Lab Assignment 9 Due on Oct 14, 2pm

In this assignment, we will try to find a journey from a given source station to a given destination station using a modification of Breadth-First Search (or BFS), which you would have already studied in class.

You have to implement the function printDirectJourneys in file printJourney.cpp This function takes the source and destination station names as strings, and you are required to print out in a nice formatted manner the list of all distinct direct journeys (i.e. no change of trains) from source station to destination station, if available. Otherwise, you must print out a message saying "No direct journeys available".

The connection between stations through journeys or trains is given by the "adjacency" array in the Planner class (see planner.h). Please feel free to look up the declarations of relevant classes in dictionary.h Each entry of the adjacency array is a pointer to an object of type StationAdjacencyList. You can see declaration of this class in dictionary.h Each object of class StationAdjacencyList has two lists, called "toStations" and "fromStations". Each of these is a list of pointers to objects of class StationConnectionInfo (also defined in dictionary.h). Each object of class StationConnectionInfo has a station index (recall all station names are mapped to unique indices through the dictionary stnNameToIndex in the class Planner (see planner.h), and a list of pointers to TrainInfoPerStation, which we have already used in Lab 7 (you can see its definition in dictionary.h as well).

Be aware that a simple BFS may not give direct journeys since there may be multiple trains connecting stations A to B and B to C, and unless the same train connects A to B and B to C, we can't use the train to go from A to C through B.

Happy planning journeys for our users!