

	i	*	
^^^*	0	1	3
^^***	1	3	2
^*****	2	5	1
*****	3	7	0

*个数 = 2*i+1, 空格个数= n-1-i

输入 4 输出上面图形

冒泡排序

```

8 5 1 3 7 6      5 1 3 7 6 | 8    0    5
  \              1 3 5 6 | 7 8    1    4
5 8 1 3 7 6      1 3 5 | 6 7 8    2    3
  \              1 3 | 5 6 7 8    3    2
5 1 8 3 7 6      1 | 3 5 6 7 8    4    1
  \
5 1 3 8 7 6
  \
5 1 3 7 8 6
  \
5 1 3 7 6 8
  
```

```

# 冒泡排序
# 8 5 1 3 7 6

nums = [8, 5, 1, 3, 7, 6]
n = len(nums)
for i in range(0, n): # 表示冒泡的次数
    for j in range(0, n-1-i): # 每次冒泡时两两比较次数
        if nums[j] > nums[j+1]:
            nums[j], nums[j+1] = nums[j+1], nums[j]

print(nums)

```

函数

```

def qidao(name, adj):
    print(name + "老师太" + adj + "了")
    print(name + "老师太" + adj + "了, 我的天呢! ")
    print("受不了了, " + name + "老师真" + adj + "! ")
qidao("李", "丑")

```

- 第一行: def 表示当前语句在定义一个函数
- 第一行: qidao 表示函数名

- 第一行: (name, adj) 表示形式参数列表, 简称形参列表, 可以理解成是变量
- 第2-4行: 表示函数的函数体, 也就是函数要执行的代码, 在函数体中可以直接使用形参列表中的变量
- 第5行: 函数调用语句, 调用 qidao 这个函数, 并且在调用时传递两个参数, 两个参数分别为 "李", "丑", ("李", "丑") 可以称为实际参数列表, 简称实参列表
- 函数调用时会完成实参往形参的赋值, 赋值时根据顺序——赋值

```
def get_sum(arr):  
    n = len(arr)  
    s = 0  
    for i in range(0, n):  
        s += arr[i]  
    return s  
  
nums = [2, 3, 1, 5, 7, 8]  
a = get_sum(nums)    # a = 26  
print(a)
```

- return 语句: 结束函数的执行
- return 后面如果接有 数值、变量、或表达式, 会将其值作为返回值返回, 返回值返回给函数调用语句位置, 可以理解成返回值代替函数调用语句出现在其位置