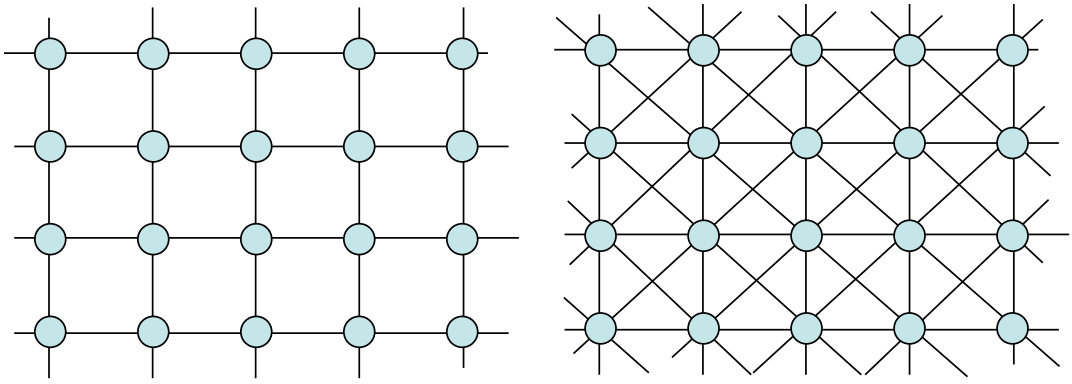
Midterm solution (do not distribute!)

SI 508

NAME _____

This exam is open book, open cTools, open web. But it is not open-human. You are to work on this entirely on your own without the help of others. The exam is worth 25% of your final grade. Good luck!

1. (10pts) **Compute the Watts-Strogatz clustering coefficient for each of the following two infinite lattices. In the first, each node is connected to its 4 closest neighbors, and in the second, to its 8 closest neighbors.**

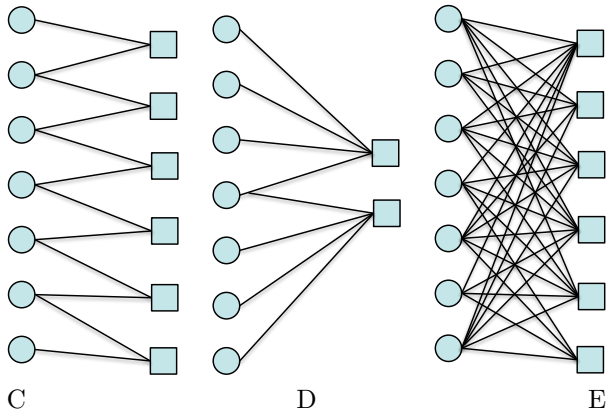


A B

lattice A: The clustering coefficient is 0. None of the nodes' neighbors are directly linked.

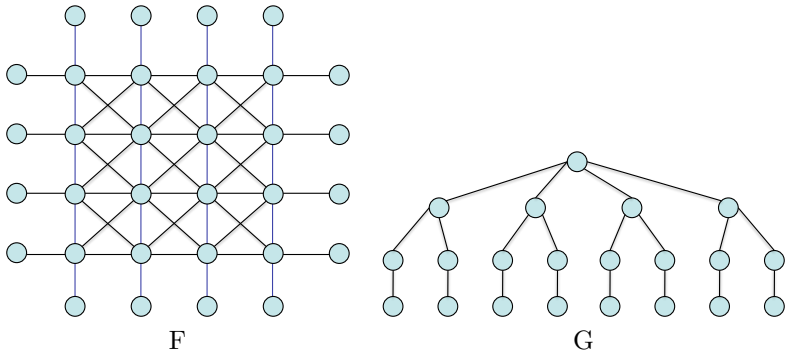
lattice B: The clustering coefficient is 3/7. Each node has 8 neighbors. This means that there are $\binom{8}{2=\frac{8*7}{2}=28}$ possible connections between them. There are 12 connections present, therefore the clustering coefficient is $\frac{12}{28} = \frac{3}{7}$.

