北京郵電大學

Queen Mary University of London

# EBU7405 B

## Joint Programme Examinations 2020/21

**EBU7405 3D Graphics Programming Tools**

**Paper B**

**Time allowed 2 hours**

**Answer ALL questions**

## Only use Answer book to answer the questions.

**NOT allowed: electronic calculators and electronic dictionaries.**

**INSTRUCTIONS**

1. **You MUST use the supplied Answer book to answer the questions.**
2. You must NOT take **question papers or answer books**, used or unused, from the examination room.
3. Write only with a black or blue pen **and in English**.
4. Do all rough work in the answer book – **do not tear out any pages**.
5. If you use Supplementary Answer Books, tie them to the end of this book.
6. Write clearly and legibly.
7. **Read the instructions on the inside cover.**

**Examiners**

Dr Marie-Luce Bourguet, Dr Pengwei Hao



Filename: 2021_EBU7405_B

# Instructions

**Before the start of the examination**

1) Place your BUPT and QM student cards on the corner of your desk so that your picture is visible.

2) Put all bags, coats and other belongings at the back/front of the room. All small items in your pockets, including wallets, mobile phones and other electronic devices must be **placed in your bag in advance**. **Possession of mobile phones, electronic devices and unauthorised materials is an offence.**

3) Please ensure your mobile phone is switched off and that no alarm will sound during the exam. **A mobile phone causing a disruption is also an assessment offence.**

4) Do not turn over your question paper or begin writing until told to do.

**During the examination**

1) You must not communicate with or copy from another student.

2) If you require any assistance or wish to leave the examination room for any reason, please raise your hand to attract the attention of the invigilator.

3) If you finish the examination early you may leave, but not in the first 30 minutes or the last 10 minutes.

4) For 2 hour examinations you may **not** leave temporarily.

5) For examinations longer than 2 hours you **may** leave temporarily but not in the first 2 hours or the last 30 minutes.

**At the end of the examination**

1) You must stop writing immediately – **if you continue writing after being told to stop, that is an assessment offence.**

2) Remain in your seat until you are told you may leave.

**Question 1**

a) For each OpenGL function below, state if it is a "primitive generating" function or a "state changing" function. Justify your answers.

**[8 marks]**

i) glOrtho

**(3 marks)**

ii) glScale

**(2 marks)**

iii) glVertex

**(3 marks)**

b) Consider the code shown in **Code Box 1** on the next page.

**[11 marks]**

i) Briefly describe what appears on screen when executing this code.

**(2 marks)**

ii) Select four statements in this code that contribute to the creation of a smooth animation and explain their respective purpose.

**(8 marks)**

iii) Modify one line in the code to make the animation go faster.

**(1 mark)**

c) This question is about the OpenGL camera.

**[6 marks]**

i) What are the properties of the default OpenGL camera?

**(3 marks)**

ii) Write the arguments of the glOrtho function that correspond to the default camera.

**(3 marks)**

```
#include <GL/glut.h>
#include <math.h>

#define DEGREES_TO_RADIANS 3.14159/180.0

static GLfloat spin = 0.0;
GLfloat a, b;

void square()
{
    glBegin(GL_QUADS);
      glVertex2f(a , b);
      glVertex2f(-b , a);
      glVertex2f(-a , -b);
      glVertex2f(b , -a);
   glEnd();
}

void display()
{
    glClear (GL_COLOR_BUFFER_BIT);
    square();
    glutSwapBuffers ();
}

void spinDisplay (void)
{
    spin = spin + 5.0;
    if (spin > 360.0) spin = spin - 360.0;
    a = 0.5 * cos(DEGREES_TO_RADIANS * spin);
    b = 0.5 * sin(DEGREES_TO_RADIANS * spin);
    glutPostRedisplay();
}

int main(int argc, char** argv)
{
   glutInit(&argc,argv);
   glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);
   glutInitWindowPosition(500,0);
   glutCreateWindow("double buffered");
   glutDisplayFunc(display);
   glutIdleFunc (spinDisplay);
   glutMainLoop();
}
```

**Code box 1.**

**Question 2**

a) This question is about modelling.

**[5 marks]**

i) Name three classes of methods for geometric modelling.

**(3 marks)**

ii) Name four file formats for 3D models.

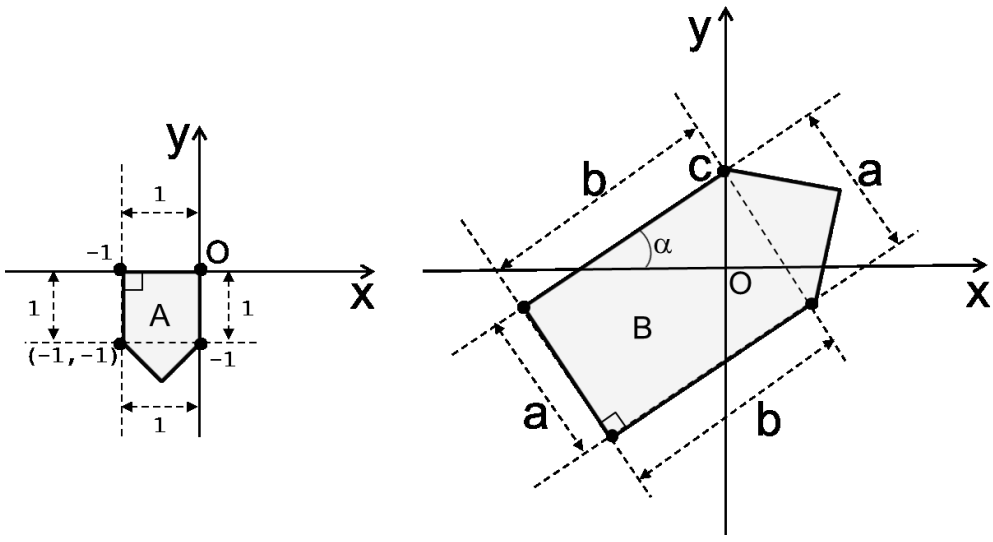
**(2 marks)**

b) This question is about geometric transformations.

