D a) Used DFS Traversal. Whenever for something I got to do a dfs. I saved it in a list (which would like a priority queue) which saves the top sort order.
Then I checked if the derived top sort order was valid by checking if a nodes connection is behind it or if a node is there twice.

b) Used BFS. and ~~deleted~~ decreased the number of indegree. Whenever in degree of a node is zero enque it. and remove the corresponding edges. after each loop dequee and keep the ans in a list.

2) Same as task ~~1(b)~~ 1(a)

3) I sorted the nodes according to their finishing times and put them in a stack. Then I reversed the edges. Did dfs accoreding to the ordere in the stack.