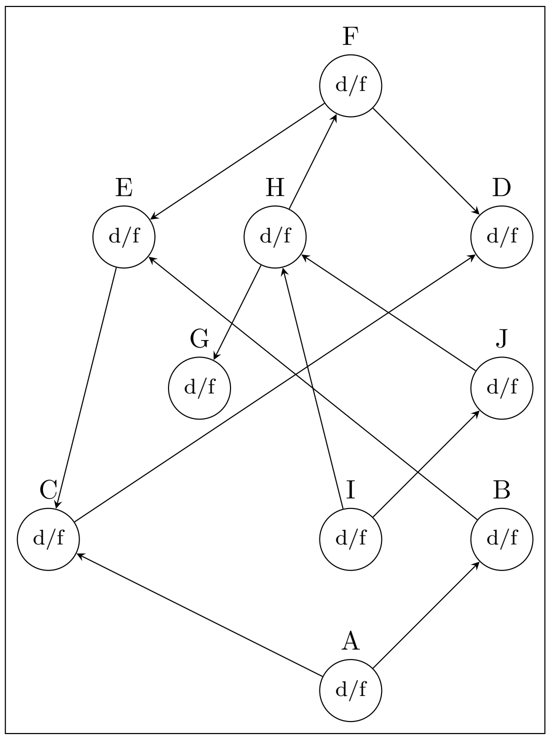
Algorithms: Design and Analysis - CS 412

Weekly Challenge 06

Ali Muhammad Asad - aa07190

Consider the graph, $\mathcal{G}$, below with 10 nodes and 13 edges.



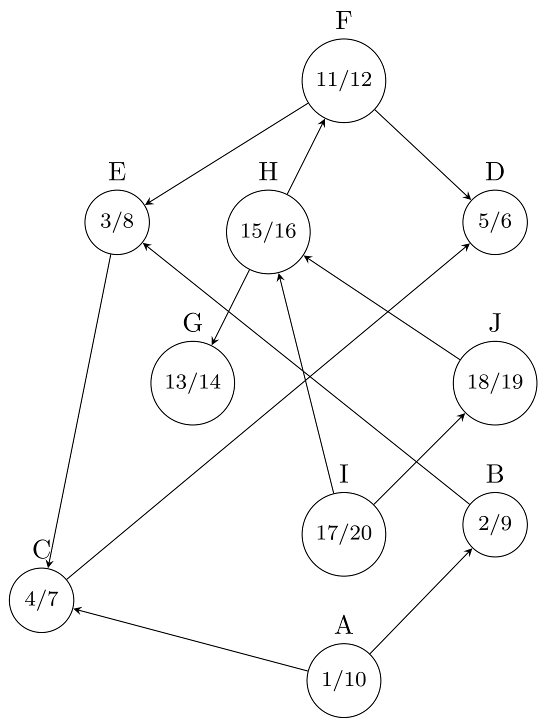
The procedure, DFS($\mathcal{G}$), is executed on the graph such that ties are resolved in alphabetical order.

(a) Redraw the graph below such that each node, n, contains $n.d/n.f$, where $n.d$ and $n.f$ are the node's discovery and finalization times respectively. Mention your starting nodes/nodes under the graph.

(b) Draw below the corresponding DFS-forest.

Solution:

(a) The graph with discovery and finalization times is shown below.



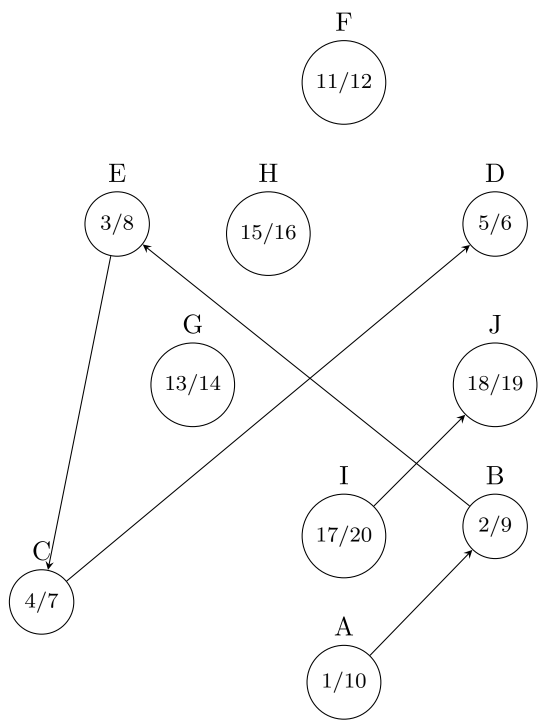
We are performing a DFS on $\mathcal{G}$, hence, we go as deep as we can before backtracking and moving onto any other node.

Since the ties are resolved in alphabetical order, we start with the node A, since lexographically, it is the smallest node. From A, we move to B, then to E, then to C, then to D, marking the times as 1, 2, 3, 4, and 5 respectively. After D, we can't go any deeper, hence we mark its end time 6, and backtrack on our end times in the similar manner, marking times of 7, 8, 9, and 10 on nodes C, E, B, and A respectively.

Now there are no more nodes to go into, hence, we move onto the lexographically smallest node left from our set of nodes that the graph $\mathcal{G}$ is comprised of; F. We discover F, and mark its start time as 11, but we can't go deeper hence its end time becomes 12.

We follow the same pattern again, with new starting nodes as G, H, and I. We can move from I to J, hence we don't include J in the starting nodes. The set of starting nodes becomes $\{A, F, G, H, I\}$.

(b) The DFS-forest is shown below.



Our paths were $\{A \to B \to E \to C \to D\}$, $\{F\}$, $\{G\}$, $\{H\}$, and $\{I \to J\}$, hence the above becomes are forest.