Assignment 1—Hello 3D World

Due: Thursday September 7 at 11:59pm

1. Implement a 3-D vector class that stores 3 floats and has the following functionality:
   1. Constructor from 3 floats
   2. Read/write access through square brackets operator “[]” (operator)
   3. Addition of two vectors (operator)
   4. Subtraction of two vectors (operator)
   5. Dot product (operator)
   6. Cross product (operator)
   7. Multiplication by scalar (operator)
   8. Division by scalar (operator)
   9. Normalization
   10. Length computation
   11. Rotation of “this” point about arbitrary axis
   12. Rotation of “this” vector about arbitrary direction
   13. Output to ostream, input from istream (operator)
2. Implement a 3x3 matrix class that stores 3 3-D vectors as rows and has the following functionality:
   1. Constructor from 3 3-D vectors
   2. Read/write access to rows through square brackets operator “[]” (operator)
   3. Function to get column
   4. Function to set column
   5. Multiplication with 3-D vector (operator)
   6. Multiplication with another matrix (operator)
   7. Inversion
   8. Transposition
   9. Function to set matrix as rotation about principal axis by theta degrees
   10. Output to ostream, input from istream
3. Implement a framebuffer class that stores unsigned int pixels and the resolution of the frame buffer and that has the following functionality:
   1. Constructor from input resolution
   2. Set all pixels to given color
   3. Set one pixel to given color
   4. Drawing of axis aligned rectangle
   5. Drawing of circle
   6. Drawing of disk
   7. Load / save from tiff file
4. Implement a planar pinhole camera class with the following functionality
   1. Constructor from input resolution and horizontal field of view
   2. Projection of 3D point
   3. Translation
      1. Front-back
      2. Left-right
      3. Up-down
   4. Rotation
      1. Pan
      2. Tilt
      3. Roll
5. Demonstrate your code.
   1. Choose a point and an arbitrary axis and rotate the point full circle by 1 degree increments (i.e. 360 steps). Draw the point in the framebuffer as a circle with a radius of 5 pixels. Make sure the camera is positioned such that the point stays on screen at all times, and such that the trajectory of the point is an ellipse (and not a segment or a circle). Also show the axis, by drawing it one 3D point at the time.
   2. Render a 12s 30Hz 720p video sequence illustrating your rotating point. The video will have one frame for each position, i.e. 12x30=360 frames in total. The video file should be in a popular format. Use the video making software of your choice.
6. Turn in via Brightspace one zip archive that contains
   1. Source code
   2. Executable
   3. Video file

© Popescu 2023, popescu@purdue.edu