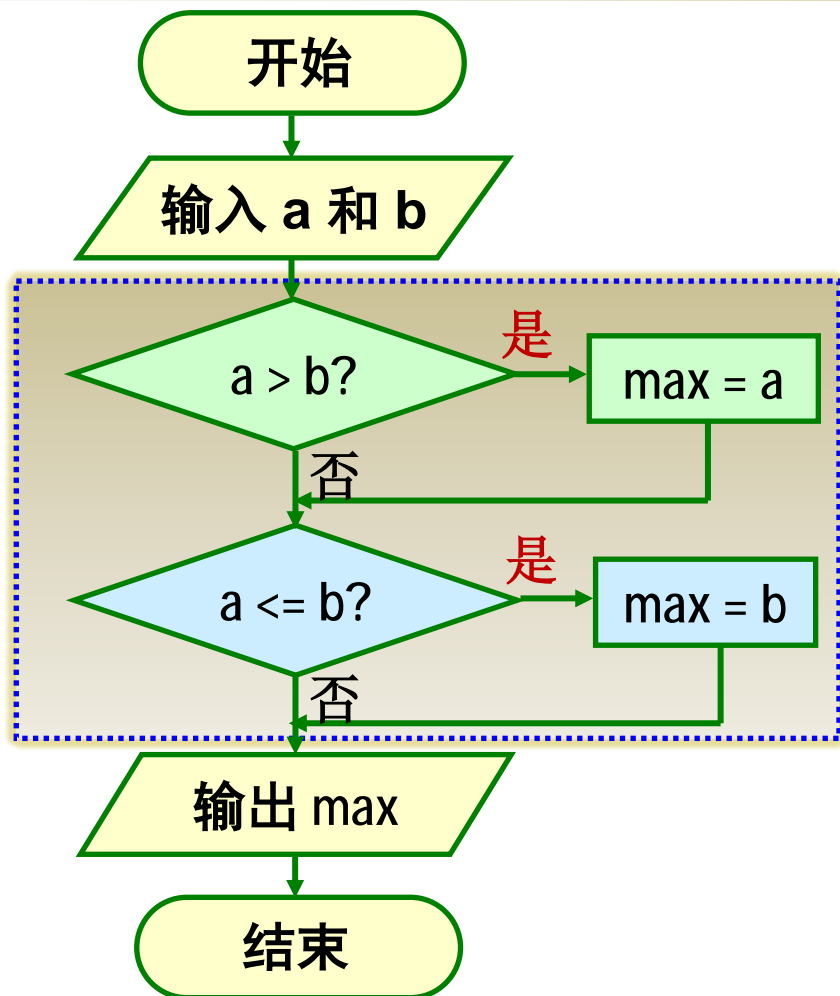
\* 级联式if语句：else-if



```
if (表达式1)
{
    语句A
}
else if(表达式2)
{
    语句B
}
else if (表达式3)
{
    语句C
}
else
{
    语句D
}
```

