**The Use of Map**

**Objective**

The objective of this exercise is to practice on the use of map.

**Problem Description**

Every bus stop in Singapore is uniquely identified with a 5-digit number. For ease of description, it is also identified with a string describing its location, with the first part of this string being the name of a road and the second part a qualifier to make the string unique, and the 2 parts are separated by “**-.**” in the display. For example,

Adam Road**-.**Bef SICC (B41141) is the bus stop with bus stop number 41141 (B41141), at Adam Road (the first part being the name of the road) before Singapore Island Country Club (SICC) (with the second part “Bef SICC”, and “**-.**“ being the separator)

Adam Road-.Opp SICC (B41149) is the bus stop with bus stop number 41149, at Adam Road opposite Singapore Island Country Club (SICC)

Refer to <http://www.sbstransit.com.sg/iris3/busstopno.aspx> to find out the meaning of 1 and 9 and other choice of digit as the last digit in the bus stop number, and <http://www.streetdirectory.com/asia_travel/travel_sites/whats_nearby/cat/93/bus_stop/A/> on the bus stop name.

In order to facilitate the computation of bus fare as well as to locate the bus position with respect to the bus stops, each bus stop should have its GPS position. Assuming a GPS position is (x, y, z) used in a 3D coordinate system and they are double fields.

Given 2 bus stops with GPS positions (x1, y1, z1) and (x2, y2, z2) respectively, the distance between the 2 positions can be worked out as

In this exercise, you are required to design the class BusStop to include whatever attributes needed, the getter and setter functions and any utility (helper) functions needed, and a test driver to show the working of the class. You are also required to organize all BusStop objects created in a Map. **The test driver will create a number of BusStop objects, store them into a Map, and process a number of commands to manipulate these objects.**

**Input Data**

The test driver will expect some commands and data as follow:

The first line is the number of bus stops to be created, say n, followed by lines describing the bus stops.

For each bus stop, there are 5 lines:

road name

qualifier

bus stop number

GPS

Bus services stop at this bus stop (strings separated by blanks)

**Input Commands**

After the bus stop objects have been created and stored in a **map**, the program will print the full list of all bus stops in ascending order of the bus stop number. It will then proceed to act on the following commands:

getRoadName k

To display the road name of bus stop object k

getQualifier k

To display the qualifier of bus stop object k

getBusStop k

To display the bus stop object with bus stop number k in the format of

roadname.-qualifier(Bbusstopnumber) (x,y,z) [bus#1, bus#2, bus#3, …]

getGPS k

To display the GPS of bus stop object k

getBusStopAtThisStation k

To display all buses that stop at bus station k

distance k m

To display the distance between bus stop object k and m

You may assume that all input data are valid and therefore there is no need to verify the data.

**Output**

The output contains the result of command processing.

**Sample Input**

2

Adam Road

Bef SICC

41141

0 0 0

95B 2

Adam Road

Opp SICC

41149

0.5 0 0

95A 1

getRoadName 41141

getQualifier 41149

getGPS 41141

getBusStopAtThisStation 41141

distance 41141 41149

**Sample Output**

Adam Road-.Bef SICC(B41141)(0.0,0.0,0.0)[95B, 2]

Adam Road-.Opp SICC(B41149)(0.5,0.0,0.0)[95A, 1]

getRoadName 41141

Adam Road

getQualifier 41149

Opp SICC

getGPS 41141

(0.0,0.0,0.0)

getBusStopAtThisStation 41141

[95B, 2]

distance 41141 41149

0.5

**Note**:

1. To ease the reading of the output of this program, the result of processing each command must be prefixed by a line feed and the command itself as shown in the sample output.

2. Currently we use 3 doubles to represent GPS. The representation might be changed to the most appropriate format in future.

**Program Submission**

To submit the program, zip the 3 folders: solution, input and output into a file named ***Lab09g<LabGroupNo><MatricNo>.zip*** and upload it into the correct folder for this lab in the workbin of your lab group.