1. Gambling with runtime and correctness. (SOLO)

Suppose you are given a randomized algorithm that solves $f : \Sigma^* \rightarrow \Sigma^*$ in expected time $T(n)$ and with $\varepsilon$ probability of error. Show that for any constant $\varepsilon' > 0$, there is a Monte Carlo algorithm computing $f$ with running time $O(T(n))$ and error probability $\varepsilon + \varepsilon'$.

2. Extending randomized min-cut. (SOLO)

(a) Consider a variant of the minimum cut problem where each edge has a cost (positive number), and our goal is to output a cut with minimum total cost (the cost of a cut is the sum of the costs of the cut edges). The usual minimum cut problem corresponds to the case where each edge has equal cost.

Suppose we modify the randomized algorithm seen in class so that in each iteration, the probability of picking an edge to contract is proportional to its weight. (We are not giving a full description here, but we would like you to figure out exactly what this means.) Show that the success probability of this algorithm is at least $1/n^2$. Apply boosting to get an algorithm with error probability at most $1/2^{300}$.

(Hint: Can the analysis done in class be adapted to this setting?)

(b) Using the analysis of the randomized minimum cut algorithm seen in class, show that a graph can have at most $n(n - 1)/2$ distinct minimum cuts.

(c) The minimum $k$-cut problem is a generalization of the minimum cut problem in which we are trying to find a partition $V_1, V_2, \ldots, V_k$ of the vertex set of the input graph such that the number of edges that have its endpoints in different $V_i$’s is minimized. Modify the contraction algorithm seen in class in order to solve this problem. Give a bound on the success probability.

3. Communication upper and lower bounds. (SOLO)

(a) Let $GT : \{0, 1\}^n \times \{0, 1\}^n \rightarrow \{0, 1\}$ be defined as follows: $GT(x, y) = 1$ if $x \geq y$ when $x$ and $y$ are viewed as binary numbers; $GT(x, y) = 0$ otherwise. Show that $D(GT) = \Theta(n)$.

(b) Show that $R^{1/3}(GT) = O(\log^2 n)$.

(Hint: What randomized protocols can you use?)

(c) Let $V$ be a public vertex set. Alice and Bob have sets $E_A, E_B$ of weighted edges. Give a tight $\Theta(\cdot)$ upper and lower bound on computing the MST of $G = (V, E_A \cup E_B)$, assuming the weights have value polynomial in $V$ (i.e. they are $O(\log(|V|))$ bits long).

4. Quicksort analysis. (GROUP)

Complete the analysis of the Quicksort algorithm from the class. In particular, show that given an array of $n$ integers, the expected number of comparisons required by the algorithm is at most $2n \ln n$.

(Hint: The $n$-th harmonic number $H_n = \sum_{k=1}^{n} \frac{1}{k}$ satisfies the inequality $\ln n \leq H_n \leq 1 + \ln n.$)
5. **Back to DFAs. (GROUP)**  
In this problem, you will prove irregularity not by the standard pigeonhole arguments, but rather by communication lower bounds.

(a) Prove that $D(\text{MAJ}) \geq \Omega(\log n)$.

(b) Let $L = \{w \in \{0,1\}^* \mid w \text{ has at least twice as many } 1\text{'s as } 0\text{'s}\}$. Prove that if $L$ is regular then $D(\text{MAJ}) = O(1)$, thus showing via (a) that $L$ is irregular.

6. **Anti-symmetry. (OPEN)**  
Let $F : \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}$ be a communication problem. Consider the following model for computing $F$. Alice sends a message to Bob and then Bob decides the output. The cost is the length of Alice’s message. Let $D^{A \rightarrow B}(F)$ denote the cost of the most efficient deterministic protocol that computes $F$ in this model. Give an example of a function $F$ with $D(F) = O(\log n)$ but $D^{A \rightarrow B}(F) = \Omega(n)$. 