CS106 W19 - Assignment 06

Due: Friday, March 8, 11:59 PM

QUESTION ONE: Hierarchical Drawing

Write a Processing sketch that draws a scene of your invention, making use of geometric context functions including `beginShape()`, `vertex()`, `endShape()`, `pushMatrix()`, `popMatrix()`, `translate()`, and `scale()`. The scene can consist of anything you want, as long as it satisfies the conditions below.

a) Your sketch window should have size at least 400×400, and no more than 1000×1000. Choose a size that's most appropriate for your scene.

b) The scene should be *figurative*: it should contain clear objects (people, animals, plants, buildings, vehicles, etc.), and not consist entirely of abstract shapes.

c) Don't use any data from external files (images, SVG files, text files, etc.). All drawing should be based on calls to built-in drawing functions in the sketch itself.

d) At least one object must be an "advanced shape" (i.e., it must be drawn using `beginShape()`, `vertex()`, and `endShape()`).

e) There must be at least one object that's *repeated*: it should appear at least three times with different transformations, carried out using geometric context. Put the code to draw the object into a helper function, and call that function three times wrapped inside different transformations. At least one of the copies should be scaled.

Note that "repetition" doesn't necessarily imply the use of a loop. Suppose you have a helper function called `myShape()`. Then both of the following would be considered to contain three calls to the function:

```
pushMatrix();
// Perform some transformations
myShape();
popMatrix();

pushMatrix();
// Perform some transformations
myShape();
popMatrix();

pushMatrix();
// Perform some transformations
myShape();
popMatrix();
```

```
for ( int idx = 0; idx < 3; ++idx ) {
   pushMatrix();
   // Perform some transformations
   myShape();
   popMatrix();
}
```

Note that the demo code "HierarchicalStreet.pde" is a good example of Hierarchical Drawing.

If you are struggling to come up with a scene then consider having a snowman made up three large balls of snow, a belly button, and carrot nose. In your scene, have multiple copies of the snowman at different sizes. Perhaps the carrot nose can be draw using beginShape(), vertex(), and endShape().

QUESTION TWO: Video Processing

This is version 2 of Assignment 6 for students who do not have a working video camera. Instead of a camera we'll use the video flyboard.mp4.

Start with the code videopixels.pde, and the video flyboard.mp4 in your data directory.

Modify videopixels.pde so that it uses the flyboard.mp4 video rather than the camera feed. Your modified code will be as shown below.

```
// CS106 A07Q02 W19 VERSION 2

// import the video library
import processing.video.*;
int row; // this will be a row that is shown as a black line

// this is the movie object
Movie mov;

void setup() {
  size(320, 240);

// load a movie and play it looping
  mov = new Movie(this, "flyboard.mp4");
  mov.play();
  mov.loop();
}
```
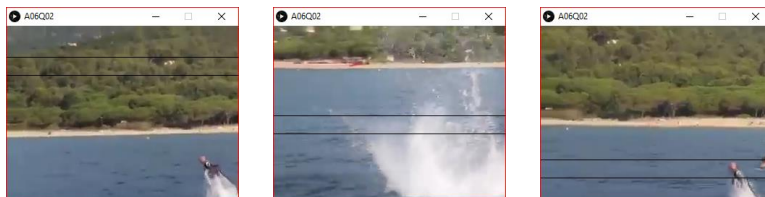
Modify the code so that the result mimics what you see in this youtube video, with the exception that the video shown on the sketch window is the flyboard.mp4.

https://youtu.be/pBOdkq9wPpM

It is meant to represent an older television where we see horizontal lines on the sketch window.

QUESTION THREE: Video Processing

This is version 2 of Assignment 6 for students who do not have a working video camera. Instead of a camera we'll use the video flyboard.mp4.

Start with the code videopixels.pde, and the video flyboard.mp4 in your data directory.

