Computer Science 220 S2 (2019)

Assignment 4 (Graph Algorithms)

Due date: Oct 25th

Goals

In this assignment we want you to write a program that solves an applied problem dealing with shortest paths in edge-weighted graphs.

Ploblem Statement

Tamati needs to get back home to see his parents. He needs to travel from Auckland to Wellington, New Zealand as quickly as possible. Being a poor university student, he cannot afford to fly, so he plans to drive there. However, there are many routes he could take. Also, despite having a Smart Car, he cannot directly travel from one city to the next without stopping for fuel.

He has a list of cities and knows their latitude and longitude as well as how far he can travel on a single tank of petrol. But, like many university students, he has forgotten how to program! He needs your help to determine the shortest possible way to travel.

To simplify things, Tamati has decided to fill up his gas tank at every city he stops at. You may assume that if the distance between two cities is less or equal to his gas tank distance, then he can travel between those two cities. To shorten the trip he is going to off-road his Smart Car by taking the straight-line distance between any two cities. Tamati has already figured out how to determine this distance in kilometers taking into account the spherical nature of the planet (thanks Google!).

R = Earth's radius (6371 KM)
$\Delta lat = lat_2 - lat_1$
$\Delta lng = lng_2 - lng_1$
$a = \sin^2(\Delta lat/2) + \cos(lat_1) \times \cos(lat_2) \times \sin^2(\Delta lng/2)$
$c = 2 \times \arctan 2(\sqrt{a}, \sqrt{1-a})$
$d = R \times c$

NOTE: use double precision, and $\arccos(-1)$ for the value of π.

Input

Input begins with an integer n representing the number of test cases. Each test case begins with a number C representing the number of cities. Following this will be C lines, each line contains floating point values lat and lng, representing the latitude and longitude of the city location in decimal degrees, followed by the name of a city. To generalize for other trips, his travel always starts in the first city of the list and his destination is the last city listed. The last line of input will be the distance in kilometers that Tamati can travel with a single tank of petrol.

Output

For each test case, produce on a separate line a comma separated list of the names of the cities Tamati should travel through to get home as quickly as possible. If it's not possible for Tamati to make it home without running out of petrol, print "Not possible". You can assume there will not be any ties.

Sample Input	Sample Output
2 8 -36.8533 174.749 Auckland -37.788 175.265 Hamilton -37.7012 176.154785 Tauranga -38.677 176.0669 Taupo -39.926 175.045 Wanganui -39.4956 176.9 Napier -40.3465 175.6055 Palmerston North -41.2943 174.7595 Wellington 300 2 48.4275 -123.367259 Victoria 44.653 -63.588867 Halifax 1000	Auckland, Hamilton, Wanganui, Wellington Not possible

Submission and Due Date

Submit your source code to the automated marker `www.cs.auckland.ac.nz/automated-marker`.

The deadline is 11.30pm (automarker time) on the 25th of October. This assignment is worth 7.5% of your course grade. Some partial credit may be given if you do not acheive "Correct" (green) on the automarker. Note we use plagerism detection software so do not share any source code with your fellow students.