Workshop - 3

Workshop Value: 10 marks (4.375% of your final grade)

Learning Outcomes

Upon successful completion of this workshop, you will have demonstrated the abilities:

- to decipher and identify a problem
- to analyze and decompose a problem
- to identify the required detailed steps to solve a problem
- to communicate the solution to fellow peers and non-technical businesspersons

Please review the following documents:

1. Workshop **Grading Policies**
2. Workshop **Submission Procedures**
3. Workshop **Group Breakdown**

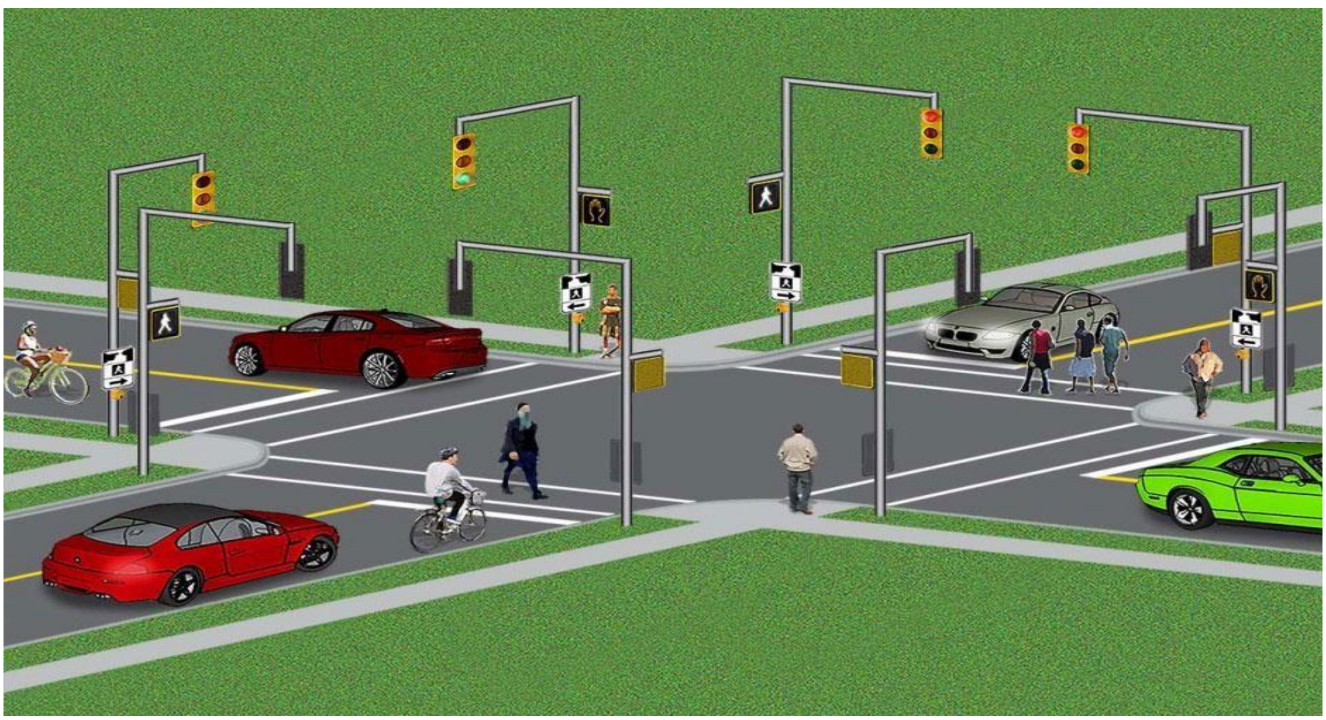
Workshop Overview

Finite state machines are around us everywhere. State machines put simply, are logical units that are limited to being in only one state at any given time. A finite state machine is one that is limited to a defined set of states (there are defined limits to the number of possible states it can be in).

Traffic lights are a good example of finite state machines. In a typical traffic light, there are 3 possible states: Green (go), Amber (don't begin to cross the intersection because the red light is about to trigger), and Red (stop). Each state is strictly timed (can be different in duration for each state) and synchronized with the opposing set of lights.

