Solution 1

```
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

// Best: O(n) time | O(1) space
// Average: O(n) time | O(1) space
// Worst: O(n^2) time | O(1) space
function quickselect(array, k) {
  const position = k - 1;
  return quickselectHelper(array, 0, array.length - 1, position);
}

function quickselectHelper(array, startIdx, endIdx, position) {
  while (true) {
    if (startIdx > endIdx) {
      throw new Error('Your algorithm should never arrive here!');
    }
    const pivotIdx = startIdx;
    let leftIdx = startIdx + 1;
    let rightIdx = endIdx;
    while (leftIdx <= rightIdx) {
      if (array[leftIdx] > array[pivotIdx] && array[rightIdx] < array[pivotIdx]) {
        swap(leftIdx, rightIdx, array);
      }
      if (array[leftIdx] <= array[pivotIdx]) {
        leftIdx++;
      }
      if (array[rightIdx] >= array[pivotIdx]) {
        rightIdx--;
      }
    }
    swap(pivotIdx, rightIdx, array);
    if (rightIdx === position) {
      return array[rightIdx];
    } else if (rightIdx < position) {
      startIdx = rightIdx + 1;
    } else {
      endIdx = rightIdx - 1;
    }
  }
}

function swap(i, j, array) {
  const temp = array[j];
  array[j] = array[i];
  array[i] = temp;
}

exports.quickselect = quickselect;

```