CS648 : Randomized Algorithms
CSE, IIT Kanpur

Practice sheet 4

The topics are:

- Min-cut
- Randomized Incremental Construction and Backward Analysis

1. **(Min-cut)**
   Recall the first basic algorithm for min-cut that carries out a sequence of $n-2$ contractions on graph G. We showed that the probability that it outputs a min-cut is at least $\frac{2}{n(n-1}$. If there were k min-cuts in the given graph G, what would be the success probability of outputing a min-cut ?

2. **(Revisiting Convex hull)**
   You have a friend who is not doing CS648 in this semester. But he often goes through the course material. Though he liked many topics in the course, he still does not believe in the magical role of backward analysis. He has seen why forward analysis does not work for the closest pair problem. But he is not able to see why forward analysis approach does not work for bounding the expected time complexity of ith step in the RIC based algorithm for convex hull. Give arguments to convince your friend that forward analysis won't work in this case as well.

3. **(Revisiting closest pair problem)**
   Recall the closest pair problem discussed in the class. While analyzing the randomized incremental algorithm for this problem, we assumed that distance between each pair of points is distinct. Carry out the analysis without this assumption. What bound do you get for the expected running time of the algorithm if you avoid the assumption ?

4. **(Revisiting Randomized Quick Sort)**
   Backward analysis may be used to analyse algorithms that don't have anything to do with randomized incremental construction. You will be surprised to see that it can be used to analyse the expected number of comparisons during randomized quick sort. Here is the sketch:

   Consider the following variant of randomized quick sort. Let S be a set of n numbers. We generate a uniformly random permutation of S. Let e_i be the element appearing at place i in this permutation. This is how we proceed. We use e_1 as the first pivot element and compare it with every other element of S. In this way, we split the set S into 2 subsets - one subset with elements smaller than e_1, and another with elements greater than e_1. In the beginning of the ith step, there will be i subsets formed as a result of the partitions of set S by $i-1$ pivot elements selected from the permutation. During the ith step, we compare e_i with all the elements of the subset containing e_i, and thus split this subset into the two sets accordingly. Finally we stop when we have processed the entire permutation.

   (a) Convince yourself that the above process completely captures the quick sort (at least from the perspective of number of comparisons performed).

   (b) Analyse this process using backward analysis and find the expected number of comparisons during ith step. Use it to calculate the expected number of comparison during the randomized quick sort.

5. **(Smallest Enclosing Circle)**
Recall the problem of Smallest Enclosing Circle: Given a set of n points, if we sample half of them randomly uniformly, let C denote the smallest radius circle enclosing them. The expected number of unsampled points lying outside the C is less than 3. In the class, we gave a sketch of this solution based on backward analysis. Provide the complete details of the solution.