(5pts) Consider the following 3 bipartite networks. Consider their 1 mode projections into networks of circles. Which of the following will have the strongest community structure using the modularity criterion? Just pick one, no need to explain _____.



Only network D would yield meaningful community structure once projected to a network of circles. Network C forms a line, network E is a complete clique. Network D consists of two 4-cliques that are joined on one node.

2. **Consider the networks F and G. Answer the following, and explain your answer.**



For this question, the correlations actually end up being quite close. I was looking for good rationalization of your answer, rather than necessarily the correct answer.

(5pt) In which of the networks would you expect there to be the *greater* (positive) correspondence between closeness and betweenness?

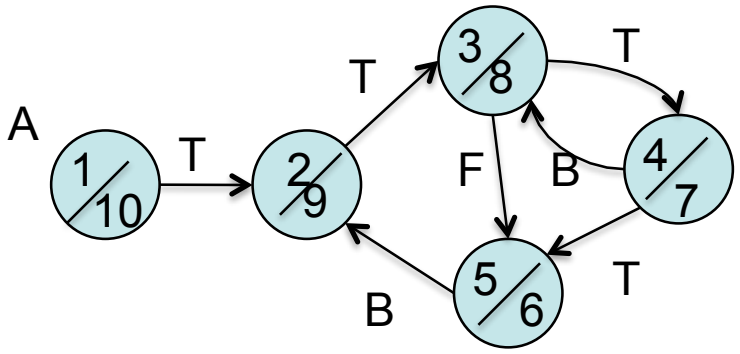
In network F, some nodes that are peripheral to the network act as gatekeepers for the leaf nodes attached to them. They have relatively low closeness but also relatively high betweenness. In G, the closer a node is to the root of the tree, the greater both its closeness and betweenness.

(5pt) In which of the networks would you expect there to be the *lesser* (positive) correspondence between degree and betweenness?

Strictly speaking (if you were to do the calculations), these two networks are rather close in this respect. The intuition for choosing F would be that the nodes in the middle of network F have higher degree, but possibly lower betweenness because other nodes can go around them.

The reason which degree and betweenness are not perfectly aligned (using a Pearson correlation coefficient, as you did in the physician's adoption study) is that the variation in degree (from 1 to 2 to 3 to 4 edges) is not as dramatic as the variation in betweenness (the top level node mediates many more paths than any of the nodes below it, etc.).

3. **(10pts) Do a depth-first traversal of the following graph starting with vertex A. Write the start and finish times of each vertex and label the edges as tree (T), back (B), forward (F), or cross (C).**



4. **(20pts) Consider the following 3 graph traversal methods:**

   - **(i) BFS (breadth first search)**
   - **(ii) DFS (depth first search)**
   - **(iii) random walk**

   Answer how you would do the following tasks, taking into account that you may need additional actions before (e.g. making the graph undirected), during or after the traversal. No need to explain your choice.

   (5pts) To find the weakly connected components, you could use __D__.

   (A) only (i) (B) only (ii) (C) both (i) and (ii) would work (D) all three would work

   Either BFS or DFS would explore the connected components most efficiently, however, even a random walk would be contained within a connected component, and so either selecting (C) (both DFS and BFS) or (D) all three, was an acceptable answer.

   (5pts) To find the strongly connected components, you could use __B__.

   (A) only (i) (B) only (ii) (C) both (i) and (ii) would work (D) all three would work

   Only DFS is used in the algorithm we covered in class. Remember how DFS finishing times were crucial in determining where to start the traversals on the transpose graph. Although you could in principle use all 3, starting from every single vertex, and in the case of the random walk, repeating it many, many times, the DFS method is the only efficient one in this case.

   (5 pts) To find the betweenness of each vertex, you could use __A__.

   (A) only (i) (B) only (ii) (C) both (i) and (ii) would work (D) all three would work

   Recall that betweenness is defined in terms of the number of *shortest* paths between pairs of nodes going through a vertex. Therefore only a method for calculating the shortest paths (i.e. BFS) would yield the betweenness of the different vertices.

