CS 308 Group T08 Class Diagram Description by

Alin Vlad Bulai, Tomas Chmelevskij, Scott Hurley, Kelly Eve Taylor, Dimitar Plamenov Todorov

Roles of Interfaces and Classes

INode - Node interface hides non-pertinent information from map and graph.

Node - Represents nodes in graph.

IEdge - Edge interface hides non-pertinent information from map and graph.

Edge - Represents lines on map and edges in graph.

GraphADT - Describes graph data structure.

MetroMapParser - Takes text file and parses it to be constructed as a graph.

BostonMetroSystem - Constructs a Simulation of the Boston Metro System as a graph.

MainApp - Starts the program.

BadFileException - Used when a file has incorrect formatting.

Search - Searches through the graph for the shortest path.

Relationships between Interfaces and Classes

Implementation - Node implements the INode interface. Edge implements the IEdge interface. BostonMetroSystem implements the GraphADT interface.

Dependency - BostonMetroSystem depends on INode and IEdge. MetroMapParser is dependent on GraphADT. MainApp is dependent on MetroMapParser and BostonMetroSystem.

Composition - BostonMetroSystem is composed of multiple node and edge objects. Each node and edge object is associated with one GraphADT or BostonMetroSystem object.

Method descriptions

Node

Node (constructor) - Sets the node’s id and station name as the strings given to it as input.

getId - Returns the node’s id as a integer.

getStationName - Returns the node’s station name as a string.

Edge

Edge (constructor) - Sets the two nodes the edge connects and its line colour as the two integers and string passed to it.

getFirstID - Returns the id of the first node the edge connects to.

getSecondNode - Returns the id of the second node the edge connects to.

getLineColour - Returns the line colour of the edge as a string.

MetroMapParser

MetroMapParser (constructor) - Creates a new parser that will read the file’s filename unless it doesn’t exist. Throws java.io.IOException and returns a MetroMapParser object.

generateGraphFromFile - Parses the file and generates a graph from it, unless it can’t read it (throws java.io.IOException) or there are format issues (ex3.BadFileException). Returns the generated graph.

BostonMetroSystem

addNode - Adds a station to the map. Takes a node as input.

addEdge - Adds a line to the map. Takes an edge as input.

getSuccessors - Returns a list of all station ids directly connected to the station whose id integer is passed as the input.

BFS - Searches the map (breadth first) for the shortest path from the starting station to the destination station. Takes starting and end nodes as input and returns a list of node id’s visited.

processPath - Helper method for finding the shortest path. Takes source node id, destination node id and the path as inputs, returns an ArrayList of node id’s.

MainApp

main - Starts the program. Throws IOException if the input file is incorrect.

BadFileException

BadFileException - Prints a message when file has incorrect formatting.