

```

def countOperationsToEmptyArray(self, nums: List[int]) -> int:

    # [-15, -17, -19, 5, 6, 7]
    # [-15, -19, -17, 5, 6, 7]
    n = len(nums)
    res = n
    # ob1. when x is not removed, any y>x can not be removed. -> if x
    need to be moved to the end. Then any y > x is required to moved to the end
    before x is finally removed 但是不一定会加上y move to the end operation。譬如
    [-17, -15, -19, 5, 6, 7] -15会随-17移除而移除，真正导致要将大于自己的数都move2end一
    次的是-17》-19这个关系
    # ob2. when x is removed? any z < x is removed then x can be removed
    or say, after the one z' just smaller than x is removed
    # -17 is the number immediately preceding -15.

    A =nums
    val2idx = {val:idx for idx,val in enumerate(A)}

    A.sort()
    for i in range(1, n):
        if val2idx[A[i]] < val2idx[A[i-1]]:
            res += n - i # all y (y>x) and x will be moved to the end
for once

    return res

#https://leetcode.com/problems/make-array-empty/solutions/3466800/java-
c-python-easy-sort-solution/

```