

# 第4章

## 选择结构程序设计

### 4.1 项目一：单分支 if 语句

#### 4.1.1 任务：字符检测与输出

##### 一、任务分析

首先应该定义一个 char 类型变量用于保存从键盘输入的字符,然后判断该字符是否是小写字母,如果是则输出它不是就结束程序。

##### 二、任务实施

该任务的实现流程如图 4-1 所示。程序清单如下：

```
#include "stdio.h"
main()
{
    char c;
    c = getchar();
    if(c >= 'a' && c <= 'z')
        printf("\n%c\n", c);
}
```

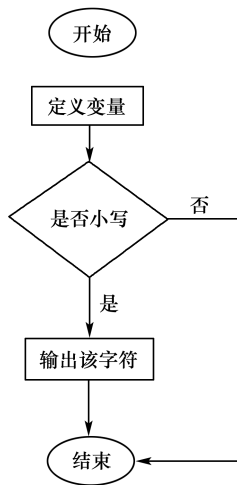


图 4-1 字符检测与输出的实现流程

#### 4.1.2 知识准备

##### 1. 单分支 if 语句的书写格式

if(条件) {语句组;}

说明：

- (1) 条件通常是一个关系表达式或逻辑表达式,当然也可以是一个逻辑量。
- (2) 语句组可以是一条语句也可以是多条语句,当只有一条语句时,花括号可以省略。

## 2. 单分支 if 语句的执行流程

单分支 if 语句的执行流程如图 4-2 所示。当条件为真时,执行语句组,然后结束 if 语句,如果条件为假,则直接结束 if 语句,语句组不被执行。

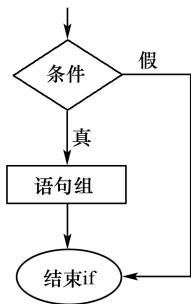


图 4-2 单分支 if 语句的执行流程

## 4.2 项目二:双分支 if 语句

### 4.2.1 任务 1:任意输入两个不相等的整数,输出较大的数

#### 一、任务分析

定义两个整型变量 a,b 用于存储输入的两个不相等整数,然后对两个数进行比较,如果  $a > b$  则输出 a 的值,否则输出 b 的值。

#### 二、任务实施

该任务的实现流程如图 4-3 所示。程序清单如下:

```

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

### 4.2.2 知识准备

#### 1. 双分支 if 语句的书写格式

```

if(条件)
    {语句组 1;}
else
    {语句组 2;}
  
```

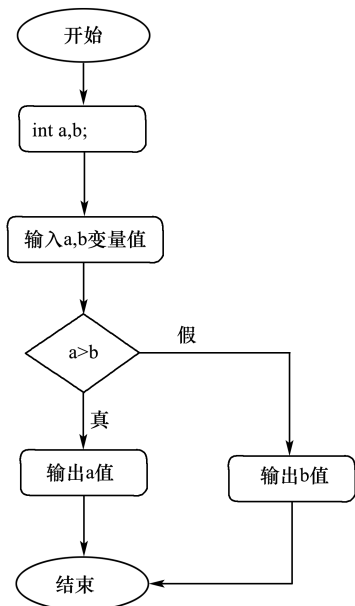


图 4-3 任务的实现流程

格式说明：

(1)语句组 1 和语句组 2 可以是一条语句也可以多条语句,当只有一条语句时花括号可以省略。

(2)else 后没有条件

## 2. 双分支 if 语句的执行流程

双分支 if 语句的执行流程如图 4-4 所示。当条件为真时,执行语句组 1,然后结束 if 语句;当条件为假时,执行语句组 2,然后结束 if 语句。

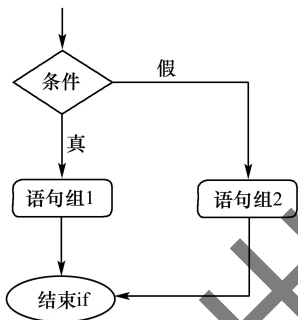


图 4-4 双分支 if 语句的执行流程

## 4.3 项目三:if 语句的嵌套

### 4.3.1 任务:检测键盘输入的字符类型

#### 一、任务描述

检测键盘输入的字符类型,如果输入的是字母则输出“输入为字母字符”,若输入的是数字则输出“输入为数字字符”,否则输出“输入为其他字符”。

#### 二、任务分析

定义一个 char 类型变量来存放从键盘输入的一个字符,然后判断该字符是否属于字母(大写和小写都可以),若是则输出“输入为字母字符”,否则继续判断该字符是否为数字字符,若是则输出“输入为数字字符”,否则输出“输入为其他字符”。

#### 三、任务实施

该任务的实现流程如图 4-5 所示。程序清单如下：

