Anna Kurchenko and Lindsay Cagarli
Project B Writeup
CSCI 420

Assignment Walkthrough and Explanations:

Data parsing + cleaning :
We organized the data as a list of parsed fields for each instance of GPS data. Each element of the list contains a dictionary of the important fields we collected from the NMEA GPS lines.
The NMEA information comes in with 3 different location descriptors per location instance in the format GPRMC, GPGGA and lng, where GPRMC contains most of the relevant fields (lat, long, directions, etc), GPGGA contains the altitude, and lng contains a summary of the data. We chose to ignore the lng line and instead go through the data keeping only those data instances for which both GPGGA and GPRMC entries existed.
We chose not to extract information from lng to fill in missing fields of either GPGGA or GPRMC because it is ambiguous for us to know if the data was omitted from error and therefore if the lng line is credible. Therefore if either the GPGGA or GPRMC specs are missing for a given data entry we are ignoring this data entry.
Following this we parsed through the file looking for GPRMC and GPGGA lines specifically. The order of these was also arbitrary so we accounted for all combinations as well as Arduino errors where new line characters were eaten and both GPS specs were on the same line. This was carefully distinguished to make sure we weren't overflowing into the next dat entries' specs.

Q. 3-6 What is 'near-by'?
We defined that near-ness should be within a certain meter range of our target location, but the range depends on the location. For Dr. Kinsman's house, we defined 'near' should be within 2 blocks, which is ~ 280m away. For the RIT campus, we wanted to capture the entire campus + a 2-block radius around it. We took the center of campus, 90 Lomb Memorial Drive and using Google Earth determined this nearness should be within ~930 m our center.
Based on the questions and start/end locations we played with this near-ness range and Google Earth. For instance 'near' is more flexible than the definite question 'start at', which we calculated with a tighter bound.

Q.7 Check if a full trip occurred
To be considered a full trip, the start and end need to have a valid location lock which is checked using the GPRMC status field where "A" is valid and "V" is invalid. We set the "sudden jump" threshold to 35m. Mathematically, if a car is going 80mph it can travel 36m in 1 second. Knowing that on average an Arduino reports data points under 1 second, and that the area of our GPS data is constrained to Henrietta, it is extremely improbable that such a jump in data points is possible. Next we iterate through consecutive data points checking for distances which exceed this threshold, if found the function returns early and concludes that the trip is not a 'full' trip.

Q.8 Trip Duration

We iterate through the parsed GPS data checking for movement above the minimum speed threshold. This marks the start of the trip. We follow the same process for determining the end point in reverse.

After finding the start and end points we find the difference in time, ensuring we exclude any additional stationary data points around the trip.

Q.9 How many stops

We define a stop to be a data point with a speed of less than or equal to 5 mph or where the vehicle is stationary between 30 seconds and 5 minutes. We decided to use 5 minutes as the upper threshold since most trips use local roads rather than highways. Local roads have less traffic and are very unlikely to have standstill traffic, thus long stops are unlikely. Stops longer than 5 minutes could occur if there was an accident or road work. However, these are special cases which we concluded are unlikely to be present in Dr. Kinsman's data.

Q.10-11 Duration travelled uphill

We solved this by making use of the total trip duration from the previous question and used this as a basis from which to compute the uphill percentage. We computed the uphill time by analyzing the altitude changes in the trip data. For each consecutive pair of points, we calculate the change in altitude and the horizontal distance using the haversine distance. If there was distance traveled (dist > 0), and the altitude angle was above 15m (the given threshold), we accumulate the time traveled between these two points to our tracked uphill time. Once the total uphill time is determined we find the ratio to the total trip time and report it as a percent.

Q.12 Meters climbed uphill - per hill

We tracked each hill by the distance traveled that also had an altitude > 15, which had no flat sections longer than 50m. This threshold was chosen by examining the data and some of the different altitude variations, in combination with researching the average hill variations in Rochester.

The code processes each data point by calculating altitude change and horizontal distance between consecutive points also using the Haversine formula. If the altitude increases, the distance is added to the current hill climb. If the altitude decreases, the code checks whether the section is flat for more than 50m and ends the current hill if necessary. We also made sure to keep a flat distance accumulator to account for flat sections that spanned multiple points, and checked against this to see if a hill has met its 50m flat section limit. The function also ensures that any ongoing hill climb is added at the end of the trip.

We gather the distance climbed per hill as well as the total distance climbed over the hills and report this.

