Task-1a:

This code is basically on topological sort in DFs approach. where the code reads input from the input file and search weither there is a cycle or not. if found returns impossible to do top sort and if not found then return the topological sort.

Task-1b:

same as task 1a but in BFs approach.

Task-2:

The code reads a directed acyclic graph's course dependencies from input file then applies a BFs using a priority queue (min-heap) to find the lexicographically smallest valid course-sequence.

Task-3:

The code reads the input file and applies kosaraju's algorithm to find the strongly connected components (sccs) and writes the lexicographically smallest sccs to an output file.