```
#include "stdio.h"

main()
{
    char c;
    c = getchar();
```

```

if(c >= 'a' && c <= 'z' || c >= 'A' && c <= 'Z')
    printf("输入为字母字符");
else
    if(c >= '0' && c <= '9')
        printf("输入为数字字符");
    else
        printf("输入为其他字符");
}

```

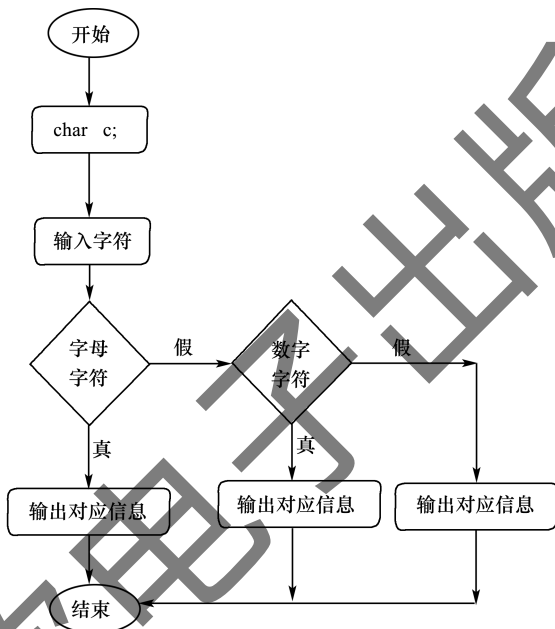


图 4-5 任务的实现流程

### 4.3.2 知识准备

if 语句实现 3 选 1 的两种嵌套形式及执行流:

嵌套形式 1 及执行流程:

```

if(条件 1)
    if(条件 2)
        {语句组 1}
    else
        {语句组 2}
else
    {语句组 3}

```

该嵌套形式的执行流程如图 4-6 所示。

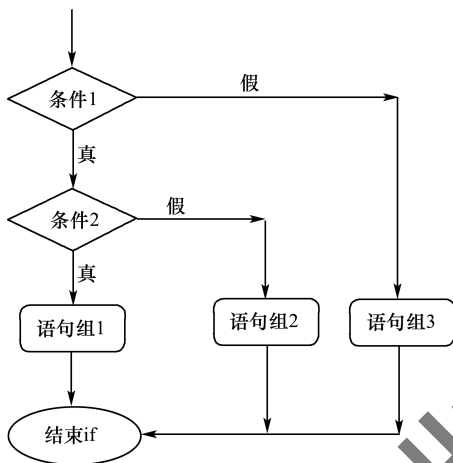


图 4-6 嵌套形式的执行流程图

嵌套形式 2 及执行流程:

```

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

该嵌套形式的执行流程如图 4-7 所示。

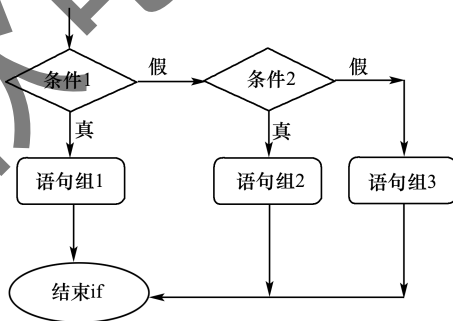


图 4-7 嵌套形式的执行流程图

### 4.3.3 任务 2: 检测键盘输入的字符类型

#### 一、任务描述

检测键盘输入的字符类型,若为小写字母则输出“输入为小写字母”,若为大写字母则输出“输入为大写字母”,数字字符则输出“输入为数字字符”,其他符号则输出“输入为其他字符”。

