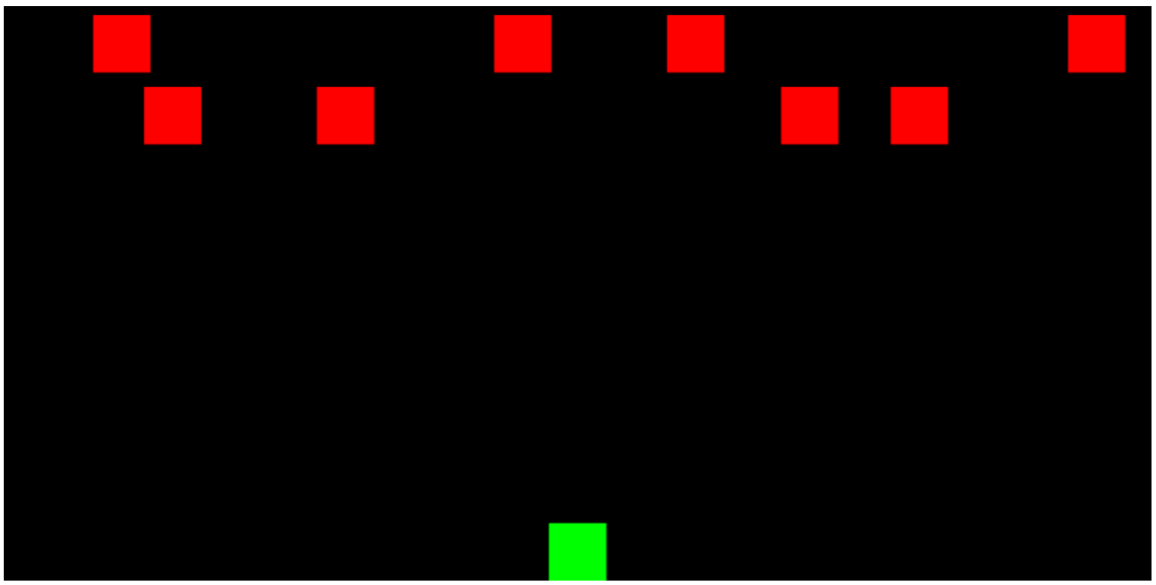
Computing Science CMPT 361 **Summer 2019**
Instructor: KangKang Yin **Simon Fraser University**

Programming Assignment #1 (20 marks)
Due on: Wednesday, June 12, at 11:59 p.m.

Plagiarism is a serious academic offense: passing off someone else's work or ideas as your own in order to get a higher mark. Plagiarism is treated very seriously. The assignment you hand in must represent your own work. Submitting source code downloaded/copied from the WWW or your classmates' solutions as your own is deemed cheating, and an F grade will be awarded. However, reading and studying code from the web and discussing ideas with your classmates is allowed, but you must acknowledge their help, and still do your own work. In the README file you submit with your source code, you must list all the open source code that you have studied, and all the people's names with whom you have discussed your assignment.

Late Policy: Late submissions are possible, but they will be penalized. One day late: 10% penalty; Two days late: 20% penalty; Three days late: 40% penalty; Four days late: 60% penalty; Five days late: 80% penalty; Six or more days late: 100% penalty.



Problem 1 (20 marks): Space Invaders!

You are to implement a simplified version of the game Space Invaders as demoed in class and described below. Any visual flare that you wish to add to the appearance of your game will be judged by the grader and may be credited at his discretion. The game window consists of a rectangle of appropriate size, e.g., so that the window will fit in the screen comfortably. You are advised to complete this problem in several steps.

(a) [4 marks] Red Aliens Movement

Draw two rows of red aliens at the top of the screen. They should move left and right randomly, and move downward in uniform speed. If any alien reaches the bottom of the screen, the aliens win and the game is over.

(b) **[2 marks] Green Cannon Movement**

Draw one green cannon at the bottom of the screen. The cannon can be moved by the left and right arrow keys on the keyboard to move left and right.

(c) **[4 marks] Red Aliens's Shooting**

The red aliens shoot bullets downward continuously in uniform speed. If any of the bullets touch the cannon, the cannon is destroyed and the aliens win. The aliens should not collide into each other. The aliens should not shoot each other either. A simple strategy is that the top row aliens only start to shoot after all lower row aliens are killed.

(d) **[3 marks] Shooting Cannonball**

The cannon shoots cannonballs upward when the user clicks on the left mouse button. If any cannonball touches an alien, the alien dies and disappears. The aliens lose if they are all killed by cannonballs before any of them reaches the bottom of the screen.

(e) **[2 marks] Faster Movement**

The aliens move left and right faster and faster, when they get closer and closer to the bottom of the screen.

(f) **[3 marks] Balance the Game Difficulty**

Tune the movement and shooting parameters so that the player does not win or lose all the time. She should win sometimes and lose sometimes.

(g) **[2 marks] Additional Game Logic**

Press 'q' to quit and 'r' to restart the game. Report in a pop-up window if the player win or lose at the end of the game.

Note that the above steps build on top of each other, in order. You need not submit individual programs to correspond to these steps. If you can implement all the required parts, a single, complete program is sufficient. **No skeleton code** is provided.

Submission: Please submit a zip file with student number and your name (i.e., `300000001_TerryFox.zip`). The zip file contains an HTML, a JavaScript, and a README file. The README should acknowledge any help you have received and any discussion you have participated, and document any steps not completed, additional features, and any extra instructions for the TA to mark your assignment. Also note that **you must use shaders** to do the drawings. HTML provides simple functions to draw 2D shapes but you will not get any marks if you use them.