INSTRUCTIONS TO CANDIDATES

(a) The exam accounts for 50% of the total marks for the subject.

(b) Candidates should attempt all questions 1 to 5 and EITHER question 7 or 8.

(c) Marks for each question are shown. Total marks for the exam is 60.

Question 1

(a) State whether each of the coordinate systems (i)-(iv) below is right or left handed. Assume all horizontal and vertical axes lie in the plane of the page and diagonal axes point out of the page.

(b) A planar polygon has the five vertices A(4,4,-1), B(8,4,-3), C(8,10,6), D(6,12,15) and E(4,10,8). Work out the plane equation of the plane in which the polygon lies.

(c) A different planar polygon has the three vertices F(3,4,2), G(10,6,18) and H(7,9,15). Work out the plane equation of the plane in which the polygon lies.

(d) Which plane from (b) and (c) has the largest z value at x=5, y=5? Show your working.

(1+3+3+1 = 8 marks)

Question 2

(a) Explain the following line rasterization algorithms: (i) basic incremental and (ii) Bresenham’s. Discuss why Bresenham’s algorithm is more efficient, including assumptions about the relative cost of operations.
(b) Give an efficient scan-line algorithm for rasterizing polygons. Explain how it achieves its efficiency, including the major data structures it uses and how it exploits coherence.

(c) Show the state of the major data structures in the algorithm you gave as answer to (b) at scan lines $y = 5$, $y = 6$ and $y = 11$ for the polygon given below. Indicate the pixels which will be intensified on each of those scan lines.

(d) Discuss the simplifications to the algorithm you gave as answer to (b) if only convex polygons are to be handled.

(4+4+4+2 = 14 marks)

**Question 3**

(a) An important aspect of the Sutherland-Hodgeman polygon clipping algorithm is the identification of different *cases*, that is, different relationships between an edge of the polygon and an edge of the window. Give the different cases and the output in each case.

(b) Demonstrate your understanding of the Sutherland-Hodgeman algorithm by showing the steps the algorithm goes through to clip polygon $DEFGHIJK$ against the polygon $ABC$, as shown in the figure below. Start wherever you deem convenient. Include in your answer the case (from your answer to (a)) which applies for each edge-edge comparison.

(c) Show the calculations necessary to determine the case which applies when clipping the edge $DE$ against the edge $AB$.

(d) Can the Sutherland-Hodgeman clipping algorithm be used to clip the polygon $ABC$ against the polygon $DEFGHIJK$? Explain.

(1+4+2+2 = 9 marks)

**Question 4**

(a) What are homogeneous coordinates and why do we use them in computer graphics?
(b) Give the steps, including associated transformation matrices, required to build a combined transformation matrix (CTM) which transforms the object ABCDEFG into the object A'B'C'D'E'F'G' as shown below. Only multiply the matrices together if you have time.

![Diagram showing transformations](image)

(2+4 = 6 marks)

**Question 5**

A games programmer is writing a game which requires a two-legged “bug”. She is starting with a simple wireframe box model. The legs consist of three segments, the first of which is able to rotate around a hip joint, the second of which is able to rotate about a knee joint and the third of which is able to rotate around an ankle joint, as shown below. Rotation about each joint is restricted to be about the z-axis. The torso’s position is to be fixed whilst the segments in the legs are free to rotate around their joints. One rotation angle is stored for each joint. The depth of the torso and the legs is 2, centred on the y axis.

![Diagram showing joint rotations](image)

Assuming a function `glutWireCube(GLfloat size)` which draws a cube of side length `size` centred at the origin, write the following functions:

(a) `myWireBox(GLfloat width, GLfloat height, GLfloat depth)`

which draws a wire box at the current drawing position with the current drawing orientation in the current drawing colour

and

(b) `void display()`
which uses OpenGL modelling transformations, OpenGL modelview matrix stack manipulation functions and myWireBox to draw the bug. Assume the viewing parameters, window initialisation, etc., have all been correctly set-up. Your display function may call other functions.

(1+7 = 8 Marks)

**Question 6**

(a) Explain the difference between an image space and an object space hidden surface algorithm.

(b) BSP trees are one approach to the hidden surface problem. Draw a sequence of diagrams showing the BSP tree for the line segments shown below inserted in the order 1, 2, 3, 4, 5. Give the order in which the line segments in your BSP tree will be drawn to solve the hidden surface problem for a viewer positioned at P. Is the BSP tree algorithm an object space or image space algorithm?

(c) Discuss the z-buffer hidden surface algorithm. Is it an image space or object space algorithm? Discuss situations where the z-buffer approach is likely to be (i) inferior and (ii) superior to the BSP tree algorithm.

(2+3+3=8 Marks)

Answer either question 7 or question 8.

**Question 7**

Discuss 3D viewing including the terms: perspective projection, parallel projection, view volume and near and far clipping planes. Explain the OpenGL functions: glFrustum, gluPerspective, glOrtho and gluLookAt. Also explain the difference between viewing, modelling and projection transformations. Use diagrams to clarify your answer.

(7 Marks)

**Question 8**

Give and discuss a simple illumination (lighting) model that includes ambient, diffuse and specular components. Explain the difference between flat and smooth (Gouraud) shading. Discuss what needs to be done to smooth shade a polygonal model using OpenGL, including the use of the OpenGL functions: glShadeModel, glMaterial, glLight, glLightModel and glEnable. Also explain what is meant by two sided lighting. Use diagrams to clarify your answer.

(7 Marks)