CSE 3101: Design and Analysis of Algorithms
Assignment on graph algorithms - 2

1. The Strongly-connected-components (SCC) algorithm in the book uses two DFS calls and a graph transposition. Here is an alternative algorithm suggested by a student that does not use transposition.

   1. Run DFS on the graph.
   2. Order nodes in the order of earliest finish times.
   3. Run DFS on the graph, choosing white nodes in this order. Output each DFS tree as a SCC.

   Come up with an input that shows this algorithm does not always work.

2. Some security experts are monitoring a computer network with n computers $C_1, \ldots, C_n$, tracking the spread of worms. The way worms spread is as follows. If an infected computer C_i communicates with an uninfected computer C_j at time t_k, the C_j becomes infected as well starting at time t_k. Infection can spread from one machine to another in a sequence of communications as long as time is non-decreasing in that sequence. It is acceptable for any two successive communication steps in that sequence to have the same timestamp. This means a computer C_j had open connections to both C_i, C_q at the same time and so a worm could move from C_i to C_q.

   The experts gather a trace indicating the times at which pairs of computers communicated. Thus the data is a sequence of ordered triples (Ci, C_j, t_k) that indicates that C_i, C_j exchanged data at time t_k. There are m such triples. Assume that the triples are sorted by time and that each pair communicates at most once.

   Your algorithm should take this data and be able to answer questions of the form "If a worm infected C_a at time x, could it have infected C_b by time y"? The algorithm should run in time $O(m + n)$.

3. A brokerage firm trades shares in n companies. For each pair they mention a ratio r_{ij} that indicates that one share of company i trades for r_{ij} shares of company j. We allow the ratio to be fractional.

   Trading a sequence of shares $i_1, \ldots, i_k$ successively in that order, and then trading shares for company i_k for company i_1, it may be possible to end up with more shares of i_1 than we started with. Give a polynomial time algorithm that finds such a cycle if one exists.

   Note: This is very similar to the arbitrage problem from an earlier assignment.

4. Some scientists are trying to analyze historical data about the lives of people who lived in a region over the last two centuries. This data was gathered orally and is prone to errors. The data pertains to n people, $P_1, \ldots, P_n$. Each data item is of one of the following forms:

   - For some i, j, P_i died before P_j was born,
   - For some i, j, the lifespans of P_i, P_j overlapped.

   Give an efficient algorithm to decide if the facts are consistent. More specifically, your algorithm should produce a possible date of birth and a possible date of death for each person that is consistent with the facts, or if no set of dates exist (that is consistent with the data) then it reports that fact.