

Due: 9/27 (11:59PM)

Requirements:

- Write a WebGL program that creates and visualizes random polygons (n-gons). Name your source code `hw2.html` and `hw2.js`. The program should meet the following requirements:
 - Set the title of the program to “hw2” (must appear as such on title bar).
 - The program generates 20 random polygons at random positions with random colors (see Fig. 1).
 - Each polygon must have a random number of vertices (between 3 and 9).
 - Each polygon must continuously rotate and scale (see accompanying video on Canvas). Note that the rotation must be done *in-place* about its own center, not about the origin. *Hint: first translate the polygon to the origin, then rotate, then translate it back to its original position.*
 - The scaling must go back and forth between scaling factors of 1 and 0 as upperbound and lowerbound, respectively.
 - Each time you hit F5 (refresh), the canvas must display a new set of 20 random polygons rotating and scaling in-place (See Fig. 1 – 2).

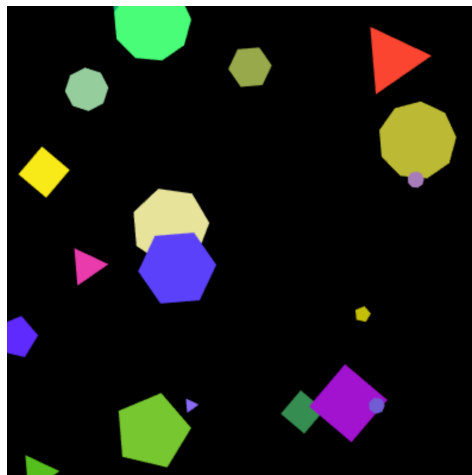


Figure 1: Rotating and scaling polygons 1

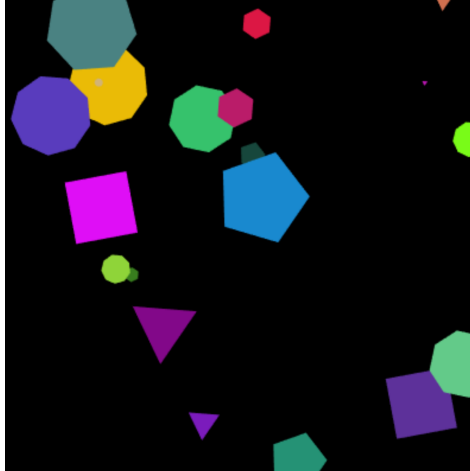


Figure 2: Rotating and scaling polygons 2

What to submit:

- Submit all your **source files (.html, .js)** that are needed for compilation, including **library files/folders**. *Missing library files/folders will lead to point deduction.*

How to submit:

- Use Canvas Assignment Submission system to submit your source files.
- Make sure to zip all your files/folders into `hw2.zip`, then submit your `hw2.zip` as a single file.

Policy

- Do all the assignments on *Chrome Development Tools* using HTML, JavaScript, and GLSL ES.
- At the top of each source file, provide comments specifying the author, date, and a brief description of the file.
- Source code must contain enough comments here and there to make it easy enough to follow. Insufficient comments could lead to point deduction.
- Incomplete program will get almost no credit (e.g., program does not run due to compile errors or program terminates prematurely due to run-time errors).
- *Thou shall not covet thy neighbor's code.* If identical (or nearly identical) submissions are found among students, every student involved will get automatic zero for the assignment. The same goes for copying existing code from online source.
- If a student makes multiple submissions, only the last submission will be considered valid.