CS 106 Winter 2019

Lab 03: Input and Output

Due: Wednesday, January 23th, 11:59pm

Summary

This lab will allow you to practice input and output. Each question is on a separate page.

SAVE each sketch as “L03Q01”, “L03Q02, “L03Q03”, etc.

QUESTION ONE: Open and display a graphics file.

Cut and paste the following code into a Processing sketch called L03Q01.

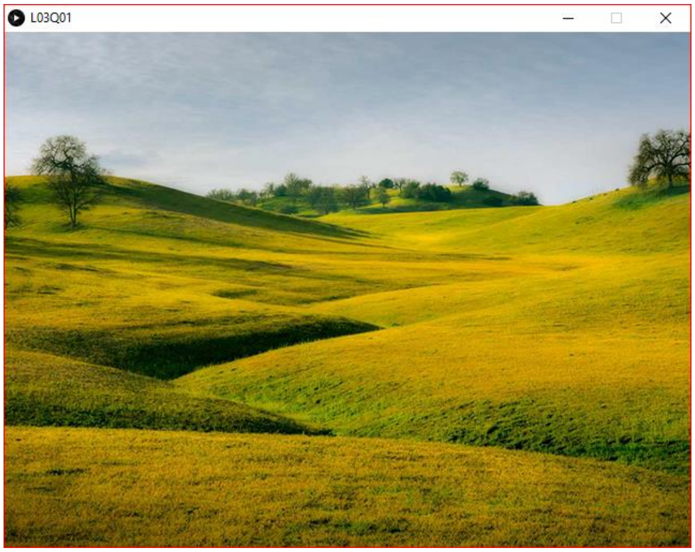
In the directory L03Q01, add a subdirectory called data, and in the subdirectory put the file pasture.jpg.

Look at your directory structure to ensure it is correct. You should have a directory L03Q01. Within that directory should be your Processing file L01Q01.pde. There should also be a directory data and in the data directory should be the graphics file pasture.jpg. The directory structure is shown in the box to the right.

```
L01Q01 (directory)
      L01Q01.pde (Processing file)
      data (directory)
            pasture.jpg (graphics file)
```

Run the sketch. Your result should be as shown below. That is all there is to L03Q01.

```
PImage img;

void setup()
{
  size(640, 480);
  img = loadImage("pasture.jpg");
}

void draw()
{
  background(200);
  image(img, 0, 0);
}
```



QUESTION TWO: Save a sketch to an output file.

Cut and paste the following code into a Processing sketch called L03Q02.

In the directory `L03Q02`, add a subdirectory called `data`, and in the subdirectory put the file `ball.png.`

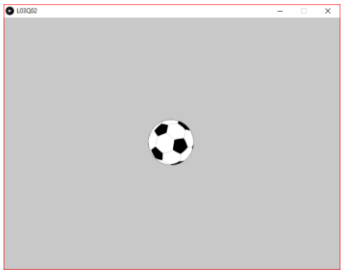
Modify the `draw()` function below to display the ball in the middle of the screen. You'll do this with a line that starts with: `image(img,`

Run the sketch. Your result should be as shown below.

```
PImage img;

void setup()
{
  size(640, 480);
  img = loadImage("ball.png");
}

void draw()
{
  background(200);
  // Add a line of code here to display the ball
  //     in the middle of the sketch window.

}
```

Also, and a function KeyPressed(). If the user presses ‘s’ it saves the contents of the sketch window to a file called `screenshot.jpg`. We have seen the code for this previously.

```
void keyPressed () {
  if ((key == ‘s’) || (key == ‘S’)) {
    save("screenshot.jpg");
  }
}
```



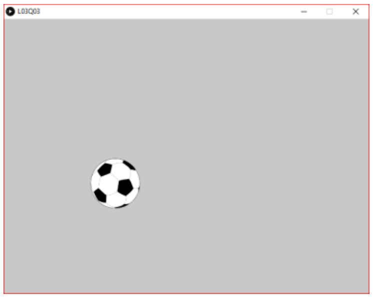
QUESTION THREE: Load and animate a graphics file.

Cut and paste the following code into a Processing sketch called L03Q03.

