**Title Page (KG)**

## **1 Introduction**

## **Application Requirements (DC, SD, SL)**

## **Technical Approach (DC, SD, SL)**

## **2 Design**

## **Logic Flow Diagram (KG)**

## **3 Implementation**

## **Tools, Libraries, Platforms (ALL)**

## **User Interfaces (KG)**

## **Algorithms (DC, SD, SL)**

## **Runtime Environment (KG)**

**D.C. - Introduction** :

*Requirements*:

* To implement part of a bus management system based on Vancouver bus system data
* Implement an algorithm to search for a bus stop by full name or by the first few characters in the name, using a ternary search tree (TST) and returning the full stop information for each stop matching the search criteria (which can be zero, one or more stops)
* I had to move keywords *flagstop, wb, nb, sb, eb* from start of the names to the end of the names of the stops when reading the file into a TST (eg “WB HASTINGS ST FS HOLDOM AVE” becomes “HASTINGS ST FS HOLDOM AVE WB”), in order to provide meaningful search functionality.

*Technical approach:*

* I planned out what I wanted to do for my implementation and how I wanted to do it.
* I examined the requirements carefully.
* Researched more about ternary search trees (my algorithm)
* I simplified it in different parts and worked each one out one by one
* Created a really simple version of the algorithm and over time I added more and more to it until it met all the requirements.

*Implementation*:

*Tools, Libraries, Platforms:*

* Eclipse on MacOS

*Algorithm*:

* Ternary search tree was used.
* A ternary search tree is a type of prefix tree where nodes are arranged as a binary search tree.
* There are two classes, a class for the Node and a class for the TernarySearchTree.
* The node class has a constructer for the nodes in the tree.
* I have an insert function in the TernarySearchTree class which is used to insert the list of bus stops into the tree
* The average Time Complexity for the insert function is of O(log n) and the worst case is O(n)
* There is a search function in the TernarySearchTree class which is used to search for the word (bus stop name).
* The average Time Complexity for the search function is of O(log n) and the worst case is O(n)
* There is a function in TernarySearchTree class to traverse the tree to be able to find the searched word
* The average Time Complexity for the transverse function is of O(log n) and the worst case is O(n)
* This algorithm has a Space Complexity of O(1).