Your task is to compute the total number of comparisons used to sort the given input file by QuickSort. As you know, the number of comparisons depends on which elements are chosen as pivots, so we'll ask you to explore three different pivoting rules.

The input file contains all of the integers between 1 and 10,000 (inclusive, with no repeats) in unsorted order. The integer in the i^{th} row of the file gives you the i^{th} entry of an input array.

QuickSortFirst:

This implementation always use the first element of the given array as the pivot element.

QuickSortLast:

This implementation always use the final element of the given array as the pivot element.

QuickSortMedianOfThree:
This implementation uses the "median-of-three" pivot rule. Consider the first, middle, and final elements of the given array. (If the array has odd length it should be clear what the "middle" element is; for an array with even length $2k$, use the k^{th} element as the "middle" element. So for the array 4 5 6 7, the "middle" element is the second one ---- 5 and not 6!) Identify which of these three elements is the median (i.e., the one whose value is in between the other two), and use this as your pivot.