Shaders - Advanced Homework Assignment

(check lcm.liacs.nl for duedate)

This assignment should only be done by students who have not been assigned a paper to present.

Goal: Increase understanding Open GL Shaders

No teams allowed. All source code should be written to compile and work on Linux at LIACS, e.g. room 302

This assignment uses as a starting point the Preworkshop 2, which you have become an expert on. As with the other assignments, the general plan is to give you code which works (or that you have used already) and expand upon it. In Workshop 3, you explored and implemented diverse lighting (eg. cartoon and diffuse) on a Teapot model. In Workshop 4, you did simulations of static water, specular reflection, earthquakes and waves. Unlike the workshops which you had about 2 hours, for this assignment you have 2+ weeks and it is significantly more work.

(1) Basics (medium difficulty)

Note that completing "Problem (1) Basics" is sufficient for a pass (grade of 6)

(1a) User viewpoint movement (medium)

(you may have already done this as a bonus problem earlier) – implement normal viewer viewpoint movement similar as Workshop 1 with the keys “i” (move forward), ”j” (rotate left), “k” (move backwards), “l” (rotate right), “s” (up), “x” (down)

(1b) Driving the Teapot (medium)

This part of the problem is inspired by the Landspeeders in the movie Star Wars, which are essentially hovercrafts that glide above the ground. Your goal is to use the Teapot model as the Landspeeder where the spout of the Teapot is the front of the vehicle. First, adjust either the size of the terrain or the size of the Teapot so that you have a space to drive in. Second, toggle the Teapot mode on and off with the key “t”. When the Teapot mode is on, the Teapot should be shown (off = no Teapot) and the following controls should work

```
case ‘t’:  //toggle Teapot mode on/off
case ‘2’: //when Teapot mode is on, toggle between normal viewer perspective and the perspective
          of a Teapot driver (facing forward) while sitting in the teapot.
case 'w':  //apply force to move the Teapot vehicle forward
case 'q':  //rotate the teapot left
case 'e':  //rotate the teapot right
```

Use the Newtonian force equations to make the Teapot movement realistic. Several important factors are

- There are no wheels so the Teapot should not grip the ground – think roughly of spaceship movement but where there is also friction/drag from air to slow it down over time.
- When going downhill (e.g. side of a mountain), the Teapot should be angled downwards roughly parallel with the ground underneath
- When going over a small hill or cliff or bump, the Teapot should work with gravity and do a jump.

Grading will emphasize showing knowledge of forces and physics realism.

(2) Pyramids and Spheres (medium to high difficulty)

Note: for drawing the pyramids and spheres, please feel free to use any implementation/code you prefer and as usual just give credit to the original writer/coder. The main goal of this problem is (2c). *Grading will emphasize realism of the explosion.*

(2a) Particle Cannon and Pyramids (medium)

In a visually similar way to Workshop 1, implement the particle cannon with 1000 triangle-pyramids with the “f” (fire) and “g” (gravity) keys working.

(2b) Lighting and Spheres (medium)

Implement spheres (see attached sphere tutorial for some different methods of drawing spheres – note its not as easy as one might think) and then light them with ambient, diffuse and specular lighting (visually similar to

homework 2 but using GLSL shaders). Add the “p” key which should toggle between triangle-pyramids. When you do this toggle, you may change the number of spheres so that you get reasonable frame rates.

(2c) Deathstar Explosion (medium-high)

When the user hits the 'b' (boom) key, make each sphere explode in a visually similar way to the following image (you can think of it as 2 kinds of explosions: one explosion is on the plane of a disk and a second slower explosion in “all directions”/spherical)

Keyboard assignments (it is necessary to implement these):

```
case 's': // up   (viewer is moving up)
case 'x': // down   (viewer is moving down)
case 'j': // left  (viewer rotates left)
case 'l': // right (viewer rotates right)
case 'i': // forwards (viewer is moving forwards)
case 'k': // backwards  (viewer is moving backwards)
case 'f': // fire (fire the particle cannon)
case 'g': // toggle the gravity key on/off
case 'd':   //toggle diffuse lighting
case 'a':  //toggle ambient lighting
case 'm':  //toggle material properties
case ‘t’:  //toggle Teapot mode on/off
case ‘2’: //when Teapot mode is on, toggle between normal viewer perspective and Teapot driver
          facing forward perspective
case 'w':  //apply force to move the Teapot vehicle forward
case 'q':  //rotate the teapot left
case 'e':  //rotate the teapot right
case 'p':  //toggle between pyramids and spheres
case 'b':  //Deathstar explosion
```

Submission Checklist

Place in a ZIP file the following and submit on the LML Course Manager zip file where the top level contains a directory called **firstname.lastname.project** Inside the directory are:

(1) a file named "**AnswerJournal.txt**" which should list
- Your name and student ID
- The name of the machine you had it working on. e.g. a machine in rm. 302.
- Mention which of the problems you solved.

--> *"make" - Compiling should always be done in this class using "make" (If you received special permission to use a different method, you should mention it here and who you spoke to.)*

(2) The source code, Makefile and

(3) Working executable of your solution

Good luck!