CS 174A — Introduction to Computer Graphics: Assignment 2

Let your imagination and creativity fly! Due: End of Friday May 19th

Weight: 15 % *Maximum points:* 37

Note: We will screen the games and animations in-class and hold a vote on the best ones! The winners will receive a 0.2% (4th place), 0.45% (3rd place), 0.75% (2nd place), or 1.5% (1st place) bonus to their course grade.

Collaboration: **None**. If you discuss this assignment with others you should submit their names along with the assignment material. ***Start working on this assignment early. You will not have time to do a satisfactory job at the last minute.***

Write a program that displays an animated scene. Your scene should include a combination of hierarchical objects that move around. Required elements:

- [4 points] Make use of hierarchical objects with at least three levels (e.g., a human arm)
- [4 points] Demonstrate the camera tracking a moving object sometime, by overwriting the camera matrix using *lookAt()*.
- [6 points] Design polygonal objects of your own to supplement the existing ones. To specify these shapes you must provide novel positions, normals, and texture coordinates to the graphics card by extending class Shape().
- [2 points] Assign reasonable texture coordinates to, and texture, an instance of your custom polygonal object. Either texture it by mapping an image file, or procedurally (more like the Funny_Shader demo).
- [2 points] Your texture coordinates from the previous step must be designed to create an abrupt transition (discontinuity) along some edge(s) of the shape. Flat shading should be evident there when drawn and lit with the provided Phong reflection model. Explain in your README where to look for it.
- [2 points] Real-time speed. Make sure that your animation runs at the same speed i.e., one simulated second corresponds roughly to one real second regardless of the machine your program runs on (even a ridiculously fast one from the future). The variable animation_time inside the graphics_state object is your gauge of the passage of real seconds.
- [2 points] Display the frame rate of your program on the graphics window, taking advantage of the update_strings() method that each Scene_Component object runs each frame.
- [4 points] Creativity (story, aesthetic style, etc).
- [4 points] Complexity and impressive underlying mechanics.
- [5 points] Overall quality: Fluidity of object and camera motion, attention to detail in scene construction, etc.
- [For participation in the in-class screening and contest] Make and submit a movie of your animation (length 90 sec or less) using your favorite screen recording application (e.g., camstudio/quicktime). If your program is interactive, submit a video of it being used. Make sure you encode your movie to be below the CCLE limit of 100MB, and observe the 90s limit.

Special instructions:

- Your video must be only of what your program can output given certain user inputs. The video should not be edited.
- Note that creativity and quality amount to 9 points. You will not get a perfect score if your scene is complex, but not creative.
- You must do the assignment from scratch using the provided template code (modification OK). JavaScript libraries that do not make GPU calls are OK. Using outside help or any piece of code from any previous offerings of the course will be considered plagiarism.
- Please refrain from the following scenes that have been submitted too many times: 1. A rocket launches off the ground. 2. A basketball free throw / soccer penalty kick. 3. Baymax finds a thing. 4. A tree grows. 5. A disco dancer shows off. 6. Planets orbit a sun. 7. The bee from Assignment 1 is visible for more than 1 second. 8. Walls trace out a standard maze.
- Some examples of animations made for previous offerings of this course are at: http://web.cs.ucla.edu/~dt/courses/CS174A/animations/assignment2-best-16s/

Submission guidelines:

- Submit your movie with the name *<uid>.ext (e.g., 802870392.mp4)*, where *<uid>* denotes your 9 digit bruin ID and *ext* can be any common video format*.*
- Submit all the files related to your project in a single archive named *<uid_yes/no>.zip (e.g., 802870392_yes.zip)*. Here the *<yes or no>* indicates whether or not you want us to host your work as an example for future quarters' students.
- Include in the top level of your *<uid_yes/no>.zip* archive a *README.TXT* file that summarizes your animation, describes your custom shape and where to spot its discontinuity, and explains anything else that might be helpful to know in grading and understanding your project. Make it presentable -- if you've chosen "yes,", it will accompany your work. So will your name and identifying info if you leave it there, so include or exclude it according to what you want publicly displayed.