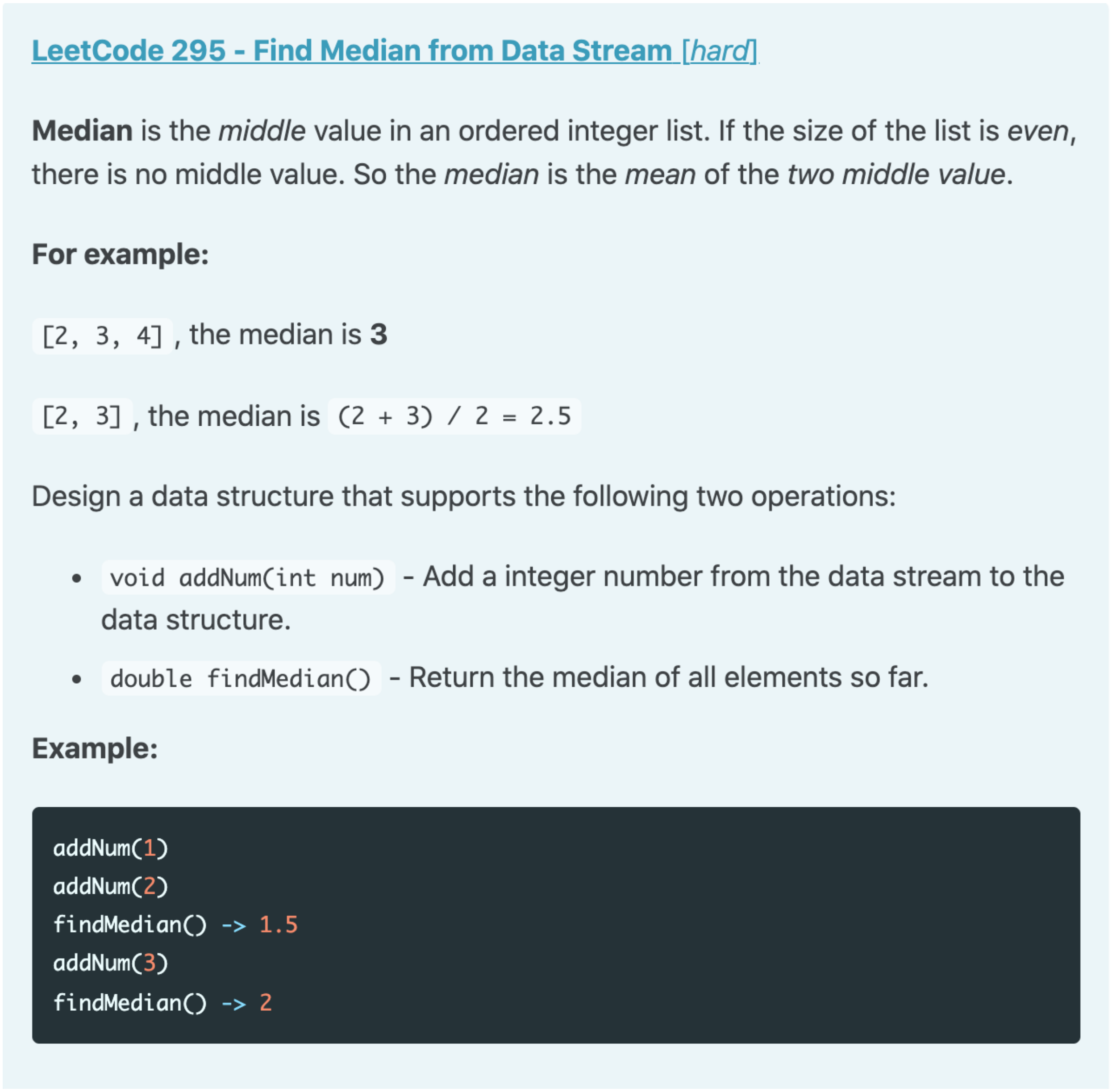
Two Heap (For find median)



LeetCode 295 - Find Median from Data Stream [*hard*]

Median is the *middle* value in an ordered integer list. If the size of the list is *even*, there is no middle value. So the *median* is the *mean* of the *two middle value*.

For example:

`[2, 3, 4]`, the median is **3**

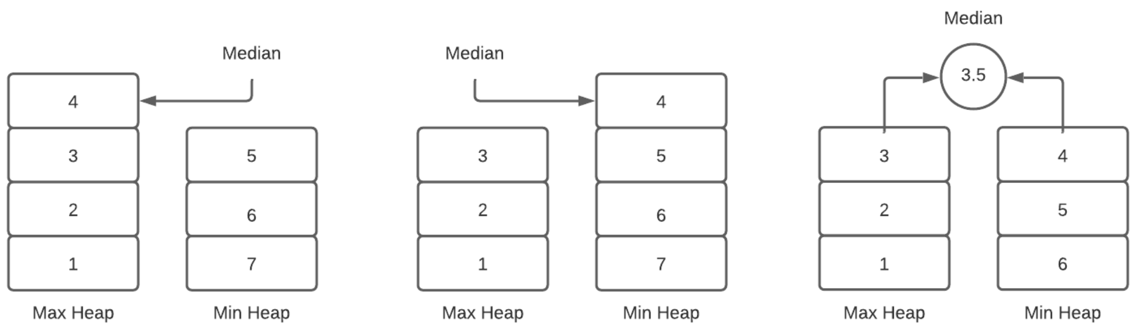
`[2, 3]`, the median is `(2 + 3) / 2 = 2.5`

Design a data structure that supports the following two operations:

- `void addNum(int num)` - Add a integer number from the data stream to the data structure.
- `double findMedian()` - Return the median of all elements so far.

Example:

```
addNum(1)
addNum(2)
findMedian() -> 1.5
addNum(3)
findMedian() -> 2
```


Median
4
3
2
1
5
6
7
Max Heap
Min Heap
Median
3
2
1
4
5
6
7
Max Heap
Min Heap
Median
3.5
3
2
1
4
5
6
Max Heap
Min Heap

```java
public class TwoHeap {

    private Queue<Integer> minHeap, maxHeap;

    TwoHeap() {
        minHeap = new PriorityQueue<>();
        maxHeap = new PriorityQueue<>(Comparator.reverseOrder());
    }

    void add(int num) {
        if (!minHeap.isEmpty() && num < minHeap.peek()) {
            maxHeap.offer(num);
            if (maxHeap.size() > minHeap.size() + 1) {
                minHeap.offer(maxHeap.poll());
            }
        } else {
            minHeap.offer(num);
            if (minHeap.size() > maxHeap.size() + 1) {
                maxHeap.offer(minHeap.poll());
            }
        }
    }

    double getMedian() {
        int median;
        if (minHeap.size() < maxHeap.size()) {
            median = maxHeap.peek();
        } else if (minHeap.size() > maxHeap.size()) {
            median = minHeap.peek();
        } else {
            median = (minHeap.peek() + maxHeap.peek()) / 2;
        }
        return median;
    }

    public static void main(String[] args) {
      TwoHeap heap = new TwoHeap();
        heap.add(2);
        heap.add(3);
        heap.add(4);
        heap.add(5);
        heap.add(6);
    }
}
```

LeetCode 480 - Sliding Window Median [*hard*]

LeetCode 502 - IPO [*hard*]