École Polytechnique Fédérale de Lausanne
Eisenbrand, Moldenhauer

September 22, 2011

**Randomized Algorithms** (Fall 2011)

# Assignment 1

**Due date**: 10:00am, October 4, 2011 — 15 points

**Problem 1 (Tight bound)** — 2 points

Show that the lower bound of $\frac{2}{n(n-1)}$ on the probability that Karger's algorithm returns a fixed min-cut is tight.

**Problem 2 (Randomized Algorithms, Problem 1.2)** — 4 points

(a) Suppose you are provided with a source of unbiased random bits. Explain how you will use this to generate uniform samples from the set $S = \{0, \dots, n-1\}$. Determine the expected number of random bits required by your sampling algorithm.

(b) What is the worst-case number of random bits required by your sampling algorithm? Consider the case when $n$ is a power of 2, as well as the case when it is not.

(c) Solve (a) and (b) when, instead of unbiased random bits, you are required to use as the source of randomness uniform random samples from the set $\{0, \dots, p-1\}$. Consider the case when $n$ is a power of $p$, as well as the case when it is not.

**Problem 3 (Randomized Algorithms, Problem 1.8)** — 4 points

Consider adapting the min-cut algorithm of the lecture to the problem of finding an $s-t$ min-cut in an undirected graph. In this problem, we are given an undirected graph $G$ together with two distinguised vertices $s$ and $t$. An $s-t$ cut is a set of edges whose removal from $G$ disconnects $s$ from $t$. We seek an $s-t$ cut of minimum cardinality. As the algorithm proceeds, the vertex $s$ may get amalgamated into a new vertex as a result of an edge being contracted. We call this vertex the $s$-vertex (initially the $s$-vertex is $s$ itself). Similarly, we have a $t$-vertex. As we run the contraction algorithm, we ensure that we never contract an edge between the $s$-vertex and the $t$-vertex.

(a) Show that there are graphs in which the probability that this algorithm finds an $s-t$ min-cut is exponentially small.

(b) How large can the number of $s-t$ min-cuts in an instance be?

**Problem 4 (Approximate cuts)** — 5 points

Let $\alpha$ be a positive integer, $c$ the cost function on the edges and OPT be the weight of a minimal cut. A cut $S$ is $\alpha$-approximate if $c(\delta(S)) \leq \alpha \,\mathrm{OPT}$. Show that after $n-2\alpha$ iterations of the random contraction algorithm, any $\alpha$-approximate cut $S$ is still present with probability $\binom{n}{2\alpha}^{-1}$.