HW3: Geospatial data

Total points: 6

In this homework, you are going to work with **spatial data** - you will create (generate/sample) some data, visualize it (in SIX different ways - OpenLayers, ArcGIS, Google Earth, an 'appspot' app, QGIS, and raw JavaScript+HTML). Hope you have fun with this!

The exercise will give you a taste of working with spatial data, use of a spatial file format and spatial-related software/code, all of which are quite useful from a real-world (or job interview) perspective.

You are going to be visualizing, over a map of the USC campus, the following: a collection of locations [2 ways, 2 points], a convex hull polygon (1 point), straight lines (1 point), a pretty curve [2 ways, 2 points].

Ready? Let's start!

You need to collect a set of location data [a (long,lat) pair for each location] - library locations (https://libraries.usc.edu/libraries-overview) on UPC (but if you want, pick one on HSC!). Ideally you'd do this by walking around (and using your phone's GPS) to get the (long,lat) values, **but on account of #StayAtHome, you would do this virtually, ie. using Google Maps (https://www.google.com/maps)** - if you long-click on any location (with your left-mouse-button, press for about 1 sec, release), you can see the coordinates for the location (lat,long), eg:

0:00 / 0:14

Record these to a text file (or just write them down!), along with the library name (shorten it to 15 characters). Collect the locations of 12 libraries. Next, grab Tommy Trojan's location, make a note.

Q1 [1 point]: Now that you have 13 locations in your little 'database', you are going to visualize them on a map. Take a look at this (OL.html) web page, where a single (non-USC :)) location is shown. Save the .html page to your hw area - **RIGHT-MOUSE-CLICK on the link (to OL.html) above, that will let you save it on to your machine**. Make sure you save it correctly - the html you save, should look like this (pics/html.png). Edit your downloaded copy of OL.html with a text editor - you will see, at the very bottom where I've added the one point. Put in the 13 points you collected, remove mine, save, view in your browser - voila, your locations on a map! Take a screenshot.

Q2 [1 point]: Next, you are going to visualize the location data, using ArcGIS (https://www.arcgis.com/home/index.html) this time - please register for a free 'public' account. ArcGIS uses a format called .shp (https://doc.arcgis.com/en/arcgis-online/reference/shapefiles.htm), to store location data. We are going to place our location data in a .kml file first, then use an online converter to convert the data to .shp, download the .shp, and open it in ArcGIS. Here (starter.kml) is a starter .kml file - download it, edit it, put in your 13 locations in place of the ones already there. Upload your .kml to https://mygeodata.cloud/converter/kml-to-shp (https://mygeodata.cloud/converter/kml-to-shp), have it be converted to .shp, and download it (the result might be called .zip, that's ok, it will work; you can unzip it if you like, to see how your data is represented). Go to https://www.arcgis.com/home/webmap/viewer.html (https://www.arcgis.com/home/webmap/viewer.html), upload your .shp or .zip (hint: look under 'Add'), to again see your locations over the campus map :) Take a screenshot.

If the above 'mygeodata' converter didn't work well for you, you can try using http://zonums.com/online/kml2shp.php (http://zonums.com/online/kml2shp.php) instead, or, use QGIS to convert (you'll need QGIS for Q5 as well).

Q3 [1 point]: Look at the 12 library locations you picked, and visually create a convex hull (https://www.google.com/search?q=convex+hull&rlz=1C1CHBF_enUS723US723&sxsrf=ALeKk02isOdM6mxnNxRGoQ728QP82wX9YA:1592601627 for the points - it is simply a convex polygon, made up of a subset of your locations, which encloses the remaining locations. You do NOT need to use any software for this, just look at your locations distribution and visually determine the hull! Make a copy of your starter .kml, call it hull.kml, and in it, specify the convex hull polygon data. You would specify your polygon as a 'Placemark', as documented here (https://developers.google.com/kml/documentation/kml_tut#polygons) [you would skip the 'innerBoundaryIs' section]. Download Google Earth (https://www.google.com/earth/download/ge/agree.html) on your laptop, install it, bring it up. Load your .kml file into it - that should show your sampled locations and your convex hull, on Google Earth's globe :) Take a snapshot of this. **Note: don't leave a gap between long,lat! In other words, -118,38 for ex is ok, but -118, 38 is not - if you insert a space, your polygon won't be displayed (it's an Earth parser bug).**

Q4 [1 point]: Visually estimate the 4 closest libraries to Tommy. Draw a line between Tommy and each library - do this in another copy of your starter KML, call this lines.kml. Copy and paste your .kml into http://display-kml.appspot.com/ (http://display-kml.appspot.com/) to see your connecting lines (along with all the 13 locations). Take a screenshot.

Q5 [1+1 points]: You are going to need QGIS (https://qgis.org/en/site/forusers/download.html) for this question. Look at this (spiro.html) page, it shows a pretty Spirograph curve and underneath, the (x,y) values for the curve :) Download the page, edit it, look at my prtt() call that outputs the (x,y) points - change it so that it outputs the curve as (long,lat) points, centered at Tommy Trojan :) Copy and paste the coords array result into this (spiro.kml) KML file container. Bring the .kml up in QGIS (hint: look in Layer->Add Layer) - voila, your Spiro curve! Take a screenshot. Note that just your curve is displayed, not the map underneath. View this .kml in Google Earth, to see the curve centered at Tommy - take a screenshot. Also, in order to center the Spirograph at a given location [TT, Bovard or other], you need to ADD each (x,y) curve point to the (long,lat) of the centering location - that will give you valid Spiro-based spatial coords for use in your .kml file.

Note (fyi) - for the Spirograph curve point creation, we use the following parametric equations (with R=5, r=1, a=4):

```
x(t) = (R+r)*cos((r/R)*t) - a*cos((1+r/R)*t)
y(t) = (R+r)*sin((r/R)*t) - a*sin((1+r/R)*t)
```

Instead of JavaScript (which I'm using in the spiro.html page above), you can use any other coding language you want, to generate (and visualize) the curve's coords: C/C++, Java, Python, SQL (https://docs.oracle.com/cd/B28359_01/server.111/b28285/sqlqr02.htm), MATLAB, Scala, Haskell, Ruby, R.. You can also use Excel, SAS, SPSS, JMP etc., for computing [and plotting, if you want to check the results visually] the Spirograph curve points.

PS: Here (https://www.google.com/search?q=Spirograph+curve&ie=utf-8&oe=utf-8) is MUCH more on Spirograph (hypocycloid and epicycloid) curves if you're curious. Also, for fun, try changing any of R, r, a in the code for the equations above [you don't need to submit the results]! And, for extra fun, do this (pics/101computing.png) :)

OK, that was a bunch of geospatial data visualizing - hope you enjoyed it!

What to submit:

Q1: browser screenshot, .html

Q2: ArcGIS screenshot, .kml, .shp/.zip

Q3: Google Earth ("Earth") screenshot, .kml

Q4: appspot page screenshot, .kml

Q5: QGIS screenshot, .kml, Earth screenshot

After completing this course, you can consider doing the following, to enhance your understanding: use PostGIS to store your location data, and write SQL queries (which would call spatial functions) to have the convex hull and the nearest neighbors be output - these of course should be identical to what you'd determined 'by hand'.
