```
#include <windows.h>
#include<stdio.h>
#include<GL/glut.h>

//////////////////// Declaration of global variables   ////////////////////////
float y=0,ang=0,i=0,k=0,n=0;

float a=900,b=880,c=900,d=900,p,q=0,s;

float m=.80,j=.50,o=.15;


////////// sea function to display river   ////////////


void sea()
{

    glBegin(GL_POLYGON);
        glColor3f(0.0,0.50,1.0);
        glVertex2f(0.0,0.0);
        glVertex2f(2000.0,0.0);
        glVertex2f(2000.0,1600.0);
        glVertex2f(0.0,1600.0);
    glEnd();


  glPushMatrix();
  glTranslatef(0,q,0);

    glBegin(GL_LINES);
        glColor3f(1.0,1.0,1.0);
   for(p=0;p<20000;p=p+100)
       for(s=0;s<20000;s=s+100)
        glVertex2f(100.0+s,100.0+p);
        glVertex2f(200.0+s,100.0+p);
    glEnd();


  glPopMatrix();
}

////////////////////  Bridge function  ////////////////

void bridge()
{
    glBegin(GL_POLYGON);
        glColor3f(0.40,0.40,0.40);
        glVertex2f(0.0,900.0);
        glVertex2f(500.0,900.0);
        glVertex2f(500.0,1200.0);   //bridge top 1
        glVertex2f(0.0,1200.0);
    glEnd();

    glBegin(GL_POLYGON);
        glColor3f(1.0,1.0,1.0);
        glVertex2f(100.0,1030.0);
        glVertex2f(200.0,1030.0);
        glVertex2f(200.0,1040.0);   //strip1
        glVertex2f(100.0,1040.0);
    glEnd();


    glBegin(GL_POLYGON);
        glColor3f(1.0,1.0,1.0);
        glVertex2f(300.0,1030.0);
```

```
    glVertex2f(400.0,1030.0);
    glVertex2f(400.0,1040.0);   //strip2
    glVertex2f(300.0,1040.0);
glEnd();


glBegin(GL_POLYGON);
    glColor3f(1.0,1.0,.0);
    glVertex2f(0.0,1170.0);
    glVertex2f(500.0,1170.0);
    glVertex2f(500.0,1175.0);   //yellow strip1
    glVertex2f(0.0,1175.0);
glEnd();


glBegin(GL_POLYGON);
    glColor3f(1.0,1.0,0.0);
    glVertex2f(0.0,920.0);
    glVertex2f(500.0,920.0);
    glVertex2f(500.0,930.0);   //yellow strip2
    glVertex2f(0.0,930.0);
glEnd();

//   brige up

glPushMatrix();

glBegin(GL_POLYGON);
    glColor3f(0.46,0.46,0.46);
    glVertex2f(500.0,900.0);             //bridge top 2
    //up
    glVertex2f(900.0-k,900.0+n);
    glVertex2f(900.0-k,1200.0+n);
    //up
    glVertex2f(500.0,1200.0);
glEnd();




glBegin(GL_LINES);
    glColor3f(0.0,0.0,0.0);
    glVertex2f(20.0,1400.0);
    glVertex2f(900.0-k,900.0+n);    //pole thread front
    glVertex2f(0.0,1400.0);
    glVertex2f(900.0-k,880.0+n);
glEnd();


glBegin(GL_LINES);
    glColor3f(0.0,0.0,0.0);
    glVertex2f(30.0,1550.0);
    glVertex2f(900.0-k,1200.0+n);   //pole thread back
    glVertex2f(50.0,1550.0);
    glVertex2f(900.0-k,1203.0+n);
glEnd();



glBegin(GL_POLYGON);
    glColor3f(0.0,0.0,0.0);
    glVertex2f(500.0,880.0);
    glVertex2f(900.0-k,880.0+n); //base1
    glVertex2f(900.0-k,900.0+n);
    glVertex2f(500.0,900.0);
glEnd();
```

```
    glBegin(GL_POLYGON);
        glColor3f(0.46,0.46,0.46);
        glVertex2f(900.0+k,900.0+n);
        //up
        glVertex2f(1300.0,900.0);    // bridge top3
        glVertex2f(1300.0,1200.0);
        //up
        glVertex2f(900.0+k,1200.0+n);
      glEnd();


    glBegin(GL_POLYGON);
        glColor3f(0.0,0.0,0.0);
        glVertex2f(900.0+k,880.0+n);
        glVertex2f(1300.0,880.0);   // base 2
