You are allowed to discuss with others but not allowed to use any references except the course notes and the books "Probability and Computing" and "Randomized Algorithms". Please list your collaborators for each question. This will not affect your marks. In any case, you must write your own solutions.

The full mark is 50.

---

1. **Minimum Cuts**

   (10 marks) One advantage of Karger's random contraction algorithm for the minimum cut problem is that it can be used to output all minimum cuts. In this question, we assume Karger's algorithm as a black box, which can be used to output a minimum cut with probability at least $2/n(n-1)$ in time $O(n^2)$, where $n$ is the number of vertices in the input graph. Explain how Karger's algorithm can be used to output all minimum cuts and analyze its running time to output all minimum cuts with success probability at least 0.9999.

2. **Minimum $k$-cut**

   (10 marks) Generalizing on the notion of a cut-set, we define a $k$-way cut-set in an undirected graph as a set of edges whose removal breaks the graph into $k$ or more connected components. Show that the randomized contraction algorithm can be modified to find a minimum $k$-way cut-set in $n^{O(k)}$ time.

3. **Graph Drawing**

   (10 marks) A graph is *planar* if it can be drawn on the plane such that the edges do not intersect with each other (except that they meet at the vertices). It is a well-known result that a simple planar graph with $n$ vertices can have at most $3n-6$ edges. We say a graph $G$ has intersecting number $k$ if $k$ is the maximum number such that any drawing of $G$ on the plane has at least $k$ pairs of edges intersecting. By the above result, a simple bound is that the intersecting number is at least $t$ if the graph has at least $3n-6+t$ edges. Prove that the intersecting number is at least $m^3/(64n^2)$ for any simple graph with at least $m \geq 4n$ edges. (Hint: Do random sampling and apply the simple bound.)

4. **Quicksort**

   (10 marks) The expected runtime of the randomized quicksort algorithm is at most $2n \ln n$ steps where $n$ is the number of elements to be sorted. Prove that the probability that the actual runtime is more than say $100n \ln n$ is at most inverse polynomial in $n$.

5. **Graph Sparsification**

(10 marks) We analyze Karger's uniform sampling algorithm on complete graphs.

- Given a complete graph $G$ with $n$ vertices where every edge is of the same weight, explain that Karger's uniform sampling algorithm will succeed with high probability in giving a $(1 \pm \epsilon)$-cut approximator $H$ of $G$ with $H$ having only $O((n \log n)/\epsilon^2)$ edges.
- Consider a random graph $G_{n,p}$ with $p = c \ln n/n$ for a constant $c$. Prove that if $c < 1$ then, for any constant $\epsilon > 0$ and for $n$ sufficiently large, the graph has isolated vertices with probability at least $1 - \epsilon$. Conclude that Karger's uniform sampling algorithm requires $\Omega(n \ln n)$ edges to form a reasonable sparsifier if the input graph is a complete graph.