Exercise 1

Stable partitioning. Implement a stable partitioning algorithm that rearranges the elements of an array so that all elements that are less than or equal to pivot come before all elements that are greater than or equal to pivot.

The relative order of the elements should be preserved. The algorithm should run in $O(n)$ time and use $O(n)$ additional space.

Exercise 2

Dual-pivot quicksort. Implement a version of Yaroslavskiy's dual-pivot quicksort. See lecture (and demo) slides for details.

Compare the performance of your implementation with the standard quicksort algorithm from the lecture.

Exercise 3

Hoare's original quicksort. Implement a version of Hoare's original quicksort algorithm. It's similar to our two-way partitioning algorithm except that the pivot is not swapped into its final position. Instead, the pivot is left in one of the two subarrays, no element is fixed in its final position, and the two subarrays where the pointers cross are sorted recursively.