In the directory `L03Q03`, add a subdirectory called `data`, and in the subdirectory put the file `ball.png.`

Create a `draw()` function that causes the ball to go back and forth on the screen. Your result should mimic what is in this video: https://youtu.be/vlQdFXSZAMY

```
PImage img;
int xPos = 0;
int speed = 3;

void setup()
{
  size(640, 480);
  img = loadImage("ball.png");
}
```



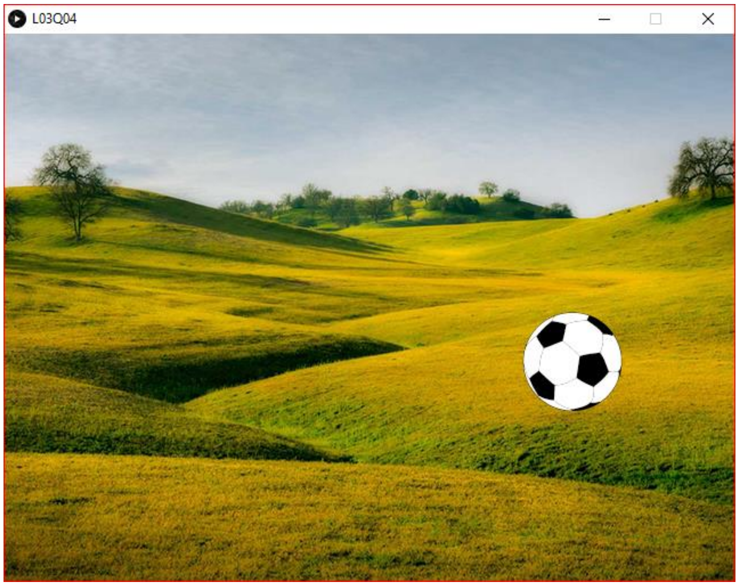
QUESTION FOUR: Load and display two graphics files at once. Animate one of them.

This question is similar to Question Three except you will use two images instead of one image.

Cut and paste your answer from Question 3 above into a Processing sketch called L03Q04.

In the directory `L03Q04`, add a subdirectory called `data`, and in the subdirectory put the file `ball.png` and `pasture.jpg.`

Create a `draw()` function that causes the ball to go back and forth on the screen as in Question 3 above, but in this Question 4 the only difference is that the background is the pasture.



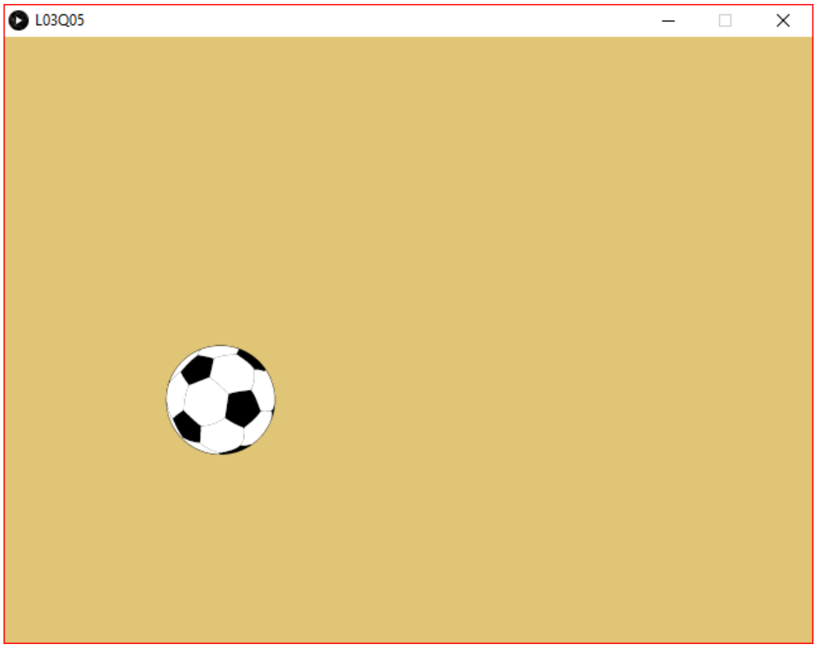
QUESTION FIVE: Load a graphics image and have Hit Tests on it.

