Testing individual classes:

Class Name	Function	Sample Parameters	Expected Output
RouteApp	public RouteApp()	None	Prompts user to enter origin and destination and controls flow of program and invokes other objects, as required
Map	Public Map (String townFileName, String roadFileName)	"towns.txt" "roads.txt"	A Map object containing list of Towns and Roads
	private void readTowns(String filename)	Sibu 11158.58 21551.12	Reads data from file and creates town object with below attribute values: Name: Sibu Longitude: 11158.58 Latitude: 21551.12 Roads:null
	private void readRoads(String filename)	Sibu Selangau	Reads data from file and creates road object with below attributes: Origin: Sibu Destination: Selangau Distance: 400km roadType: "1-way" id: 0 Also, the *roads* attribute of Town object created above will be updated to point to this Road object
	public Town searchTown (String TownName)	"Sibu"	Town object should be returned with below attributes: Name: Sibu Longitude: 11158.58 Latitude: 21551.12
	public HashSet<Town> getTownList()	None	HashSet with all the Town objects read from text file, should be returned
	public ArrayList<Road> getRoadList()	None	ArrayList with all the Road objects should be returned
Town	public Town(String name, double theLatitude, double	Sibu 21551.12 11158.58	A new Town object should be created with below attributes:

	theLongitude)		Name: Sibu Longitude: 11158.58 Latitude: 21551.12
	public String getName()	None	"Sibu" should be returned
	public double getLongitude()	None	11158.58 should be returned
	public double getLatitude()	None	21551.12 should be returned
	public ArrayList<Roads> getRoads()	None	An ArrayList with all road objects connecting this Town to other towns Eg: Road object connecting Sibu to Selangau Road object connecting Sibu to Daro
	public int getTotalRoads()	None	Should return total number of Road objects, which connect this Town to other towns Eg: 5
	public void addRoad(Road theRoad)	Road object with below attributes: Origin: Sibu Destination: Selangau Distance: 400km roadType: "1-way" id: 0	The Road object passed in as parameter will be added to the *roads* attribute of this Town object
RouteCalc			
	public void setOrigin(Town theOrigin)	Town object with below attribute values: Name: Sibu Longitude: 11158.58 Latitude: 21551.12 Roads:Conatins a pointer to a road object connecting Sibu and Selangau Conatins a pointer to a road object connecting Sibu and Daro	origin attribute of this class will be updated with the Town object passed in as parameter
	public void setDestination(Town theDest)	Town object with below attribute values: Name: Selangau Longitude: 1122755 Latitude: 22949.79	destination attribute of this class will be updated with the Town object passed in as parameter

		Roads:Contains a pointer to a road object connecting Selangau and Dalat Contains a pointer to a road object connecting Selangau and Sibu Contains a pointer to a road object connecting Selangau and Mukah	
	public void calculate()	None	*routeList* attribute will be updated with all possible routes between origin and destination
	public void checkRoute()	None	Checks each Route object in *routeList* attribute to see if the route is complete This is done by calling getStatus() method of the Route object.
	public void updateRouteList()	None	findNextTown() function of Route object returns an ArrayList<Route> for a town with more than one possible exit road updateRouteList() updates *routeList* attribute of this class in such a case
	public Route getShortestRoute()	None	Reference to the object pointed to by *shortestRoute attribute* is returned
	private void calcShortestRoute()	None	Determines shortest route and updates *shortestRoute attribute*
	private ArrayList<Route> getRouteList()	None	Returns an ArrayList<Route> containing Route objects, which connect origin and destination
Road	public Road(Town theOrigin, Town theDestination, int theID, String roadType)	Town object with below attribute values: Name: Sibu Longitude: 11158.58 Latitude: 21551.12 Roads:Conatins a pointer to a	Road object with below attributes is created: Origin: Sibu Destination: Selangau Distance: 400km roadType: "1-way"

		road object connecting Sibu and Selangau Conatins a pointer to a road object connecting Sibu and Daro Town object with below attribute values: Name: Selangau Longitude: 1122755 Latitude: 22949.79 Roads:Contains a pointer to a road object connecting Selangau and Dalat Contains a pointer to a road object connecting Selangau and Sibu Contains a pointer to a road object connecting Selangau and Mukah 1 "1-way"	id: 1
	public Town getOrigin()	None	Town object with below attributes is returned: Name: Sibu Longitude: 11158.58 Latitude: 21551.12 Roads:Conatins a pointer to a road object connecting Sibu and Selangau Conatins a pointer to a road object connecting Sibu and Daro
	public Town getDestination()	None	Town object with below attributes is returned: Name: Selangau Longitude: 1122755 Latitude: 22949.79 Roads:Contains a pointer to a road object connecting Selangau and Dalat

