TopK问题指的是在一个序列中，求出该序列的第k个最大或最小，或者前k个最大或最小的元素的问题。如果使用排序算法求解，则至少需要O(nlgn)的时间复杂度。但这个问题可以使用一些排序方法的部分关键思想来解决，从而优化时间复杂度。

1. 快排：快排中partition操作将数组序列分割成[l,mid-1]<[mid]<[mid+1,r]的形式，并且可知mid，即在数组中的位置，所以nums[mid]就是数组中的第k大或第k小的数，通过比较mid和k的关系就可以知道真正的第k位数位于哪一个序列中，并采取进一步分割的操作。时间复杂度上来说，第一次分割需要遍历n个元素，而后平均每次需要遍历的元素为n//2, n//4…所以总复杂度为O(n)。
2. 堆排序：堆排序中利用最大最小堆，可以直接存储k个满足题意的元素，如果后续有元素比堆根节点更满足条件，则替换根节点并重新调整堆。每次对堆的操作复杂度为O(lgk)，故总复杂度为O(nlgk)。