CS:5620 Homework 4, Fall 2016

Due in class on Thu, 10/27

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.

1. Below I have cut-and-pasted Problem 4 on the midterm. I would like you to solve the problem again, but this time with variables instead of specific numbers. Assume that you are given distributed system with k machines and the input graph G has n nodes and m edges. Finally, assume that the maximum size of a message that can be sent or received by a machine in a round is W words.

   (a) Design an MST algorithm that runs in $O\left(\log k \cdot \frac{n}{W}\right)$ rounds.

   (b) Analyze your algorithm to show that it has the correct round complexity.

   Cut-and-Paste starts.
   You have a distributed system with 10 machines that can directly communicate with each other. Thus the underlying communication network is a size-10 clique (complete graph). You may assume that the machines have distinct IDs: 1 through 10. You want to use this system to compute an MST of an edge-weighted graph G with 100,000 nodes and 10 million edges. The 10 million edges of G are distributed evenly among the 10 machines and so each machine is given one million edges (along with their weights). Besides the fact that the distribution of edges is even, you cannot make any other assumptions about the edge distribution (for example, you cannot assume that the lightest million edges go to machine 1!).

   Assume that the 100,000 nodes of G are uniquely numbered 1 through 100,000. Thus each edge $\{u, v\}$ can be represented by a 3-tuple:

   $$(u\text{'s number}, v\text{'s number}, \text{weight of edge } \{u, v\}).$$

   Assume that edge-weights are not too large and the 3-tuple representing an edge can be stored in a single *word* of memory (e.g., a 64-bit word).

   In each round, a machine can send one message to any one of the other 9 machines. However, this message can be fairly large – of size at most 10,000 words. Similarly, in each round a machine can receive a 10,000-word message from any one of the other 9 machines. In other words, a machine cannot, in one round, receive messages from or send messages to two or more distinct machines. For example, in some round it may be the case that machine 1 sends a message to machine 2, machine 2 sends a message to machine 3, machine 3 sends a message to machine 4, and so on and machine 10 can send a message to machne 1. Note that in this example, every machine sends a message to exactly one machine and every machine receives a message from exactly one machine.

It is required that when the algorithm ends, the MST (i.e., all edges of the MST) are known to machine 1. Your task is to design an algorithm that solves this problem in 40 rounds.
Note: Even though the model of computation in this problem is neither the LOCAL model nor the CONGEST model, we can still use ideas from CONGEST model algorithms to solve this problem. Specifically, ideas used in Distributed Kruskal's algorithm are useful here.
Cut-and-Paste ends.

2. Show how to solve the MST problem on an input graph that is a clique in the CONGEST model in $O(\log n)$ rounds.
(**Hint:** Try to efficiently simulate the GHS algorithm.)

3. This problem is on the *message complexity* of the GHS algorithm. Recall that the GHS algorithm has $O(\log n)$ phases. Further recall that in each phase the following two tasks are performed: (i) each component finds an MWOE (in parallel) and (ii) components are merged using the MWOEs (in parallel). My claim is that $O(n)$ messages suffice for all components to find MWOEs and $O(n)$ messages suffice to perform component merging and thus the message complexity of the algorithm is $O(n \log n)$. There is something wrong with this claim. Identify what is wrong and explain your answer. Restate these claims correctly and write down the message complexity of the GHS algorithm.
(**Hint:** The solution to this problem is in Section 7.1 of Professors Pandurangan's notes. You just have to extract the answer and write it in your own words.)

4. Describe a randomized algorithm that computes a *maximal matching* in a graph in expected $O(\log n)$ rounds.

5. Prove the following facts about computing maximal matching in trees: (a) If the given tree is oriented, then there is a deterministic $O(\log^* n)$-round algorithm for this problems and (b) if the tree is not oriented then there is a deterministic $O(\log n)$-round algorithm for the problem.