			Contains a pointer to a road object connecting Selangau and Sibu Contains a pointer to a road object connecting Selangau and Mukah
	private void calculateDistance()	None	Distance between origin and destination is calculated and *distance* attributes is updated Eg: distance=400;
	private int getID()	None	Value of *id attribute* is returned Eg: 1 is returned
Route	public Route (Town theOrigin, Town theDest)	Town object with below attribute values: Name: Sibu Longitude: 11158.58 Latitude: 21551.12 Roads:Conatins a pointer to a road object connecting Sibu and Selangau Conatins a pointer to a road object connecting Sibu and Daro Town object with below attribute values: Name: Mukah Longitude: 1120537.10 Latitude: 2543.32 Roads:Contains a pointer to a road object connecting Selangau and Dalat Contains a pointer to a road object connecting Selangau and Sibu Contains a pointer to a road object connecting Selangau and Mukah	A new Route object is created, with *origin* and *destination* attributes set to respective parameters
	public void calcTotalDistance()	None	Calculates total distance between origin and destination and updates

			totalDistance attribute Eg: totalDistance=700;
	private void addRoad()	None	Loops through all the road objects pointed to by *roads* attribute of *origin* For each road object, verifyRoad() method is called, which verifies the road, so that the route calculation is correct If a road object passes verifyRoad(), then it is added to *routeRoads* attribute of this class Eg: A road object connecting Mukah and Oya is added to *routeRoads* attribute
	public void setStatus(Boolean newStstus)	"open"	Eg: status="open" status attribute can have three possible value: "open", which means that destination has not yet been reached "close", which means destination has been reached and route is complete "deadEnd", which means that a dead end has been reached.
	public ArrayList<Route> findNextTown()	None	ArrayList<Route> containg route from Sibu to Daro and route from Sibu to Selangau is returned
	private void verifyRoad(Road theRoad)	Road connecting Sibu and Daro	If verifyRoad() returns true, road is added to *routeRoads* attribute by addRoad()

	public String getStatus()	None	"open", which means that destination has not yet been reached "close", which means destination has been reached and route is complete "deadEnd", which means that a dead end has been reached.
appGUI	public drawTowns(Map theMap)	Map object, created in RouteApp class	Draw the towns on the screen / on the GUI
	public drawRoads(Map theMap)	Map object, created in RouteApp class	Draw the roads on the screen / on the GUI
	public drawRoutes(RouteCalc routeCalc)	RouteCalc object, created in RouteAppp class	Draw all possible routes between origin and destination on the screen / on the GUI

Testing the entire system:

Sample Data to test the system:

Town Locations:

Town	Longitude (Deg Min Sec)	Latitude (Deg Min Sec)
Sibu	111 58 58	2 15 51.12
Selangau	112 27 55	2 29 49.79
Mukah	112 05 37.10	2 54 3.32
Oya	111 52 50.88	2 51 35.10
Dalat	111 56 20.12	2 44 35.79
Igan	111 42 39.24	2 49 27
Matu	111 31 52.31	2 40 40.79
Daro	111 25 32.87	2 31 1.12

Connections between towns:

	0 Sibu	1 Selangau	2 Mukah	3 Oya	4 Dalat	5 Igan	6 Matu	7 Daro
0 Sibu	0	y						y
1 Selangau	y	0	y		y			
2 Mukah		y	0	y				
3 Oya			y	0	y	y		
4 Dalat		y		y	0	y		
5 Igan				y	y	0	y	
6 Matu						y	0	y
7 Daro	y						y	0

Program output	User input	Sample Data	Expected Output
List of available towns is displayed	User selects origin and destination towns	User selects Sibu as origin and Mukah as destination	
Program displays all possible routes between Sibu and Mukah (Please note only 2 of possible routes are calculated manually in this case)			<u>Route 1:</u> Sibu to Selangau Selangau to Mukah Total distance: 500 km <u>Route 2:</u> Sibu to Daro Daro to Matu Matu to Igan Igan to Oya Oya to Mukah Total Distance: 700 km
Program highlights shortest route			<u>Route 1:</u> Sibu to Selangau Selangau to Mukah Total distance: 500 km