## 二、任务分析

首先定义一个字符型变量 C 用于保存从键盘输入的一个字符,然后对这个字符进行判断,若为小写字母则输出“输入为小写字母”,否则继续判断输入是否为大写字母,若是则输出“输入为大写字母”,否则再继续判断输入是否为数字字符,若是则输出“输入为数字字符”,否则输出“输入为其他字符”。

## 三、任务实施

该任务的实现流程如图 4-8 所示。程序清单如下:

```
#include "stdio.h"
main()
{
    char c;
    c = getchar();
    if(c >= 'a' && c <= 'z')
        printf("输入为小写字母\n");
    else
        if(c >= 'A' && c <= 'Z')
            printf("输入为大写字母\n");
        else
            if(c >= '0' && c <= '9')
                printf("输入为数字字符\n");
            else
                printf("输入为其他字符\n");
}
```

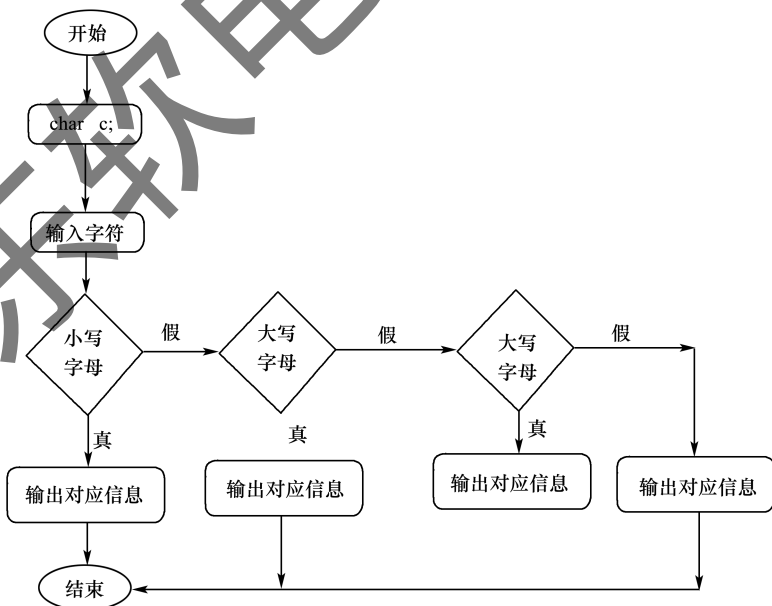


图 4-8 任务实现流程图

### 4.3.4 知识准备

if 语句实现多选一的一般嵌套形式及执行流程:

一般嵌套形式:

```
if(条件 1)
    {语句组 1}
else
    if(条件 2)
        {语句组 2}
    else
        .....
    if(条件 n)
        {语句组 n}
    else
        {语句组 n+1}
```

执行流程如图 4-9 所示。

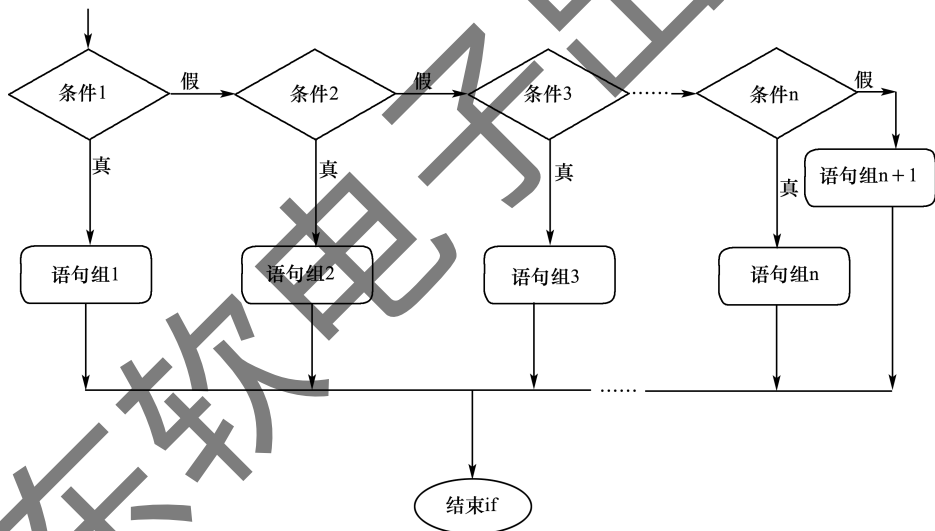


图 4-9 if 语句实现多选一的一般嵌套形式的执行流程

## 4.4 项目四:switch 和 break 语句

### 4.4.1 任务:用 switch 语句将百分制成绩转换成五级制输出

#### 一、任务描述

将百分制成绩转换成五级制成绩输出。即 90~100 分输出“A”,80~89 分输出“B”,70~79 分输出“C”,60~69 分输出“D”,60 分以下输出“E”。

