CS2001 – Data Structures Semester Project

Food Delivery Routing Optimization

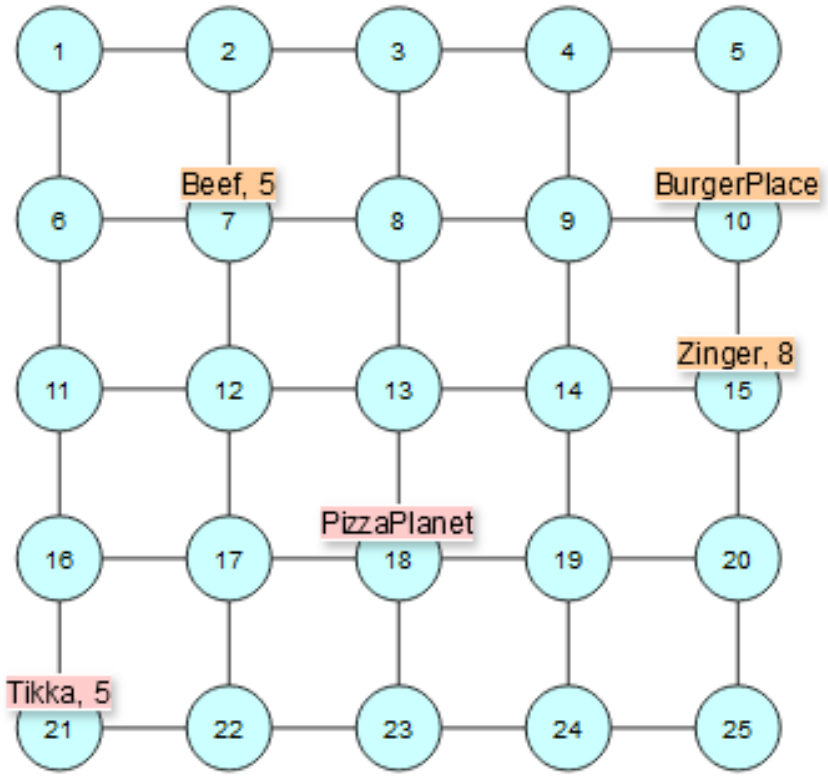
Due Date: Sunday, May 5th, 2024 11:59pm

Group Members: 3

Food Polar Bear, a delivery company operating in Grid City wants to optimize its route planning for delivering orders to customers efficiently. Grid City consists of multiple restaurants and customers scattered across a grid similar to Manhattan. The company needs to deliver orders from different restaurants to customers while considering various constraints, such as time windows for deliveries and minimizing travel time.

Suppose the company receives orders from two different restaurants: Burger Palace and Pizza Planet. Each restaurant has its own set of orders with specific delivery locations. The delivery personnel must plan their routes to cover the maximum number of deliveries in a day while adhering to the time constraints.

For instance, the delivery personnel start from Burger Palace and deliver orders to nearby customers. Then, they move to Pizza Planet and deliver orders from there. They optimize their routes to avoid backtracking and ensure timely deliveries. The total minimum time required to deliver all orders efficiently is calculated based on the route plan.



Instructions

- Represent the city map as an N x N grid where nodes should be numbered from 1 to N^2, going from left to right and top to bottom.
- Restaurants and customers are located at the nodes in a graph, and edges represent road connections.
- Each edge takes 1 unit time to travel.
- Consider time constraints for each delivery and plan routes accordingly to ensure timely deliveries.
- Optimize the routes to cover the maximum number of deliveries in a day while minimizing travel time.

Input

The input to the algorithm will be contained in a text file in the following order:

- The first line contains the number of test cases
- For each test case:
    - The first line contains grid size (N), number of riders (I), and number of restaurants (R)
    - For each restaurant:
        - The first line contains its name, its location, and the number of orders (O)
        - The next O lines contain the name, location, and delivery time limit of each order

Output

For each test case, the algorithm should output the total minimum time required to deliver all orders efficiently.

Sample Input	**Sample Output**
2 5 2 2 BurgerPalace 10 2 Beef 7 5 } Order 1 } Restaurant 1 Zinger 15 8 } Order 2 } Test Case 1 PizzaPlanet 18 1 Tikka 21 5 } Restaurant 2 5 2 1 CurryHouse 10 3 Chicken 2 7 } Test Case 2 ButterChicken 18 5 Biryani 15 2	10 8