**Lab 5 Solutions**

**Problem1: Solution**

Diagram, engineering drawing

Description automatically generated

**Problem2: Solution**

Given array A = [ 5, 1, 4, 3, 6, 2, 7, 1, 3 ]

n = 9

Sorted : [ 1, 1, 2, 3, 3, 4, 5, 6, 7 ]

a) Good pivot elements in the above array are 2, 3, 3, 4, 5 ( total = 5 ) because only these pivots satisfy the conditions

number of elements < pivot is less than 3n/4 and

number of elements > pivot is less than 3n/4.

b) Yes. At least half of the elements of array A are good pivots.

Here, 5(2, 3, 3, 4, 5) out of 9(n) elements are pivot elements.

**Problem3: Solution**

For this problem, we can use a binary search algorithm(recursive) because array is already sorted. The base will examine if Array[mid] == mid and if so, returns Array[mid]. And induction case will be if Array[mid] > mid, use binary search in left side otherwise search the right side. If lower > upper(if no such element found), return -1.

This solution solves the problem is O(logn) times.

Now, suppose f(n) = O(logn) and g(n) = o(n).

To show this algorithm runs in o(n),

Lg f/g should be 0.

n-> ∞

Using L’Opital rule, we have,

Letter

Description automatically generated with low confidence

**Problem4: Solution**

For this problem, we can use QuickSelect algorithm having worst case running time of O(n) to select pivot elements on each recursive pivot selection.

This adds O(k) running time whenever section of the area has length k, so has the same cost as the partition step.

Using this algorithm, guarantees that all pivot elements are good pivots, so the recursion tree has height O(log n) and running time is O(nlog n) in the worst case.

**Problem5: Solution**

Text, letter

Description automatically generated