# 问题：计算两个数的最大值

## ——if语句实现

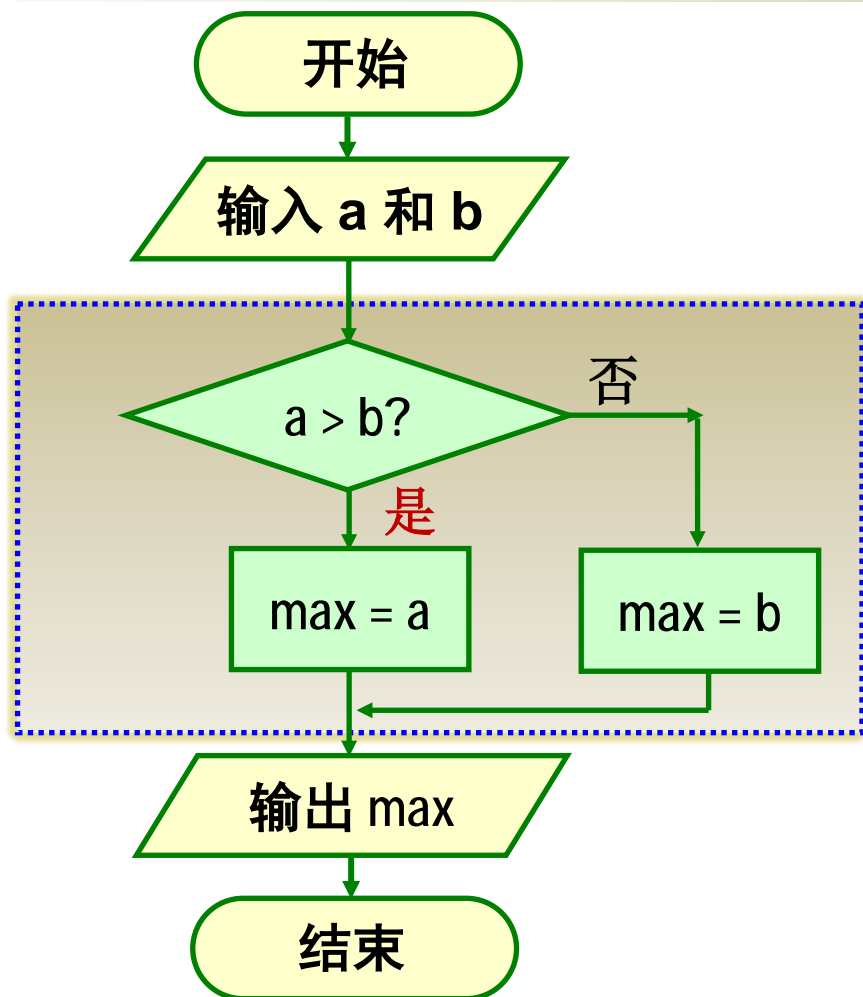


```
#include <stdio.h>
int main()
{
    int a, b, max;
    printf("Input a,b:");
    scanf("%d,%d", &a, &b);
    if (a > b)    max = a;
    if (a <= b)  max = b;
    printf("max = %d\n", max);
    return 0;
}
```

```
Input a,b: 20, 15
max = 20
—
```



## 问题：计算两个数的最大值

——**if-else**语句实现

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

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



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

# 条件运算符和条件表达式

```
#include <stdio.h>
int main()
{
    int a, b, max;
    printf("Input a, b:");
    scanf("%d,%d", &a, &b);

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

    printf("max = %d\n", max);
    return 0;
}
```

表达式1 ? 表达式2 : 表达式3

max = a > b ? a : b;

# 条件运算符和条件表达式

```
#include <stdio.h>
int main()
{
    int a, b, max;
    printf("Input a, b:");
    scanf("%d,%d", &a, &b);
```

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

```
    printf("max = %d\n", max);
    return 0;
```

```
}
```

表达式1 ? 表达式2 : 表达式3

```
printf("max = %d\n", a > b ? a : b);
```

# 复合语句 (Compound Statement)

```
#include <stdio.h>
int main()
{
    int a, b, max;
    printf("Input a, b:");
    scanf("%d,%d", &a, &b);

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

    printf("max = %d", max);
    return 0;
}
```

被当作一条语句看待

在分支中使用  
花括号的好处?

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

# 在分支中使用复合语句的好处

- 向if和else子句中添加语句时不易出错，能保证程序逻辑的正确性
  - \* 编译错误：illegal else without matching if

```
if (a > b)
    max = a;
    printf("max = %d\n", a);
else
    max = b;
    printf("max = %d\n", b);
```

```
if (a > b)
{
    max = a;
    printf("max = %d\n", a);
}
else
{
    max = b;
    printf("max = %d\n", b);
}
```

# 在分支中使用复合语句的好处

## ■ 使代码结构更清晰

```
if (a != 0)
    if (b != 0)
        c = a / b;
else
    printf("Division by zero!\n");
```

else子句究竟属于哪个if语句?



```
if (a != 0)
{
    if (b != 0)
    {
        c = a / b;
    }
    else
    {
        printf("Division by zero!\n");
    }
}
```

# 在分支中使用复合语句的好处

## ■ 使代码结构更清晰

```
if (a != 0)
{
    if (b != 0)
    {
        c = a / b;
    }
}
else
{
    printf("Division by zero!\n");
}
```

```
if (a != 0)
{
    if (b != 0)
    {
        c = a / b;
    }
    else
    {
        printf("Division by zero!\n");
    }
}
```

# 讨论——回头看scanf()

问题：根据scanf()的返回值判断scanf()是否成功读入了指定的数据项数。  
修改程序，使程序在用户输入123a时，能输出如下运行结果：

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

123a✓  
Input error!





# 讨论

用级联形式的`else-if`语句编程实现如下功能：

从键盘任意输入一个年号，判断它是否是闰年。

若是闰年输出“`Yes`”，否则输出“`No`”。

已知符合下列条件之一者是闰年：

- (1) 能被4整除，但不能被100整除；
- (2) 能被400整除。