        glVertex2f(1300.0,900.0);
        glVertex2f(900.0+k,900.0+n);
    glEnd();
glPopMatrix();



    glBegin(GL_POLYGON);
        glColor3f(0.40,0.40,0.40);
        glVertex2f(1300.0,900.0);
        glVertex2f(2000.0,900.0);  //bridge top 4
        glVertex2f(2000.0,1200.0);
        glVertex2f(1300.0,1200.0);
    glEnd();

    glBegin(GL_POLYGON);
        glColor3f(1.0,1.0,0.0);
        glVertex2f(1300.0,1170.0);
        glVertex2f(2000.0,1170.0);
        glVertex2f(2000.0,1175.0);   //yellow strip3
        glVertex2f(1300.0,1175.0);
     glEnd();
    glBegin(GL_POLYGON);
        glColor3f(1.0,1.0,0.0);
        glVertex2f(1300.0,920.0);
        glVertex2f(2000.0,920.0);
        glVertex2f(2000.0,930.0);   // yellow strip4
        glVertex2f(1300.0,930.0);
    glEnd();

    glBegin(GL_POLYGON);
        glColor3f(1.0,1.0,1.0);
        glVertex2f(1400.0,1030.0);
        glVertex2f(1500.0,1030.0);
        glVertex2f(1500.0,1040.0);   //strip3
        glVertex2f(1400.0,1040.0);
    glEnd();

    glBegin(GL_POLYGON);
        glColor3f(1.0,1.0,1.0);
        glVertex2f(1600.0,1030.0);
        glVertex2f(1700.0,1030.0);
        glVertex2f(1700.0,1040.0);   //strip4
        glVertex2f(1600.0,1040.0);
    glEnd();

    glBegin(GL_POLYGON);
        glColor3f(1.0,1.0,1.0);
        glVertex2f(1800.0,1030.0);
        glVertex2f(1900.0,1030.0);
        glVertex2f(1900.0,1040.0);   //strip5
```

```
        glVertex2f(1800.0,1040.0);
    glEnd();


    glBegin(GL_LINES);
        glColor3f(0.0,0.0,0.0);
        glVertex2f(1725.0,1550.0);
        glVertex2f(900.0+k,1200.0+n);    //rite pole thread
        glVertex2f(1745.0,1550.0);
        glVertex2f(900.0+k,1200.0+n);
        glEnd();

    glBegin(GL_LINES);
        glColor3f(0.0,0.0,0.0);
        glVertex2f(1750.0,1400.0);
        glVertex2f(900.0+k,900.0+n);    //right pole thread front
        glVertex2f(1770.0,1400.0);
        glVertex2f(900.0+k,880.0+n);
    glEnd();

    glBegin(GL_POLYGON);
        glColor3f(0.25,0.25,0.25);
        glVertex2f(200.0,800.0);   //6 point polygon 1
        glVertex2f(200.0,700.0);
        glVertex2f(300.0,700.0);
        glVertex2f(300.0,800.0);
        glVertex2f(350.0,880.0);
        glVertex2f(150.0,880.0);
        glEnd();

    glBegin(GL_POLYGON);
        glColor3f(0.0,0.0,0.0);
        glVertex2f(0.0,880.0);
        glVertex2f(500.0,880.0);   //base3
        glVertex2f(500.0,900.0);
        glVertex2f(0.0,900.0);
    glEnd();
//
    glBegin(GL_POLYGON);
        glColor3f(0.0,0.0,0.0);    //base4
        glVertex2f(1300.0,880.0);
        glVertex2f(2000.0,880.0);
        glVertex2f(2000.0,900.0);
        glVertex2f(1300.0,900.0);
    glEnd();

    glBegin(GL_POLYGON);
        glColor3f(0.25,0.25,0.25);
        glVertex2f(1500.0,800.0);
        glVertex2f(1500.0,700.0);
        glVertex2f(1600.0,700.0);  //6 point polygon2
        glVertex2f(1600.0,800.0);
        glVertex2f(1650.0,880.0);
        glVertex2f(1450.0,880.0);
    glEnd();


}

//////////////////////  Boat function  ////////////////////

void boat()
{
        //glTranslatef(0,-250,0);
        glPushMatrix();
```

```
    glTranslatef(0,y,0);
    glPushMatrix();
glBegin(GL_POLYGON);