Create a Processing sketch called L03Q05.

In the directory `L03Q05`, add a subdirectory called `data`, and in the subdirectory put the file `ball.png.`

Create a `draw()` function that causes the ball to go back and forth on the screen. Your result should mimic what is in this video: **https://youtu.be/DN59Vf5rLBA**

- If the user presses the mouse on the ball (i.e. a hit test) the background becomes a random color.
- If the user presses 'w' on the keyboard the background becomes white.
- If the user presses 'b' on the keyboard the background becomes black.



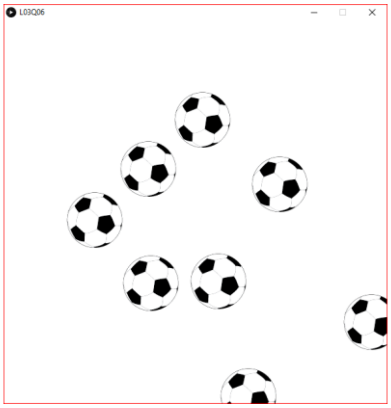
QUESTION SIX: A painter program to draw graphics from a file.

Create a Processing sketch called L03Q06.

In the directory `L03Q06`, add a subdirectory called `data`, and in the subdirectory put the file `ball.png.`

Create a `mousePressed()` function. Each time the user presses the mouse the ball is displayed on the screen at the mouse location. Starter code is below and a screen shot is below. I YouTube video is at: https://youtu.be/axYwKnBW0JI

```
PImage img;

void setup()
{
  size(600, 600);
  img = loadImage("ball.png");
  background(255);
}

void draw() {
}
```



If you forget how to solve problems like this, look at Example 3-5 from the textbook and online at: http://learningprocessing.com/examples/chp03/example-03-05-mouse-key-events

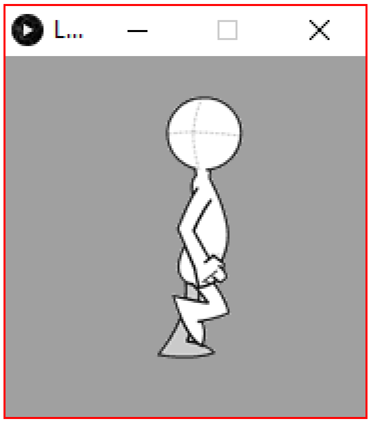
QUESTION SEVEN: Walk cycle animation.

Create a Processing sketch called L03Q07.

In the directory `L03Q07`, add a subdirectory called `data`, and in the subdirectory put the files `1.png, 2.png, 3.png, 4.png, 5.png, 6.png, 7.png, 8.png,` and `9.png.`

You are to write a `draw()` function so that each image shows in order so that we see an animation of the character walking. It should mimic what we see in the YouTube video: https://youtu.be/1XI2ubmcuFY Starter code is below and a screen shot is below.

```
// Original code from Professor Dan Vogel in CS105.
// the walkcycle image sequence
PImage[] walkcycle = new PImage[9];
// the index of the current walkcycle element to show
int frameToShow = 0;
// position of image
float xPos = width / 2;
float yPos = height / 2;

void setup() {
  size(180, 180);

  // load all the walk cycle images
  walkcycle[0] = loadImage("1.png");
  walkcycle[1] = loadImage("2.png");
  walkcycle[2] = loadImage("3.png");
  walkcycle[3] = loadImage("4.png");
  walkcycle[4] = loadImage("5.png");
  walkcycle[5] = loadImage("6.png");
  walkcycle[6] = loadImage("7.png");
  walkcycle[7] = loadImage("8.png");
  walkcycle[8] = loadImage("9.png");

  // slow down draw to see the animation
  frameRate(8);
}
```



Hint: Each time through `draw()` you need to display one image. The first time through `draw()` you display `1.png`. The second time through `draw()` you display `2.png`. The third time through `draw()` you display `3.png`. And so on until you display `9.png`. Then you start over again with `1.png`. You will want to use the global variable `frameToShow` to keep track of the index of the image to be displayed.

