

# GRAPHICS PROGRAMMING UNIT PROJECT

## DESIGN & IMPLEMENT AN INTERACTIVE ANIMATION

### Overview

- This is the project for the Interactive Graphics unit. This project covers interactive graphics programming.
- The project will be assessed with a letter grade, and contribute toward ~11% of your final grade.
- PLEASE use a zip utility to bundle your program files together and send them as ONE attachment to the email. Hand in the written part on hard-copy

### Project Description

- The purpose of the project is to create an interactive animation graphics application in **Processing** that involves user independent animation as well as user dependent interactivity.
- It is up to you to decide what the animation will be—a drawing tool, a short story, a game, a simulation, whatever! Be creative!
- A simple example would be to simulate a ball bouncing up and down, and if the user clicks on the ball, then the ball stops bouncing or bounces in a different direction. (Your project should be a bit more complicated than that, but you could start with this idea and build on it.)
- General Advice.
  - COMPLETE ALL OF THE LABS, AND THE PROJECT-2 LABS. THEY WILL GIVE YOU ALL THE SKILLS THAT YOU NEED!!!
  - Save often!
  - Try to get a **simple working version** up as fast as you can, then go back and add details and options later (this is called “iterative development”).
  - Make copies of intermediate versions of your project, just in case your code starts developing in a direction that you don’t like or can’t get to work. Then you can easily go back to a previous intermediate version.
  - Start on your project right away! Don’t leave it till the night before.
  - **If you need help ASK!**

### Grading and Requirements

- The project has two parts: **(1) design**, and **(2) application**, with the design worth 20% and the application worth 80% in determining your grade for this project.
- The design part involves **written documentation**, to be written using a word processor (e.g., in Word) and submitted as **printout** (preferably).
- The second part is to be written using Processing.
- I very strongly recommend that you start by working on the design part first. Draw out your ideas on paper. Think carefully about what you want your application to look like and how you want the user to interact with it.
- Think about what the user might do wrong and how you would deal with that. Then, after your design seems solid, start programming.

**(1) Design**

1. Describe (in words) the application that you are going to create.
2. Plan out what the Processing window will look like, how it will change when it is animated, and what the user will do to interact with it.
3. Be sure to include a drawing in your documentation that illustrates your design ideas, as well as instructions for the user.
4. Submit a 1-page document that contains all of the information listed above.

**(2) Application**

1. Using **Processing**, implement the application that you have designed.
2. Your application must contain the following **required elements**:
  - Comments!
  - Different shapes (e.g., lines, ellipses, triangles, etc.)
  - Different colors.
  - Animation—at least one of the shapes in your application must move around, change shape, or change color, independently.
  - Interactive aspects—the application must respond when the user presses particular keys or moves/clicks the mouse.

## PROJECT-2 Submission Checklist

<b>Do you have?</b>	
<p><b>A one-page word document describing and defending the choices you made in your program. Your paper should contain:</b></p> <ol style="list-style-type: none"> <li>1. Name, Date, Class in top left corner and then program title centered</li> <li>2. 1 paragraph general description of program</li> <li>3. 1 paragraph description of animation element and interaction element</li> <li>4. Simple drawing of what the program window will look like</li> <li>5. 1 paragraph description of problems that you encountered.</li> </ol>	
<p><b>A program, written in Processing that implements the design discussed in your document. It should:</b></p> <ol style="list-style-type: none"> <li>1. Include comments</li> <li>2. Include 3 or more different shapes</li> <li>3. Include 3 or more different colors</li> <li>4. Include an animated element that moves and/or changes shape and/or changes color on its own (without user interaction)</li> <li>5. Include an interactive element - some part of the application <b>MUST</b> respond to input from the user (mouse movement, keystroke).</li> <li>6. Be <b>NON-TRIVIAL</b> - it should be more complex than the application you created in lab #3.</li> <li>7. Be <b>ORIGINAL</b> - don't just hand in the results of expanding on the previous labs in class.</li> <li>8. Include <b>SPECIFIC</b> instructions regarding how the user can interact with your program. <b>IF POSSIBLE</b> put those instructions directly into your application. If you are unable to include instructions in your application, make sure you include in your <b>PAPER</b> how the user will interact with your program.</li> <li>9. <b>IMPORTANT</b>: do not copy other's code. It must reflect your original/independent work</li> </ol>	