

# 第五讲

## 选择语句

# 5、选择语句

## ■5.1 逻辑表达式

■表达式的值：真or假

■若为真,则结果为数值 1

■若为假,则结果为数值 0

■关系运算符

< > <= >=

■优先级

■算术>关系>判等>赋值

■ $i < j < k \rightarrow (i < j) < k$

■ $i < j \ \&\& \ j < k$

■判等运算符

= = ! =

■ $1 < 2.5$	1
■ $3 < 4 > 2$	0
■ $3 + (4 < 3) * 4$	3
■ $3 + 4 < 3 * 4$	1

# 5、选择语句

## ■5.1逻辑表达式

### ■逻辑运算符

!    &&    ||

■操作数的真假判定: 非零值为真, 零值为假

■优先级: ! > && > ||

■求值规则:

条件a	条件b	a&&b	a  b	!a
1	1	1	1	0
1	0	0	1	0
0	1	0	1	1
0	0	0	0	1

```
/* 05-01-expression.c */
#include <stdio.h>
int main(void)
{
    int y, leap;
    printf("Please enter any year:");
    scanf("%d", &y);
    leap = y%4==0&&y%100!=0 || y%400;
    printf("leap:%d\n", leap);
}
```

# 5、选择语句

## 5.2 if语句

- if语句先求出测试表达式的值,然后根据值执行两种操作之一

`if ( expression ) statement`

- 求表达式的值, 如果不为0, 则执行相应的语句

```
if (line_num == MAX_LINES)
    line_num = 0;
```

- 判等: `==`

- 赋值: `=`

- 判断i是否处于某个区间

- $0 \leq i < n$

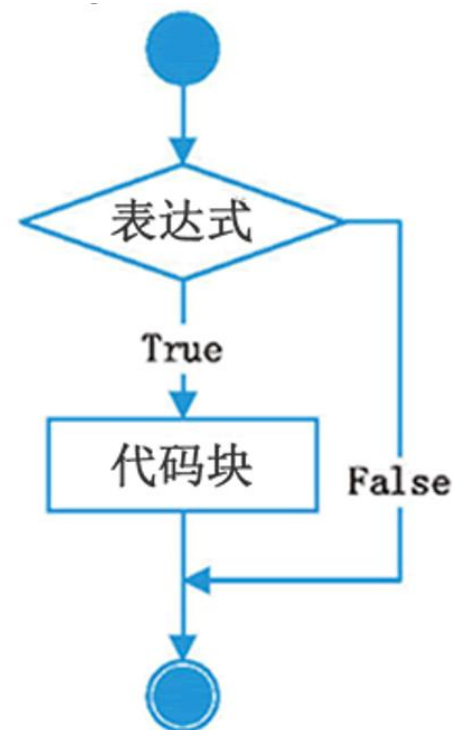
- `if (0 <= i && i < n)`

- 不在区间内

- `if (i < 0 || i >= n)`

```
/* 05-02-if.c */
#include <stdio.h>
int main(void)
{
    int y, leap;
    printf("Please enter any year:");
    scanf("%d",&y);

    if(y%4==0&&y%100!=0||y%400==0){
        printf("%d年是闰年\n",y);
    }
}
```



# 5、选择语句

## ■5.2 if语句

■else子句

■`if ( expression ) statement else statement`

■例

```
if (i > j)
    max = i;
else
    max = j;
```

## ■5.2 if语句

### ■嵌套语句

```
if (i > j)
    if (i > k)
        max = i;
    else
        max = k;
else
    if (j > k)
        max = j;
    else
        max = k;
```

■将其他与匹配的 if 对齐可以使嵌套更容易看到

## ■5.2 if语句

### ■嵌套语句

### ■为避免混淆，加上 “{}”

```
if (i > j) {  
    if (i > k)  
        max = i;  
    else  
        max = k;  
} else {  
    if (j > k)  
        max = j;  
    else  
        max = k;  
}
```

```
if (i > j) {  
    if (i > k) {  
        max = i;  
    } else {  
        max = k;  
    }  
} else {  
    if (j > k) {  
        max = j;  
    } else {  
        max = k;  
    }  
}
```

尽量多添加 “{}”

## ■5.2 if语句

### ■ “级联” if 语句

- 测试一系列条件，一旦其中一个条件成立就停止

```
if (n < 0)
    printf("n is less than 0\n");
else
    if (n == 0)
        printf("n is equal to 0\n");
    else
        printf("n is greater than 0\n");
```

```
if (n < 0)
    printf("n is less than 0\n");
else if (n == 0)
    printf("n is equal to 0\n");
else
    printf("n is greater than 0\n");
```



