



United International University (UIU)
Dept. of Computer Science and Engineering (CSE)
CSI 422, Computer Graphics Lab
Section A, Spring 2023, MHAK
Marks: 25 Time: 40 minutes
Final Exam, Set A

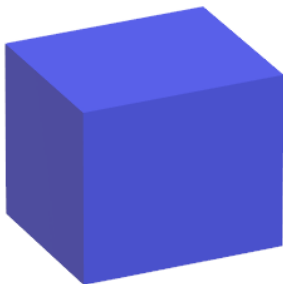
Name:

Student ID:

Answer all the questions below.
Follow the reference code at the last page
Each question carries 5 marks

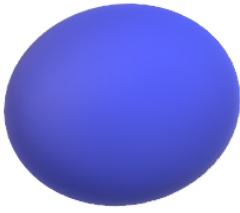
Q1. The length of the following cube is 8 centimeters. **How can you convert it into a cuboid containing length 16 centimeters, height 4 centimeters and width 2 centimeters?** Assume length, width and height are aligned with x,y and z axis respectively. Write the corresponding ***Display Code*** in OpenGL. The method for drawing cube is

void drawCube(double length)

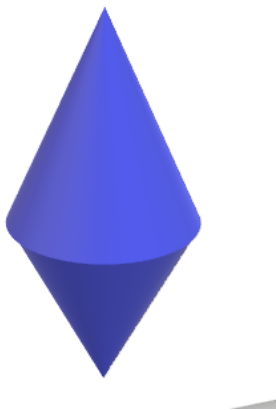


Q2. The radius of the following sphere is 10 centimeters and center is located at (3,4,5) point currently. Now you want to draw a new sphere with radius 15 centimeters at (7,8,9) point. **How can you draw this using multiple transformations?** Write the corresponding ***Display Code*** in OpenGL. The method for drawing sphere is

void drawSphere(double radius, int slices, int stacks)



Q3. There is a complex object like below. The upper cone has radius 5 cm, height 10 cm and the center of base is (0,0,0). **How do you draw this complex object by transforming the upper cone?** Write the corresponding *Display Code* in OpenGL. The method for drawing cone is
void drawCone(double radius, double height, int slices)



Q4. The radius and height of the following cylinder are 5 cm and 10 cm respectively. **How can you convert it into a pillar with radius 2 cm and height 100 cm?** Write the corresponding *Display Code* in OpenGL. The method for drawing cylinder is
void drawCylinder(double radius, double height, int slices)



Q5. There is a car like below at (0,0,0) initially. You want to move the car along the **Y** axis. **How can you animate moving the car? Write the corresponding *Display and Animate Code* in OpenGL.** The method for drawing car is

void drawCar()

and it's position value is stored as

float position = 0;



```

#include<windows.h>
#include <GL/glut.h>
#include <math.h>
#include "customObjects.h" //Contains all draw methods
struct point{
    double x,y,z;
};
void display(){
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glClearColor(0,0,0,0);    //color black
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    gluLookAt(eye.x, eye.y, eye.z,    0,0,0, 0,0,1);
    glMatrixMode(GL_MODELVIEW);
    drawAxes();
    drawGrid();
    //Display Code is written here normally
    glutSwapBuffers();
}
void animate(){
    //Animate Code should be written here normally
    glutPostRedisplay();
}
void init(){
    glClearColor(0,0,0,0);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluPerspective(80, 1, 1, 5000.0);
}
int main(int argc, char **argv){
    glutInit(&argc,argv);
    glutInitWindowSize(800, 800);
    glutInitWindowPosition(0, 0);
    glutInitDisplayMode(GLUT_DEPTH | GLUT_DOUBLE | GLUT_RGB);
    glutCreateWindow("My OpenGL Program");
    init();
    glEnable(GL_DEPTH_TEST);
    glutDisplayFunc(display); //display callback function
    glutIdleFunc(animate);
    glutMainLoop();          //The main loop of OpenGL
    return 0;
}

```



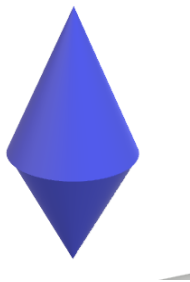
United International University (UIU)
Dept. of Computer Science and Engineering (CSE)
CSI 422, Computer Graphics Lab
Section A, Spring 2023, MHAK
Marks: 25 Time: 40 minutes
Final Exam, Set B

Name:

Student ID:

Answer all the questions below.
Follow the reference code at the last
Each question carries 5 marks

Q1. There is a complex object like below. The lower cone has radius 5 cm, height 10 cm and the center of base is (0,0,0). **How do you draw this complex object by transforming the lower cone?** Write the corresponding *Display Code* in OpenGL. The method for drawing cone is
void drawCone(double radius, double height, int slices)

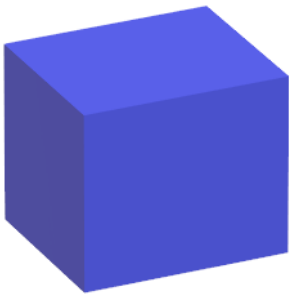


Q2. The radius and height of the following cylinder are 4 cm and 8 cm respectively. **How can you convert it into a pillar with radius 2 cm and height 80 cm?** Write the corresponding *Display Code* in OpenGL. The method for drawing cylinder is
void drawCylinder(double radius, double height, int slices)



Q3. The length of the following cube is 10 centimeters. **How can you convert it into a cuboid containing length 20 centimeters, height 2 centimeters and width 5 centimeters?** Assume length, width and height are aligned with x,y and z axis respectively. Write the corresponding ***Display Code*** in OpenGL. The method for drawing cube is

void drawCube(double length)



Q7. The radius of the following sphere is 5 centimeters and center is located at (1,2,3) point currently. Now you want to draw a new sphere with radius 15 centimeters at (3,2,1) point. **How can you draw this using multiple transformations?** Write the corresponding ***Display Code*** in OpenGL. The method for drawing sphere is *void drawSphere(double radius, int slices, int stacks)*



Q5. There is a car like below at (0,0,0) initially. You want to move the car along the **X** axis. **How can you animate moving the car? How can you animate moving the car?** Write the corresponding *Display and Animate Code* in OpenGL. The method for drawing car is

```
void drawCar()
```

and it's position value is stored as

```
float position = 0;
```



```

#include<windows.h>
#include <GL/glut.h>
#include <math.h>
#include "customObjects.h" //Contains all draw methods
struct point{
    double x,y,z;
};
void display(){
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glClearColor(0,0,0,0);    //color black
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
    gluLookAt(eye.x, eye.y, eye.z,    0,0,0, 0,0,1);
    glMatrixMode(GL_MODELVIEW);
    drawAxes();
    drawGrid();
    //Display Code is written here normally
    glutSwapBuffers();
}
void animate(){
    //Animate Code should be written here normally
    glutPostRedisplay();
}
void init(){
    glClearColor(0,0,0,0);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluPerspective(80, 1, 1, 5000.0);
}
int main(int argc, char **argv){
    glutInit(&argc,argv);
    glutInitWindowSize(800, 800);
    glutInitWindowPosition(0, 0);
    glutInitDisplayMode(GLUT_DEPTH | GLUT_DOUBLE | GLUT_RGB);
    glutCreateWindow("My OpenGL Program");
    init();
    glEnable(GL_DEPTH_TEST);
    glutDisplayFunc(display); //display callback function
    glutIdleFunc(animate);
    glutMainLoop();          //The main loop of OpenGL
    return 0;
}

```