Modify videopixels.pde so that it uses the flyboard.mp4 video rather than the camera feed. Your modified code will be as shown below.

```
// CS106 A07Q03 W19 VERSION 2

// import the video library
import processing.video.*;
int row; // this will be a row that is shown as a black line

// this is the movie object
Movie mov;

void setup() {
  size(320, 240);

// load a movie and play it looping
  mov = new Movie(this, "flyboard.mp4");
  mov.play();
  mov.loop();
}
```

Modify the code so that the result mimics what you see in this youtube video, with the exception that the video shown on the sketch window is the flyboard.mp4.

https://youtu.be/gFUCOf7LZ5E

It is meant to represent a newer digital television where we constantly see pixilation of parts of the sketch window.

QUESTION FOUR

This is version 2 of Assignment 6 for students who do not have a working video camera. Instead of a camera we'll use the video flyboard.mp4.

Start with the code videopixels.pde, and the video flyboard.mp4 in your data directory.

Modify videopixels.pde so that it uses the flyboard.mp4 video rather than the camera feed. Your modified code will be as shown below.

```
// CS106 A07Q04 W19 VERSION 2

// import the video library
import processing.video.*;
int row; // this will be a row that is shown as a black line

// this is the movie object
Movie mov;

void setup() {
  size(320, 240);

// load a movie and play it looping
  mov = new Movie(this, "flyboard.mp4");
  mov.play();
  mov.loop();
}
```

Modify the code so that the result mimics what you see in this youtube video, with the exception that the video shown on the sketch window is the flyboard.mp4.
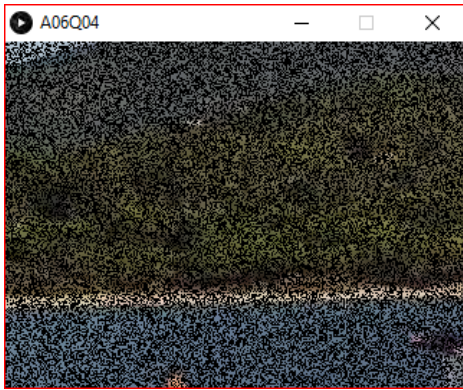
https://youtu.be/dYByfYzIoQg

As the user moves the mouse from left to right the amount of random pixels that are black goes from 20% to 80%.
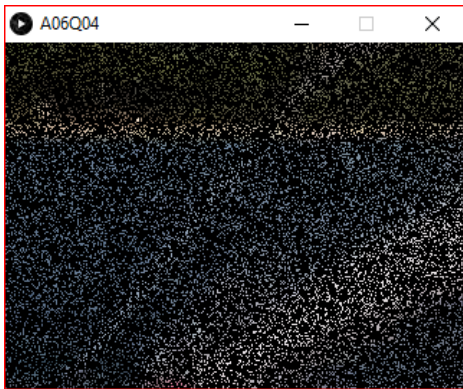
The following is an example when the mouseX is near the left-hand side of the sketch window.

This following is an example when the mouseX is halfway across the sketch window.

This following is an example when mouseX is at the right-hand side of the sketch window.

Submitting

Create a folder "A06_username", but replace "username" with your UW id. So if your email is "jac926@edu.uwaterloo.ca" you would create a folder "A06_ jac926".

SAVE your sketches in that folder as "A06Q01_username", "A06Q02_username", etc. Again, replace username with your UW id.

Zip your "A06_username" folder (with "username" replaced by your UW id) and submit it the correct assignment dropbox.

It is your responsibility to submit to the correct dropbox with the correct files before the deadline. Otherwise you will have marks deducted.

Academic Integrity

All assignments in CS106 are done individually. Group work and sharing of code is not allowed.

Detecting Plagiarism:

- We monitor Reddit, File Trading Sites, past year CS106 assignments, etc.

- We use Measure Of Software Similarity (MOSS)

    - automatic system for determining the similarity of code

Discipline

Discipline (Policy 71)

- https://uwaterloo.ca/secretariat-general-counsel/policies-procedures-guidelines/policy-71