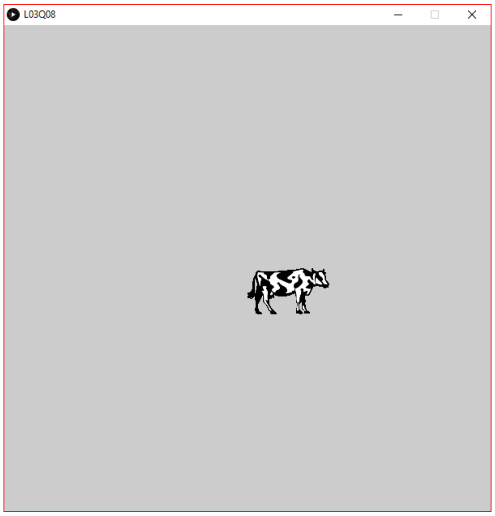
QUESTION EIGHT: Load and display and svg vector-based graphics file.

Create a Processing sketch called L03Q08.

In the directory `L03Q08`, add a subdirectory called `data`, and in the subdirectory put the file `cow.svg.`

Run the sketch. Your result should be as shown below. That is it. That is all there is to L03Q08.

```
PShape animal;

void setup()
{
  size(600, 600);
   animal = loadShape("cow.svg");
}

void draw()
{
  shape(animal, width/2, height/2);
}
```



Note: `cow.svg` is a **vector-based graphic** whereas all other graphics above such as (`ball.png, pasture.jpg, and 1.png`) were all **raster-based graphics**.

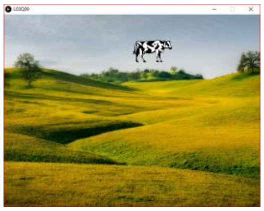
QUESTION NINE: Animate an image from a svg file.

Create a Processing sketch called L03Q09.

In the directory `L03Q09`, add a subdirectory called data, and in the subdirectory put the file `cow.svg` and `pasture.jpg.`

You are to write a `draw()` function so that we see the cow dropping from top to bottom. Also, the user can press any key and the cow jumps up about 50 pixels. Your result should mimic what we see in the YouTube video: https://youtu.be/IG_Ss5wUFQA Starter code is below and a screen shot is below.

```
PShape animal;
PImage img;
int xPos;
int yPos = 0;
int speed = 3;

void setup()
{
  size(640, 480);
  xPos = width / 2;
  img = loadImage("pasture.jpg");
  animal = loadShape("cow.svg");
}
```



Starter code for you `keyPressed()` function is as follows. All you need to do is add a line to change the value of `yPos`. You don't need an `if` because `yPos` should degrease regardless of which key is pressed.

```
void keyPressed() {

}
```

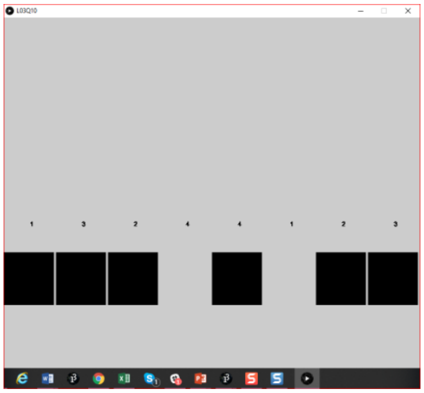
QUESTION NINE: For The Memory Game add graphics.

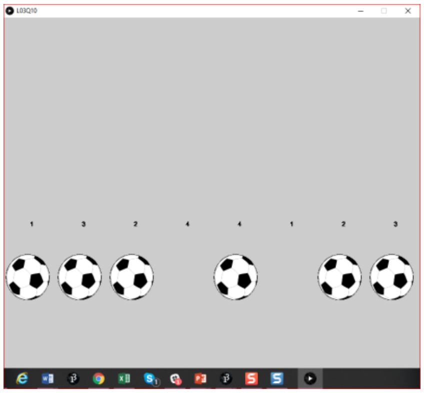
Create a Processing sketch called L03Q10.

