

703. Kth Largest Element in a Stream



Kth Largest Element in a Stream

```
class KthLargest:

    def __init__(self, k: int, nums: List[int]):
        nums=heapq.nlargest(k,nums)
        heapq.heapify(nums)
        self.heap=nums
        self.k=k

    def add(self, val: int) -> int:
        if len(self.heap) < self.k:
            heapq.heappush(self.heap, val)
        elif val > self.heap[0]:
            heapq.heapreplace(self.heap, val)
        return self.heap[0]

# Your KthLargest object will be instantiated and called as such:
# obj = KthLargest(k, nums)
# param_1 = obj.add(val)
```

1046. Last Stone Weight



Last Stone Weight

```
class Solution:
    def lastStoneWeight(self, stones: List[int]) -> int:
        stones.sort()
        while(len(stones)>1):
            stone1=stones.pop()
            stone2=stones.pop()
            diff=abs(stone1-stone2)
            stones.append(diff)
            stones.sort()
        return stones[0]
```
