Assignment #3:

This assignment is due in two phases:

- Parts I-IV are due Friday Oct 21
- Part V is due Friday Nov 11 (after break)

To complete this assignment, you will email your solutions to the TA prior to the deadline. You may work in pairs. We recommend each member of the pair attempt the theory questions separately at first to maximize learning and then come together to write up an answer. Be sure the full names and logins of each student are on all files.

Part I-III: Type Classes

The file a3-typeclasses.lhs associated with this assignment is a literate Haskell file (find the file on the course website, assignments and notes page). Follow the directions within the file, filling in details where necessary. Make sure your final file type checks and compiles. Submit the file.

Part IV : Animate the Solar System

Read the literate Haskell file a3-solar.lhs. Fill in the details and create a solar system animation.

Part V: Create Your Own Animation

Create your own animation of some kind. Submit a file movie.hs. That file should contain a definition main :: IO () that depends on the SOE libraries and that when run creates the main animation you would like to present. (It should be possible to stop the animation with the space bar.) If your solution involves multiple files and modules, submit those. Also submit a README file describing the structure of your solution. The best solutions will create elegant, reuseable infrastructure or libraries. (It seems likely that such infrastructure would be divided in to its own modules.) Be sure to explain the reuseable infrastructure in your README. If you want to produce multiple demos that show off your infrastructure, create definitions main1, main2, etc. Give instructions in your README. Ideas:

- Animate your fractal in some interesting way. Possibly play with time as an additional axis for recursion. Better yet, develop infrastructure for animating fractals in general.
- Animate some sorting algorithms: http://www.sorting-algorithms.com/ (Create infrastructure to animate any kind of sort and show that it works with 3 different kinds of sorts on lists.)
- Animate some search tree algorithms or graph algorithms (or see COS 126, 226 demos)
- Animate the towers of Hanoi, a DFA, percolation, or anything else you've seen in another class
- Create rotating galaxies of stars. Simulate a physical system with objects and gravity.
- Create stick figures or stick animals and make them do something.