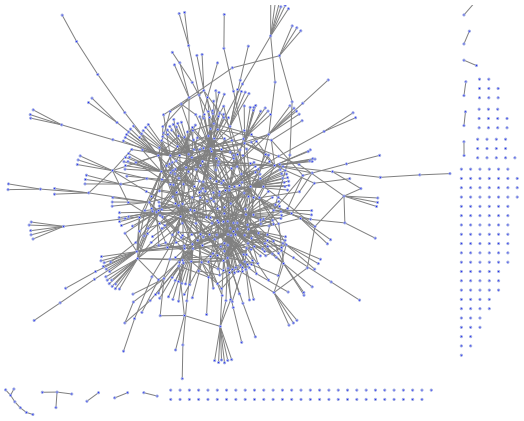
   For the Googling exam takers among you, there is such a thing as 'random walk betweenness' but it is a different, and relatively recently invented animal. When one says 'betweenness', one is referring to the standard shortest path definition.

   (5 pts) To find the average shortest path, you could use __A__.

   (A) only (i) (B) only (ii) (C) both (i) and (ii) would work (D) all three would work

   For the same reason as above, only BFS will yield shortest paths (consider for example the DFS traversal in the previous problem - the node visited at time 5 is distance 4 away from the starting point).

5. **(30pts) Consider the gnutella network you've become familiar with over the course of the semester. For each of the following models in a few words express what features of the real world gnutella network are captured by the model and in which respects the model falls short.**



random graph
features captured:

- average degree
- short path lengths (though Gnutella's ASP is even shorter)
- presence of giant component & smaller components

features not captured:

- degree distribution
- clustering (although Gnutella has very little clustering, it is still greater than that of a random graph)

Watts-Strogatz
features captured:

- short path lengths
- clustering (although Gnutella has very little clustering, it is still greater than that of a random graph)

features not captured:

- degree distribution
- presence of giant component & smaller components

Barabasi-Albert
features captured:

- degree distribution
- ultra small shortest paths

features not captured:

- presence of giant component & smaller components
- clustering (although Gnutella has very little clustering, it is still greater than that of a random graph)

6. **(10pts) Imagine you're working in an organization where being the source of gossip is rewarded through different perks: people will like you, do favors for you, etc., all in exchange for gossip. You want to achieve three things. First, you'd like to hear as much gossip as possible, so that you can pass it on. Second you want to be the person spreading the gossip (if one of your contacts has already heard the gossip, you get no credit for spreading it). Third, you want to be close to the sources of gossip because the gossip gets less accurate or stops altogether with each additional step it has to make. Explain which *two* of the following centrality measures you would try to maximize in positioning yourself in the network by forming contacts (assume contacts are bidirectional, that is if you share gossip with a person, they reciprocate by sharing gossip with you). Your options are: clustering coefficient, degree, closeness, betweenness, proximity prestige, constraint. Explain your rationale for choosing the measures you did and not the others.**

   In order to satisfy both hearing as much gossip as possible (1), and being close to its sources(3), I will want to optimize my *proximity prestige*. Proximity prestige (which can be applied to both directed and undirected graphs) means that there are many people connected either directly or indirectly to me, and that my distance to them is short. Proximity prestige is also nice because you can limit the depth of the input domain you consider. So maybe I only care about people who are within 3 hops of me, because any gossip that starts from further away is going to be too distorted or get lost. On the other hand, if you are optimizing for closeness, it may be that your average distance to anyone else in the network is 4, but that there are not too many individuals closer than that. Nevertheless, closeness was also a good choice for making sure that (3) is satisfied.

   Betwenness is an essential choice because it means that my contacts are not sharing gossip among themselves, rather they need to go through me, which means that I get credit for spreading the gossip (2).

   Degree may be a good way of attaining either closeness or betweenness, but it isn't really an end in and of itself.

   Both high clustering and constraint would give me the opposite of high betweenness - my contacts would be sharing gossip amongst themselves, and I would play a less significant role.