CS:5620 Homework 5, Fall 2016

Due in class on Tue, 11/29

Collaboration: You are welcome to form groups of size 2 and work on your homeworks in groups. Of course, you are not *required* to work in groups. Every group should make one submission and names of both group members should appear on the submission and both students in a group will receive the same score. Other than the TA and the professor, you can only discuss homework problems with your group partner. Collaboration can be positive because talking to someone else about these problems can help to clarify your ideas and you will also (hopefully) get to hear about different ways of thinking about the problem. On the other hand, collaboration can be negative if one member of the group rides on work being done by the other member – please avoid this situation. If your solutions are (even partly) based on material other than what has been posted on the course website, you should clearly acknowledge your outside sources.
Late submissions: No late submissions are permitted. You will receive no points for your submission if your submission is not turned in at the beginning of class on the due date.
Evaluation: Your submissions will be evaluated on correctness *and* clarity. Correctness is of course crucial, but how clearly you communicate your ideas is also quite important.
Who submits what: Undergraduate students are required to solve Problems 1-4, Master's students Problem 2-5, and PhD students Problems 3-6.

1. We have studied three MST algorithms in the CONGEST model: (i) Distributed Kruskal, (ii) Gallagher-Humblet-Spira, and (iii) Garay-Kutten-Peleg.
   (a) Suppose that the input is an n-vertex graph with diameter $O(n^{1/3})$. Which of these three algorithms will be asymptotically the fastest? Explain your answer.
   (b) Suppose that the input is an n-vertex graph whose MST has diameter $O(n^{1/3})$. Which of these three algorithms will be asymptotically the fastest? Explain your answer.
2. Consider the Garay-Kutten-Peleg MST algorithm. Recall that the second phase of this algorithm is a version of the *Distributed Kruskal* (or *Pipelined*) algorithm and this phase runs in $O(\sqrt{n} + \text{diameter}(G))$ rounds on an n-vertex graph G. Also recall that this phase runs in $O(\sqrt{n} + \text{diameter}(G))$ rounds because there are $O(\sqrt{n})$ MST fragments after Phase 1 and we need to gather $O(\sqrt{n})$ additional edges in Phase 2 to complete the MST construction. Describe this version of the Distributed Kruskal algorithm using pseodocode. Use the pseudocode that we wrote for the original version of the Distributed Kruskal algorithm and modify it to the current setting in which we are starting with a bunch of already-computed MST fragments.
3. Exercise 7.5 on *Spanning Tree Verification* from Professor Pandurangan's notes.
4. Exercise 5.6 on Leader Election in a complete graph from Professor Pandurangan's notes.
5. Exercise 5.8 on Leader Election in general graphs from Professor Pandurangan's notes.
6. Exercise 5.11 on estimating network size in general graphs from Professor Pandurangan's notes.
