CARDIFF UNIVERSITY
EXAMINATION PAPER

Academic Year: 2003/2004

Examination Period: AUTUMN

Examination Paper Number: CM0304

Examination Paper Title: Graphics

Duration: 2 hours

Do not turn this page over until instructed to do so by the Senior Invigilator.

Structure of Examination Paper:

There are THREE pages.

There are FOUR questions in total.

There are no appendices.

The maximum mark for the examination paper is 100% and the mark obtainable for a question or part of a question is shown in brackets alongside the question.

Students to be provided with:

The following items of stationery are to be provided:

ONE answer book.

Instructions to Students:

Answer THREE questions.

The use of translation dictionaries between English or Welsh and a foreign language bearing an appropriate departmental stamp is permitted in this examination.

1. Graphics Pipeline Architectures

(a) Describe the principle of graphics pipeline architectures. Draw a simple diagram to illustrate the general operation of a pipeline architecture (e.g. consisting of two components). How does the graphics pipeline speed up graphics computations?

[8]

(b) Which factors determine the performance of pipeline architectures? What does this mean for the design of an efficient graphics pipeline?

[7]

(c) A simple way to reduce the number of calculations in a graphics pipeline when displaying polygonal meshes is to use a mesh primitive, such as a triangle strip. A triangle strip is a sequence of three or more vertices, in which every consecutive set of three vertices $[(v_1,v_2,v_3),(v_2,v_3,v_4),(v_3,v_4,v_5)$, etc.] defines a triangle. Assume you intend to process n three-dimensional triangles by a graphics pipeline. What is the maximum speedup you could obtain in the front-end subsystem (transformations on the geometry) by converting triangles into triangle strips? Compute the number of multiplications and additions required to perform a transformation on a set of separate triangles and a triangle strip to support your answer.

[10]

2. Polygons and Orientation/Sidedness

(a) What conditions does an OpenGL polygon primitive have to fulfil?

[6]

(b) Given n three-dimensional points $p_1,\ldots,p_n$ in this sequence, what is the standard convention in OpenGL to determine if a viewer looks at the front or the back side of the polygon? Assume the unit vector v points towards the viewer from point p_1. How can we determine if the viewer is looking at the front or the back of the polygon?

[9]

(c) Let $(x_0,y_0),\ldots,(x_{n-1},y_{n-1})$ be the n vertices of a convex, two-dimensional polygon. Derive a formula to compute the area of this polygon which can also be used to determine the orientation of the polygon. Explain how we can determine the orientation of a polygon from this formula. Note that the area of a triangle with vertices (x_1,y_1), (x_2,y_2) and (x_3,y_3) is

$$\frac{1}{2}\det\begin{pmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{pmatrix} = \frac{1}{2}(x_1y_2 + x_3y_1 + x_2y_3 - x_3y_2 - x_1y_3 - x_2y_1).$$

[10]

3. Convex Hull

(a) Carefully describe an efficient, incremental algorithm for finding the convex hull of a set P of n two-dimensional points. Assume that none of the points in P have the same x coordinate.

[15]

(b) Now let P contain the vertices of a simple, non-convex polygon in clockwise or anti-clockwise order. Explain how to modify the algorithm from (a) to compute the convex hull of the polygon more efficiently (find a more efficient method for the part of the algorithm from (a) which determines the time order).

[5]

(c) What is the time order of the convex hull finding algorithms from (a) and (b)? Why does the algorithm from (b) have a lower time order?

[5]

4. Phong's Illumination Model

(a) List the different types of light used by Phong's illumination model to model the light emitted by a point light source and explain their basic properties. For each type of light, on what does the intensity of the light observed by the viewer depend on?

[9]

(b) Using Phong's illumination model, we wish to render a scene containing a single point light source, which emits monochromatic light of all types at the same intensity L. For each type of light, explain how to compute the intensity of the light observed by the viewer, which is reflected at a point p of the polygon. (Assume that the intensity L does not depend on the distance from the light source, the polygon reflects R percent of the light independently of the light type, and all vectors used by Phong's illumination model are known for the point p).

[7]

(c) When using Phong's illumination model in graphics pipelines, why do we not consider light sources being obscured from the surface by other surfaces? Describe the principle of a method for simulating simple shadows using Phong's illumination model (note that shadows are generated by projections). For what type of scenes is your method suitable? What are its limitations?

[9]