UBİ602 Algorithm Complexity Analysis

**Homework #5**

1. Show (prove) that QuickSort runs in O(n log n) time in average for random arrays.
2. Design a bottom-up non-recursive Merge Sort algorithm where
3. n (input size) is a power of 2
4. n is any positive integer greater than 1

Assume that you have a procedure (or subroutine) that merges to sorted sub segments of a given array. Name of this merge procedure is ***Merge(A,l,m,r)*** that takes an array A with two sorted subsegments A[***l*** … ***m***] and A[***m+1*** *…* ***r***] and returns the array A with a sorted subsegment A[***l*** … ***r***].

1. **a.**For the one-dimensional version of the closest-pair problem, i.e., for the

problem of finding two closest numbers among a given set of *n*real numbers,

design an algorithm that is directly based on the divide-and-conquer

technique and determine its efficiency class.

**b.**Is it a good algorithm for this problem?