

## # 第三周

两个有序序列的中位数

根据教材3.3.4节的内容，先完成“答题”模块的填空题。然后完成力扣

4. 寻找两个正序数组的中位数

要求用3种方法分别实现，说明思想，粘贴代码、截取力扣的时间效率图到word中。若方法太慢，力扣上运行超时，也做出这样的说明。

方法1:2个数组作为1个数组，排序后输出中位数。

方法2:2个数组有序合并到另1个数组，输出中位数。

方法3:用教材例3.2的方法，寻找中位数输出。

C Kernel in jupyter notebook不需要主方法

## ## 方法1：先对合并后的数组使用冒泡排序，排序后进行折半查找，寻找中位数

In [14]:

```
a = [1, 2, 3, 5, 6, 7, 8, 9, 10]
b = [2, 3, 4, 5, 9, 10, 12]

c = a+b
print("原生数组:")
print(c)

def bubbleSort(array, start):
    if len(array) == start:
        return array
    for i in range(len(array) - start - 1):
        if array[i] > array[i + 1]:
            temp = array[i]
            array[i] = array[i + 1]
            array[i + 1] = temp
    return bubbleSort(array, start + 1)

print("排序后数组:")
print(bubbleSort(c, 0))

print("中位数:")
print(c[int(((len(c) - 1) / 2))])
```

原生数组:

[1, 2, 3, 5, 6, 7, 8, 9, 10, 2, 3, 4, 5, 9, 10, 12]

排序后数组:

[1, 2, 2, 3, 3, 4, 5, 5, 6, 7, 8, 9, 9, 10, 10, 12]

中位数:

5

执行结果:

![imgs/imgs\_20220315214826.png](imgs/imgs\_20220315214826.png)

**## 方法2：利用归并排序，将两个数组有序合并到另一个数组，然后输出中位数**