    glColor3f(m,j,o);
    glVertex2f(900.0,700.0);
    glVertex2f(800.0,620.0);
    glVertex2f(750.0,500.0);
    glVertex2f(750.0,200.0);     //ship
    glVertex2f(900.0,50.0);
    glVertex2f(1050.0,200.0);
    glVertex2f(1050.0,500.0);
    glVertex2f(1000.0,620.0);

glEnd();


glBegin(GL_POLYGON);
    glColor3f(0.0,0.0,0.0);     // ship back  1
    glVertex2f(750.0,200.0);
    glVertex2f(900.0,0.0);
    glVertex2f(900.0,50.0);
    glVertex2f(751.0,200.0);
glEnd();

glBegin(GL_POLYGON);
    glColor3f(0.1,0.1,0.1);
    glVertex2f(901.0,0.0);      //ship back  2
    glVertex2f(1050.0,200.0);
    glVertex2f(901.0,50.0);
glEnd();


glBegin(GL_LINES);
    glColor3f(0.0,0.0,0.0);
    glVertex2f(900.0,700.0);
    glVertex2f(820.0,600.0);  //boat grill
    glVertex2f(820.0,600.0);
    glVertex2f(800.0,620.0);
    glVertex2f(820.0,600.0);
    glVertex2f(770.0,500.0);
    glVertex2f(770.0,500.0);
    glVertex2f(750.0,500.0);
    glVertex2f(770.0,500.0);
    glVertex2f(770.0,200.0);
    glVertex2f(770.0,200.0);
    glVertex2f(750.0,200.0);
    glVertex2f(770.0,200.0);
    glVertex2f(900.0,70.0);
    glVertex2f(900.0,70.0);
    glVertex2f(900.0,50.0);
    glVertex2f(900.0,70.0);
    glVertex2f(1030.0,200.0);
    glVertex2f(1030.0,200.0);
    glVertex2f(1050.0,200.0);
    glVertex2f(1030.0,200.0);
    glVertex2f(1030.0,500.0);
    glVertex2f(1030.0,500.0);
    glVertex2f(1050.0,500.0);
    glVertex2f(1030.0,500.0);
    glVertex2f(980.0,620.0);
    glVertex2f(980.0,620.0);
    glVertex2f(1000.0,620.0);
    glVertex2f(980.0,620.0);
    glVertex2f(900.0,700.0);
    glVertex2f(770.0,350.0);
    glVertex2f(750.0,350.0);
```

```
        glVertex2f(770.0,450.0);
        glVertex2f(750.0,450.0);
        glVertex2f(770.0,250.0);
        glVertex2f(750.0,250.0);
        glVertex2f(1030.0,250.0);
        glVertex2f(1050.0,250.0);
        glVertex2f(1030.0,350.0);
        glVertex2f(1050.0,350.0);
        glVertex2f(1030.0,450.0);
        glVertex2f(1050.0,450.0);
        glVertex2f(840.0,130.0);
        glVertex2f(820.0,110.0);
        glVertex2f(975.0,110);
        glVertex2f(955.0,125.0);
    glEnd();



        glBegin(GL_POLYGON);
            glColor3f(0.10,0.10,0.);
            glVertex2f(850.0,400.0);   //boat inside polygon
            glVertex2f(950.0,400.0);
            glVertex2f(950.0,500.0);
            glVertex2f(850.0,500.0);
        glEnd();

        glBegin(GL_POLYGON);
            glColor3f(0.0,0.0,0.0);
            glVertex2f(850.0,400.0);//table on ship1
            glVertex2f(850.0,350.0);
            glVertex2f(860.0,350.0);
            glVertex2f(860.0,400.0);
            glEnd();

        glBegin(GL_POLYGON);
            glColor3f(0.0,0.0,0.0);
            glVertex2f(920.0,400.0);//2
            glVertex2f(930.0,380.0);
            glVertex2f(930.0,380.0);
            glVertex2f(920.0,400.0);
        glEnd();

        glBegin(GL_POLYGON);
            glColor3f(0.0,0.0,0.0);
            glVertex2f(950.0,400.0);//3
            glVertex2f(950.0,350.0);
            glVertex2f(940.0,350.0);
            glVertex2f(940.0,400.0);
        glEnd();

        glBegin(GL_POLYGON);
            glColor3f(0.0,0.0,0.0);
