**Question 2:** **Application of prefix sum - Quicksort**

In our study of parallel algorithms, we have studied the parallel prefix algorithm. In this algorithm, the input is a vector of n integers, and the output vector is one where each element is the sum of all the preceding elements.

One application of a parallel prefix sum is the quicksort algorithm. Sequentially, the quicksort algorithm is a recursive sort in which a single element is chosen as a pivot, and the rest of elements are marked as being greater than or less than the pivot and adjusted accordingly. The algorithm is then called again on both the greater values and the lesser values.

The parallel version is similar to the sequential version in that it too harnesses the power of divide and conquer. In this method, we pick one value is picked as the pivot and construct a second array of 0 and 1s, where a 1 indicates that the respective value is less than the pivot, and 0 indicates greater than. We also construct a third array where 1 indicates greater than pivot and 0 indicates less than pivot. We then find the prefix sum of these arrays. The resulting prefix sum array will give us the array indices for each value in the original array, with respect to the pivot. Each subarray construction and parallel prefix sum can be executed in a parallel for loop.

With randomly selected pivots, this gives us a runtime of O(lg n).

Blelloch, G.E, “Prefix Sums and Their Applications,” CMU, Pittsburg, PA, pp. 35-57.

<http://www3.cs.stonybrook.edu/~rezaul/Spring-2013/CSE638/CSE638-lectures-8-9.pdf>

Chowdhury, R.A., CSE 628: Advanced Algorithms, Topic: “Parallel Quicksort and Selection”. SUNY Stony Book. 2013.

<https://www.cs.cmu.edu/~guyb/papers/Ble93.pdf>