

Project 1: Electric Car Traveler

Team Members:

Jayraj Arora Email: jayraj.arora@csu.fullerton.edu

Manan Hasmukhbhai Patel Email : mananpatel@csu.fullerton.edu

Sravani Kallempudi Email: sravanik@csu.fullerton.edu

Summary:

Given the range of car C , number of cities n , and distances(edges) between the cities, problem is to find the minimum stops needed to recharge an electric car to reach from the source to the destination; such that if the current city's charger is broken, car can return back to the previous city to recharge there.

Pseudocode:

- Create a doubly linked list of distances and cities (e.g. A | 0 <-> B | 90 <-> C | 60 <-> D | 70 <-> E | 65 <-> F | 83 <-> G | 75 <-> H | 72).
- stops would be the result that is the no. of minimum stops needed
- Append the first node to stops list(as it is the first node where car starts)
- Let $curNode = head$ // pointer to the head of linked list
- Let $curStopDistance = 0$ // this will contain the distance to reach the current city(e.g. 90 to reach city C)
- Iterate till we traverse the entire linked list ($curNode.next == null$)
 - Let $distanceTravelledSoFar = 0$ // this will contain the total distance travelled by the car in one charge. Resets to 0 once a stop is made.
 - Iterate till $distanceTravelledSoFar + curStopDistance > C$ (car range) or we reach the last node
 - $curNode = curNode.next$
 - $curStopDistance = curNode.distance$
 - $distanceTravelledSoFar = distanceTravelledSoFar + currentStopDistance$
 - if $distanceTravelledSoFar + currentStopDistance > carRange$
 - Go to the previous stop ($cur = cur.previous$) // Here car needs to be recharged again.
 - Add this stop to the stops list
- Add the last stop to the stops list
- return stops

Input Parameters:

C = capacity of the car

n = no of cities/nodes

distances between the cities

Utility Class:

// class representing the node of a doubly linked list

DistanceNode:

char city

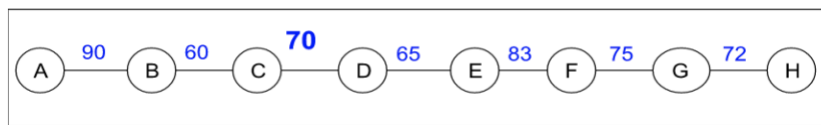
int distance

DistanceNode previous

DistanceNode next

Example:

The capacity is C = 300 miles, the starting city = A and the destination city = H, the miles between cities are shown below.



We need to compute the list of stops starting with A and ending in H such that the number of stops is minimized, in case the charge station in a stop city is broken, one can make it back to the previous city.

Output: [A, D, G, H]

How code runs:

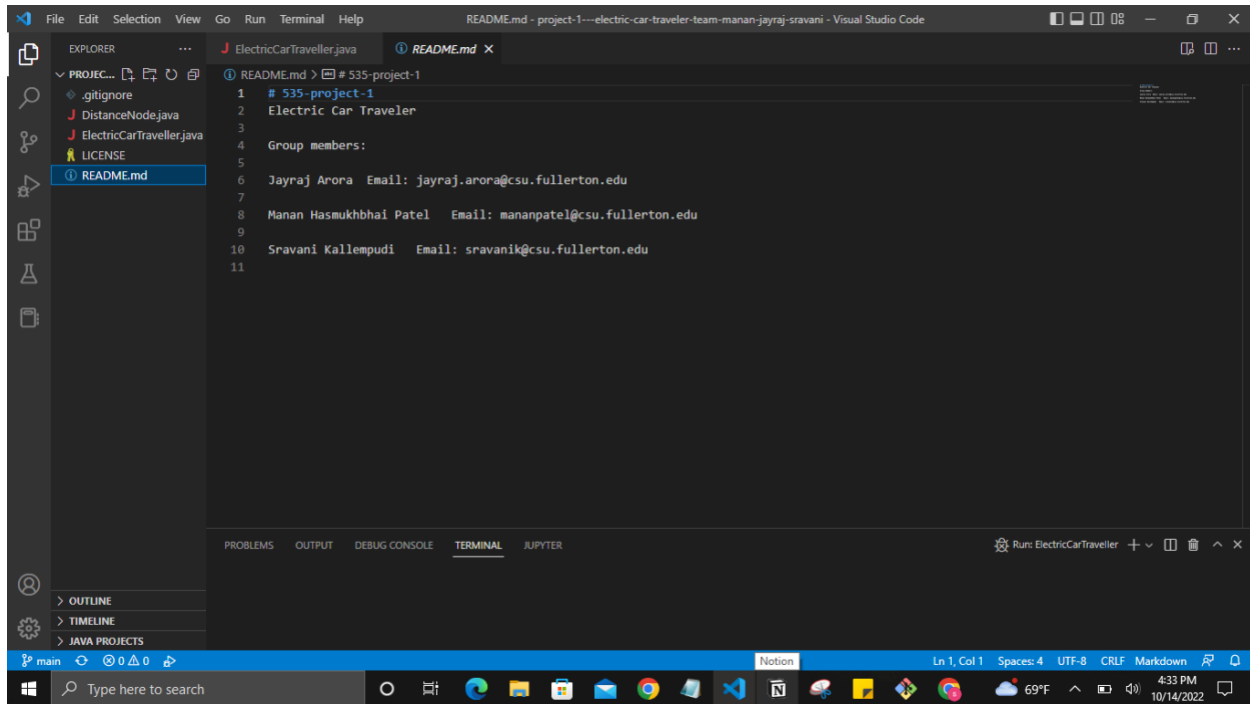
```
cd project-1---electric-car-traveler-team-manan-jayraj-sravani/
```

```
javac ElectricCarTraveller.java
```

