

General guidelines and starting KML File (1 point):

- It should have 13 different GPS locations (1 should be home location and rest can be anywhere)
 - Should have a convex hull mapped (The convex hull should contain all 13 points and a few should be on the boundary). Incorrect **deduct 0.25**
 - Should have 4 nearest neighbors from their home location or they could have chosen any location. But there should be a starting point and for which they should have found 4 nearest neighbors. If not done accordingly **deduct 0.25**
 - If the convex hull and nearest neighbors are split into two separate files, it's fine. -
- If either the convex hull or 4 nearest neighbors are missing, **deduct 0.5** points each.
- If there aren't 13 locations, **deduct 0.25** for each missing location.

Selfie pics

No points. But deduct for following reasons

Check if the locations map to the locations given in the KML file

Note: The students can take selfies with half covered faces. It's to prove that they have gone to the location and taken the selfie.

- **deduct 2** if 0-4 selfies are submitted.
- **deduct 1** if 5-8 selfies are submitted.

SQL part (2 points):

A text file (.txt or .sql) with your two queries should be uploaded (a file for each query is okay too).

[Saty] Some students might have used sqlite, Oracle, etc. Please don't deduct points on account of that [Postgres doesn't need to be the only choice!]

compute the convex hull (1 points)

- You basically can call ST_CONVEXHULL (in postgres) to generate a convex hull for points.
- -0.5 point If only table creation and data insertion commands are present without the rest of the query

- -1 if convex hull generated is not convex (i.e. there are points outside the boundary)
- If you are not sure about the correctness of queries, please run them

Nearest Neighbors (1 points)

- You basically can call ST_Distance (in postgres) to order points based on their distance.
- -0.5 point If table creation and data insertion commands are present without the rest of query
- -0.25 If the result is not limited to the 4 nearest neighbors (e.g. if LIMIT 4 is not present)
- -1 if nearest neighbors are not sorted (i.e. random points are reported)
- If you are not sure about the correctness of queries, please run them

Please note that:

- it is okay to hardcode points, in the above queries!
- It is okay to create and use a table to store your 13 points in it, then write queries against the table.
- It is okay to report the entire table row, name, ID, coordinates or other unique attributes as the results for nearest neighbors.

Sample Queries:

Create table

```
CREATE TABLE MyNeighborhood (name VARCHAR(1000), gps GEOMETRY);
```

Insert points

```
INSERT INTO MyNeighborhood VALUES
```

```
('Home',ST_GeomFromText('POINT(-118.291565 34.035194)')),
```

```
...
```

```
('LOCATION13',ST_GeomFromText('POINT(-118.243432, 34.2324)')),
```

Calculate convex hull

```
CREATE TABLE ConvexHull AS (SELECT
```

```
ST_CONVEXHULL(ST_MULTI(ST_COLLECT(gps))) Hull from MyNeighborhood);
```

```
SELECT ST_ASTEXT(Hull) Hull FROM ConvexHull;
```

Calculate 4 nearest neighbors

```
SELECT nn.name, ST_ASTEXT(local.gps) as Home, ST_ASTEXT(nn.gps) as Neighbor
```

```
FROM MyNeighborhood local, MyNeighborhood nn
```

```
WHERE local.name='Home' and nn.name <> 'Home'
```

```
ORDER BY ST_Distance(local.gps, nn.gps) ASC LIMIT 4;
```

Screenshots

[Saty] A few students might have used Google Maps instead of GE - that's ok, too.

Google Earth Screenshot of sampled locations - 0.5 point

-deduct 0.25 points if there are lesser than 13 locations, and more than 6 locations

0 points if there are lesser than 6 locations

Google Earth Screenshot of Convex Hull and Nearest Neighbors - 0.5 point

Convex Hull - 0.25 point

Nearest Neighbors (All 4 lines visible) - 0.25 point

0 point if Convex Hull is not Convex - some points outside boundary

0 point if nearest neighbors (lines) are not visible in the screenshot

0 point if nearest neighbors are incorrect (random points chosen, no sorting done)

Html file or CodePen/jsFiddle link

- should be able to view 13 sampled points

- deduct 0.25 point for missing points

- deduct 0.5 points for not using localStorage (if the points directly get plotted, with being stored and retrieved).

Sample Code only JS part:

```
// your JS code
console.log("Hola, all!");
alert("My DB Homework!");
var d = [{"k1":[{"a":34.018725, "b":-118.286611},
{"a":34.021743, "b":-118.282786},
{"a":34.020057, "b":-118.283693},
{"a":34.019211, "b":-118.28555},
{"a":34.021276, "b":-118.283993},
{"a":34.01876, "b":-118.284367},
```

```

{"a":34.020069, "b":-118.289994},
{"a":34.022894, "b":-118.287091},
{"a":34.020159, "b":-118.283309},
{"a":34.01872, "b":-118.282454},
{"a":34.019516, "b":-118.289478},
{"a":34.021963, "b":-118.284722},
{"a":34.022128, "b":-118.286126},,]);
localStorage.setItem("myData",JSON.stringify(d));
// you'd need to modify the above, to store all your points
var dataStored = JSON.parse(localStorage.getItem("myData"));
// verify that we fetched our data correctly
console.log(dataStored);
// we can iterate through our array [of points], like so:
var a = dataStored.k1; // 'a' will contain our array
initMap();
for(var indx in a) {
  addMarker(a[indx].a,a[indx].b);
}

```

Spirograph Part (1 Point): Question 7

[Saty] Some students might have used an alternative site to visualize the .kml - that's ok.

- If missing any file reduce or if the contents are wrong (deduct .25 point each)
- Spirograph point generation code (can be in Python, Java, JS, Excel...)

