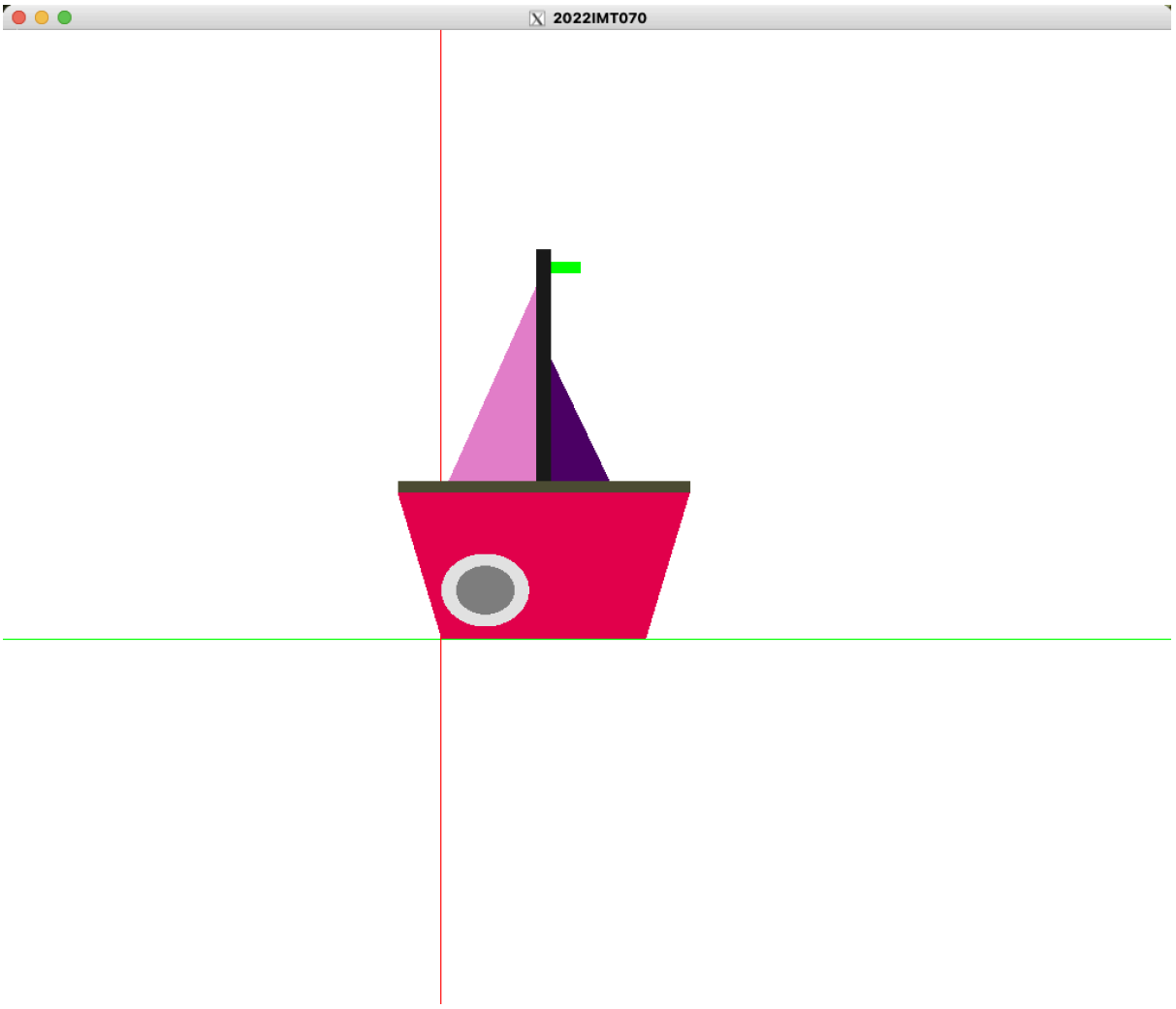
```
//2022IMT-070 Manoj Shiv
```




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

int graphSize = 250;


void drawCircleLL(double cx, double cy, double radius) {
   const int num_segments = 90;
   const double step = 2 * M_PI / num_segments;

   glBegin(GL_POLYGON);
   for (double theta = 0; theta < 2 * M_PI; theta += step) {
       double x = cx + radius * cos(theta);
       double y = cy + radius * sin(theta);
       glVertex2d(x, y);
```

```
    }
    glEnd();
}


void windowdisplay() {
    glClearColor(1.0f, 1.0f, 1.0f, 1.0f);
    glClear(GL_COLOR_BUFFER_BIT);


    glPointSize(2.0f);

    glColor3f(1.0f, 0.0f, 0.0f);
//AXES
    glBegin(GL_LINES);
        glColor3f(0.0f, 1.0f, 0.0f);
        glVertex2f(-graphSize*1.0f, 0.0f);
        glVertex2f(graphSize*1.0f, 0.0f);
    glEnd();
    glBegin(GL_LINES);
        glColor3f(1.0f, 0.0f, 0.0f);
        glVertex2f(0.0f, -graphSize*1.0f);
        glVertex2f(0.0f, graphSize*1.0f);
    glEnd();


//BASE penta
glColor3f(0.9f, 0.0f, 0.3f);
glBegin(GL_POLYGON);
        glVertex2d(0,0);
        glVertex2d(70,0);
        glVertex2d(85,60);
        glVertex2d(-15,60);
    glEnd();

//flat rectangle
glColor3f(0.3f, 0.3f, 0.2f);
    glBegin(GL_POLYGON);
    glVertex2d(-15,65);
        glVertex2d(85,65);
        glVertex2d(85,60);
        glVertex2d(-15,60);
    glEnd();

    //tall rectangle
```

```
glColor3f(0.1f, 0.1f, 0.1f);
   glBegin(GL_POLYGON);
   glVertex2d(32.5,65);
       glVertex2d(37.5,65);
       glVertex2d(37.5,160);
       glVertex2d(32.5,160);
   glEnd();


//flag
glColor3f(0.0f, 1.0f, 0.0f);
   glBegin(GL_POLYGON);
       glVertex2d(37.5,155);
       glVertex2d(37.5,150);
       glVertex2d(37.5+10,150);
       glVertex2d(37.5+10,155);
   glEnd();

//sail one
   glColor3f(0.9f, 0.5f, 0.8f);
   glBegin(GL_POLYGON);
       glVertex2d(32.5,65);
       glVertex2d(2.5,65);
       glVertex2d(32.5,65+80);
   glEnd();

   //sail 2

   glColor3f(0.3f, 0.01f, 0.4f);
   glBegin(GL_POLYGON);
       glVertex2d(37.5,65);
       glVertex2d(57.5,65);
       glVertex2d(37.5,115);
   glEnd();

   //cirlce one:



   glColor3f(0.9f, 0.9f, 0.9f);
drawCircleLL(20-5,20, 15);

glColor3f(0.5f, 0.5f, 0.5f);
   drawCircleLL(20-5,20, 10);
```

```
    glFlush();



}

void init() {
    // Set up the coordinate system
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(-graphSize+100, graphSize, -graphSize+100, graphSize);
    glMatrixMode(GL_MODELVIEW);
}

int main(int argc, char** argv) {
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize(1024, 1024);
    glutCreateWindow("2022IMT070");
    init();
    glutDisplayFunc(windowdisplay);
    glutMainLoop();
    return 0;
}
```