

Engineering Problem Solving II
Spring 2017
Homework 6
Due: Friday April 21 @ 5pm

Homework Description

You are an engineer at a wireless phone company that has just expanded its coverage. Given the new tower configuration, you are tasked with writing a program that determines the best tower to serve each customer's mobile device. Ideally the tower with the strongest signal should be assigned to serve each mobile device. However each tower can serve a limited number of mobile devices. Thus, if the tower that has the strongest signal for mobile device X is already serving at its full capacity, then mobile device X gets served by the tower with the next strongest signal that is not at full capacity.

For this assignment you are going to get the tower configuration from a file that contains the tower ID, 2D coordinates and the cell tower mean radius separated by one space. Each line except the first line describes a cell tower. The first line contains the number of towers in the file. The file has the following structure:

File name: towers.txt
First line: Number of towers in the file
All other lines: Tower data (one tower per line, attributes separated by space).
Tower attribute information:
1. Tower_ID
2. X_coordinate
3. Y_coordinate
4. Radius

The mobile devices list is given in another file, which contains their ID and the 2D coordinates. Assume that the mobile devices given in the phones.txt file connect to the strongest tower in the order given by the phones.txt file. The phones.txt file has the following structure:

File name: phones.txt
First line: Number of phones in the file
All other lines: Phone data (one phone per line, attributes separated by space).
Phone attribute information:
1. Phone_ID
2. X_coordinate
3. Y_coordinate

Example input files above can be found on ICON.

In both files the distance units are given in kilometers (km). Compute the estimated signal strength (in decibels relative to milliwatts) between a phone and a tower using the following estimation formula:

$$\text{Signal Strength} = -113.0 - 40.0 \log_{10}(r/R)$$

Where r is the distance of the mobile device to the tower, and R is the radius of the tower. You can compute the distance r between a mobile device and a tower using Pythagoras's theorem and the tower and mobile device coordinates.

All towers have the same service capacity (i.e. number of mobile devices it can serve). However it can be changed when upgrades are made. Thus your program should take the tower capacity number as an input from the user.

You should output to the screen a mapping of each mobile device ID to the tower ID that serves it. Also include the signal strength, the distance to the tower, and whether or not this is the closest tower to the device.

Here is an example of the output to be printed to the screen using the inputs found on ICON and a tower capacity of 2:

Tower capacity: 2

Phone_ID	Tower_ID	Distance	Signal_strength	Closest_tower
0	1	4.24264	-74.0231	no
1	1	4.24264	-74.0231	no
2	0	1.41421	-79.0206	yes
3	0	1.41421	-79.0206	yes
4	2	7.07107	-106.979	no
5	2	7.07107	-106.979	no
6	3	9.89949	-112.825	no
7	3	9.89949	-112.825	no

For grading purposes, your program will be tested with a set of input files other than the one provided to you.

You MAY assume that there will only be a maximum of 1000 towers and a maximum of 1000 phones.

GRADING Rubric (100 points total):

You will receive a **zero** on your homework if your program does not compile on an ECS Linux machine. Remember: late homework is not accepted.

75 points for working code:

- 5 points reading the user's tower capacity number
- 10 points for correctly reading the input files into arrays
- 15 points for correctly calculating the distance and identifying the closest tower to each mobile device
- 15 points for correctly calculating the signal strength and identifying the tower with the strongest signal for each mobile device
- 15 Points for correctly identifying the best tower to serve each mobile device given the power capacity constraints.
- 15 Points for correct output.

25 points for style (See the Style guide under Content on ICON):

- 10 points for comment blocks for each function including main
- 4 points for indenting
- 7 points for sufficient in-line comments
- 4 points for meaningful variable names

NOTE: Submit a *.cpp file to the dropbox. Late homework will not be accepted. Contact the grader for possible regrade if your program received a zero for code that does not compile.

DO NOT WORK TOGETHER! Students caught working together on this assignment will **drop a whole letter grade** for the course!