## 二、任务分析

首先定义一个整型(约定百分制成绩是 0—100 之间的整数)变量 score 用于存放学生的成绩值,然后判断该成绩在哪一个范围,然后输出对应的等级制。

## 三、任务实施

```
#include "stdio.h"

main()
{
    int score;
    printf("输入 0 - 100 之间成绩值:\n");
    scanf("%d",&score);
    switch(score/10)
    {
        case 10:
        case 9: printf("\nA\n");break;
        case 8: printf("\nB\n");break;
        case 7: printf("\nC\n");break;
        case 6: printf("\nD\n");break;
        default: printf("\nE\n");
    }
}
```

### 4.4.2 知识准备

#### 1. switch 语句的书写格式

```
switch(表达式 0)
{
    case 常量表达式 1: 语句组 1;break;
    case 常量表达式 2: 语句组 2;break;
    case 常量表达式 3: 语句组 3;break;
    .....
    case 常量表达式 n: 语句组 n;break;
    default: 语句组 n+1;
}
```



## 2. 执行流程

switch 语句的执行流程如图 4-10 所示。

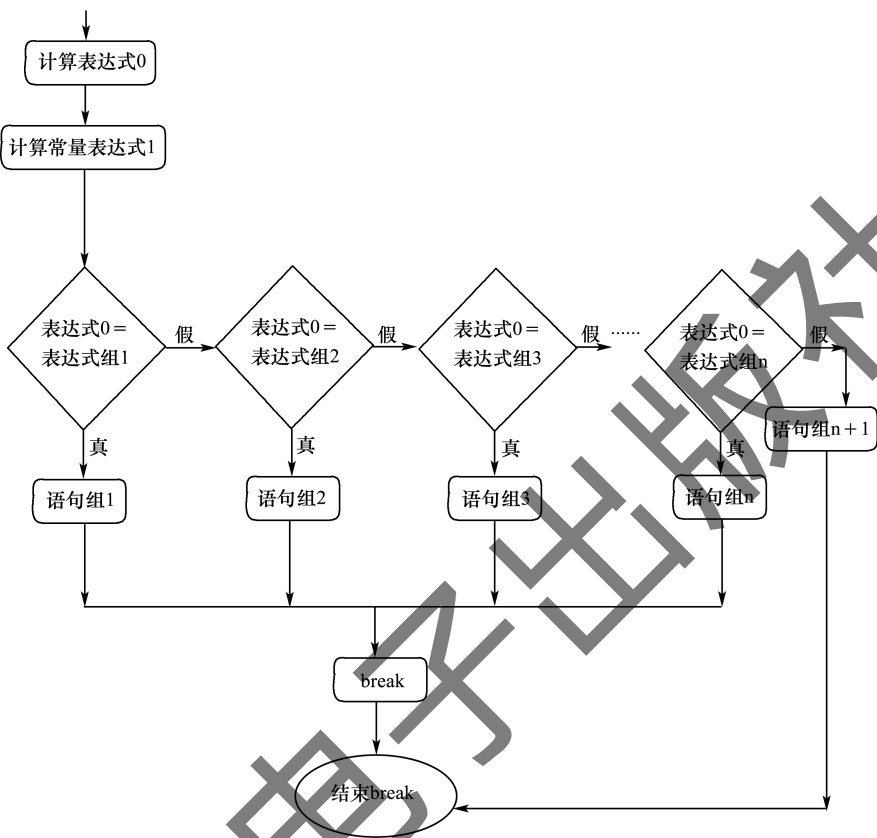


图 4-10 switch 语句的执行流程

## 3. break 语句的功能

跳出 switch 语句即强制结束 switch 语句的执行。

## 4.5 本章知识强化

**【实例 1】** 输入三个任意的整数,按从大到小的顺序输出。

解析:定义 a,b,c 三个变量用于保存键盘输入的三个整数,然后对这三个数进行比较,将最大数存放在 a 中,较大数存放在 b 中,最小数存放在 c 中即可。

程序清单:

```
#include <stdio.h>

main()
{
    int a,b,c,temp;
    scanf("%d,%d,%d",&a,&b,&c);
    if(a<b) {temp = a; a = b; b = temp;}
    if(a<c) {temp = a; a = c; c = temp;}
```

```
if(b<c) {temp = b; b = c; c = temp;}
printf("%d, %d, %d", a, b, c);
}
```

**【实例 2】** 编程实现三分段函数:当  $x < 0$  时,  $y = -1$ ; 当  $x = 0$  时,  $y = 1$ ; 当  $x > 0$  时,  $y = 1$ 。

程序清单:

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

**【实例 3】** 输入一个任意的三位正整数, 然后逆序输出该三位数。如输入 347, 输出为 743。

程序清单:

```
#include <stdio.h>
main()
{
    int a, b, c, x;
    scanf("%d", &x);
    a = x/100;
    b = x/10 % 10;
    c = x % 10;
    printf("%d %d %d\n", c, b, a);
}
```

**【实例 4】** 输入三角形的三条边长, 输出其面积。

```
#include "stdio.h"
#include <math.h>
main()
{
    int a, b, c;
    float l, s;
