Problem 9: Tiger Rides (4+24 points) [File Upload]

By Pedro Paredes

Problem Statement

It is no secret that the university campus has been expanding a lot lately. There are so many buildings and some of them are so far away that walking between classes takes too long. To help mitigate this issue, the university is creating Tiger Rides, a ridesharing service to take students between classes.

The university's campus is represented by a **connected, undirected** graph with N vertices (representing buildings), and M weighted edges (representing paths/roads between buildings). The weight of an edge is a positive integer representing the time to traverse a vehicle needs to traverse that edge (assume there are no fluctuations because of traffic or lights). The university has a list of D demands, which are pairs of distinct vertices (u, v) that indicate that there is some student that needs to go from a class in vertex u to a class in vertex v. It is guaranteed that initially no two students are in the same place.

Tiger Rides has a fleet of V vehicles, each of which can transport two people at a time, in addition to a driver. Vehicles start in some vertex of the graph (parked near the building) and they can move from vertex to vertex, following the edges. If a vehicle is currently on vertex v and transporting less than two people and there is a student on vertex v that wants to go somewhere, the vehicle can pick them up (assume this takes no time). If a vehicle is currently on a vertex v transporting a student that want to go to vertex v, it can drop them off (assume this takes no time). For safety reasons, once a vehicle picks up a student, it can only drop them off at their desired destination. Also, you can assume that multiple vehicles can be in the same vertex or edge at the same time, there is no restriction on this.

Given the Princeton building graph, the list of demands, the initial location of each vehicle, your goal is to find a plan to complete each demand, i.e. move each student between their desired locations. A plan is preferred to another plan if the total distance traveled by all the vehicles is minimized.

Task summary: Given a graph, a list of demands, and a fleet of vehicles, find a plan to complete each demand.

Test Case Format

The format of each test case is as follows. The first line contains 4 integers separated by spaces, N, M, D, V. Then follow M lines, each containing three integers u, v, w, representing an edge between buildings u and v (both integers between 1 and N) of weight w. Then follow D lines, each containing two distinct integers between 1 and N, representing a demand between the corresponding vertices. Finally follows a line with V space separated integers, the ith of which represents the initial location of the ith vehicle.

Note that there might be multiple demands ending in the same vertex, meaning multiple students might want to end up at any one vertex. Multiple vehicles can also be start in same vertex. But no to students start at the same vertex.

You should output a plan with one step per line. A step should be one of the following:

- `M` c v, where `M` is the uppercase letter "M", c is a number between 1 and V that represents a vehicle and v is a number between 1 and V representing a vertex. This step means "move vehicle c to vertex v". If the current location of vehicle c isn't adjacent to vertex v, then your plan is considered invalid.
- `P` c, where `P` is the uppercase letter "P" and c is a number between 1 and V that represents a vehicle. This step means "pick up a student on vehicle c". If the current location of vehicle c doesn't contain a student waiting for a ride, or if vehicle c is currently transporting 2 other students, then your plan is considered invalid.
- `D` c, where `D` is the uppercase letter "P" and c is a number between 1 and V that represents a vehicle. This step means "drop off all valid students on vehicle c", so it drops off all students currently in c that want to end up in its current location. If the current location of vehicle c doesn't contain a student wanting to be dropped off there, then your plan is considered invalid.

Grading

You will be given 4 test cases, with possibly different graphs, demands and fleet. For each one you should compute a valid plan in the format specified above. You will be awarded 1 point per test case if your solution leads to a plan with total distance traveled by all the vehicles less than the following:

- Test 0: 600000
- Test 1: 100000
- Test 2: 500000
- Test 3: 550000

You can win an additional 6 points per test. We will use a type of "grand prix" model to assign these points. If your plan is better than the above threshold and is better than the plans submitted by all the other teams, you win 6 points for that test. If your plan is the second best, then you win 5, etc. Note that only your best plan is going to be counted (so you can't win points for 1st and 2nd best plans). This means that your score for this problem might change as other teams submit solutions.

You can find all the the test cases in the `trides_data.zip` file listed on the problems section of the website. This zip file contains 4 files, `trides_T.txt`, where `T` is the test case number, from 0 to 3, containing the respective input.

You should submit one file called `XXXXX_p9.zip`, where `XXXXX` is your team ID. This zip file should contain files following the format `XXXXX_T.txt`, where `T` is the test case number, from 0 to 3, containing the respective plan. If you don't have a solution to one of the tests you don't have to include it.

Tip: The 4 test cases are not arbitrary. To be successful in this problem you should analyze them and try to take advantage of any special structure you find.