LAB 1: 2D DRAWING

Overview

Write a simple 2D drawing program using WebGL.

Tasks

Your drawing program will support the drawing functionality of points, lines, triangles, squares, and disks. Points are already implemented as an example.

- For lines and triangles, the user can specify the vertices by clicking on the canvas, the line or the triangle are drawn when the user input two or three vertices.
- For squares and disks, the user only needs to click on the canvas once. The square or the disk are centered at the click position. You can specify the size of square or the disk as you desire.
- You need to change the uniform color in the code to be per vertex color. For points, lines, and triangles, the colors are per vertex. For squares and disks, the colors are per object.
- Find the TODO tasks in the provided code example.

Submission

What to submit?

- Your program source files
    - lab1.html
- Any images, data files, or setup scripts needed for your program to run (including any supplied by the assignment)
    - setupShader.js
- A readme file which:
    - Describes the TODO tasks you finished for this lab and anything that is not achieved for the lab
    - Lists which browser/OS you developed your code on (just in case)

Combine these into a single ZIP archive file and submit that ZIP file to BB. (I should be able to unzip this file and immediately run your program, so ensure you include everything needed.)

Late Penalty

You should submit your lab on time. Being late for one lab could affect your time to complete subsequent labs. All labs are due at 11:59 pm (Eastern) of the specified due date, and there is **a 10% penalty each day for up to 50%**. After that, you get zero.

AI Policy

Using AI help for the labs are ALLOWED. But you must quote the code from AI help.

Quote it like this:

// --- start AI code ---

// --- end AI code ---

Using AI generated code without quoting is considered academic misconduct and will result in 0 for the lab. Mistakes in the code from AI help will results in doubled point deduction according to the rubrics.

e.g. rubrics states -1 point for incorrect object rotation. The point deduction will be doubled to -2 if AI help is used for object rotation.

Reuse of any code of previous examples are ALLOWED and encouraged.

Grading Criteria

Grading of the labs will base on the following:

- 90%: Correctness and adherence to assignment specification.
    - Find the detail rubrics for each TODO task in the provided code
- 10%: Readability, the structure of code, use of comments, adherence to lab procedures (submitting, naming conventions, etc.)