第六章—循环结构进阶

注意：本次课是面向所有想学和爱学的同学们，所以，希望大家能真真正正的在这里学到技能，这是我们安排的相应课后作业，希望大家认真完成，及时提交到 QQ 群里，QQ 群号：344401117，加群请备注。不然我们不会通过哦，因为这个群也是服务于我们此次直播课的学生。最后，大家有任何的疑问和建议都可以在群里提出来哦，我们虚心倾听。也希望大家帮助我们扩散、宣传，我们希望让更多想学和爱学的同学们听见我们的声音。

跟我练一

指导-统计大写字母和小写字母的个数

需求说明

编写一个程序，用于接收用户输入 10 个字符，统计其中大写字母和小写字母的个数。比较大写字母与小写字母的个数，并显示相应的消息。

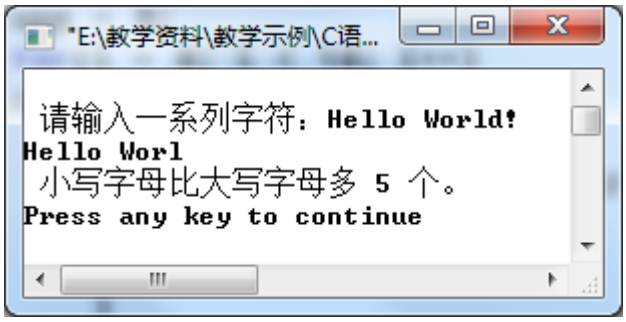
实现思路

此程序需要 3 个整型变量和一个字符型变量。程序的输入将存储在字符型变量中。一个整型变量作为 for 循环的计数器，统计接收的字符数，直到字符的个数达到 10，循环结束。另外两个整型变量用作小写字母和大写字母的计数器。用户可输入任何值，程序将利用输入字符的 ASCII 值检查这些字符是否为大写字母或小写字母。

参考解决方案

```
#include <stdio.h>
void main()
{
    char inp;
    int i, low, upp;
    /*
        for循环的计数器是 i,
        统计大写字母的计数器是 upp,
        统计小写字母的计数器是 low
    */
    printf("\n 请输入一系列字符：");
    low = 0;
    upp = 0;
    for(i = 0; i < 10; i++)
    {
        inp = getchar();
        if(inp >= 'a' && inp <= 'z')
        {
            low++;
        }
        else if(inp >= 'A' && inp <= 'Z')
        {
            upp++;
        }
        putchar(inp);
    }
    /* 比较大写字母和小写字母的个数 */
    if(low > upp)
        printf("\n 小写字母比大写字母多 %d 个。\n", low - upp);
    else if(upp > low)
        printf("\n 大写字母比小写字母多 %d 个。\n", upp - low);
    else
        printf("\n 小写字母和大写字母的个数相等，为 %d 个。\n", low);
}
```

程序的输出结果如下图所示：



跟我练二

指导—求输入的数字之和

需求说明

编写一个程序，最多接收 10 个数字，求这些数字之和。用户可以通过输入 999 终止程序，并显示输入的数字之和。

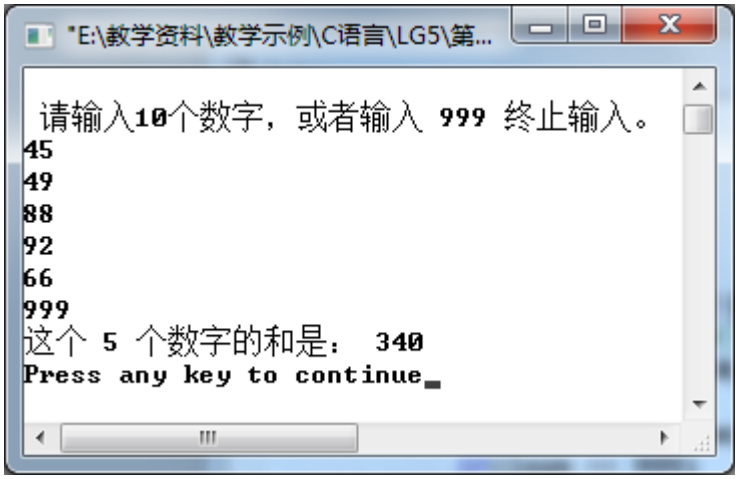
实现思路

该程序需要 3 个整型变量，一个用作循环计数器，一个用来存储用户输入的数字，另一个用来存储累加和。如果用户输入 999，则终止程序，即如果输入 999，使用 break 语句跳出循环。

参考解决方案

```
#include <stdio.h>
void main()
{
    int inum, sum, cnt;
    sum = 0;
    printf("\n 请输入10个数字，或者输入 999 终止输入。\n");
    /* 使用for循环接收10个数字 */
    for(cnt = 0; cnt < 10; cnt++)
    {
        scanf("%d", &inum);
        if(inum == 999)
        {
            /* 如果该数是999则退出for循环 */
            break;
        }
        /* 将输入的数字累加到sum变量中 */
        sum = sum + inum;
    }
    printf("这个 %d 个数字的和是： %d\n", cnt, sum);
}
```

