**Data Science – 720** **Project 1** **GPS Data Visualization and Convex Optimization**

You are provided with a set of GPS data tracks in which Dr. K drove from home to RIT. Given the GPS data please achieve the following goals. Ultimately, we want one KML file with the best track from the starting location (home) to a parked position at RIT, with all of the left turns and stops marked on it. Report the track of the tracks provided. Do not worry about finding the way points and pulling each track into different trip segments.

1. Write a program to convert the GPS data into KML data that can be viewed in Google Earth. (60%)
   During the process find a way to clean the data by filtering out and ignoring any redundant or erroneous data.
   For example, if the vehicle is parked, you do not need multiple data points at that same location.

2. Locate in the data any left-hand turns. (10%)
   That is, turns crossing across traffic.
   You must define what a left hand turn is, and then locate it in the data.
   At a minimum, it should be at least a 25 degree turn within at-most a hundred feet, or one minute.
   You may use any data in the RMC or GGA sentences of the GPS file.
   Do not use any debugging lines that Dr. Kinsman put in the GPS file if they are present. (lng, lat… in the file.)
   Label the left turns, at each long and lat, with a yellow pin in KML, or (for 5% instead of 10% points)
   print the (long,lat) points out sorted from highest to lowest longitude.

   ```
   Example default yellow Placemark:
   <Placemark>
      <description>Default Pin is Yellow</description>
      <Point>
        <coordinates>-77.6805,43.0867,0.0</coordinates>
      </Point>
   </Placemark>
   ```

3. Locate in the data any stops for stop signs or stoplights. (10%)
   Again, you must define what a stop is. Sometimes a car only slows down for a stop sign, and rolls through it.
   There is still a stop sign there. What attributes would indicate this?

   A stop might only occur in one of the tracks, so you might need to collect them in a separate file or cache them, and then add them back on to the final track displayed.

   Put a red placemark at each stop. Or, print out one set of (lat,long) per stop (for 5% instead of 10%)
   Again, sorted largest long to lowest long.

   ```
   Example Red Placemark:
   # The Hex values are the colors: Alpha, Blue, Green, Red.  Delete this line.
   <Placemark>
      <description>Red PIN for A Stop</description>
      <Style id="normalPlacemark">
        <IconStyle>
          <color>ff0000ff</color>
          <Icon>
            <href>http://maps.google.com/mapfiles/kml/paddle/1.png</href>
          </Icon>
        </IconStyle>
      </Style>
      <Point>
          <coordinates>-77.6800,43.0867,0.0</coordinates>
      </Point>
   </Placemark>
   ```

4. Finding the optimum track. Of the routes provided, determine the best route to use. (10%)
   This route includes the travel time from the starting location to the final location.
   In order to define "best" we need a formal definition. It is neither just the fastest, nor just the shortest.

   The cost function to evaluate the routes is
   Cost = (Travel Time (mins) / 30 (mins) ) + ½ (Maximum Velocity (mph) / 60 (mph) ).

   Notice how this is designed:
   a. The total cost function is composed of two parts: the objective function and the regularization.

   b. The objective function is what we really want to minimize – the travel time to work.

   c. The regularization is something that would be *nice* to minimize, but is not strictly required.
      The regularization helps break ties if two different trips take the same amount of time.

      Notice also that the regularization is less then the objective function to keep the regularization part from dominating the objective function. That way you cannot pick the slowest trip in by default.

   d. The usual travel times is 30 minutes, so the travel time is normalized by 30.
      The usual maximum velocity is about 60 MPH, so the max velocity is normalized by 60.
      This way the two normalized components of the objective function and the regularization are *commensurate* – or *reasonably comparable.*

      Then the factor of ½ assures that the regularization does not dominate the objective function.

      You might convert MPH to knots to do the math in knots instead of MPH, for easier computing.

   e. The final computed cost has no units on it.

5. Write-up your results, with the following information in it. (10%)
   a. Background section (three or four paragraphs), with *references.*
      Discuss the background of GPS: When did GPS come about? How many satellites are there? What other relevant details should we know about GPS? What else did you find out that is interesting about GPS?

   b. Describe how you wrote your program. What program pattern did you use?
      Did you hold all the information in memory at once? Did you parse all of the files at once?

   c. How did you define a left-hand turn, and then detect it in the data? What problems did you find with the approach? Did you have to do any noise removal or signal processing?

   d. How did you define a stop, and then detect it in the data? What problems did you need to overcome?

   e. Describe what the "best" trip to work was. Which file?

   f. A screen capture of image of the best track or trip to work, with annotations if you did them.

   g. A summary conclusion of what you learned overall, and how this might be useful for some commercial application.

6. **Turn into the group dropbox:** your well commented program, your write-up in PDF format, and your resulting kml file.