Animation

CS 3451: Project 2B

1 - Objective

The goal of this project is to learn how to use transformations and camera position to create an animated scene. This project is an extension of Project 2A, and you will incorporate your object from Project 2A into this assignment. Like Project 2A, creativity and effort is a part of this project.

2 - Deadline

Your project solution should be submitted on T-Square by 11:55 PM on Saturday, February 24, 2018.

3 - Process

3.1 Source code

You should extend your code from Project 2A to complete this project.

3.2 Project Description

Now that you have created an object for Project 2A, you should have some ideas of how you wish to incorporate it into a fully animated scene. The most interesting animated scenes, no matter how short, seek to tell a story. Introduce your character, have your character carry out an action, and then resolve the scene.

You will most likely want to create more objects to populate your scene, but these new objects can be more simple than Project 2A's object. Your main goal for this assignment will be to make one or more of the objects in the scene move, and to also move the virtual camera through the scene. You will probably want to use a "time" variable to help control motion of your objects.

Below is a checklist of elements that you must include in your scene:

Camera Motion

You should move the camera smoothly through the scene, rather than keeping it in one place. Just rotating the entire scene does not count as moving the camera.

Please note that having the user press keys or move the mouse to control the camera does NOT count towards automatic motion of the camera. If you want to include user controls, have an automatic motion of the camera in the first part of the scene, before handing over controls to the user.

Include Project 2A Object

You must incorporate Project 2A's object somewhere in your scene.

Object Animation

Include at least two object motions in the scene (distinct from the camera motion). One of these motions should include translation, and another, different motion should include rotation. If you wish, these two different motions

can be for two different parts of the same object, or they can be motions of different objects. Make sure it is clear that these objects are moving, and not just changing their apparent positions due to camera motion.

Object Instancing

At least one of the objects that you create should be replicated using *instancing* so that it appears in more than one location in the scene. You must **not** use duplicate lines of code to achieve this, but rather you must use the matrix stack and procedural encapsulation to accomplish this. The object that you instance should be composed of several sub-parts. For example, just placing a sphere in several places doesn't count as instancing. Place a comment at the very start of your program that explicitly states which object you are replicating using instancing. You do not need to use Project 2A's object as the object that you instance – you can instance any object that you want. You should instance a *whole* object; calling the left and right arm two instances of an arm will not count.

Lighting and Shading

You must include at least one light source in your scene. Surfaces in the scene should be illuminated by the light source. Do not use only ambient light.

Duration

Your animation should create more than 300 frames of images. Please create an animation that finishes in a reasonable amount of time on your computer.

If you find that your animation runs slowly, it is most likely that this is due to use of many polygons in the scene. By far, the most common reason for having many polygons in this assignment is the use of lots of spheres. If you want to use many spheres, consider using the built-in Processing command sphereDetail() that specifies how many polygons a sphere uses.

We will be looking for each of the above items in your animation. Omitting any of the above elements will cause a deduction in your grade for this assignment.

3.3 Effort is Part of the Grade

This assignment will be graded partially based on our assessment of the amount of care, effort, and creativity that you put into creating your animated scene. If you choose a simple scene and throw it together, you will not get a high score on the "effort" component of this project. **20% of the project will be based on our assessment of your effort.**

3.4 Optional Elements

If you wish to, you may add textures to your scene. It is NOT necessary to add textures in order to have a successful animated scene.

Unlike part 2A, you are allowed to load external models as part of your scene. These might be objects that you have created using modeling programs such as Blender or Maya. Note, however, that in this project we are looking for how you animate your scene, and not for your skill in modeling in these other systems. Your emphasis should be on scene creation and animation within Processing.

3.5 Authorship Rules

The code that you turn in should be your own. You are allowed to talk to other members of the class and to the Professor and the TA about general implementation of the assignment. It is also fine to seek the help of others for general Processing/Java programming questions. You can and should use your Project 2A code as a starting point for this project. You may not, however, use code that anyone other than yourself has written, with two exceptions. First, you may use parts of the example code from 2A such as the code for cylinders. Second, you may use outside code **only** for the purpose of reading in an external polygonal model (such as from Blender or Maya). If you choose to do this, be sure to put a comment in your code giving proper credit for the source of this

code. As always, you should not show your code to other students. Feel free to seek the help of the Professor and the TA's for suggestions about debugging your code.

3.6 Submission

In order to run the source code, it must be in a folder named after the main file. When submitting any assignment, leave it in this folder, compress it into a zip file (not tar or rar) and submit via T-square.