Q.13 Measure of Impurity

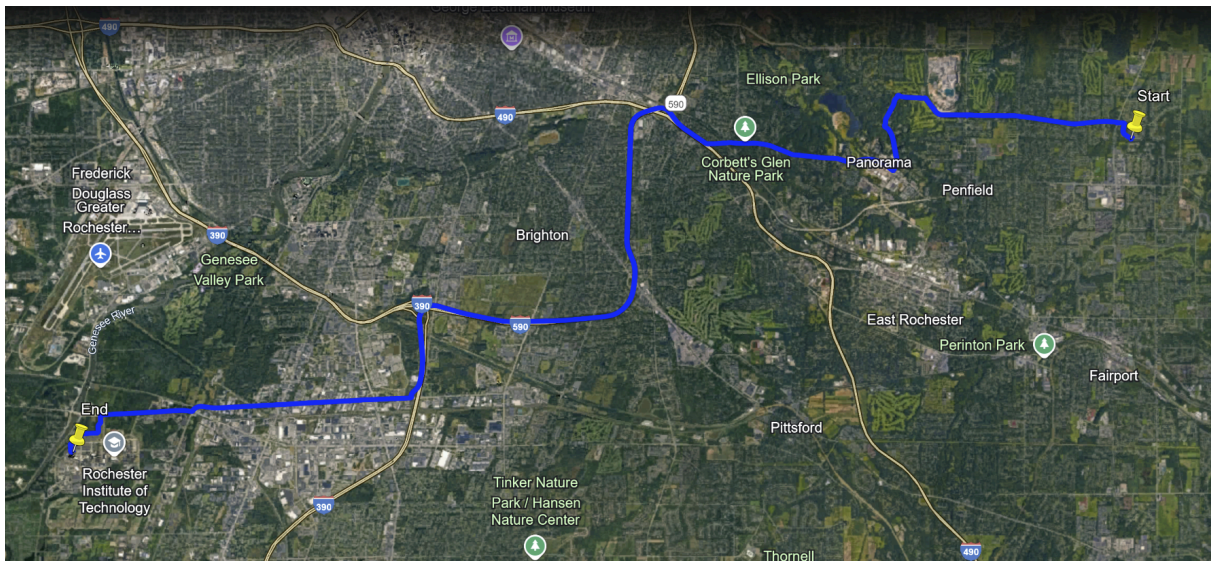
We chose to determine how often the brakes were 'slammed' for the task of measuring an impurity and providing Dr. Kinsman with a driving recommendation.

Brake-slamming is a metric that should be minimized since hitting the brakes suddenly degrades the brakes and is also a characteristic of a bad/unsafe driver.
To calculate this metric, we found all the data points that decelerated from one another by examining the points' speeds. We then found the deceleration rate with a speed decrease/distance calculation. If the rate was over 0.5 m/s we considered it to be a brake 'slam'. This threshold was chosen by trial and error after running our data and researching common rates online. Generally the threshold is set higher between 3 and 5 m/s to be considered a slam, but there were no instances of brake slamming captured with these higher rates. We therefore increased the sensitivity of our threshold to capture more data points of brake slamming. However even with a 0.5m/s threshold there were very few slams across all the trips. We can therefore conclude that Dr. Kinsman is a cautious driver who feathers his brakes well, and doesn't have much room for improvement in this area.

Q.14 Best time to leave to get to RIT

We defined the best time of day for the professor to leave his house by finding the fastest trip that delivered him from home to RIT. We first excluded the files which did not have the correct start and end positions. This left us with 6 files with trip durations:
28:10, 34:04, 28:04, 28:51, 27:49, 29:18. File '2024_09_12__143509_gps_file.txt' had the best time of 27:49.

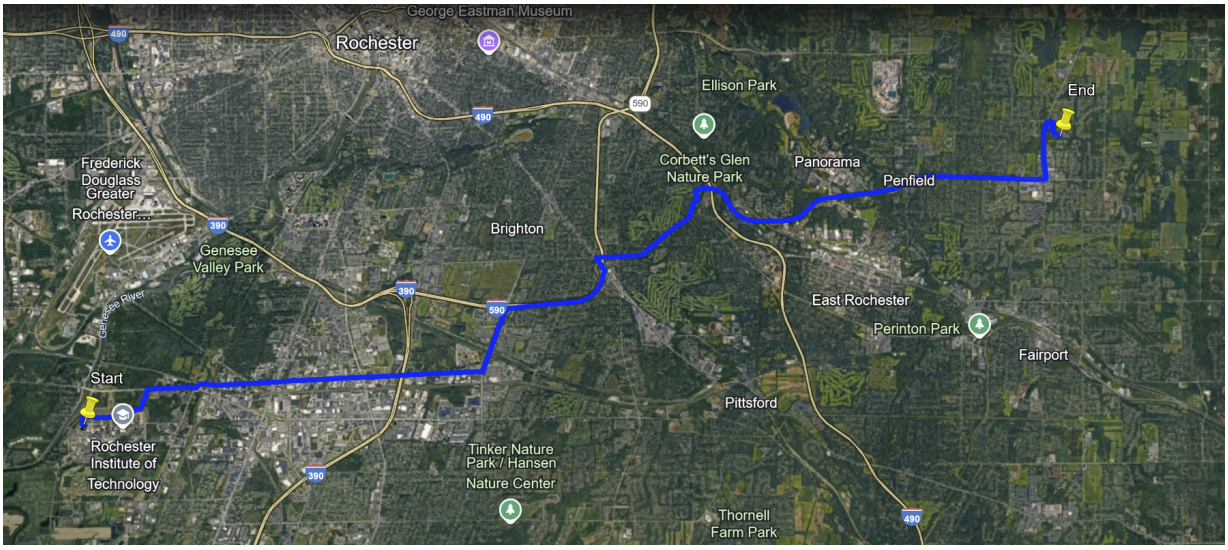
This trip began at 14:32. This is right in the middle of the workday after lunch, one of the least busiest times of day to commute, so it makes sense that this start time generates the fastest trip. If Dr. Kinsman wants to get to RIT while driving the least amount of time and probably being most fuel efficient, he should leave after lunch time and before the 5pm rush.



