Tutorial: Randomized Algorithms

1. An “approximate median” of an unsorted array A[1..n] is an element of A with rank between n/4 and 3n/4 (inclusive of both). Give a fast Monte Carlo algorithm for finding the approximate median of the array.

2. Suppose that you are given a bolt and a set of n nuts of distinct size, exactly one of which fits the bolt. To find the fitting nut, a simple Las Vegas randomized algorithm is designed as follows:

Repeat until a fitting nut is found

Pick a random nut and try it

If it fits the bolt stop

Else remove the nut from the set of nuts

What is the <u>exact</u> expected number of nuts that will be compared with the bolt before the proper nut is found?

3. Given three $n \times n$ matrices A, B, and C, we need to check if AB = C. Design an $O(n^2)$ time Monte Carlo algorithm for the problem.

4. Suppose that at each step of Karger's min-cut algorithm, instead of choosing a random edge for contraction, two vertices are chosen at random and are merged into a single vertex. This is repeated till only 2 vertices are left. Show that there exists an input graph for which the probability that the modified algorithm finds a min-cut is exponentially small.

5. There are n boxes (numbered 1 to n), with exactly one box containing Rs. 10000. The other boxes are empty. To find the money, each box will have to be opened to see if it contains the money until the money is found. Opening a box counts as one "probe". We want to find the money while minimizing the number of probes performed. The following Las Vegas algorithm is designed for it.

Select x in [0, 1] uniformly at random.

if x = 1

then Probe boxes in order 1, 2,…, n and stop if bill is located

else Probe boxes in order n, …, 1 and stop if bill is located

Show that the expected number of probes before the money is found is exactly (N+1)/2.