**Problem Specification**

We want to create a traffic simulator to simulate traffic flows of vehicles in a city and visualize them. The simulator can be broken down into two parts: the actual simulation algorithms for vehicles, and an user interface to create, modify, save and load a city and its traffics to run with the algorithms.

**Problem Decomposition (UML Diagrams)**

The problem can be broken into several objects that interact with each other to achieve the desired behavior.

There will be two UML diagrams addressing simulation and user interface respectively. Please refer to the actual UML files in design folder.

**Car**

The only dynamic class concept in this system, a car can change directions and move on road lanes. It has following attributes:

Position - a 2d int pair that represents the coordinate of a car on the map

Id - a unique string identifier for each instance

Length - physical length of the car

Width - physical width of the car

Direction - the current direction (up, down, left, right) the car is travelling toward

Speed - how many segments (one car length) the car will move each update cycle

currentTerrain - current map segment the car is on, could be traffic light, road or intersection

The car will be able to move using the move() method, in which the speed, decision to turn or stop will be determined. Turning is achieved using the turn() method. The car will stop when it has only one unit length before the next vehicle. It will turn at intersections and follow speed limit when it’s on road. It will stop at red/yellow traffic light and move at green light. Upon hitting the end of a road, it will destroy itself and disappear from the map.

**Bus**

Bus is a subclass of car, with an increase in length and no changes in other attributes and behaviors.

**Motorbike**

Motorbike is another subclass of car, with a decrease in length and no changes in other attributes and behaviors.

**Road**

Road describes a simulated road where vehicles can run on. As a standard terrain segment it can connect with intersections. It has the following attributes:

Id - a unique string identifier for each road instance

Length - length of a road

Start - a 2d int array representing the start coordinate of a road on map

End - a 2d int array representing the end coordinate of a road on map

Direction - a 2d vector specifying a road’s direction and applied on cars to change directions

Speed\_limit - maximum speed a vehicle can achieve on this road

Intersections - intersections this road is connected to, first one is start and second one is end

The length of the road is from user input, with a minimum length of 5 units. A road is responsible to spawn car at rates from user input, and always spawn car at the start.

**Intersection**

An intersection represents connection hub of two or more roads. It can be I shaped, T shaped or a cross. Upon hitting the intersection the car will randomly decided whether to turn or go straight. It has the following attributes:

Id - a unique string identifier for each intersection instance

Type - a string identifier for the shape of this intersection for cars to take decisions

Position - 2d coordinate of this intersection on map

An intersection can be rotated to fit the incoming road connections.

**Traffic Light**

A traffic light represents a simulated traffic controlling light. It will signal the car to stop or continue when the car is on it.

Status - current status of the traffic light, can be green, red or yellow

Id - a unique identifier to represent each traffic light instance

Position - 2d coordinate of this traffic light on map

A traffic light changes its status using the operate() method in the update cycle. If the light is green the car will continue to move. If it’s red or yellow the car will stop and wait for it to be green.

**Main**

The Main class represents the simulator itself and has entry point of the program. It will have a life cycle for the simulator programs to run in and interfaces to communicate with the GUI to take user inputs. It will run the vehicles, update the traffic lights and manage the car list.