Starter code in JavaScript is given. You might want to run it once and check if you doubt the code (unless they have written it in a weird language that you can't compile and run). You can use <https://www.codechef.com/ide> (or other online IDEs) to run it and see if you don't have a particular compiler or interpreter.
- A .kml file (pdf is also acceptable) [If the spirograph image is wrong then it is because of KML file, check the KML file and spirograph point generation file. If both are wrong deduct .25 for each]
- Shapefile (this is a zip) [You don't need to unzip. If zip file is present give the allotted marks]
- Screenshot with Spirograph visualized using ArcGIS Online.

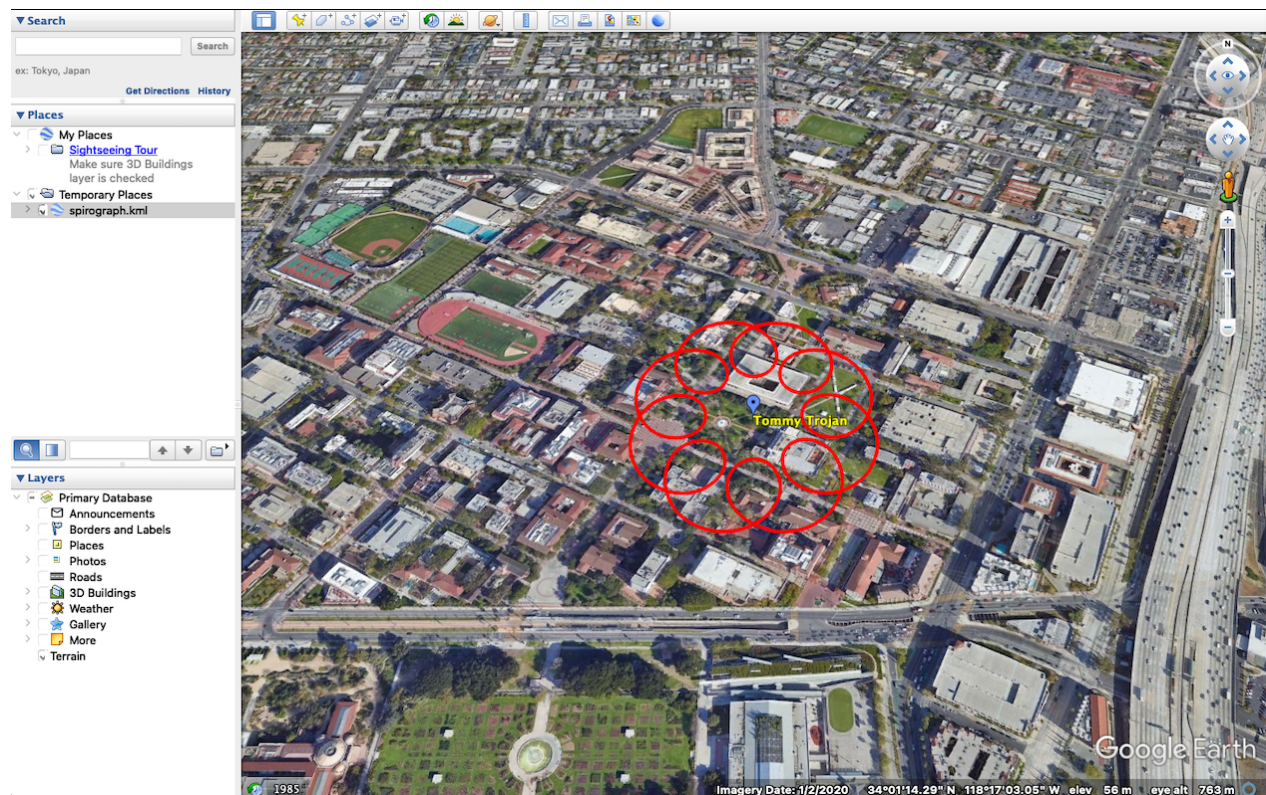
Sample Python Code Generation:

```
R=5
r=1
a=4
x0 = R+r-a, y0 = 0
#ctx.moveTo(150+10*x0,150+10*y0)
f = open("spiro.kml","w")
pi = math.pi
nRev = 10
t = 0.0
while t < (pi*nRev):
    x = (R+r)*math.cos((r/R)*t) - a*math.cos(((1+r/R)*t));
    y = (R+r)*math.sin((r/R)*t) - a*math.sin((1+r/R)*t);
    f.write("\n")
    f.write("          " +
str((y/10000)+-118.285558)+", "+str((x/10000)+(34.020844))+", "+str(30))
    t += 0.01
```

Sample KML file without co-ords:

```
1  <?xml version="1.0" encoding="UTF-8"?>
2  <kml xmlns="http://www.opengis.net/kml/2.2"
3  <Document>
4      <Style id="line">
5          <LineStyle>
6              <width>5</width>
7              <color>#ff0000ff</color>
8          </LineStyle>
9      </Style>
10     <Placemark> <!--Not Mandatory-->
11         <name>Bovard Location</name>
12         <Point> /Users
13             <coordinates>
14                 -118.285558,34.020844
15             </coordinates>
16         </Point>
17     </Placemark>
18     <Placemark>
19         <name>Spiral</name>
20         <styleUrl>#line</styleUrl>
21         <LineString>
22 >             <coordinates>--
3165         </coordinates>
3166         </LineString>
3167     </Placemark>
3168 </Document>
3169 </kml>
```

Sample Spirograph: [The center should be **Tommy Trojan**] Do not deduct points if there is no point near Tommy



Allowed corner cases:

If students haven't played with the r , R , and other params, it would go into the Pacific ocean.

If the curve in KML is above the ground by some distance.