Topological sort/ Course Schedule-II- Leetcode #210

Topological sort means in a graph if there is an edge u->v, in topological order u should come before v and this is true for all pairs of edges.
We can observe that this can be done for only Directed Acyclic Graph(DAG) as if a cycle exists like u->v->x->u then u should come before v and v should come before x and x should come before u which is impossible.

Method-1: Using DFS
In DFS, the nodes which have no further descendents to explore will be pushed in stack and hence the node with maximum descendents will be pushed last. Hence the topological order will be the stack contents.
Note: In this method, detection of cycle needs to be done if the graph is not acyclic.

Method-2: Using BFS
This method is popularly known as “Kahn’s Algorithm”.
In this an indegree vector is maintained which has the indegree(edges that have the node as destination) values of each node. Now we can say that the nodes with no indegree have to be at the beginning of the sort as it comes before the other nodes.
A queue is maintained for BFS operation and with processing each node, we have to delete the edges from that node(i.e., the indegree of other nodes from that node decreases). If any node is left with indegree >0 and all the nodes are explored then it shows that the graph has a Cycle. Hence sorting is impossible.