Assignment 3

- Due Date: **10:00AM, Apr 16 (Fri), 2021**.
- Please submit one solution sheet per group.
- Please mention the names and roll numbers of all the group members on the first page of the answer sheet.
- Discuss only with your fellow group members.
- In case you use external sources for deriving your solutions, please specify them clearly.
- We will be able to cross-check your understanding of the solutions in the viva. For this it is important to know and understand every solution you write in your submission.

---

1. There are n jobs and m persons available to do the job.
   (a) You are given a list of pairs (J_i, P_j) which indicate that person P_j is willing to do job J_i. You have to determine whether it is possible to assign all n jobs to persons so that each person is assigned at most one job, and the person is willing to do the assigned job. Show how this problem can be solved using max-flow.
   (b) Modify the algorithm when each person i specifies an upper bound d_i on the number of jobs that he/she is willing to do.
   (c) Suppose now the jobs are classified as "boring" and "interesting". Each person specifies an upper bound d_i on the total number of jobs that he/she is willing to do, and also an upper bound b_i on the number of boring jobs that he/she is willing to do. Show how max-flows can be used to find a feasible assignment of jobs, if it exists.
   (d) Finally, consider a variation where the classification of jobs as boring or interesting is done separately by each person. Thus each person indicates which jobs he/she is willing to do, and whether the job is boring or interesting for him/her. Given upper bounds on the total number of jobs, and the number of boring jobs that each person is willing to do, use max-flows to find a feasible assignment of jobs to persons, if it exists.

2. Given a directed acyclic graph, it is required to find minimum number of paths in the graph such that every vertex is contained in at least one path.
   (a) Show how this can be formulated as a flow problem.
   (b) Modify the solution if every vertex is required to be in exactly one path.
   (c) Do the same if the condition is required for every edge, rather than every vertex.
   (d) Why does this not work if the graph contains cycles?

   [Hint: Use lower bound n flow to ensure every vertex/edge is covered.]

3. Read the definition of reductions from the Lecture 26 and read pages 454 – 459 of the textbook by Kleinberg and Tardos. For each of the problems below, show that they can be reduced to the Independent Set decision problem and vice-versa.
   (a) Disjoint Sets: Given a set S and a collection $C = \{S_1, S_2, \ldots, S_m\}$ of subsets of S, and a number k, are there k pairwise disjoint subsets in C?

(b) Given a graph G and a number k, does G contain a dominating set of size at most k? A dominating set is a subset of vertices such that every vertex not in the subset is adjacent to some vertex in the subset.

(c) Given a graph G, and a number k, is there a subset of k vertices in G such that there is no path of length at most 2 between any two vertices in the subset.

(d) Given a tree T, and a collection $\{T_1, T_2, \ldots, T_m\}$ of subtrees of T, are there k edge-disjoint subtrees in the given collection? Show that if the subtrees are required to be vertex-disjoint, the problem can be solved in polynomial-time.