Kashif Ahmad
kac160230
Lab2

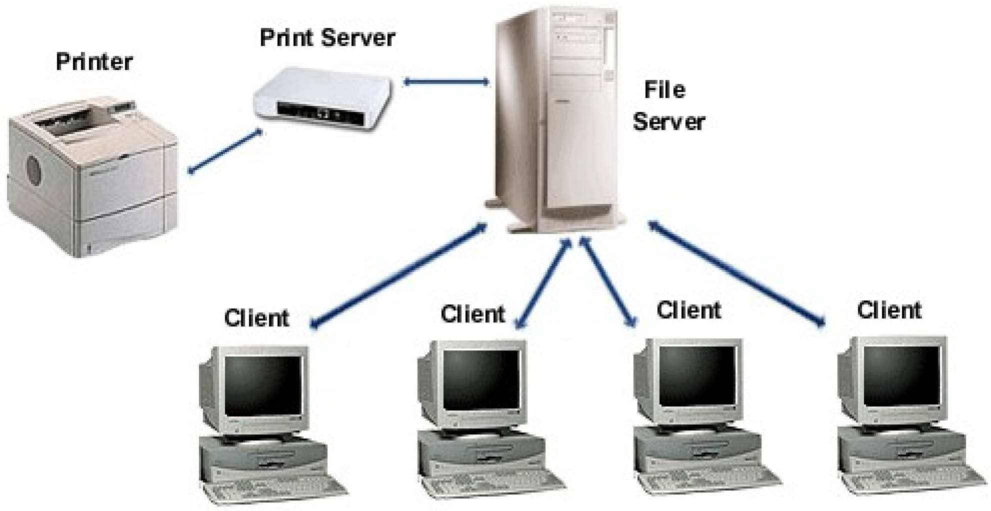
Q) Identify an example type of network of when it is a good thing to be a high degree node and an example of when it is a bad thing to be a high degree node in a network.

Ans) A main airport is a high degree node and it is a bad thing to arrive on that airport or that terminal because it will be busy and it will have a lot of wait time for baggage clearance and check outs. A printer in a server can also be an example. The queues may be long and hence wait times for printing

It is also a good thing to be a high degree node for example a recruiter is connected with a lot of people and he may point you to the right job or the job that suits a person's skill set.

Q) Describe or draw an example of a network in which a particular node has very few connections, but it could be argued that it is a very important node. Justify your reasoning.

Ans) A printer connected to a server, only has a connection to the server but it is an important node since all the network sends data to it.



Q) Explain what equation (6.25) in Newman actually means - i.e., what is a simpler way to explain what the two summations mean?

Ans) The two summations are in going edges and outgoing edges. The point is that every ingoing edge is also an outgoing edge too, so the ingoing edges is equal to out going edges. In short they tell the total number of edges in a network.
Ingoing edges = outgoing edges = total edges

Q) write down the adjacency matrix

A
|0 1 1 0 0|
|0 0 1 1 0|
|0 0 0 1 1|
|0 0 0 0 1|
|0 0 0 0 0|
A^2
|0 0 1 2 1|
|0 0 0 1 2|
|0 0 0 0 1|
|0 0 0 0 0|
|0 0 0 0 0|

each non zero element of A^2 tells us the number of path that has length 2 that correspond to those nodes

0-2 1 **0->1->2**
0-3 2 **0->1->3   0->2->3**
0-4 1 **0->2->4**
1-3 1 **1->2->3**
1-4 2 **1->3->4 1->2->4**
2-4 1 **2->3->4**

Q) Multiply Ax A^2 x=A(Ax)and A^3x=A(A(Ax))

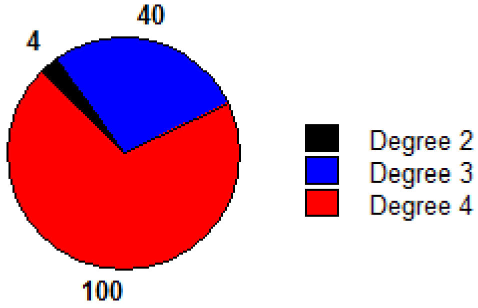
```
Ans) Ax,(A^2)x,(A^3)x =
          |0|
          |0|
          |0|
          |0|
          |0|
```

```
Ans) Ax,(A^2)x,(A^3)x =
            |1|    |1|    |0|
            |1|    |0|    |0|
            |0|    |0|    |0|
            |0|    |0|    |0|
            |0|    |0|    |0|
```

it tells us about the node 2. Ax tells us the number of 1 edged paths the node 2 has incoming**(which is 2 in this case 0->2, 1->2)** has A^2 x tells us the number of 2 length path it has incoming**(1 in this case 0->1->2)** and A^3 x the number of 3 length paths, which is **zero** in this case.

Q)looking at the network, write down the degrees of the nodes (you can aggregate your answer, so write down how many nodes have degree k). Pick a way to visualize this aggregated degree data (a graph of some sort) -just draw something by hand.

Ans) 4 of them have degree 2
     40 of them have degree 3
     100 of them have degree 4



Q) Briefly describe why you see the pattern that you do and indicate why the process stops where it does. How could you make the propagation go further?

Ans) This shows how the information is propagated in a network and how it reaches the last node.It Stops when the 10 steps are finished, you can make it go further by inputting 12 or a greater number in the propagation function

Q) Now add code to propagate this network starting at node 0 for 10 steps. What is the end result? What is A_{10}(Ais the adjacency matrix)? Describe why this is the case from the point of view of network paths and also from linear algebra (i.e., what special name do we give such matrices?).

Ans) The end result is that the propagation reaches a leaf, which doesnt have any outgoing edges and hence terminates there. A is the NULL matrix. Eventually there is no path that can be traced with an edge length of 10. I.e there is no path or route from one node to the other of length 10(10 edges). From the point of linear algebra it also makes sense because the priciple diagonal and half the matrix was zero to begin with. If you keep on multiplying its going to shift up and eventually zeros will be filled in.

0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0
0 0 0 0 0

Q) Run this code -is the out- component what you predicted?
Ans) Yes it was the component I predicted, it ends in node 4, because no outgoing from node 4

Q) Without changing the `propagate` function, how could we use it to find in-components instead of out-components?
Ans) You can input a transpose of the graph while loading the edgelist or an inverse, not sure really.

Q)What is the size of the largest strongly connected component?
Ans) node 1 is the most strongly connected and it is connected to every component so its size is 3

Q)Start by finding the out-components of nodes with index 2 and 16. Have your program print out the size of the component.

Ans) I used the instructions as provided however the size is coming out to be zero

Q)What is the minimum value of `steps` that guarantees that you will find the entire out component? What network has the structure such that you need to use this many steps in order to find the entire out-component, since most graphs will usually require a number smaller than this?
Ans) The steps can be the square of the number of nodes. A network that has to be traversed with each node

Q) First, reason why this graph of activities should be an acyclic directed graph. Explain why cycles in a project activity network would be a bad way to organize it.

Ans) If there are cycles the project can continue for years and not have a completion time. The graph should be acyclic because otherwise the activities would never end and would never be completed.

Q)Can you think of a potential use case for analyzing the minimum cut of an activity network?

Ans) For calculating the bottleneck or the activity which needs to be completed in time otherwise the project will be delayed