CS510 Midterm, Spring 1998

Name: _____

ID # _____

Question	Pts	Max
1		10
2		20
3		20
4		30
5		20
Total		100

1. True/False Questions. Circle the appropriate answer and elaborate if you feel necessary. (10 Points)

(a) `true/false`: The look and feel of a GUI is prescribed by the X Windows System, specifically by the X-library.

(b) `true/false`: A Bezier surface can intersect itself.

(c) `true/false`: Colors in X-windows must always be specified in terms of a single index into a colormap.

(d) `true/false`: Double buffering in combination with z-buffered rendering requires allocation of two color frame buffers but only one z-buffer.

(e) `true/false`: Two adjacent (sharing a common face in 3D space) cubic volumes representing leaf nodes in an Octree must also be adjacent (sharing a common parent node) in the Octree.

2. Short Answer Questions. (20 Points)

Each of the following should require only a brief answer.

a) Explain the X Windows Systems graphics context and why it is kept on the X-server.

b) Are there any control points for a Bezier Surface for which the point is guaranteed to lie on the surface?

c) Name one significant advantage of an Octree as a solid modeling representation when compared to a polygonal (boundary) representation. Name one significant disadvantage.

d) Explain in simple English the difference between object-space and image-space visible surface determination.

3. Visible Surface Determination. (20 Points)

There are circumstances where the simple List Priority Algorithm will produce a correct 3D rendering of the visible surfaces in a scene. Conversely, there are other cases where it will fail.

a) Describe in English how the List Priority Algorithm works. (7 Points)

b) State a condition sufficient to guarantee that the List Priority Algorithm will produce a correct rendering when drawing multiple faces. (6 Points)

c) Draw a simple example for which the List Priority algorithm will fail. (7 Points)

4. Bicubic Surfaces. (30 Points)

Perhaps the simplest way to write a Bezier Surface is in terms of three 4x4 geometry matrices, $\mathbf{G}_x$, $\mathbf{G}_y$ and $\mathbf{G}_z$.

a) Give the general linear algebraic expressions for the three components of the Bezier Surface in terms of $\mathbf{G}_x$, $\mathbf{G}_y$ and $\mathbf{G}_z$ below in terms of **S**, **T** and **M**. (8 Points)

$x(s,t) =$

$y(s,t) =$

$z(s,t) =$

b) For the x component expand the matrices to show the elements of **S**, **T**, **M** and $\mathbf{G}_x$. Feel free to pick whatever integers you like for $\mathbf{G}_x$. Do not actually do any of the matrix multiplications. (7 Points)

$x(s,t) =$

c) There are some particularly simple geometry matrices which in turn lead to simple forms for the parametric components $x(s,t)$, $y(s,t)$ and $z(s,t)$. Provide the geometry matrix $\mathbf{G}_x$ which leads to the following parametric equation. (*you can answer this question by inspect, but you will be called to demonstrate the correctness of your answer in part* d) *below*) (8 Points)

$$x(s,t) = 3t + 1$$

$$\mathbf{G}_x =$$

d) Prove that your matrix $\mathbf{G}_x$ above is correct by carrying the necessary multiplications to verify the result. (7 Points)

5. Solid Modeling - Constructive Solid Geometry. (20 Points)

a) Regularized set operations are necessary because the combination of solids using standard set union, intersection and difference can create new sets which are not solids. Specifically, there are three well know geometric forms which can result when two solids are combined using standard operations. State what each of these three forms are and illustrate with an example how each might be produced. In your examples, draw both of the solids being combined and indicate the set operation being used to combine them. (10 Points)

b) Draw a diagram of the CSG tree that could be used to make the 2D solid object model shown on the right using the primitives shown on the left. Use the symbols

$\cup^*$, $\cap^*$ and $-^*$ to indicate the operation carried out at each node in your tree. For the sake of this illustration, do not worry about the fact that in practice you would also need to specify a 2D transformation to place the two solids being combined into proper relative positions. It is sufficient just to show what primitives are being combined and how. Do include next to each node a small sketch of the solid which is produced by the regularized set operation at that node. Obviously primitives may be used more than once. (10 Points)