1. Consider the following endorelation on the set $D = \{ \text{Anna, Bates, Carson, Daisy, Edith, Mrs. Hughes, Ivy, Jimmy, William, Alfred} \}$:

(a) Express this endorelation as a graph.
(b) Is it a connected graph?
(c) What’s the longest path in the graph?
(d) Are there any cycles? (If so, name one.)
(e) Which vertex has the largest in-degree?
(f) Which vertex has the largest out-degree?
(g) Which vertex has the smallest in-degree?
(h) Which vertex has the smallest out-degree?

2. Suppose you had a graph of 7 nodes.

(a) What’s the largest number of edges it could have? (Justify your answer.)
(b) How many edges will be in its minimum spanning tree? (Justify your answer.)
(c) What’s the ratio between these two quantities?
(d) If you made this graph bigger by giving it 8, 9, 10, 11, . . . nodes, does the ratio get larger or smaller?

3. Phyllis is a classic rock fan, and is creating a movie database program to keep her albums and songs organized. One of the things she wants to be able to do is print out a list of all the albums she owns by a particular band, in addition to other meta information about them (the year they formed, how many members, what record studios, etc.) For this, she creates a binary search tree of nodes, each of which represents a band. The nodes, once accessed, will contain everything about the Band that Phyllis needs – the trick is making her system work efficiently so that a Band can be quickly pulled up by name.
Suppose that Phyllis bootstrapped up her binary search tree by adding the following band nodes, in this order: *Led Zeppelin, The Beatles, Crosby Stills and Nash, Jimi Hendrix, Eagles, Boston, Rush, The Who, Bob Dylan, Pink Floyd, Queen.* (Do this literally, so that *Eagles* is literally “Eagles” whereas *The Beatles* is literally “The Beatles.”) Note that the binary search property here is in reference to band names: the nodes in the left subtree of a node are “less than” that node in the sense that all of them come before the node in alphabetical order.

a. Draw what the search tree would look like at this stage.

b. If we were looking for a band in this binary search tree (starting from the root and being smart about traversing the tree in the optimal way) what’s the greatest number of nodes we would have to visit before we either found what we’re looking for or knew for a fact it wasn’t in the tree? What’s the least number of nodes we might have to visit? What’s the average number we would have to visit? (Think carefully about this last one. Hint: it isn’t simply the average of the least and the greatest. Think of it this way: picking a band at random, what’s the average number of nodes required to reach it?)
c. Suppose the initial bands were inserted in a different order. Start over with an empty tree, and insert them in this order: The Beatles, The Who, Rush, Queen, Pink Floyd, Led Zeppelin, Jimi Hendrix, Eagles, Crosby Stills and Nash, Boston, Bob Dylan. Now what does the tree look like?

d. Same questions as above: what’s the greatest number of nodes we’d have to visit, what’s the least number, and what’s the average number?

e. What does all this tell you about binary search tree performance? What is it sensitive to?
4. During a friendly game of Mario Party, Daisy\(^*\) finds herself cast into a Tangled Web minigame. She has to spend coins to connect all the spaces in the Web together with rope in 30 seconds! Every inch of rope that she uses costs her coins (the amount for each connection is shown on the diagram) so she wants to be as efficient as possible with her connections. What’s the \textbf{minimum} number of coins she could spend to ensure that every space is directly or indirectly connected to every other space through a path of rope? (Show the configuration of how she should lay out her rope, darkening the edges she chooses).

\*Not the same Daisy as in problem 1.