    scanf("%d, %d, %d", &a, &b, &c);
```

```
l = (a + b + c) / 2.0;
if(a + b > c && a + c > b && b + c > a)
    {s = sqrt(l * (l - a) * (l - b) * (l - c));
    printf("s = %f", s);}
else
    printf("不能构成三角形! \n");
}
```

**【实例 5】** 输入圆的半径和一个选项整数,当选项整数为 0 时,输出圆的面积;当选项整数为 1 时,输出圆的周长;当选项整数为 2 时,输出圆的面积和周长;否则输出“没有定义这个功能!”。

程序清单 1:

```
#include <stdio.h>
#define P 3.14
main()
{
    int n;
    float r, l, s;
    scanf("%d, %f", &n, &r);
    if(n == 0)
        {s = P * r * r;
        printf("%.2f\n", s);}
    else
        if(n == 1)
            {l = 2 * P * r;
            printf("%.2f\n", l);}
        else
            if(n == 2)
                {s = P * r * r;
                l = 2 * P * r;
                printf("%.2f\n", s);
                printf("%.2f\n", l);}
            else
                printf("没有定义这个功能! \n");
}
```

程序清单 2:

```
#include "stdio.h"
#define P 3.14
main()
{
    int n;
    float r, l, s;
```

```
scanf("%d,%f",&n,&r);
s=P*r*r;
l=2*P*r;
switch(n)
{
    case 0:printf("%.2f\n",s);break;
    case 1:printf("%.2f\n",l);break;
    case 2:printf("%.2f,%.2f\n",s,l);break;
    default:printf("没有定义这个功能!\n");
}
}
```

## 4.6 本章小结

本章主要讲述了选择结构的程序设计方法和控制结构,分述如下:

1. 单分支 if 语句的格式和执行流程;
2. 双分支 if 语句的格式和执行流程;
3. if 语句的嵌套形式和执行流程;
4. switch 语句的格式和执行流程。

## 习题四

### 一、填空题

1. 写出下列命题的 C 语言描述。

- (1) x 小于 y 或小于 z ( );
- (2) x 和 y 都小于 z ( );
- (3) x 或 y 中有一个小于 z ( );
- (4) x 是偶数 ( )。

2. 根据以下 if 语句写出与其功能相同的 switch 语句(x 的值在 0—100 之间的正整数)。

```
if 语句:
if(x<40)
b=1;
else
if(x<50)
b=11;
else
if(x<60)
b=111;
else
if(x<70)
```

```
b = 1111;
    else
if(x<80)
b = 11111;
switch 语句:
switch(      )
{
    b = 1;break;
case 4: b = 11;break;
case 5: b = 111;break;
case 6: b = 1111;break;
    b = 11111;
}
```

