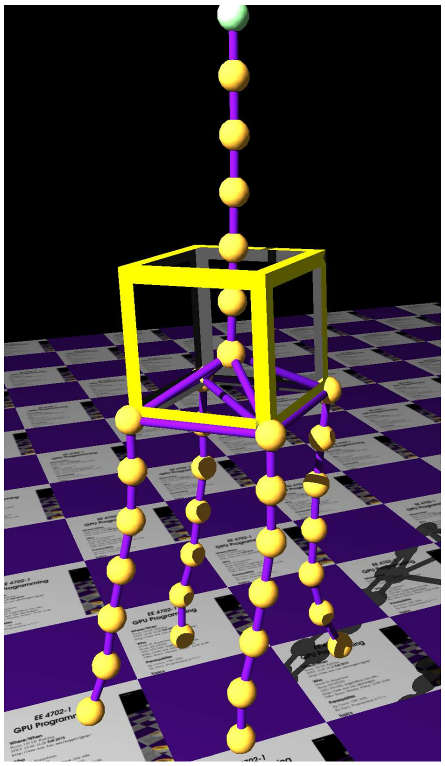
LSU EE 4702-1 Homework 2 Due: 21 September 2016



Problem 0: Follow the instruction on the `http://www.ece.lsu.edu/koppel/gpup/proc.html` page for account setup and programming homework work flow. Compile and run the homework code unmodified. It should initially show some objects on the left and a swinging thing of beads on the right. The screenshot to the right shows what the thing should look like after the assignment is solved correctly.

Press digits `1` through `2` to initialize different scenes, the program starts with scene 2. Scene 1 shows a plain string of beads. *Promptly report any problems.*

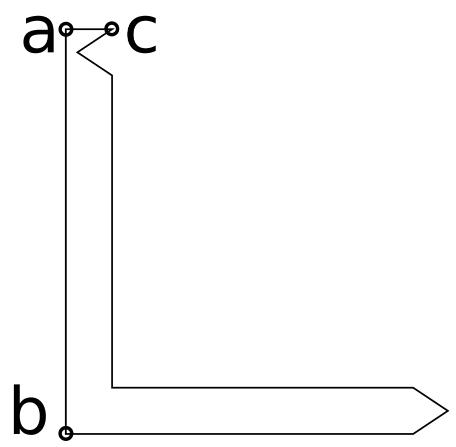
Use key `h` to toggle between the first (head) ball being locked in place and free. Use key `t` to do the same for the last (tail) ball.

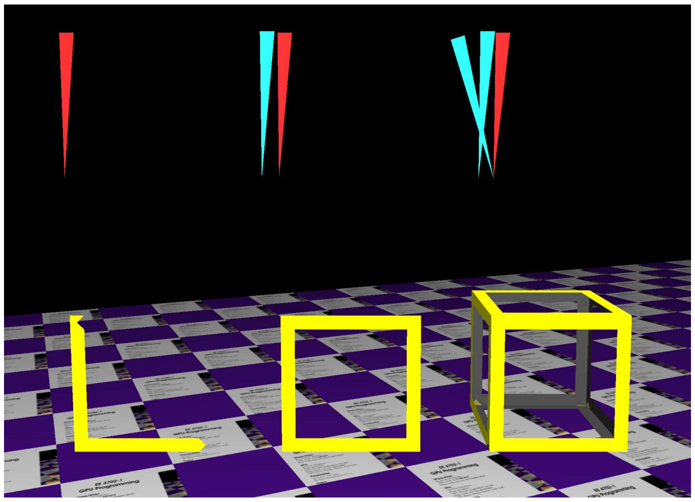
Initially the arrow keys, PageUp, and PageDown can be used to move around the scene. Press (lower-case) `b` and then use the arrow and page keys to move the first ball around. Press `l` to move the light around and `e` to move the eye (which is what the arrow keys do when the program starts).

Look at the comments in the file `hw02.cc` for documentation on other keys.

There is a problem on the next page.

Problem 1: The scene contains a swinging thing of beads on the right-hand side (see the screenshot above) and six *samples* on the left-hand side (see the screenshot below). Both screenshots show the appearance after the problem is solved correctly. The six samples are arranged into two rows of three. The upper row is generated by code in `render_it_demo`, that code does not need to be changed for this assignment. The lower row is generated by code in `render_it`, that routine should be changed. The leftmost sample is generated when the `version` argument of `render_it` is 0, the middle sample is generated when `version` is 1, and the rightmost sample is generated when `version` is 2. After the assignment is solved correctly, the leftmost sample should look like the shape illustrated below, the middle sample should look like a square (or frame), and the rightmost sample should be a cube. Routine `render_it` with `version` set to 2 is also used to render the object about halfway down the thing of beads.





(*a*) The illustration on the upper left shows a shape with points labeled, a, b, and c. Routine `render_it` (in hw02.cc) has three coordinate arguments by the same name. Modify `render_it` so that when argument `version` is 0, it draws the illustrated shape. The routine currently draws a triangle with a, b, and c as vertices (shown in the upper row of the screenshot). The shape can be drawn using individual triangles or using a triangle strip.

The labeled vertices must correspond to the coordinates passed to `render_it`. The other vertices must be chosen so that when rotated it fits together to form a square (see the next subproblem). The front faces of the triangles must be facing the user, and should be yellow. The back faces should be gray.

Remove the original code in `render_it` (if it's not needed).

(*b*) Further modify `render_it` so that when `version` is 1 it draws the shape twice, once in its original orientation, and a second time rotated so that it and the original form a square. Perform the rotation by transforming the coordinates that you found for the previous subpart. Don't individually find coordinate locations for the rotated shape. The routine has sample code showing how to transform coordinates placed in a list.

Try to render the entire square in a single rendering pass. (That is, one list should have the coordinates for the original and rotated shape.)

(*c*) Finally, modify the `render_it` so that when `version` is 2 the vertices from the previous subpart are transformed so that they form the faces of a cube. This can be achieved by rotating the vertices from the previous part. Be sure to appropriately rotate the normal too. It is okay to do a render pass for each face.

Rotate the faces so that the outsides are yellow and the inside is gray.