Overview: In this project, you will work as a team to write a C++ implementation of a traffic intersection simulation.

Details:

- Each lane (eastbound, westbound, northbound, southbound) in the roadway must be made up of individual “sections” appropriately linked together. (Note that, at the intersection, different lanes will have to share the same “section”.)

- You must implement three different types of vehicles: cars, SUVs, and delivery trucks.
  - Cars will occupy exactly two sections of the roadway. SUVs will occupy exactly three sections of the roadway.
  - A given vehicle may either proceed straight through the intersection or turn right. (Challenge: implement left turns for vehicles.)
  - When a vehicle arrives (see more below), the vehicle will intend to turn right with probability $r > 0$, turn left with probability $l \geq 0$, or go straight with probability $1 - r - l$.
  - When a vehicle approaches the intersection and is clear to proceed (see traffic light and section-ahead rules below), it may advance straight or turn right unimpeded. If a vehicle is turning left, it may proceed only if no oncoming vehicle will enter the lane to be crossed until after the turning vehicle has completely exited the intersection. (How do you handle simultaneous left turns?)

- The intersection will be controlled by a traffic light in each of the four directions.
  - The traffic light will be green for a period of time ($g$ clock ticks, when traffic may proceed uninterrupted, subject to left-hand turns), yellow for a period of time ($y$ clock ticks, when traffic may proceed only when the corresponding vehicle can completely exit the intersection by the time the light becomes red), and red ($r$ clock ticks) for a period of time.
  - The length of the red for the eastbound/westbound lights will be the sum of the lengths of green and yellow for the northbound/southbound lights, and vice-versa.
  - The northbound and southbound traffic lights will be exactly synchronized with one another. Similarly, the eastbound and westbound lights will be exactly synchronized.
  - When the northbound and southbound lights are green or yellow, the eastbound and westbound lights must be red, and vice-versa.

- Your simulation will be driven by an integer-valued clock.
  - At each clock tick, your simulation must attempt to advance any vehicles present exactly one “section” (either straight, right, or left). You must do this for all four directions of travel.
  - A vehicle may proceed only if the road section ahead in its intended direction of travel is open, and subject to rules of the traffic light and (if appropriate) the rules of turning.
  - At each clock tick, you must advance the traffic lights appropriately toward an eventual change in color.
  - At each clock tick, for each lane a new vehicle will arrive with probability $p$.
  - The probability of that vehicle being a car will be $c$, of being an SUV will be $v$, and of being a delivery truck will be $1 - c - v$.
  - When a vehicle arrives to a particular lane, if there is an open section at the entrance to that lane, the vehicle will begin to “enter the lane”. (One possible approach: if the vehicle is a car, 1/2 of the car will be in that section; if the vehicle is a delivery truck, 1/4 of the truck will be in that section. The remainder of the vehicle will be “in the system” but not on any of your implemented road sections, i.e., hanging off the entry end of your simulated lane of traffic.)
Submitting (group):

- Create a tarball containing your Makefile, all classes, and a README explaining your design decisions and how to compile and run your code.
- Name the tarball similar to `cmsc240_final_st1ab_st2bc_st3cd.tgz` which includes all three students’ netids in the name of the tarball.
- Drop the tarball into the “final” folder within the course Box folder for one of the persons in your group.

Submitting (individual): Each member of your group must also complete an anonymous evaluation of all members of the group.

- Create a plaintext (ASCII) file named similar to `cmsc240_final_eval_st1ab.txt`, replacing st1ab with your netid.
- In the file, include the following:
  - Describe in outline form what you did on the project. Include interactions with others or shared activities. Include any comments concerning what you might have done differently in retrospect.
  - For each member of the team other than yourself:
    * Provide the team member’s name;
    * Give one of the ratings listed below based on your direct, personal knowledge (if you do not feel you know enough about what the team member did to make a rating, use the “no evaluation” rating); and
    * Provide comments to justify the rating.
  
  Ratings:
  
  “Did fair share of work”
  “Did more than fair share of work”
  “Did less than fair share of work”
  “No evaluation”

- Drop the plaintext (not Word) file directly into the “final” folder within your shared Box folder for this course.

All Materials Due: 12:00 Thu 03 May