CS535 Homework 5

Due: 6pm, Nov. 19, 2022.

1. Suppose that an undirected graph $G = (V, E)$ has a perfect matching M. A node $v \in V$ is unhappy with its mate in M and requests for the full list of its neighbors which can be matched to v in some perfect matching. Give an $O\left(n^2\right)$-time algorithm to compute such full list from M and G.

2. A carpooling club in a suburban city has n members all working at the downtown. They intend to carpool by pairs daily. By the midnight of a day, each member submits his/her departure time interval for the next day.Two members can carpool if their departure intervals overlap. The club then computes a maximum number of carpooling pairs and inform the members about the assignment. Describe an $O(n \log n)$-time greedy algorithm for finding a maximum number of carpooling pairs. [*Hint*: Binary-search tree data structures may be used to speed up the algorithm]

3. A car repair shop has m technicians and n luxury cars to be repaired on a certain day. Each technician has expertise needed for a subset of cars, and each car requires the service by two technicians of needed expertise for the whole day. The shop seeks an assignment of technicians to the largest number of cars. Describe a polynomial-time algorithm for this assignment.

4. Let $G = (V, E; w)$ be graph with positive edge weights. A matching in G can be efficiently produced as follows. Start a path with a non-singleton node, and extend the path in *one* direction for as long as possible by keeping only the *heaviest* edge incident to the endpoint and removing all other edges incident to the endpoint. If the endpoint has no incident edge, a new path is started with a new node. This process goes on until no edges are left. Finally, these paths are split into two matchings, and the larger one is output.

   (a) Give a *linear*-time implementation of the above method.

   (b) Show that the output matching has at least half weight of a maximum-weighted matching in G.

5. Let $G = (V, E; w)$ be graph with positive edge weights, and k be a positive integer. Describe a polynomial-time algorithm to compute a maximum-weighted matching with at most k matched edges.

6. [**PhD Session only**] Suppose X is a subset of vertices in a $G = (V, E)$. Give a polynomial-time algorithm to find a matching M in G covering the largest number of vertices in X.