

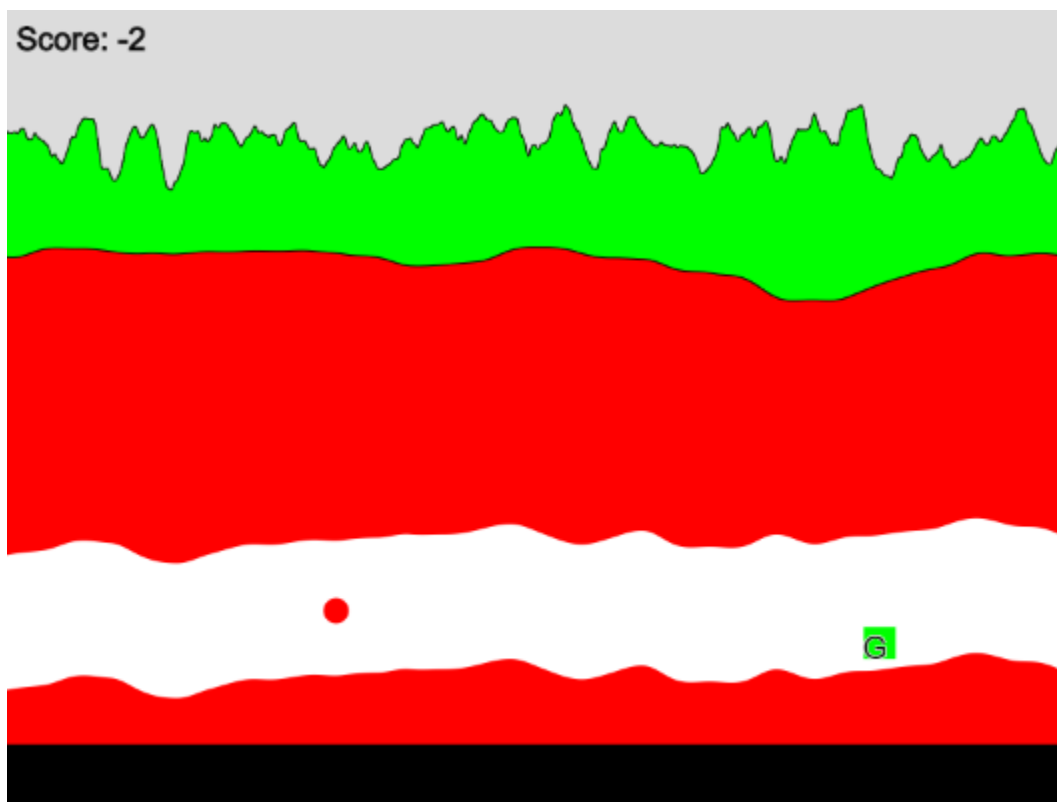
CS106 W2020 - Assignment 08

Due: Friday, March 13, 2020, 11:59 PM

You are to write a JavaScript p5 program to mimic the “Goose Game”, as seen in the video:
<https://youtu.be/HbUfjruwulA>

There is Starter Code for this assignment.

This assignment is graded out of 16 marks.



Requirements and Grading

1) Modify the Goose Game [14 marks]

The starter code contains the start of a game which we are calling “Goose Game”.

Save the starter code as “A08_GooseGame” in a directory called “CS106_A08”.

- Examine the code to understand the code.

- The starter code displays an initial “splash” screen that simply says “Press 's' to start. Have fun”. When the player presses “s” the game starts. This is completed in the starter code.
 - There is a keyPressed function that handles the key press, when the user presses “s”.
 - There is a “gameOn” global variable that determines whether the splash screen displays or the game is being played.
- Three sound files are loaded (honk, horn, and hit).
- The starter code draws a green mountain range moving from right to left on the canvas.

Modify the game as described in the following steps (and as shown in the above video). There are seven parts to this question (a to g) and each is valued at 2 marks .

- [2 marks] Add a second mountain range. This mountain range is red. It must be moving from right to left. It must be smoother and moving faster than the background green mountain range. See the video.
 - Note: Your code for drawing this second/red mountain range will be almost identical to the provided starter code that draws the green mountain range.
- [2 marks] Add the white road. It is 100 pixels wide. It must be moving from left to right. It must be smoother and faster than both mountain ranges. See the video.
 - Note: The code for this will be similar to the two mountain ranges above, and similar to the solution to question 3 on last week’s lab.
- [2 marks] Display the red ball 250 pixels from the left hand side of the screen. It does not move to the right or left. To get the 2 marks here, the ball does not have to stay inside the road (as it does in the video). As long as you have a red ball displayed 250 pixels from the left then you get the 2 marks.
- [2 marks] Allow the user/player to use the up and down arrow keys to move the red ball up and down (the starter code already does some of this for you, in the function keyPressed). Again, to get the 2 marks here the ball does not need to be inside the white road.
- [2 marks] Constrain the red ball so that it always stays within the white road. It cannot go outside the white road even if the player hits the up and down arrows many times. To get these 2 marks, the ball must always be 250 pixels from the left hand side of the canvas, and within the white road. If the ball goes outside the white road then the “horn” sound plays.

If you have completed the above bullet points you have achieved 10 of the 14 points for this question.

- f) [1 mark] In the “Goose Game” shown in the video there is a green rectangle with the letter “G”. This is meant to be a goose. Add a goose to the game. The goose must go from right to left on the white road. See the video.
- g) [1 mark] Do a hit test to determine whether the goose hits the red ball. If the goose hits the red ball do the following:
- Play the “honk” sound.
 - Restart the goose at the right hand side of the screen. See the video.
- h) [1 mark] Do a hit test to determine if the goose goes off the left hand side of the canvas. If the goose goes off the left hand side of the canvas do the following:
- Play the “hit” sound.
 - Restart the goose at the right hand side of the screen. See the video.
- i) [1 mark] Add scoring to the game. Display the score in the upper left. See the video. The scoring works as follows.
- 2 points are added if the ball hits the goose.
 - 1 point is deducted if the goose makes it to the left hand side of the screen.
 - 1 point is deducted if the ball is outside the white road.

General Correctness

Assignments that do not run may receive a grade of 0. Even if you don’t complete the entire assignment, don’t leave it in a broken state. Make sure it runs so we can find ways to give you part marks.

[2 marks] Coding Style and Efficiency

Follow the course coding style for whitespace and comments. Consult the “Code Style Guide” on LEARN. For example:

- Comment your code appropriately. Avoid superfluous comments.
- Correctly and consistently indent your code blocks.
- Use correct inline spacing for variable declaration and assignment.
- Use good line spacing to chunk sections of your code.
- Pay special attention to inline spacing for your conditional statements.
- Use semicolons.
- Use let or const, never var.
- All variables must be declared using let or const. Don’t use variables that have not been declared.

One or more marks may be deducted for solutions that have obvious inefficiencies.

- Variables that are declared or assigned, but not used.

- Unnecessarily variables that are duplicates of other variables.
- Unnecessarily repeating the same code in multiple places.
- Too many “magic numbers”: the same number appears in many places indicating a variable should have been used instead.

Restrictions

In general, you may not use any functions, libraries, or statements not covered in lecture or labs unless not specifically exempted below or in a post by a TA or instructor on this assignment discussion board.

Submitting

Zip your assignment sketch folders (CS106_A08), and submit it the correct assignment dropbox. Consult “How to Submit” on Learn for more information on how to create a ZIP.

It is your responsibility to submit to the correct dropbox with the correct files before the deadline. Otherwise you will receive a mark of 0.