CS 535, Fall 2009

Due Wednesday September 9 at 6am.

Assignment 1—Basics

Develop an application with the following functionality:

- Load, display, modify, and save a digital image. The image files loaded and saved should be in the tiff format (www.libtiff.org). Your program and other image viewers should be able to load the tiffs you save. The modification operations required are drawing and erasing a big white point represented with a 7x7pixel square.
- Rotation of a point about an arbitrary axis.

The graphical user interface (GUI) should have the following capabilities:

- Window of 640x480 pixels used as drawing canvas.
- Text input of file name, load image button, save image button. The loaded image should be shown in the window. Work with 640x480 images such that you do not need to resize the window or the image.
- Floating point input for arbitrary axis point (3 floats), for arbitrary axis direction (3 floats), and for point to be rotated (3 floats).
- Radio button group for the viewing direction with 3 buttons labeled *xy*, *xz*, and *yz*. Default selection should be *xy*.
- Render button that triggers the animated rotation of the point about the arbitrary axis and the display of the current position of the point. The point should be displayed on top of the image loaded as background. The old position of the point should be erased. Choose a rotation angle increment that produces reasonable rotation speeds. The program should automatically find a mapping from the selected 2D plane to the display window such that the entire trajectory of the point is visible and such that the bounding box of the trajectory is comparable to the display window.

Additional instructions

- Implement your own 3D vector and 3x3 matrix classes in order to solve the rotation about an arbitrary axis. It is an important step for building the infrastructure needed for future assignments.
- We have only limited TA resources so we have to all use the same platform for efficient grading. Use the Microsoft (MS) Windows operating system and the MS Visual C++ (VC++) programming environment.
- I hope to find time to actually implement some graphics techniques in class. I will be using the MS operating system, the MS VC++ programming environment, the OpenGL graphics API (www.opengl.org), the FLTK user interface library (www.fltk.org), and the Cg shading language

(www.nvidia.com). I will post all code developed in class, and you are free to use it in your assignments. Therefore it is a good idea, especially if you are new to graphics, to use these same software tools.

- No OpenGL functionality should be used in the assignment, except glDrawPixels for displaying your frame.

Extra credit

Only attempt extra credit once the required part is working. In your Readme.txt or Readme.doc file include a list of extra credit features attempted.

- Draw a circular point, shape of point selected through GUI, 1%
- Draw a trail behind the point, length of trail selected set through GUI, 1%
- Load images of arbitrary size and resample them non-uniformly to fit the 640x480 window, 3%.
- Extend your code to draw regular planar polygons (i.e. equilateral triangles, squares, pentagons, hexagons, etc), 2%

Turn in instructions

Turn in your work in an archive submitted via Blackboard Vista. The archive should contain:

- Source code and VC++ solution (please Build->Clean Solution to minimize submission size)
- Image files used
- External libraries used
- Code should compile, link, and run
- A Readme.txt or Readme.doc file that lists special GUI features and extra credit attempted