**[11 marks]**

i) Consider Model A and Model B in **Figure 1**. Give a chain of the basic transformation matrices for translation, scaling and rotation which, when post-multiplied by the homogeneous coordinates of the vertices of Model A, will transform the vertices of Model A into their corresponding vertices of Model B, such that the vertex at $(0,-1)$ of Model A is transformed to the vertex at $(c,0)$ of Model B.

**(7 marks)**




**(Model A)** **(Model B)**

**Figure 1.**

ii) Compute the composite 2D transformation matrix for the transformations found in question i) above.

**(4 marks)**

c) Write the OpenGL functions that correspond to the three transformations illustrated in **Figure 2**. Estimate their parameters.

**[6 marks]**

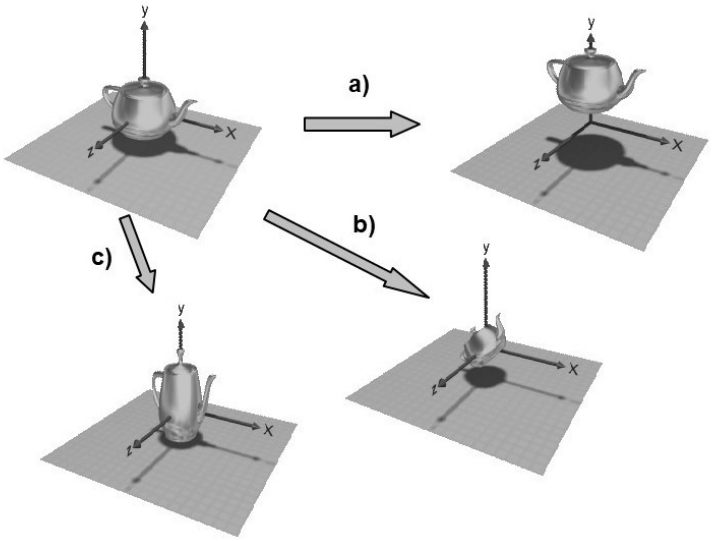



**Figure 2.**

d) What information do we need to consider for a lighting model?

**[3 marks]**

**Question 3**

a) This question is about lighting calculations in OpenGL. Describe the appearance (colour and shininess) of an object illuminated under the conditions declared in **Code box 2**.

**[6 marks]**

```
float light_ambient[] = {1.0, 0.5, 0.5, 1.0};
float light_diffuse[] = {0.0, 1.0, 0.0, 1.0};
float light_specular[] = {1.0, 1.0, 1.0, 1.0};
float material_ambient[] = {0.8, 0.3, 0.3, 1.0};
float material_diffuse[] = {0.0, 0.4, 0.0, 1.0};
float material_specular[] = {1.0, 1.0, 1.0, 1.0};
float shininess = 10.0;
```

**Code box 2.**

b) Referring to the similarities between WebGL and OpenGL, briefly explain the role of each of the following four WebGL application elements: (1) the canvas; (2) the WebGL context; (3) the VBO (Vertex Buffer Object); and (4) the IBO (Index Buffer Object).

**[9 marks]**

c) This question is about 3D modelling types and tools.

**[10 marks]**

i) What are the differences between solid and wireframe modelling?

**(4 marks)**

ii) What are NURBS?

**(3 marks)**

iii) In Blender, would you model a cube using solid, wireframe or NURBS? Justify your answer.

**(3 marks)**

**Question 4**

a) This question is about projection.

**[16 marks]**

i) Give the transformation matrix for a 2D window-to-viewport transformation. Use the variables given in **Figure 3**.

**(5 marks)**

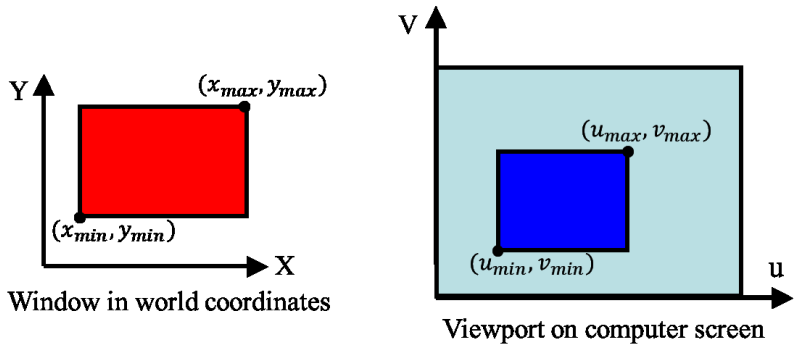



**Figure 3.**

ii) In the world coordinate system, if the view plane normal is [2,2,1], the view-up vector is [0,1,1], the view reference point is [5,6,7], and the distance from the centre of projection to the view plane is 5, find the transformation matrix for perspective projection.

**(11 marks)**

b) This question is about rasterisation.

**[9 marks]**

i) For a line from (x0, y0) to (x1, y1), give the implicit function as the line equation F(x,y)=0. Give the normal vector of the line and show which half-plane makes F(x,y)>0 and which half-plane makes F(x,y)<0.

**(5 marks)**

ii) Give a method to test if a point is inside or outside a triangle.

**(4 marks)**

**END OF PAPER**