Q.15 Best time to leave to go home

We similarly defined the 'best' for this question as the fastest trip. Once again we excluded the files with incorrect start and end locations, leaving us with 6 files with durations: 29:25, 30:36,

27:57, 30:52, 37:24, 28:38. Of these the best file was ‘2024_09_09__232616_gps_file.txt with a trip duration of 27:57 minutes. This trip started at 23:24, 11:30 pm. This time makes sense as this is well after the after work commuter rush and when most people are off the roads. If Dr. Kinsman wants to get the best mileage on his way home from work, he should leave right around midnight.



Conclusion:
Through this project we learned how to read, analyze, parse, and work with GPS files in the NMEA format. This is a useful skill as mentioned in class for professional assignments, but also interesting since GPS is something so ubiquitous and taken for granted. Few people (us included) have wondered about how GPS data is actually tracked beyond a high level overview and knowing it works with satellites that track latitude/longitude. This turned out to be an insightful and interesting task, and it was really cool to visualize data using the KML files especially with the real world and highly applicable examples of commuting around Henrietta.

Answering the meaningful questions of ‘what does it look like to slam on your breaks’ or ‘how many hills were climbed’ additionally made the assignment more personal, forced us to connect with the data, confront ambiguity by defining our own assumptions (thresholds, concepts such as nearness/best), and apply transformational thinking that connected the dots between the unintelligible raw data we started with and tangible daily actions.

<u>Implementation Details</u>
We ended up using the Haversine distance as we wanted to practice applying it, as it wasn’t really used for previous assignments. Also, we knew it would make our measurements more precise since it factors in the Earth’s curve.

In regards to data cleaning we skipped over data entries which were invalid based on the following criteria:

- If the GMPRC status listed as ‘invalid’
- If the date/time object was erroneous either missing characters or containing unsupported characters
- If either the GPRMC or GPGGA lines were missing for one data point, and if there are empty lines.

We also cleaned and included data lines that had eaten new line characters which placed either GPRMC or GPGGA lines on the same line.
It is important to note that because some data entries were skipped in the cleaning phase some of our calculations could be inaccurate. For instance when measuring flat distances on a hill a missing data point can cause our hill decider to cut the hill off early or under-measure the flat distance and join two hills. Similarly, time accumulation can be cut short.

Rush hours were found to be at typical times in the morning and late afternoon to early evening. This is consistent with standard rush hour times as people make their daily commutes to and from work. These numbers were found by examining the longest trips taken to and from RIT, similarly to how we approached questions 14-15.

Figuring out when the train goes through nearby RIT:
This can be achieved by firstly finding the location of the track intersection with the professor’s path (Lat: 43.0935153 Long: -77.6527006). Then we can write a function which captures the time spent within a nearby radius - say 100 meters of the intersection, which is near enough to sit at a red light. We would additionally collect the time at which the vehicle is found to be waiting at the intersection. By running each GPS file through this program we could determine which trips had a long duration stuck at the intersection, and cross reference this with the time of the stop. This generates a list of times when the professor stumbled across a train blocking his path, and provides a timetable for when the train passes near RIT.

Can you figure out when the Professor needed to stop for gas?
No. There are too many factors to consider to determine gas mileage or when the professor is likely to need to re-fuel. We know nothing about the make or model of his car, so we would need to go through all the data and check if he stopped at a gas station at any point. If there are multiple gas station stops we might be able to infer the range of miles he can travel between re-fuels, but this hinges on A) how many gas station points there are in the data, B) assuming that the car was not in use at all outside of the data we have. Since we cannot know these variables we cannot make a prediction for when the professor needs to refuel.

Ethical Issues
There are multiple ethical issues involved with this project. Dr. Kinsman is being very trusting to give out his personal information such as home address, driving routines, destinations and arrival times. This is all information that could be exploited and used to maliciously target Dr. Kinsman. Additionally this data could be sold to online marketers inflicting marketing schemes on Dr. K.

Do you think this will help you get a job?

We do believe that deeply interacting with GPS data as we did in this project will help us secure a career. As mentioned in class, this seems to be a hot topic for recruiters as college students don't commonly practice with GPS data even though it is common across the industry. Additionally this was a great example of forcing us to data mine and exhibit many of the learning outcomes stipulated in this course. Personally we are proud of this project and will be featuring it on our resumes/githubs when applying to jobs.