Homework Exercise #3
Due: October 20, 2009

1. In class, we discussed a synchronous distributed algorithm for constructing the minimum spanning tree of a graph. Assume that messages are short: each message can contain $O(1)$ process ids and edge weights. The implementation discussed in class uses $O(m + n \log n)$ messages, where n is the number of nodes and m is the number of edges in the network.

Consider an alternative implementation, where, at the end of each phase, the leader of each component gets complete knowledge of that component: the leader knows all the ids of nodes in its component, and the weights and endpoints of all edges incident with nodes in its component.

This knowledge allows the leader to determine the minimum weight edge that has exactly one endpoint in the leader's component, so each leader can determine which edge it wants to add to the spanning tree without doing any communication.

If several components are joined into one during a phase, the leaders of all the old components send all of their information about their old components to the new leader of the new, larger component. (This information is sent via the part of the spanning tree that has been built so far.)

Give (for all n), a weighted graph G_n with n nodes where the number of messages required to implement this approach is $\Omega(n^3)$ (regardless of how the leaders of the new components are chosen).