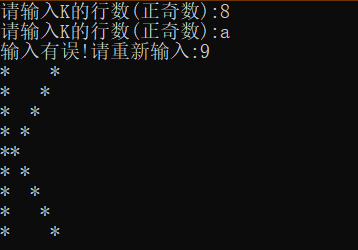
# Section1:

## 运行结果:



## 思路:

观察到字母K结构上有一个巨大的特点---上下以(n+1)/2这一行对称,十分适合使用递归。然后我的思路就此而发，以(n+1)/2行为递归的终点，从上到下进行输出，而递归函数中又包含着一个while语句来进行每行具体内容的输出，而逐步进入递归深处和逐步退出递归刚好满足输出空格从一个一个变少变成一个一个变多，于是只需要使用与上面一样的while语句即可，于是我对每行输出的while语句封装成一个新的函数。思路大概如此。

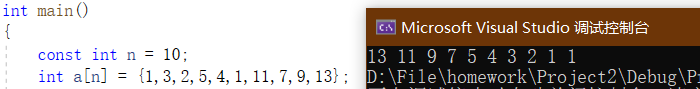
由于K是关于(n+1)/2行对称的字母，于是从输入的行数n必须为奇数，于是为了增加程序健壮性，于是就加入了如下判断如果输入数为非正数或者非数字就要求重新输入，对于输入的小数进行四舍五入。

## 代码：

Git：<https://github.com/silidada/homework.git>

# Section2：

## 运行结果：



## 思路:

对于排序第一时间想到冒泡排序法，但是由于要求只能用一层for循环，于是想到使用递归来代替其中一个for 循环，秉着与上一section一致的思路---递归递整体，循环循小处，于是就想到要通过递归来控制比较的数位，而for则将该数位下的数进行排序，近似于用递归代替原来冒泡排序两层循环中的第一层循环。

思路大概如此。

## 代码：

Git：<https://github.com/silidada/homework.git>