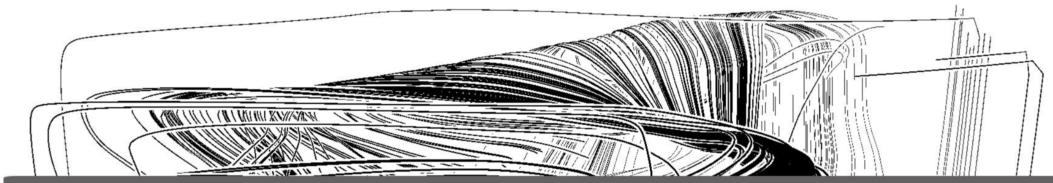

CSCI 3090: Computer Graphics and Visualization

CSCI 3090U Assignment One (6%)

Modeling and Curves

Due: February 5, 2016 (11:59pm)

1. [60 marks] Reading and Displaying a 3D Model

This part of the assignment will give you some experience with manipulating and displaying polygonal models. The bunny is stored in a standard format called PLY. This format is used for many large polygonal models, so a program that displays the bunny will display any of these models as well. The PLY file format is not that complicated, but writing a set of procedures to read it is not trivial and really not related to the main theme of this assignment, so we have provided you with a set of procedures that can be used to read models in this format. The main procedure is called `readply`, and this is the only procedure that you will need to call. The single parameter to this procedure is the name of the file where the model is stored. This procedure returns a `ply_model` structure that contains the vertices and polygons in the model. This data structure is documented in the `readply.h` include file. *Note that these models do not have normal vectors.*

The first step in this assignment is to read the model and just display the bunny in a solid colour [15 marks]. This shows that you have read and interpreted the model correctly. Next you need to compute the normal vectors for the model. You can do this by first computing the normal for each polygon, and then using these normals to compute an average normal for each vertex. You can use the algorithm presented in class to do this [25 marks]. Once you have computed the normals, display the bunny using lighting similar to what we have done in the laboratories and examples [10 marks].

Note that in these models the Y axis is up, so you will need to change the viewing transformations in order to display them correctly. Rotate the bunny about the vertical axis to show the complete model [10 marks].

Include screen shots, source code, and a discussion in your answer.

2. [40 marks] Curves

In this part of the assignment you will fit a piecewise cubic curve to a set of input data and then use OpenGL to display it.

The input to this program is stored in the file `points.txt`. The first item in this file is the number of points in the file, followed by three numbers (the x, y and z coordinates) for each of these points. Use these points as the control points for a piecewise cubic Cardinal spline and compute the polynomial coefficients (a_0, a_1, etc.) for each piece.

Then draw the curve by calculating points along the curve yourself (i.e. do your own sampling). Display the control points along the curve. Try different values for the tension parameter to determine its impact on the curve's shape. Provide screen shots of several different tension values, or render several tension values simultaneously with different colours.

Include screen shots, source code, and a discussion in your answer.

Bonus [up to 10 marks]: Describe in your report how to use the calculated coefficient vector and the Bezier constraint matrix to convert each piece into a cubic Bezier curve [5 marks] and adapt your program to evaluate the Bezier curve [5 marks]

General Assignment Guidelines

Individual Work

This is an individual assignment. While you may discuss your progress with your classmates, each member of the class is expected to hand in their own work.

Please include this statement on the cover page of your report:

"I, <insert name>, certify that this work is my own, submitted for CSCI 3090U in compliance with the UOIT Academic Integrity Policy."

In this course, you MAY NOT review, copy or otherwise use any graded academic assignments from prior semesters or other sections of this course, in written, electronic, or verbal form, used in whole or part, including formatting of any assignment.

Report Formatting

Please hand in the following with every assignment in this course:

- A written report (PDF only) describing your solution, highlighting any features of your source code which you would like to bring to the instructor's attention, describing any known problems, answering any questions posed in the assignment handout, and confirming the academic integrity statement. This report should be well-formatted, clearly organized, and use correct grammar and spelling.
- Source code which can be compiled in Windows. If your code has special compilation requirements, include them in a `readme.txt` file.
- Images you produce should be embedded in your report. Keep a high resolution copy available in case it is requested for clarification.