程序的输出结果如下图所示：



跟我练三

指导—打印空心菱形图案

需求说明

用星号打印一个如下的空心菱形图案：

```
   *
  * *
 *   *
  * *
   *
```

实现思路

从图案可以看出，每行的空格数和打印”*”的位置是有规律的，前四行和后三行可以分别打印。前四行的规律为：第 i 行（i 从 1 开始到 4），首先打印 4 – i 个空格，然后考虑下接下来从 1 到 2 * i – 1 位置处打印的内容，只在边界处打印”*”，在中间位置打印空格。后三行的第 i 行同样只在边界处打印”*”，在中间位置打印空格。
需要使用嵌套循环，外层循环控制行数，内层循环打印每一行中的内容。

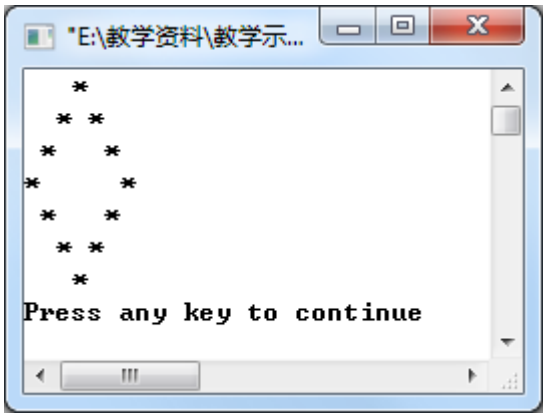
参考解决方案

```
#include <stdio.h>
void main()
{
    int i, j, k;
    /*
        i控制打印的行数
        j控制打印的空格数
        k控制打印的星号个数
    */

    //先打印上边的4行
    for(i = 1; i <= 4; i++)                    //控制要打印的行数
    {
        for(j = 1; j <= 4 - i; j++)            //控制每行要打印的空格数
        {
            printf(" ");
        }
        for(k = 1; k <= 2 * i - 1; k++)
        {
            if(k == 1 || k == 2 * i - 1)       //只在循环的边界值打印星号，否则打印空格
                printf("*");
            else
                printf(" ");
        }
        printf("\n");
    }

    //然后打印下边的3行
    for(i = 1; i <= 3; i++)
    {
        for(j = 1; j <= i; j++)
        {
            printf(" ");
        }
        for(k = 1; k <= 7 - 2 * i; k++)
        {
            if(k == 1 || k== 7 - 2 * i)
                printf("*");
            else
                printf(" ");
        }
        printf("\n");
    }
}
```

程序的输出结果如下图所示：



跟练四

练习

需求说明

编写一个程序，最多接收 10 个数，并求出其中所有正数的和。用户可以通过输入 999 终止程序，统计用户输入的正数的个数，并显示这些正数的和。

提示：

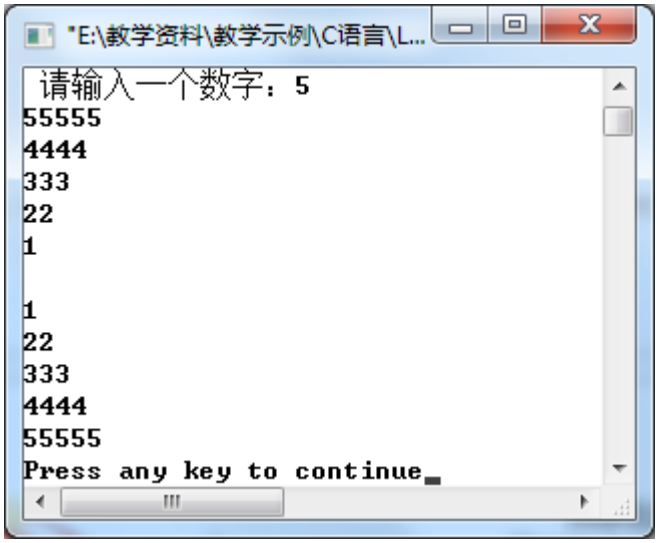
- 参考上机练习 2
- 如果输入的数是负数，则忽略该数，不做累加，然后接收下一个数（使用 continue 语句）

跟我练五

练习

需求说明

编写一个程序，用于生成如下图所示的输出结果，要求根据用户输入的一个整数，输出一个数字组成的图案。



提示：

1) 将图案分成上下两部分分别打印
2) 使用嵌套循环，外层循环控制打印的行数，内层循环控制该行打印的数字及打印的个数

课后作业

1、 求 1 + 2! + 3! + … + 10!的和
2、 以表格格式显示 1 到 10 的乘法表，如下图所示：