

第 6 章 循环语句

课后练习题

1. 填空题

(1) 执行下面程序后，变量 `result` 的值是_____，函数 `function()` 的功能是_____。

```
int function(int n){
    int k = 0;
    do {
        k = k + n%10;
        n = n/10;
    } while(n);
    return k;
}

int main(void){
    int result = function(3456);
    return 0;
}
```

(2) 填空完成下面程序，其功能是打印 100 以内个位数为 6 且能被 3 整除的所有正整数。

```
int main(void){
    int i,j;
    for (i=0;_____;i++){
        j = i*10 + 6;
        if (_____)
            continue;
        printf("%d ",j);
    }
    printf("\n");
    return 0;
}
```

2. 改写下列程序段，将其中的 for 循环改用 while 循环实现，要求功能不变。

(1) 程序段 1:

```
int fact(int n){
    int i;
    int result = 1;
    for (i=2; i <= n; i++)
        result = result * i;
    return result;
}
```

(2) 程序段 2:

```
int main(void) {
    int sum = 0;
    int i;
    for (i = 0; i < 10; i++){
        if (i % 2 == 1)
            continue;
        sum = sum + i;
    }

    return 0;
}
```

3. 按照下述思路，编写一个函数，用于计算一个正数的平方根。

正数 n 的平方根可以通过计算一系列近似值来获得，每个近似值都比前一个更加接近准确值。第一个近似值是 1，接下来的近似值则通过下面的公式来获得：

$$a_{i+1} = \frac{a_i + \frac{n}{a_i}}{2}$$

编写一个程序，读入一个值，计算并打印它的平方根。如果你将所有的近似值都打印出来，你会发现这种方法获得准确结果的速度有多快。原则上，这种计算可以永远进行下去，它会不断产生更加精确的结果。但在实际中，由于浮点变量的精度限制，程序无法抑制计算下去。当某个近似值与前一个近似值相等时，你就可以让程序停止继续计算了。

通过与 C 标准数学库中的 `sqrt()` 函数的计算结果对比来测试你的程序是否正确。

4. 编写程序，输出如下字母塔：

```
A
ABA
ABCBA
ABCD CBA
ABCDEDCBA
ABCDEFEDCBA
ABCDEF GFEDCBA
ABCDEF GHGFEDCBA
ABCDEF GHIHGFEDCBA
ABCDEF GHIJKIHGFEDCBA
ABCDEF GHIJKLIHGFEDCBA
ABCDEF GHIJKLMLKIHGFEDCBA
ABCDEF GHIJKLMNMLKIHGFEDCBA
ABCDEF GHIJKLMNOPONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRSRQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRSTSRQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRSTUTSRQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRSTUUVTSRQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRSTUUVWUITSRQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRSTUUVWXYXWUITSRQONMLKIHGFEDCBA
ABCDEF GHIJKLMNOPQRSTUUVWXYZZYXWUITSRQONMLKIHGFEDCBA
```

5. 编写一个程序，提示用户从键盘输入两个正整数，在屏幕上输入这两个正整数的最大公约数和最小公倍数。

6. 编写一个程序，要求用户输入下限整数和上限整数，然后，依次计算从下限到上限的每一个整数的平方的加和，最后显示结果。程序将不断提示用户输入下限整数和上限整数并显示出答案，直到用户输入的上限整数等于或小于上限整数为止。程序运行的结果示例应该如下所示：

```
请输入上限和下限：5 9
25 至 81 之间的完全平方数之和为 255
请输入下一组上下限：3 25
9 至 625 之间的完全平方数之和为 5520
请输入下一组上下限：5 5
退出程序！
```

7. 鸡兔同笼是我国古代著名趣题之一。大约在 1500 年前,《孙子算经》中就记载了这个有趣的问题。书中是这样叙述的:“今有雉兔同笼,上有三十五头,下有九十四足,问雉兔各几何?”这四句话的意思是:有若干只鸡兔同在一个笼子里,从上面数,有 35 个头;从下面数,有 94 只脚。问笼中各有几只鸡和兔?

1) 以下程序用于求解鸡兔同笼问题的解,请补充完整:

```
int main(void){
    int x,y;
    for (x=1;x!=35;x++){
        y = 35 - x;
        if(_____)
            printf("笼中有%d只鸡, %d只兔。",x,y);
    }
    return 0;
}
```

2) 以上程序的运行结果是: _____;

3) 这一问题也可以采用一元一次方程来求解,请说明解题思路,并给出相应的程序。

要求: 让用户分别输入笼中鸡和兔的头和足的数量,并对用户输入进行判断。若输入数据类型正确且合理,则计算结果;若输入的不是正整数,则提示输入数据类型错误,并要求用户重新输入;若输入数据类型正确但求解结果不满足条件(求解结果不是整数、鸡或兔的数量为负数),则提示输入数据不合理,并退出程序。