1. (20 points) The following statements may or may not be correct. In each case, either prove it (if it is correct) or give a counterexample (if it is not correct). Always assume that the graph $G = (V, E)$ is undirected, connected and the edge weights are positive.

1. If a graph $G$ has more than $|V| - 1$ edges, and there is a unique heaviest edge, then this edge cannot be part of a MST.
2. If $G$ has a cycle with a unique heaviest edge $e$, then $e$ cannot be part of an MST.
3. Let $e$ be any edge of minimum weight in $G$. Then $e$ must be part of some MST.
4. If the lightest edge in a graph is unique, then it must be part of every MST.
5. If $e$ is a part of some MST of $G$ then it must be the lightest edge across some cut of $G$.
6. The shortest path tree computed by Dijkstra’s is necessarily an MST.

2. (20 points) Let $G = (V, E)$ be an undirected graph. Prove that if all its edge weights are distinct, then it has a unique minimum spanning tree.

3. (20 points) Consider the following problem: We have $n$ oxen, $Ox_1, \ldots, Ox_n$, each with a strength rating $S_i$. We need to pair the oxen up into teams to pull a plow; if $Ox_i$ and $Ox_j$ are in a team, we must have $S_i + S_j \geq P$, where $P$ is the weight of a plow. Each ox can only be in at most one team. Each team has exactly two oxen. We want to maximize the number of teams. Provide an efficient algorithm to compute the optimal matching along with its runtime analysis and proof of correctness.

4. (20 points) Suppose you and your friend live on the same street $D$ miles away from each other. Between your house and your friend’s house there are $n$ motorized scooters positioned at distances $x_1, x_2, \ldots, x_n$ miles from your house towards your friend’s house.

Each scooter has a battery that may or may not be fully charged. Let’s say that the scooters can take you distances $d_1, d_2, \ldots, d_n$ miles down the road.

Any time you encounter another scooter, you may take the new scooter and leave your old scooter behind.

There is an additional scooter at your house that you start with and it can go distance $d_0$ miles.

You wish to go to your friend’s house minimizing the number of times you change scooters.

Describe a greedy strategy that finds the smallest set of scooters that you can take to get to your friend’s house. (5 points)

Prove that it is optimal. (9 points)

Describe how to implement it efficiently. (3 points)

Calculate the runtime. (3 points)

5. (20 points) We are given a collection of intervals on the line: $I_1 = [\ell_1, u_1], \ldots, I_n = [\ell_n, u_n]$. We’d like to select a small set of points on the line, such that each interval contains at least one of the points.

Give an efficient algorithm for finding the smallest possible set of such points, and justify its correctness.

Provide a runtime analysis for your algorithm.