Journal Report 5
9/30/19 - 10/6/19
Richard Zhan
Computer Systems Research Lab
Period 2, White

Daily Log

Monday September 30

I created the Event class, which stores information about traffic and slowdowns. It stores the road that has traffic. It also stores the severity of the traffic by recording the actual speed. The Event class has the method $hasExpired$, which returns a boolean depending on whether the Event is too old to be used.

Tuesday October 1

In the method $step\_simulation$, I added code for recording Events. Events are recorded if the Car's actual speed is less than 75% of the speed limit. These Events store the road, time, and actual speed of the traffic. I also had to fix a bug in my Event class due to not using pointers as class variables.

Thursday October 3

In the method $step_simulation$, after each Car checks to see if it should record its own situation as an Event, Cars will try to communicate their own Events with other Cars. Each Car would be compared to the other Cars and checked to see if they are *in range* of each other for communication. If so, they would exchange messages that had not yet expired. If a Car stores multiple Events that pertain to the same road, it uses the more recent one.

Timeline

Date	Goal	Met
9/16/19 - 9/22/19	Finish writing the simulation code and collect data on the average amount of time for each trip.	No, I corrected a bug in the navigation methods for non-DTD cars. I started but have not finished the simulation code.
9/23/19 - 9/29/19	Finish writing the simulation code and tweak variables to reach realistic settings.	Yes, I wrote the necessary code and found the correct input values to run a realistic simulation.
9/30/19 - 10/6/19	Began coding the naive (non-optimized) DTD scheme. Try to finish setting up the class *Event* and the communication system between cars	Yes, I set up the class *Event* and the DTD car communication system. I still need to incorporate these Events in the DTD navigation system
10/7/19 - 10/13/19	Finish the naive DTD scheme and begin looking into optimizations	
10/14/19 - 10/20/19	Continue looking into optimizations. Compare non-DTD times to DTD times for different maps	

Reflection

This week, I started working on the naive implementation of DTD cars. I finished creating the class *Event* and the communication system for DTD cars. I met this week's goal, so I am happy with what I accomplished this week. I have tested the program to ensure that the new modifications are working properly. Next week, I want to finish the naive implementation of the DTD cars. Even though it may run relatively slowly, I will not be simulating extremely large numbers of cars at a time. Last week, I established the baseline times of the non-DTD navigation system for my proof-of-concept scheme, so I am excited to see how the DTD navigation system compares.

```
/**
  * Event class is used to store information about traffic on the road network
  * It stores which road the traffic is on, what time it occurred, etc
  */
class Event {
public:
   Event() {};
   Event(Edge*, ld, ld, ld);

   int id;
   Edge* road;
   ld actualSpeed;
   ld speedLimit;

   ld startTime;
   bool hasExpired();
};
```