CSU22012: Final Project

Vancouver Bus Management System

Louis Clarke - 20332725

April 2022

Design Document Contents

1.1

The first query that the user can select allows them to find the shortest paths between 2 bus stops (selected by the user) and be presented with the list of stops en route as well as the associated "cost". I used the object transportMap to read in all the files necessary and created an EdgeWeightedDigraph to represent all the stops with edges representing the existing bus routes between stops and transfers between stops. The transportMap object also contains three key maps that pair Stop ID's to the vertex index of each stop, vertex index to the stop ID of each stop and each stop vertex index to the name of each stop. To find the shortest route between stops I tried to use Floyd-Warshall but with a worst case time complexity of O(V^3) where V is the number of vertices. So I decided to implement the DijkstraSP class from the Princeton library. This algorithm has a worst case time complexity of O(V + ElogV) which I deemed to be sufficient. And in order to show the user the shortest path between the two stops that they selected I called the dijkstraSP.distTo(), dijkstraSP.hasPathTo() and dijkstraSP.pathTo() methods. These of course return the index number of each vertex, which I then used to fetch Stop ID and Stop Name information using the key-maps that were stored in the transportMap object.

1.2

The second query that the user can select allows them to search for a bus stop by full name or by the first few characters in the name, using a ternary search tree (TST), returning the full stop information for each stop matching the search criteria (which can be zero, one or more stops). Using the TST class from the Princeton library and my nameSearch object class, my program reads in the file "stops.txt" and stores every stop as a key-value pair in the TST. With the stop name as the key and storing the line number in the file where the rest of the stops info is stored as a value. TST has a worst case time complexity of O(N) when inserting or deleting a node and an average case of O(logN). The method TST.keysWithPrefix() returns a list of stop names which my program then uses to print a list of stops and information about each stop. In order to refrain from printing a list of hundreds of stops that would overwhelm the user with information, it instead only prints to the user the top ten stops that match the most with the user's search query.

1.3

The third query that the user can select allows them to search for all trips with a given arrival time, returning full details of all trips matching the criteria (zero, one or more), sorted by trip id. The information needed for this query is stored in stop_times.txt. The class arrivalTime handles all functionality needed for this query. An arrivalTime object stores all information about a trip (trip ID, arrival time, departure time etc.). Initially I was going to simply store all trips from stop_times.txt in an ArrayList but searching for a list of matching stops and creating the list was way too inefficient. I then implemented the method storeArrivalTimesHashMap(). Which reads in stop_times.txt, creates an arrivalTime object for each line in the file and then stores the arrivalTimes in a HashMap. The HashMap uses the arrival time of each trip as a key and contains ArrayLists of arrivalTime objects. When the user gives a time to search for anArrayList of trips with a matching arrival time is returned from the HashMap. I then implemented mergeSort to sort the list by trip ID which has a worst case time complexity of O(NlogN). Finally the sorted arrayList is then printed so the user can view trip information for all of the trips that match their query.

2.1

I decided to use a simple command line interface in order to make the system as easy to use as possible and display queried information in a readable format. I also made use of some ASCII art generators and markdown to make the system somewhat visually pleasing. This allowed me to use a simple while loop and switch statement as the basis for my main class. I used Scanner to handle user input which made error and edge case handling quite straightforward.

Full Title of Your Paper

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat (Lorem, 20XX). Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan.

Method (Heading Level 1)

Participants (Heading Level 2)

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan.

Assessments and Measures

Ut blandit malesuada quam, ac varius tortor gravida eget. Vestibulum id ligula leo, ut accumsan mi. Sed tristique euismod convallis. Nulla facilisi. Etiam vestibulum est id orci interdum vitae porta enim blandit. Cras sit amet arcu dolor, at venenatis erat. Vestibulum accumsan placerat mauris. Morbi nec nibh nibh. Duis ultricies posuere nunc. Morbi at tellus quis magna vestibulum eleifend.

Heading Level 3

Maecenas ullamcorper bibendum consequat. Pellentesque ultrices, eros eu tincidunt pretium, magna leo volutpat libero, non bibendum diam nunc eget urna. Vivamus eu tortor et dui aliquam vestibulum at vel augue. Vivamus elit dui, porttitor eget egestas at, rhoncus in justo. Curabitur tristique, elit ac venenatis volutpat, eros mauris iaculis diam, vitae rhoncus erat metus vitae eros.

Heading level 4 is inline with the following paragraph, separated by a period. Nulla congue egestas ante, id ultricies orci dignissim commodo. Fusce placerat, libero eu pharetra pulvinar, lorem dui pulvinar nisi, et semper orci orci vitae magna. Nullam sodales, felis id feugiat scelerisque, tortor nulla interdum mauris, ac porttitor odio dolor eget eros.

Second level 4 heading. Duis sit amet ipsum pretium erat accumsan iaculis vitae eget risus. Donec ut dui in lorem volutpat fermentum bibendum pulvinar libero. Nunc imperdiet eros et mi posuere pellentesque. Donec tincidunt ipsum eget nisl ullamcorper eu placerat libero ullamcorper. Maecenas id luctus ligula. Cras condimentum eleifend nibh sit amet iaculis. Suspendisse placerat sollicitudin mi, vel ornare augue hendrerit ac. Nulla sed suscipit sapien. Cras pellentesque orci lectus, eu consequat enim.

Heading level 5 is also inline. Nulla congue egestas ante, id ultricies orci dignissim commodo. Fusce placerat, libero eu pharetra pulvinar, lorem dui pulvinar nisi, et semper orci orci vitae magna. Nullam sodales, felis id feugiat scelerisque, tortor nulla interdum mauris, ac porttitor odio dolor eget eros.

Second level 5 heading. Duis sit amet ipsum pretium erat accumsan iaculis vitae eget risus. Donec ut dui in lorem volutpat fermentum bibendum pulvinar libero. Nunc imperdiet eros et mi posuere pellentesque. Donec tincidunt ipsum eget nisl ullamcorper eu placerat libero ullamcorper. Maecenas id luctus ligula. Cras condimentum eleifend nibh sit amet iaculis. Suspendisse placerat sollicitudin mi, vel ornare augue hendrerit ac. Nulla sed suscipit sapien. Cras pellentesque orci lectus, eu consequat enim.

Results

Maecenas id luctus ligula. Cras condimentum eleifend nibh sit amet iaculis. Suspendisse placerat sollicitudin mi, vel ornare augue hendrerit ac. Nulla sed suscipit sapien. Cras pellentesque orci lectus, eu consequat enim.

Outcome 1

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Outcome 2

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Discussion

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan.

References

Lastname, C. (2008). Title of the source without caps except Proper Nouns or: First word after colon. *The Journal or Publication Italicized and Capped*, Vol#(Issue#), Page numbers.

Lastname, O. (2010). Online journal using DOI (digital object identifier). *Main Online Journal Name*, Vol#(Issue#), 159-192. https://doi.org/10.1000/182

Lastname, W. (2009). *Title of webpage*. Site Name. Retrieved July 3, 2019, from http://www.example.com