# 数据流中的中位数

## 题目

[牛客网](https://www.nowcoder.com/practice/9be0172896bd43948f8a32fb954e1be1?tpId=13&tqId=11216&tPage=4&rp=2&ru=%2Fta%2Fcoding-interviews&qru=%2Fta%2Fcoding-interviews%2Fquestion-ranking)

如何得到一个数据流中的中位数？如果从数据流中读出奇数个数值，那么中位数就是所有数值排序之后位于中间的数值。如果从数据流中读出偶数个数值，那么中位数就是所有数值排序之后中间两个数的平均值。我们使用 Insert() 方法读取数据流，使用 GetMedian() 方法获取当前读取数据的中位数。

## 解题思路

1. 同两个堆来表示中位数的左右两部分，左边是大根堆，右边是小根堆
2. 在插入元素时，两边元素个数最多只能相差1，并且要保证左边的元素均小于右边的元素
3. 当插入大堆的元素大于部分小堆元素时，需要将大堆的 top 元素移动到小堆，反之亦然

private PriorityQueue<Integer> maxHeap = new PriorityQueue<>((o1, o2) -> -o1.compareTo(o2));  
  
private PriorityQueue<Integer> minHeap = new PriorityQueue<>();  
  
private int size = 0;  
  
  
public void Insert(Integer num) {  
 if (size % 2 == 0) {  
 maxHeap.add(num);  
  
 if (minHeap.isEmpty() || num > minHeap.peek()) {  
 minHeap.add(maxHeap.poll());  
 }  
  
 } else {  
 minHeap.add(num);  
  
 if (maxHeap.isEmpty() || num < maxHeap.peek()) {  
 maxHeap.add(minHeap.poll());  
 }  
 }  
  
 size++;  
}  
  
public Double GetMedian() {  
 if (maxHeap.isEmpty() && minHeap.isEmpty()) return 0.0;  
 if (maxHeap.isEmpty()) return minHeap.peek() \* 1.0;  
 if (minHeap.isEmpty()) return maxHeap.peek() \* 1.0;  
  
 if (maxHeap.size() == minHeap.size()) {  
 return (maxHeap.peek() + minHeap.peek()) / 2.0;  
 }  
  
 return maxHeap.size() > minHeap.size() ? maxHeap.peek() \* 1.0 : minHeap.peek() \* 1.0;  
}