

TraVeling System using Dijkstra Baalti Algorithm

**Submitted to: Dr. Junaid Akhtar**

**Submitted by:**

**Group Members**

1. Ali Murtaza (39)
2. Sardar Moazzam Sajid (17)
3. Hamza Iqbal (10)
4. Muhammad Asad Ayyub (25)



July 30, 2021

course: Analysis of Algorithm

How we Solved Dijkstra Baalti Algorithm?

* Read the files of city names and distances and stored them in separate lists and after that converted into a graph using dictionary data structure.
* We used Dijkstra Baalti Algorithm with Short Distance as Priority and implemented it by changing BFS Baalti Algorithm queue to Priority Queue.
* We also keep track of paths with root as its starting city and by using tracks of paths we answered the questions of find shortest and longest paths.

**Q1: What is the Shortest Path between two cities?**

Answer:

* For shortest path between two cities we call the Dijkstra Baalti Algorithm and from there we used short path tracks which are stored in a dictionary returned by Dijkstra Baalti and by using that dictionary we find out the shortest path our origin city and destination city.
* Since we are traversing the graph in Priority Queue manner so paths are also stored on the basis of shortest distances, we only find in shortest paths that which path started from root city and has destination city at where we have to stop.
* If Direct Path is not the shortest or we put zero in shortest then we can also get shortest path available.

**Q2: What is the shortest path if we start from a city and travel all other cities in other words what is the shortest path to do world tour?**

**Answer:**

* We used Dijkstra Algorithm but with little modification, since we had to travel the shortest path so we check root adjacent cities and using priority queue find the shortest city from root city.
* When we travel the shortest city from root city then we make this city as root check its adjacent cities and find the closest city from that city and also check that if it is already travelled or not. In this way we travel all cities of graph by following shortest path.

**Q3 : What is the longest path if we start from a city and travel all other cities in other words what is the longest path to do world tour?**

**Answer:**

We solved this problem exactly same as we solved the above shortest world tour problem but the change we do is that we change priority queue condition to longest distance.

**Q4: What is the longest path between two cities?**

**Answer:**  For this we travel from origin and travelled the whole thirty cities using the longest distances priority queue and we go to our destination city only when all other cities are travelled.

**Q5: What is the best shortest path to travel all 30 cities?**

**Answer:** For this we loop on the list of all 30 cities and check the distance of shortest path with every city as source by calling the shortest path aglorithm function on every city and at the end we get the path which has the most lowest distance from all paths.

**Q6: What is the worst shortest path to travel all 30 cities?**

**Answer:** For this we loop on the list of all 30 cities and check the distance of longest path with every city as source by calling the longest path aglorithm function on every city and at the end we get the path which has the highest distance from all paths.