Workshop Details

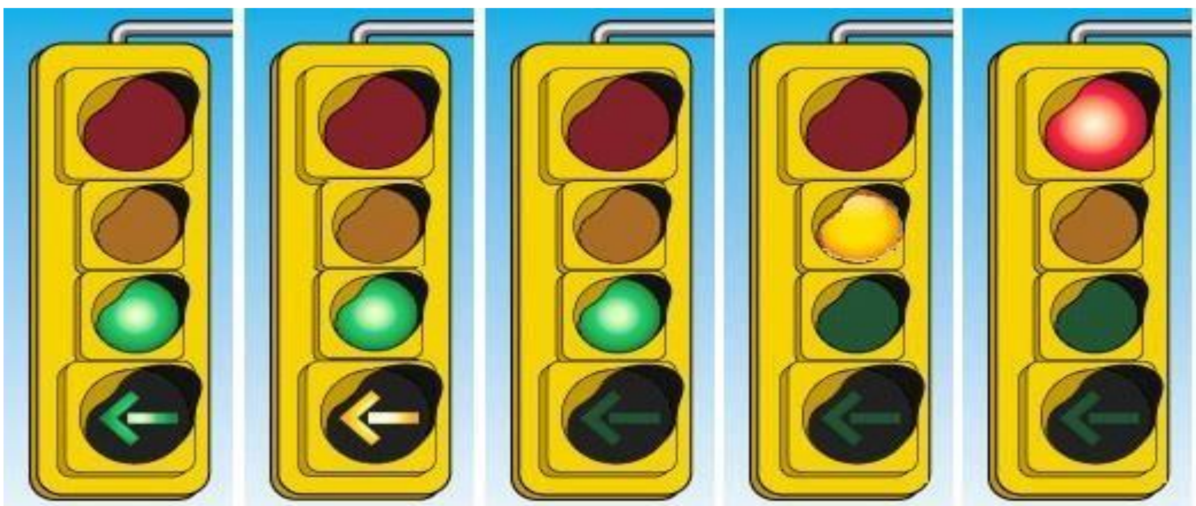
A busy intersection in downtown Toronto needs to upgrade the current 3-light 3-state traffic light system (see illustration below) to a **4-light 5-state** system to address an increasingly heavy demand for left turns. A **computer simulation** needs to be developed as a proof of concept.



Looking at the light that is red, we see the cars are stopped, waiting for the light to turn green. If 5 of more cars line up, the car sensor will signal the light to turn green sooner, as described below. The pedestrian who wants to go in the same direction as the cars stopped on the red light is also stopped. If the pedestrian presses the walk button on the light pole, this will also decrease the wait time for the red light as described below.

The 4 lights in the new system are as follows:

1. Red 
2. Amber 
3. Green 
4. Advance left turn (Green to Amber) 



4-Light 5-State Traffic Light: ***Sequence of states***

The sequence of a 4-light **5-state** system is shown in the above illustration. Below are the 5-states including the time duration for each state:

1. Solid Green + Advanced Left **Green** Arrow **[15 seconds]**
2. Solid Green + Advanced Left **Amber** Arrow **[5 seconds]**
3. Solid Green + Advanced Left Arrow (**off**) **[45 seconds]**
4. Solid **Amber** **[10 seconds]**
5. Solid **Red** **[?? seconds YOU FIGURE THIS OUT]**

Additional Features

Request-To-Walk Button

A button is provided for pedestrians to press to alert the system a person needs to cross. This is only effective when the lights are in a red-light state. This request will reduce the remaining wait time of the red-light state to 10 seconds.

Car Sensor

There is a sensor mounted on the light fixtures that monitors the number of waiting cars when in a red-light state. When the number of cars waiting reaches 5 or more, the sensor will trigger the system to reduce the remaining wait time of the red-light state to 10 seconds.

Interrupt

The defined system should take in to account an "interrupt". This is something that can occur at <u>any time</u> and will take over/supersede the active state. Examples of this type of interruption can be the traffic controllers setting all lights to flash amber (for caution) or to flash red (4-way stop) etc... **You <u>don't need to specify what the interrupt should do</u> – only account for it by checking <u>if there is</u> an interrupt to the normal process flow <u>and</u> how the system should resume when the interrupt has completed.**

Work Breakdown

[Logic 1] Define a **standard 3-light system** (red, green, amber) with an interrupt (no advance green, request to walk, or car sensor). The light durations are: Green=40 sec., Amber=15 sec., and Red=? [you figure this out].

[Logic 2] Define a **standard 3-light system** (green, amber, red) with no advance green, request to walk light, car censor, or interrupt. The light durations are: Green=45 sec., Amber=10 sec., and Red=? [you figure this out].

[Logic 3] Define a **red-light waiting sequence** (do not include the other light states) with a request to walk button and a car sensor. Determine the red light duration if the following durations are applied to the other light states: Green=50 sec., Amber=20 sec.

[Group] Define the 4-light 5-state system that supports the advanced left turn light, request to walk, car sensor, and interrupt features.

Your Task

Individual Logic Assignment

1. Determine your individual assigned logic part based on your member# (see **Group Breakdown** link at the beginning of this document)
2. Where applicable, apply the core components of the **computational thinking** approach to problem solving to help you synthesize a solution
3. Submit your individual assigned part to your professor (see **Submission Procedures** link at the beginning of this document)

Group Solution

1. In the week the workshop is scheduled, you will be working in your assigned sub-group. See **Group Breakdown** link at the beginning of this document for details on how the sub-groups are determined.
2. Please review what is expected as described in the **Grading Policies** link at the beginning of this document.
3. Submit your group solution to your professor (if you are handing in physical paper answers, follow the directions as set by your professor, otherwise, refer to the **Submission Procedures** link at the beginning of this document)

Presentation

Decide among yourselves which member among you in the **sub- group** will be doing a presentation. Priority should be given to those who have not yet done one. Refer to the **Grading Policies**, and **Submission Procedures** links for details on deadlines, expectations and how to submit your work.