**DATA STRUCTURES – FALL 2023**

**LAB 14**



**Lab Task: City Map Navigation**

**Objective:** Implement a city map navigation system using a graph representation, where intersections are vertices and roads are edges. Develop functions to find the shortest path between two locations in the city.

**Task Steps:**

**Step 1: Graph Representation (10 minutes)**

* Create a C++ class named **CityMap** to represent the city map.
* Include necessary data members to store intersections and roads (edges).
* Implement a constructor to initialize the city map.

**Step 2: Add Intersection and Road (20 minutes)**

* Implement a function **addIntersection(int intersectionID)** to add an intersection to the city map.
* Implement a function **addRoad(int source, int destination, int length)** to add a road between two intersections with a specified length.

**Step 3: Display City Map (10 minutes)**

* Implement a function **displayCityMap()** to display the intersections and roads of the city map.

**Step 4: Shortest Path Algorithm (20 minutes)**

* Implement a function **findShortestPath(int start, int end)** to find the shortest path between two intersections.
* Use priority queues to efficiently select the next intersection with the shortest path.

**Step 5: Test the Implementation (15 minutes)**

* In the **main()** function, create an instance of the **CityMap** class.
* Add a menu that will help intersections and roads to simulate a city map and display the city map.
* Test the **findShortestPath()** function for different pairs of intersections.

Below is a sample sequence of inputs you can use to test the city map navigation system:

**Add Intersections:**

Enter Intersection ID: 1

Enter Intersection ID: 2

Enter Intersection ID: 3

**Add Roads:**

Enter Source Intersection ID: 1

Enter Destination Intersection ID: 2

Enter Road Length: 5

Enter Source Intersection ID: 2

Enter Destination Intersection ID: 3

Enter Road Length: 3

**Display City Map:**

Choose option 3 from the menu to display the city map.

**Find Shortest Path:**

Enter Start Intersection ID: 1

Enter End Intersection ID: 3

The system will display the shortest path between the specified intersections.

 