CS420/520: Graph Theory with Applications to CS, Winter 2018

Homework 4

Due: Thr, 3/15/18

Homework Policy:

1. Students should work on homework assignments in groups of preferably three people. Each group submits to TEACH one set of typeset solutions, and hands in a printed hard copy in class or slides the hard copy under my door before the midnight of the due day. The hard copy will be graded.
2. The goal of the homework assignments is for you to learn solving algorithmic problems. So, I recommend spending sufficient time thinking about problems individually before discussing them with your friends.
3. You are allowed to discuss the problems with other groups, and you are allowed to use other resources, but you *must* cite them. Also, you must write everything in your own words, copying verbatim is plagiarism.
4. *I don't know policy:* you may write "I don't know" *and nothing else* to answer a question and receive 25 percent of the total points for that problem whereas a completely wrong answer will receive zero.
5. Algorithms should be explained in plain english. Of course, you can use pseudocodes if it helps your explanation, but the grader will not try to understand a complicated pseudocode.
6. **Solutions must be typeset.**

Readings:

(A) Jeff lecture notes on randomized minimum cuts: http://jeffe.cs.illinois.edu/teaching/algorithms/notes/14-mincut.pdf.

(B) Wikipedia on Eulerian paths: https://en.wikipedia.org/wiki/Eulerian_path, and Hamiltonian path: https://en.wikipedia.org/wiki/Hamiltonian_path.

Problem 1. Recall that we show in class that the following GuessMinCut procedure returns a minimum cut with probability $1/\binom{n}{2}$.

```
GuessMinCut(G):
  for i ← n downto 2
      pick a random edge e in G
      G ← G/e
  return the only cut in G
```

Now, suppose you are given a graph G with *weighted* edges, and your goal is to find a cut whose total weight (not just number of edges) is smallest.

(a) Describe an algorithm to select a random edge of G, where the probability of choosing edge e is proportional to the weight of e.

(b) Prove that if you use the algorithm from part (a), instead of choosing edges uniformly at random, the probability that GuessMinCut returns a minimum-weight cut is still $\Omega(1/n^2)$.

(c) What is the running time of your modified GuessMinCut algorithm?

Problem 2. Show that the number of minimum cuts in an unweighted graph is at most $\binom{n}{2}$.

Problem 3.

(a) Let u and v be two non-adjacent vertices of a graph $G = (V, E)$, such that $deg(u) + deg(v) \geq V$. Show that G is Hamiltonian if and only of $G + uv$ is Hamiltonian.

(b) Show that any graph with n vertices and minimum degree $\geq n/2$ is Hamiltonian.