3. 如下程序段:

```
switch(rank)
{
    case 1:printf("perfect! \n");
    case 2:printf("good! \n");
    case 3:printf("just soso! \n");
    case 4:printf("bad! \n");
default:printf("impossible! \n");
}
```

若 rank 的值为 3,则输出结果为\_\_\_\_\_。

4. 把以下两条 if 语句合并成一条 if 语句\_\_\_\_\_。

```
if(m<=n) x = 100;
else y = 50;
if(m>n) printf("y = %d\n",y);
else printf("x = %d\n",x);
```

二、阅读下面的程序,将问题的答案填入对应的划线处。

```
#include <stdio.h>
main()
{ int a,b,m,n;
    scanf("%d %d",&a,&b);
m = 1;n = 1;
if(a>0) m = m + n;
if(a<b) n = 2 * m;
else if(a == b) n = 5;
    else n = m + n;
printf("m = %d n = %d\n",m,n);
}
```

- (1) 当输入 1 - 2 回车后, 程序的运行结果为\_\_\_\_\_。
- (2) 当输入 1 0 回车后, 程序的运行结果为\_\_\_\_\_。
- (3) 为了输出  $n=4$ , 变量  $a$  和  $b$  应具备条件\_\_\_\_\_。

### 三、选择题

1. 两次运行下面的程序, 如果从键盘上分别输入 12 和 4, 输出结果是( )。

```
main()
{
    int m;
    scanf("%d",&m);
    if(m++>11)
    printf("%d",m);
    else
        printf("%d\n",m--);
}
```

- A. 13 和 5                      B. 12 和 5                      C. 12 和 6                      D. 13 和 6

2. 如下程序的运行结果是( )。

```
main()
{
    int a=5,b=2,c=0;
    if(c<0)
        if(b<0)
a=3;
    else
a=4;
    printf("%d\t",a);
    c=a;
    if((c=b)<0)
a=3;
    else
if(b==0)
a=5;
    else
a=7;
    c=a-c;
    printf("%d\t",a);
    printf("%d\t",c);
}
```

- A. 5 7 2                      B. 3 7 2                      C. 5 7 5                      D. 3 5 7

3. 以下哪一个是判断 char 变量 ch 为小写字母的表达式( )。

- A.  $ch \geq 'a'$  AND  $ch \leq 'z'$                       B.  $ch \geq 'a'$  &  $ch \leq 'z'$
- C.  $ch \geq 'a'$  &&  $ch \leq 'z'$                       D.  $ch \geq 'a'$  ||  $ch \leq 'z'$

4. 假定所有变量都已经正确定义,下列程序段运行后 m 的值是( )。

```
a = b = 0; c = 3; m = 24;
if(! a)
    m = - 1;
else
    if(b) ;
if(c)
    m = 3
else
    m = 4;
```

A. -1

B. 3

C. 24

D. 3

5. 如下程序的执行结果是( )。

```
main()
{
    int m, n = 1;
    if(n! = 0) m = 6;
    printf("% d\t", m);
    if(n = 0) m = 12;
    else m = 7;
    printf("% d\t\n", m);
}
```

A. 6 12

B. 6 0

C. 6 5

D. 6 7

#### 四、编程题

1. 编程实现求任意一个  $ax^2 + bx + c = 0$  方程的根。

2. 编程实现分段函数:当  $x < 0$  时,  $y = x$ ; 当  $0 \leq x < 10$  时,  $y = 2x + 1$ ; 当  $x \geq 10$  时,  $y = 4x - 3$ 。

3. 编程实现:输入一个不多于 5 位的正整数:

(1) 求出它是几位数;

(2) 分别打印出每一位数字;

(3) 按逆序打印出各位数字,例如原数据为 3458,则应输出结果 8543。

要求使用以下测试数据:

要处理的数为 1 位正整数;

要处理的数为 2 位正整数;

要处理的数为 3 位正整数;

要处理的数为 4 位正整数;

要处理的数为 5 位正整数;

除此之外,程序还应当对不合法的输入做不必要的处理,如当输入的是负数或大于 5 位数则输出“输入数据错误!”。