[CSN212] Assignment 2
Divide and Conquer
[Maximum Marks 100]

1 Rank in two sorted arrays [10 Marks]

Given two sorted arrays of size m and n. *Design* and *analyze* a divide and conquer algorithm to compute the k^{th} smallest number in the union of two arrays.

2 Smallest Triangle [15 Marks]

Given a set of n points on a 2D plane, *design* and *analyze* an algorithm to find three points making a triangle of the smallest size (perimeter).

3 Maximum Sum Subarray and Maximum Difference [15 Marks]

Consider the two problems described in Section 4.1 CLRS textbook.

1. Given an array A of n positive integers, the difference of the ordered pair (i, j) where $i \leq j$ is $A[j] - A[i]$.
2. Given an array A of n integers, the sum of its sub-array is $A[i, j] = A[i] + A[i + 1] + ... + A[j]$.

The textbook algorithm uses a reduction amongst the problems and gives $O(n \log n)$ solutions for them.

1. *Design* and *analyze* divide and conquer algorithms (without reduction) to solve them in $O(n)$ time.
2. *Design* and *analyze* linear sweep algorithms (without reduction) to solve them in $O(n)$ time.
   Hint: Scan the Array from left to right and maintain the necessary data structures.

4 Extended FFT [20 Marks]

In the FFT algorithm covered in the class, we divide the problems of computing A into two subproblems as of computing A_{odd} and A_{even}. This requires us to use roots of unity to ensure $|X^2| = |X|/2$.

1. Design a divide step to divide each problem into 4 subproblems. What would be the samples used? Compute the complexity of the algorithm.
2. Repeat the above when we divide it into 3 sub-problems.
3. Repeat the above for general case of k sub-problems.

5 kth quantiles, multiary Quicksort [20 Marks]

In Quicksort a *pivot* element partitions an array of numbers to those smaller than the pivot and those larger than the pivot (assume for simplicity that all numbers are unique). Consider a multiary version of Quicksort, where $k-1$ pivots are chosen to partition the numbers to k classes analogously to the standard Quicksort. Show that such multiary partitioning can be produced in $O(n \log k)$ time so that the all classes are of the same size (to within 1).

6 FFT Applications [20 Marks]

1. Recall the pattern matching algorithm from class using FFT. Modify the algorithm to report whether a pattern A and any substring of the string B matches k-cyclically for a given $k \in [0, 25]$, where k-cyclic matching of two strings P and Q over characters $[a, z]$ of length n is defined as:

   For each $i \in [0, k]$, there are at least $\lfloor \frac{n}{k} \rfloor$ matches between characters of P and i^{th} cyclic shift of character in Q along $[a, z]$, eg. third cyclic shift of a and y are d and b respectively.

   Also, analyze the complexity of the algorithm.

2. Recall the pairwise sum problem from the class. Given two arrays of numbers $A[n]$ and $B[n]$ having numbers in the range $[0, m]$. Use FFT once to report the number of elements of A and B that can be used to generate the sum k when a pair is made using one element from A and one element from B.

7 Textbook Problems (Solve don't submit)

Solve the following problems from CLRS Textbook (3rd Edition)
4-5,4-6,9-2,30.1-7,30-3,30-6