In the directory `L03Q10`, add a subdirectory called data, and in the subdirectory put the file `ball.png.`

You are to modify one line in the `draw()` function so that we see balls instead of rectangles on the screen. The two screens are shown, one with rectangles and the one with balls.

```
PImage img;

int[] pairs = {1, 3, 2, 4, 4, 1, 2, 3};
boolean[] base = {true, true, true, false, true, false, true, true};

void setup() {
  size(800, 800);
  img = loadImage("ball.png");
}

void draw() {

  fill(0);
  float rectSize = width/pairs.length;
  for (int i = 0; i < pairs.length; i++) {

    text(pairs[i], i * rectSize+rectSize / 2, height / 2);
    if (base[i]) {
      // You to add replace the following line of code
      //    with one line of code to display the soccer
      //    ball at the appropriate location and size.
      rect(i * rectSize, height / 2 + rectSize / 2, rectSize-5, rectSize);
    }
  }
}
```





QUESTION ELEVEN: Create a pdf.

Cut and paste the code below into a Processing sketch called L03Q11.

Below the comments “// Draw as usual”, add several lines of code to draw something. You must use the following at least once each: `fill(), text(), rect(), ellipse(), and triangle()`.

Your file “output.pdf” is a pdf file of the contents of your sketch window. You can open “output.pdf” in a pdf reader.

```
import processing.pdf.*;

boolean recording = false;

void draw() {

  beginRecord( PDF, "output.pdf" );

  // Draw as usual

  endRecord();

  noLoop();
}
```

QUESTION TWELVE: Reading Text

Cut and paste the code below into a Processing sketch called L03Q12.

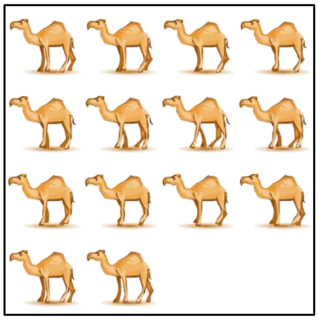
In the directory `L03Q12`, add a subdirectory called data, and in the subdirectory put the file `marley.txt.`

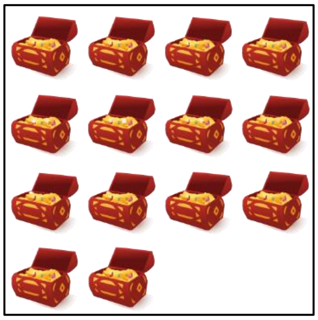
Run the sketch. The contents of the file `marley.txt` should be displayed by `printArray()`. That is it. That is all there is to L03Q12.

```
void setup()
{
  String[] lines = loadStrings( "marley.txt" );
  printArray( lines );
  println("The number of lines in marley.txt is:", lines.length);
}
```

QUESTION THIRTEEN: Animation.

This question is related to A03Q02. This question uses the files camel.png, cobra.png, carpet.png, and chest.png. The files look as follows:





Cut and paste the code on the following page into a Processing sketch called L03Q13. In the directory L03Q13, add a subdirectory called data, and in the subdirectory put the files camel.png, cobra.png, carpet.png, and chest.png.

- Document the code.
- Change the code: img = loadImage("camel.png") to load one of the following: cobra.png, carpet.png, or chest.png.

```
PImage img;
int[] sX = new int[16];
int[] sY = new int[16];
int frameIndex = 0;
int xPos;
// The following variable names were chosen based
//      on the slide this week that shows the copy command.  The slide shows:
//      copy( img, sx, sy, sw, sh, dx, dy, dw, dh );
int sW = 128;
int sH = 128;
int dW = sW;
int dH = sH;

void setup()
{
  size(512, 512);
  xPos = width;
  frameRate(18);
  img = loadImage("camel.png");
  int index=0;
  for (int c = 0; c < 4; c++) {
    sX[index] = (c * sW);
    sY[index] = 0;
    index++;
  }
}

void draw() {
  background(127);

  copy(img, sX[frameIndex], sY[frameIndex], sW, sH, xPos, height / 2, dW, dH);
  xPos -= 3;

  if (xPos < 0 - dW) {
    xPos = width;
  }

  frameIndex = (frameIndex + 1) % (4);

  fill(0);
  stroke(0);
  textSize(24);
  text("Current Frame Index is: " + frameIndex, 10, 50);
}
```

Submission

Submit all sketch directories from this lab as one ZIP file called L03.zip to the lab dropbox on Learn.

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.