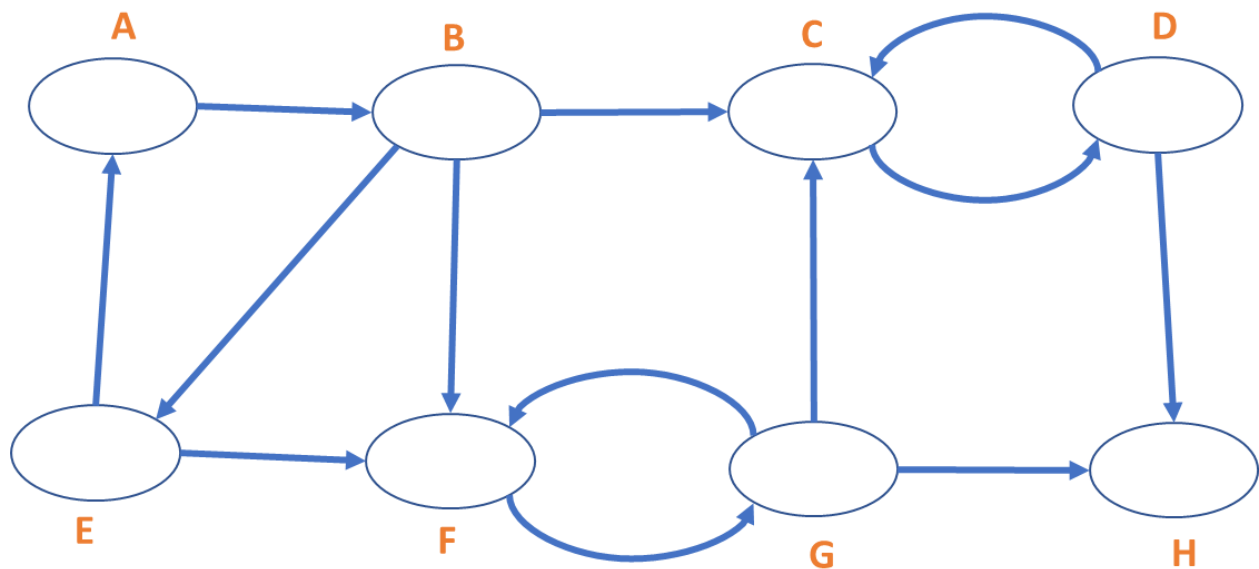


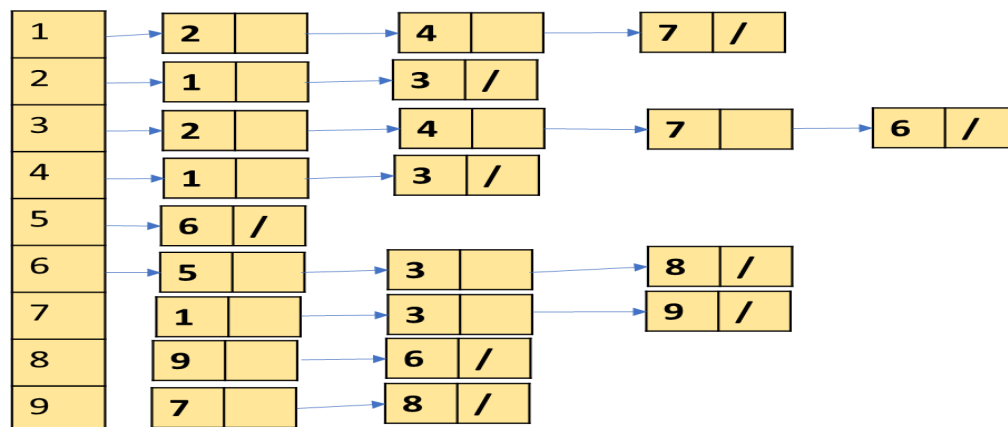
HOMEWORK-11

Total Points: 65

Q1. [20 Marks] Show all intermediate steps during the computation of strongly connected components for the following graph. Show the DFS forest with discovery and finish time for each vertex during the first DFS pass. The strongly connected components are numbered from 1 to k, where k is the number of strongly connected components. Mark the vertices with their corresponding strongly connected component number at the end.



Q2. [20 Marks] Write the DFS algorithm without recursion. Each vertex also stores its predecessor node in the DFS tree. Use the predecessor node for backtracking. The edges are stored in the adjacency list. In which order will the vertices be visited if we run your algorithm for the following graph?



Q3. [10 Marks] We can use DFS and BFS algorithms to find out whether a path exists between given two vertices.

- Give an example to demonstrate that DFS can be better than BFS for the above use case. Justify your answer.
- Give another example to demonstrate that BFS can be better than DFS for the above use case. Justify your answer.

Q4. [15 Points] Give an algorithm to print all forward edges while traversing a directed graph using DFS.