

## Homework 1

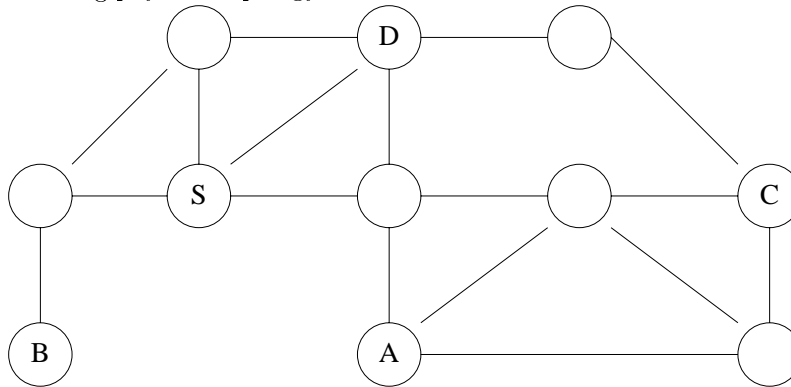
Assigned: Thursday, January 24

Due: Tuesday, February 12

1. Give an example to show that the vectoring heuristic used in the “Finding friends” paper to select the nearest node can fail (i.e., select a node that is not nearest). Use 2-d Euclidean space, where the distance between two nodes is the Euclidean distance (hence the triangle inequality does hold). Construct your example so that there are only two hosts in the candidate set  $S$  and as few beacons as possible.

Give the locations of the beacons, the hosts and the new node  $n$ . Show the heuristic calculation and illustrate how the wrong host is chosen.

2. Consider the following physical topology:



Assume the underlying unicast routing algorithm uses shortest paths and that all links have weight 1. Suppose the mesh construction algorithm has included the following *mesh* edges: (S,C), (A,B), (A,C), (C,D), (B,D). Assume the multicast group has just one source, S.

- Compute the worst case stress, the relative delay penalty (as compared to a true network-layer shortest path tree), and the network resource usage, for the routing tree constructed by Narada with source S.
  - Show the calculations (at S) to consider adding (one at a time) each of the following links: (S,D), (S,B), (S,A). In each case, begin with the original mesh shown above. State clearly whether each link is added or not.
  - Show the calculations to consider deleting (one at a time) each of the following links: (A,B), (S,C). Again, begin with the original mesh shown above. State clearly whether each link is deleted or not.
3. The Chord system described in the paper places each key at exactly one node and attempts to balance the number of keys per node. If some keys are much more popular than others, the nodes containing those keys may become “hot spots”, subject to far more request load than average nodes. Design a modification or addition to the basic Chord system to account for hot spots.

Your answer should include the following. You should describe your proposed design clearly and concisely, explaining the main idea as well as providing some detail about the modifications that would be necessary. You should argue that your design will address the hot spot problem. You should list both the strengths and weaknesses of your design. I expect that most answers to this part of the assignment will be 1-2 pages long.

4. In this part of the assignment you will write the code for a limited-functionality Chord node. We will provide the executable for a network of Chord nodes, and your node will need to interoperate with the provided network.

Once you start up the executable, you will see a message that indicates the network is ready. After that, your node should do the following:

- Send a join message on the machine where you are running the Chord network.
- Receive a message containing the finger table entries for your new node.
- Send messages as necessary to update the finger tables at the existing nodes to reflect the addition of your node.
- Send a message to obtain a set of keys that are the responsibility of your node.
- Store the new keys.
- Issue fixed queries to test the success of your node join. We will provide the queries and the expected answers.
- Issue additional queries to demonstrate the correct operation of your node. We will provide the queries (but not the answers).
- Wait for additional queries initiated by the rest of the system. Log your node's involvement in these queries (who sent the query, what was the key involved, what was the response).

You may code your node in either C or Java, though we will be able to provide more assistance for C implementations. By early next week we will provide formats for the messages and more detail about the protocol between your node and the existing nodes.

**NOTE:** There is an implementation of Chord available on the Chord website. You are not to look at or use this implementation in any way. To do so will be a violation of the Georgia Tech honor code.

More detail on the turn-in procedure for the code will be provided later.