FARHAN AKBOR KHAN
ID: 20301230 ; Section: 08 (Spring'22)

Task: 03

I used Dijkstra algorithm to the task 1 and 2. Now, assume that the places are N and roads are M. So, the time complexity of Dijkstra function is $O(M \log N)$. And the time complexity of the adjacency list is $O(M+N)$.

So, the total time complexity of task 1 is $O(M+N) + O(M \log N)$.

The time complexity of task 2 is same as task 1. Because, I used the Dijkstra algorithm in task 2.

So, time complexity of task 2 is $O(M+N) + O(M \log N)$.

If the number of titans in each road is exactly 1, there is an $O(N+M)$ algorithm which is BFS.

When the weights are same then there is no necessity of Dijkstra Algorithm, because Breadth First Search (BFS) is enough to solve the problem.