Due: 10/24 (11:59PM)

Requirements:

- Write a WebGL program that does a simple image processing task. Name your source code `hw4.html` and `hw4.js`. The program should meet the following requirements:
  - The program displays an image and lets user adjust the hue. *(Hint: you may first convert RGB color to HSV color, then adjust the hue component).*
  - The image should be accessed through its URL address. Do not include an actual image file in your submission.
  - Implement `dat.gui` scroll bar widget so the user can adjust hue interactively (see Fig. 1).
  - Also see the accompanying demo video. Your mission is to reproduce the program in the video.
  - The slider value should range from -180 to 180. This is because hue is typically expressed by an angle on color wheel, and the user-selected value represents how much the orignal pixel color shifts along the color wheel in terms of degree angle. Note that since 180 degree angle is the same as -180 degree angle, they should result in the same outputs.

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.*
- Make sure your **library folder/files** are in the right location relative to your main program (.html), such that when your main program (.html) is clicked as is, it should run without problem. *Failure to do so 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 `hw4.zip`, then submit your `hw4.zip` as a single file.

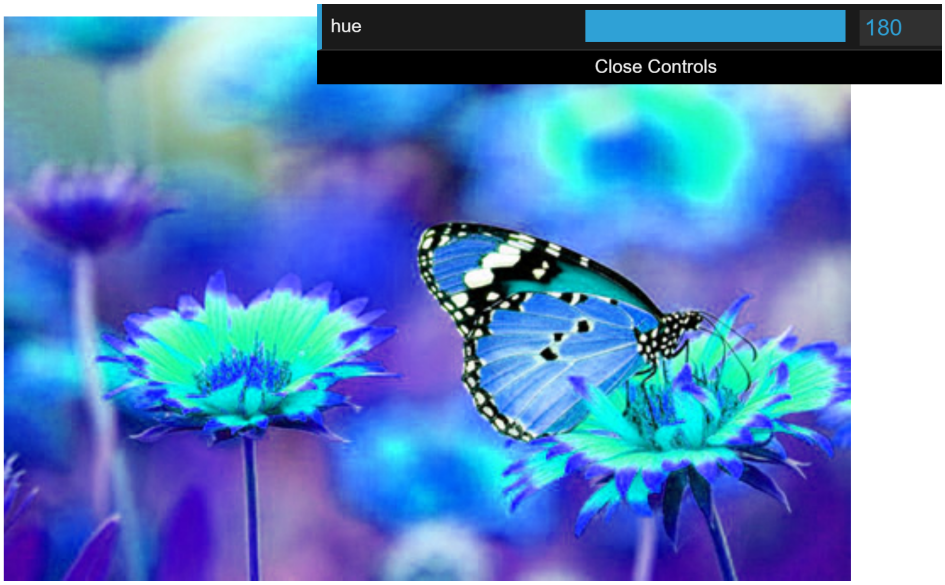

Figure 1: Interactive hue adjustment

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.