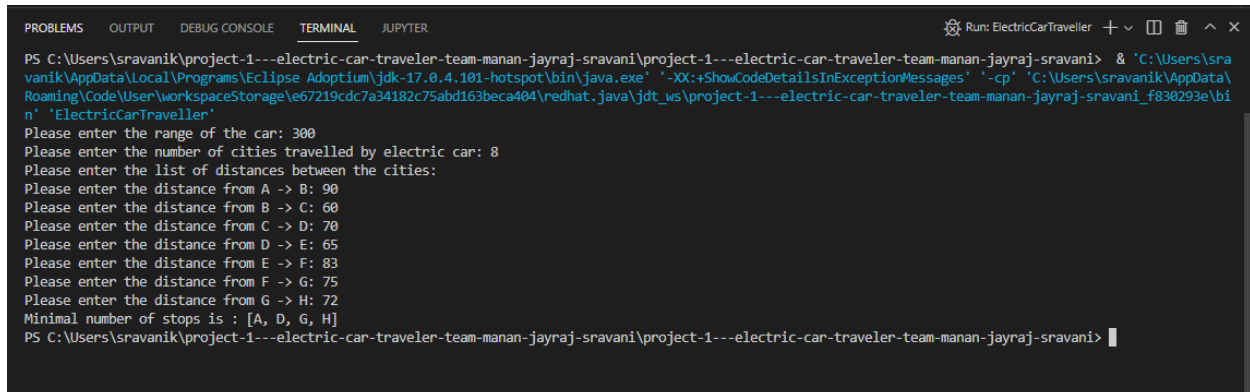
```
java ElectricCarTraveller
```

Screenshots:

1. Screenshot of group members:



2. Screenshot of Test case 1:



3. Screenshot of Test case 2:

```
PS C:\Users\sraavanik\project-1---electric-car-traveler-team-manan-jayraj-sravani\project-1---electric-car-traveler-team-manan-jayraj-sravani> c;; cd 'c:\User
s\sraavanik\project-1---electric-car-traveler-team-manan-jayraj-sravani\project-1---electric-car-traveler-team-manan-jayraj-sravani'; & 'C:\Users\sraavanik\AppData
ata\Local\Programs\Eclipse Adoptium\jdk-17.0.4.101-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\sraavanik\AppData\Roaming\Co
de\User\workspaceStorage\67219cdc7a34182c75abd163beca404\redhat.java\jdt_ws\project-1---electric-car-traveler-team-manan-jayraj-sravani_f830293e\bin' 'Electr
icCarTraveller'
Please enter the range of the car: 300
Please enter the number of cities travelled by electric car: 8
Please enter the list of distances between the cities:
Please enter the distance from A -> B: 90
Please enter the distance from B -> C: 94
Please enter the distance from C -> D: 80
Please enter the distance from D -> E: 65
Please enter the distance from E -> F: 83
Please enter the distance from F -> G: 75
Please enter the distance from G -> H: 72
Minimal number of stops is : [A, C, E, G, H]
PS C:\Users\sraavanik\project-1---electric-car-traveler-team-manan-jayraj-sravani\project-1---electric-car-traveler-team-manan-jayraj-sravani>
```

4. Screenshot of Test case 3:

```
PS C:\Users\sraavanik\project-1---electric-car-traveler-team-manan-jayraj-sravani\project-1---electric-car-traveler-team-manan-jayraj-sravani> c;; cd 'c:\User
s\sraavanik\project-1---electric-car-traveler-team-manan-jayraj-sravani\project-1---electric-car-traveler-team-manan-jayraj-sravani'; & 'C:\Users\sraavanik\AppData
ata\Local\Programs\Eclipse Adoptium\jdk-17.0.4.101-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\sraavanik\AppData\Roaming\Co
de\User\workspaceStorage\67219cdc7a34182c75abd163beca404\redhat.java\jdt_ws\project-1---electric-car-traveler-team-manan-jayraj-sravani_f830293e\bin' 'Electr
icCarTraveller'
Please enter the range of the car: 300
Please enter the number of cities travelled by electric car: 8
Please enter the list of distances between the cities:
Please enter the distance from A -> B: 90
Please enter the distance from B -> C: 60
Please enter the distance from C -> D: 80
Please enter the distance from D -> E: 94
Please enter the distance from E -> F: 53
Please enter the distance from F -> G: 75
Please enter the distance from G -> H: 72
Minimal number of stops is : [A, C, F, H]
PS C:\Users\sraavanik\project-1---electric-car-traveler-team-manan-jayraj-sravani\project-1---electric-car-traveler-team-manan-jayraj-sravani>
```