Jackson Braunschweig
Rhishabh Hattarki
SER 494
4/17/23

a) Classes:

CANFrame.java - Parent abstract class for different kinds of CANFrame. This holds the common frame information that is used by both single value and tri value frames.

CANFramesInfo.java - Holds all the information required to parse the data in the CAN frames.

CANFrameSingleVal.java - CANFrame that holds data for a single frame value.

CANFrameTriVal.java - CANFrame that holds data for a triple frame value.

CANSimulation.java - Holds the main method used to perform the simulation. Simulation includes:

1. Parsing the CAN trace file.
2. Parsing the GPS trace file.
3. Simulating real time data sensing.

CANTrace.java - Class consists of a collection of CANFrames, and also has methods to interact with them.

CANTraceParser.java - Parser that reads CAN bus data from a given file and parses into CAN Trace.

DataFieldLocation.java - class to store location of value in hex data.

FrameVal.java - class to store frame's value and unit data.

GPSTrace.java - Class consists of a collection of GPSCoordinates, also has APIs to interact with them.

GPSParser.java - Parser that reads GPS data from given file and parses into GPS Trace.

GPSCoordinate.java - class to store GPS coordinate data.

Identifier.java - Class that stores identifier constants stored in one place to avoid confusion and errors

Range.java - Class to store any kind of range, from one double value to some other

SensorDataReceiver.java - Class that receives of sensor data, prints data in the right format once received.

SensorInfo.java. - Class used to store sensor info/description that is used for parsing CAN bus data

b) Jackson Braunschweig - Worked on everything related to part 1 of the assignment this meant creating the GPS trace/parser/coordinate classes and their functionality. Also created the code that later became the parseGPSData method in the CANSimulation class.

Rhishabh Hattarki - worked on everything related to part 2 of the assignment which involved extending the CANSimulation class, reformatting how the data is printed, and creating the SensorDataReciever class and its functionality.

c)

```
Press enter to start simulation

          Current Time |   Vehicle Speed |    Steer Angle |          Yaw Rate |         Lat Accel |        Long Accel |                 GPS Lat/Long
          3718.4729 ms |       75.2 km/h |         4.0 deg |     -1.23 deg/sec |     -0.24 m/sec^s |      0.32 m/sec^s |      52.721302       13.22434
```