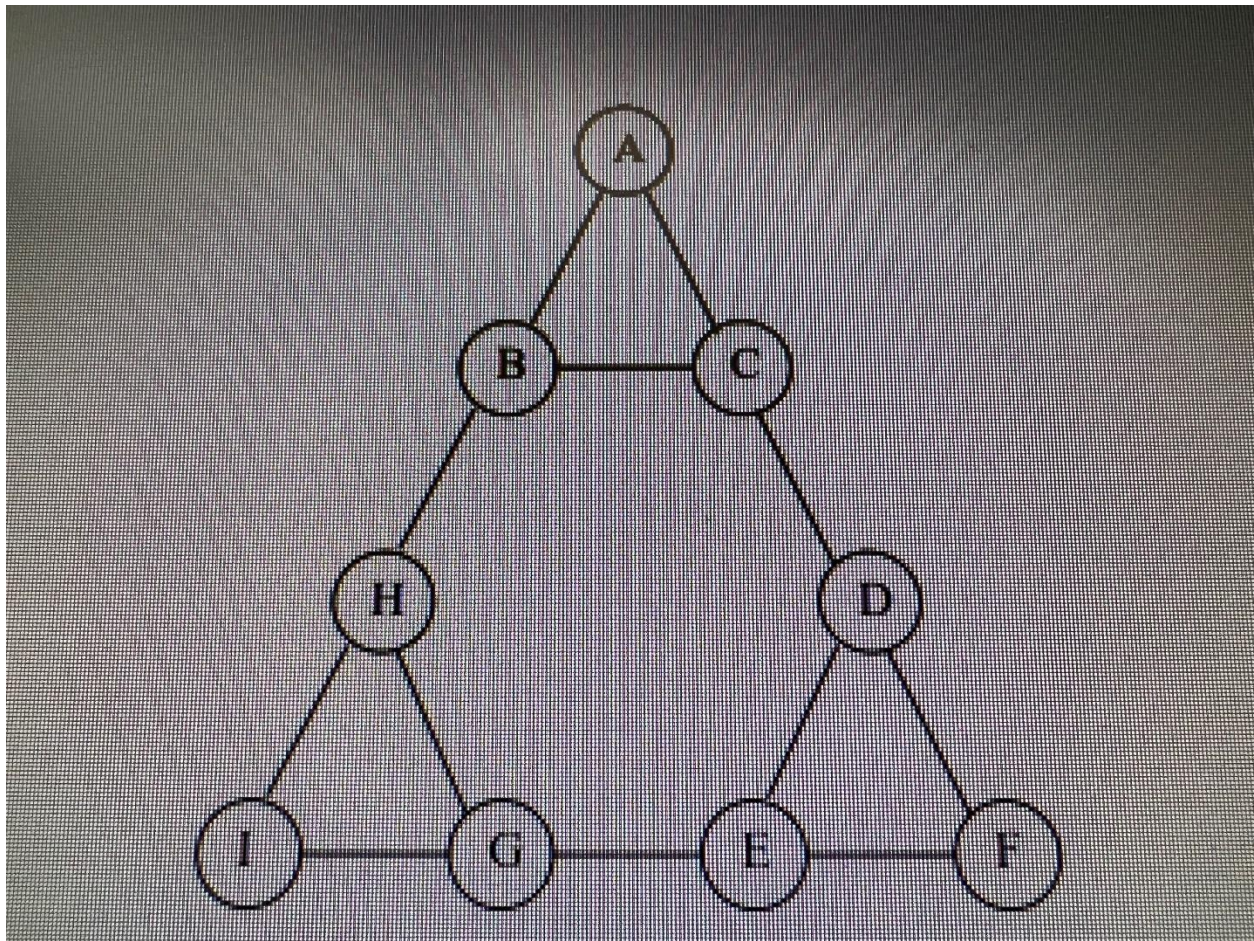


Lab-11 Analysing Social Networks: Community detection

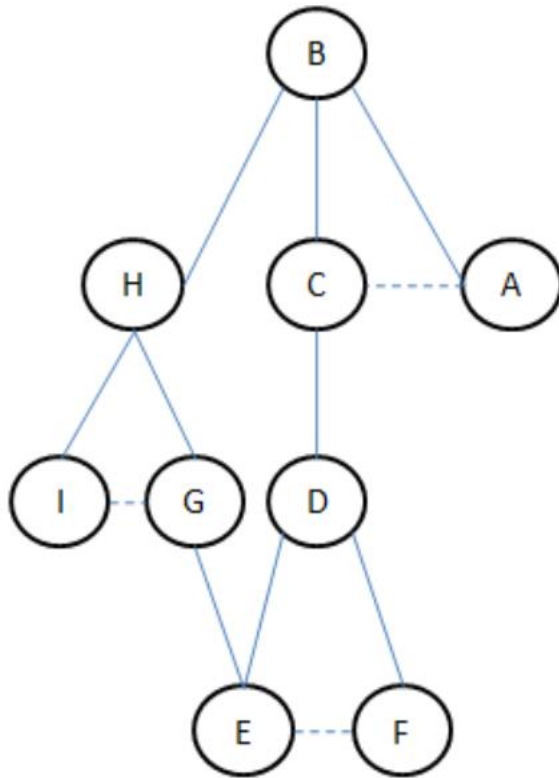
Total Marks: 19 Marks + 2 Marks (individual assessment) = 21 Marks

- i. Using Manual calculation and Python code to find the Communities by **Girvan-Newman** Algorithm for the attached graph shown below (7 marks manual calculation + 4 marks Python implementation = **11 marks**)



The Girvan-Newman algorithm calculates the number of shortest path going through edges. By calculating the shortest path we can then find the communities of a social network graph. This algorithm involves three steps: A brief description of each step is provided on the above graph :

Step I: The algorithm begins by performing a breadth-first search (BFS) of the graph, starting at the node X. Edges between different levels are called DAG edges (“DAG” stands for directed acyclic graph). As the hint, the following graph is shown which is when considering node B as root. Solid edges are DAG edges and dashed edges connect nodes at the same level:



Step II: The second step of the GN algorithm is to label each node by the number of shortest paths that reach it from the root. Start by labelling the root by 1. Then, from the top down, label each node by the sum of the labels of its parents. Show the resulted graph in each step.

Step III: The third and final step is to calculate the credit, for each node and for edges between the nodes. Show the resulted graph in each step.

Now complete the betweenness calculation, and repeat this calculation for every node as the root and sum the contributions. Note that you must divide by 2 to get the true betweenness. You need to calculate the betweenness for every node as the root. And then sum the contributions of all the edges to find the true betweenness for every edge. Finally, remove edges with the highest betweenness, until the graph has broken into a suitable number of connected components. For answering this question show the resulted communities as a set and graph, by removing edges with betweenness greater than or equal to a proper number.

For more details, of GN algorithm you can refer to Section 10-2 of book “Mining of massive datasets” by J. Leskovec, A. Rajaraman.

- ii. Using Python, find the Communities in the graph, as seen in (i) by **Louvain** Algorithm (4 marks)
- iii. Using Python, find the Communities in the graph, as seen in (i) by **greedy_modularity_communities** Algorithm (4 marks)

Tips:

- Use Python library `networkx.algorithms.community centrality` for `girvan_newman`, `community.community_louvain` for Louvain and `networkx.algorithms.community` for `greedy_modularity_communities`