# 5、选择语句

## ■ 5.2 if语句

### ■ 示例

### ■ 给出一百分制成绩,要求输出对应绩点

■  $<0 || >100$ : Illegal score

■ 90~100: GPA=4

■ 80~89: GPA=3

■ 70~79: GPA=2

■ 60~69: GPA=1

■ 0~60: GPA=0

```
/* 05-03-GPA.c */
#include <stdio.h>
int main(void)
{
    int score,gpa;

    printf("the score?");
    scanf("%d",&score);
    if (score<0 || score>100)
        printf("Illegal grade");
    else {
        if (score>=90) gpa=4;
        else if (score>=80) gpa=3;
        else if (score>=70) gpa=2;
        else if (score>=60) gpa=1;
        else gpa=0;
        printf("The GPA=%d",gpa);
    }
    return 0;
}
```

## ■5.2 if语句

### ■悬空else

- else子句总是与它上面最近的且未曾配对的if配对

```
if (y != 0)
    if (x != 0)
        result = x / y;
else
    printf("Error: y is equal to 0\n");
```

```
if (y != 0)
    if (x != 0)
        result = x / y;
else
    printf("Error: y is equal to 0\n");
```

```
if (y != 0) {
    if (x != 0)
        result = x / y;
} else
    printf("Error: y is equal to 0\n");
```

## ■ 5.2 if语句

■ 条件表达式

■ 条件运算符：  $?$   $:$

■ 必须联合使用：

表达式1  $?$  表达式2  $:$  表达式3

■ 条件运算符：三元运算符

■ 求值步骤：

■ 计算出表达式1的值

■ 如果表达式1的值不为零，那么计算表达式2的值，并且计算出来的值就是整个条件表达式的值

■ 如果表达式1的值为零，那么计算表达式3的值，并且计算出来的值就是整个条件表达式的值

■ *if expr1 then expr2 else expr3.*

```
int i, j, k;
```

```
i = 1;
```

```
j = 2;
```

```
k = i > j ? i : j;
```

```
/* k is now 2 */
```

```
k = (i >= 0 ? i : 0) + j;
```

```
/* k is now 3 */
```

## ■5.2 if语句

### ■条件表达式使用场景

- 虽然短小，但是难以阅读，建议少用
- 在简单明了的地方还是可以用

```
if (i > j)
    max= i;
else
    max= j;
max= i > j ? i : j ;
```

```
if (i > j)
    printf("%d\n", i);
else
    printf("%d\n", j);

printf("%d\n", i > j ? i : j) ;
```

```
if (i > j)
    return i;
else
    return j;

return i > j ? i : j ;
```

## ■5.3 switch语句

### ■级联if语句

```
if (grade == 4)
    printf("Excellent");
else if (grade == 3)
    printf("Good");
else if (grade == 2)
    printf("Average");
else if (grade == 1)
    printf("Poor");
else if (grade == 0)
    printf("Failing");
else
    printf("Illegal grade");
```

### ■switch语句

```
switch (grade) {
    case 4: printf("Excellent");
            break;
    case 3: printf("Good");
            break;
    case 2: printf("Average");
            break;
    case 1: printf("Poor");
            break;
    case 0: printf("Failing");
            break;
    default: printf("Illegal grade");
            break;
};
```

## ■5.3 switch语句

### ■格式

```
switch ( expression ) {  
    case constant-expression : statements  
    ...  
    case constant-expression : statements  
    default : statements
```

### ■控制表达式

- switch后面必须跟由括弧括起来的整型表达式或字符表达式, 不能是浮点数和字符串。

### ■分支标号

- case后面只能跟一个常量表达式, 该表达式只能是整型表达式或字符表达式, 不能包含变量和函数调用
- 不允许有重复的分支标号

### ■语句

- 每个分支标号后可以跟多个语句。
- 分支的顺序没有关系, default也不一定要在最后面
- 有break语句时, 分支标号的顺序不影响程序运行结果。
- break语句不是必须的, 但是一般会带

## ■ 5.3 switch语句

- 多个标签位于一组语句前

```
switch (grade) {  
    case 4:  
    case 3:  
    case 2:  
    case 1:    printf("Passing");  
                break;  
    case 0:    printf("Failing");  
                break;  
    default:   printf("Illegal grade"); }  
}
```

```
switch (grade) {  
    case 4: case 3: case 2: case 1:  
        printf("Passing");  
        break;  
    case 0:    printf("Failing");  
                break;  
    default:   printf("Illegal grade");  
                break;  
}
```

## ■ 5.3 switch语句

### ■ break语句

- 程序“跳”出switch语句，继续执行switch后面的语句
- 没有break语句，控制会从一个分支继续到下一个分支

```
grade=3;
switch (grade) {
    case 4: printf("Excellent");
    case 3: printf("Good");
    case 2: printf("Average");
    case 1: printf("Poor");
    case 0: printf("Failing");
    default: printf("Illegal grade");
}
```

```
switch (grade) {
    case 4: case 3: case 2: case 1:
        num_passing++;
        /* FALL THROUGH */
    case 0: total_grades++;
        break;
}
```

GoodAveragePoorFailingIllegal grade



## ■ 5.3 switch语句

### ■ 示例

- 用户以月/日/年的格式录入日期，然后计算机显示出“法定”格式的日期：

Enter date (mm/dd/yy): 7/19/24

Dated this 19th day of July, 2024.

- 04-date.c