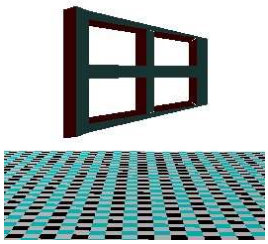
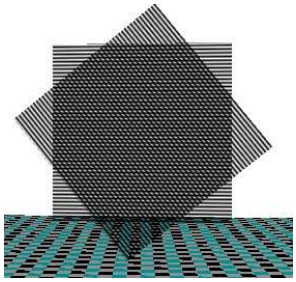
The scene is composed of 5 things that rest on a checked floor pane. The five objects are:

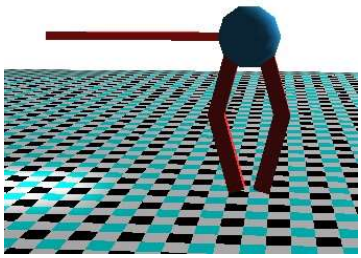
1. An Ames window that is continually rotating that is defined by coloured quads.



2. Two quad panes where one continuously is rotating, creating moori pattens.

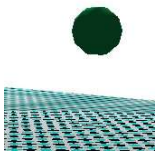


3. A ball on two legs that is moving up and down and has a retractable nose that is moving out and In independently of the legs, and on this nose is a spot light that is pointed down that moves with the nose.



4. A textured rotational sweep surface based of a sign wave for the base patten. That is continually changing the base curve.

5. Ball that is created and moves using basic physics until it is ever too old or it reaches the ground where it will then, be replaced with another ball of different velocity at the centre.



For the ball with legs, the movement of the legs was defined defining the angle and with a variable that is changed during animation and using basic trigonometry to define the position of the shear specifically it was calculated as 2 * cos (legAngle) * legLength (as there are two bits of the leg each with an angle of legAngle around the centre. Additionally the top parts of the leg were drawn by translating them by -sin(legAngle) * legLength in the x direction and cos(legAngle) * legLength in the y direction.

The sweep surfaces base curve was calculated using sin(const*y+val)/2 where vel constantly changed because of animation. Then the normal of the base curve was calculated by using the -const * cos(const*y+val) as the y direction and 2 as the x direction. As const * cos(const*y+val)/2 is the drivtave of sin(const*y+val)/2 with respect to y.

The ball was animated by defined the position as 3 floats and it's velocity as 3 floats, when it is loaded the position is set to a constant position and it velocity is set to some random value, It then moves base on this velocity, every animation fame, and it also makes the velocity in the y direction less every animation to simulate gravity. When the ball reaches 0 in the y direction it will be put back to its origin and it's velocity will be randomised. It also doses this if it goes over its max lifespan.

Controls:

The app is controlled by the arrow keys with forward moving you forward and back moving you back, well left and right rotate your view, to left and right respectively. The number buttons also are controls with 1 bringing your view to rotate around the ball with legs, 2 bringing your view to rotate around the Ames window, 3 bringing you to rotate around the moori patten, and 0 putting you back where you were.

I made this program in visual studio in windows, the source files and image files are stored in src, and the header files are stored in include, the main file is source.cpp. the library it uses are freeglut.h, time.h, glm/glm.hpp, math.h, iostream, fstream.

With the exception of loadTGA which I got from learn, I made all source and header files my self-using visual studio. I made both patten.tga and tile.tga myself using gimp, well I got rock.tga from https://www.textures.com/.

The equations used where derived by my self-using knowledge of basic trigonometry, physics and derivatives that I gained in high school.

Declaration

I declare that this assignment submission represents my own work (except for allowed material provided in the course), and that ideas or extracts from other sources are properly acknowledged in the report. I have not allowed anyone to copy my work with the intention of passing it off as their own work.

Daniel Lowe

89907250

27/03/2023