- Always try to give algorithm with best possible running time. The points that you obtain will depend on the running time of your algorithm. For example, a student who gives an $O(n)$ algorithm will receive more points than a student who gives an $O(n^2)$ algorithm.
- You are required to give proofs of correctness whenever needed. For example, if you give a greedy algorithm for some problem, then you should also give a proof why this algorithm outputs optimal solution.
- **Use of unfair means will be severely penalized.**

There are 3 questions for a total of 50 points.

---

(15) 1. Recall the critical vertex problem discussed in class. Consider two vertices s and t in a given graph. A pair of vertices (u, v) (different from s and t) are called bi-critical with respect to s and t if the removal of u and v from the graph disconnects s and t. Suppose in a given graph the shortest distance between s and t is strictly greater than $\lceil n/3 \rceil$. Prove or disprove the following statement:

There exists a pair of vertices that are bi-critical with respect to s and t.

Give an algorithm for finding this pair of vertices in case there exists one.

(15) 2. A directed graph $G = (V, E)$ is called *one-way-connected* if for all pair of vertices u and v there is a path from vertex u to v **or** there is a path from vertex v to u. Give an algorithm to check if a given graph is one-way-connected.

(20) 3. You are a party organizer and you need to solve the following problem. There are n people and you know their friendship network. Your job is to decide a subset S of people who will be invited to the party. The constraint that you need to satisfy is that every person in the subset S, is friends with at least five other people in S **and** not friends with at least five other people in S. Assume that you are given the friendship network as a graph (assume adjacency list representation) where the edges denote friendships. Design an algorithm that maximizes the size of the set S.