            glVertex2f(860.0,400.0);
            glVertex2f(860.0,380.0);
            glVertex2f(870.0,380.0);//4
            glVertex2f(870.0,400.0);
        glEnd();



     glPopMatrix();
    glPopMatrix();
}
```

```
////////////// Pole Function  ///////////
void poles()
{
    glBegin(GL_POLYGON);                //  left pole behind
        glColor3f(0.0,0.0,0.0);
        glVertex2f(30.0,1200.0);
        glVertex2f(50.0,1200.0);
        glVertex2f(50.0,1550.0);
        glVertex2f(30.0,1550.0);
    glEnd();



    glBegin(GL_POLYGON);                  //  right pole behind
        glColor3f(0.0,0.0,0.0);
        glVertex2f(1725.0,1200.0);
        glVertex2f(1745.0,1200.0);
        glVertex2f(1745.0,1550.0);
        glVertex2f(1725.0,1550.0);
    glEnd();

    glBegin(GL_POLYGON);                  //  left pole front
        glColor3f(0.0,0.0,0.0);
        glVertex2f(0.0,900.0);
        glVertex2f(20.0,900.0);
        glVertex2f(20.0,1400.0);
        glVertex2f(0.0,1400.0);
    glEnd();
//
//
//
    glBegin(GL_POLYGON);                  //  right pole front
        glColor3f(0.0,0.0,0.0);
        glVertex2f(1750.0,900.0);
        glVertex2f(1770.0,900.0);
        glVertex2f(1770.0,1400.0);
        glVertex2f(1750.0,1400.0);
    glEnd();
}

////////////// display function  ///////////////

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT);
    sea();
    bridge();
    boat();
    poles();

    glFlush();
    glutSwapBuffers();
}


////////   functio to animate bridge stripes   ////////////

void animate()
{
    q=q-.5;
    y=y+0.2;

    i+=0.2;
    if((i>=135) && (i<=439))
    {   k=k+0.1;
        n=n+0.1;
```

```
    }
    if(i>=1200 && !(k<=0 && n<=0))
    {
        k=k-0.1;
        n=n-0.1;
    }

    if(k<=0){
        g-=0.5;
    }
    if(i>1520){
        i=0;
        y=0;
        glutIdleFunc(NULL);
    }

    glutPostRedisplay();
}




void myinit()
{
    glClearColor(1.0,1.0,1.0,1.0);
    glColor3f(1.0,1.0,1.0);
    glPointSize(1.0);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(0.0,2000.0,0.0,1600.0);
}


///////   K/B function for changing boat color  ////////

void keyboard( unsigned char key, int x, int y )
{
    switch( key )
    {
        case '1':glutIdleFunc(animate);
                 break;
        case '2':
                 glutIdleFunc(NULL);
                 break;
        case '3':
                y=0;i=0;
                 break;
        case 'r':m=1.0,j=0.0,o=0.0;
                 glutPostRedisplay();
                 break;

        case 'g':m=0.0,j=1.0,o=0.0;
                 glutPostRedisplay();
                 break;

        case 'b':m=.80,j=.50,o=0.15;
                 glutPostRedisplay();
                 break;

        case 'w':m=1.0,j=1.0,o=1.0;
                 glutPostRedisplay();
                 break;

        case 'm':m=1.0,j=.0,o=1.0;
                 glutPostRedisplay();
```

```
                break;

        case 'c':m=.0,j=1.0,o=1.0;
                glutPostRedisplay();
                break;

        case 'y':m=.75,j=0.75,o=.75;
                glutPostRedisplay();
                break;

    };
}


int main(int argc,char **argv)
{
    glutInit(&argc,argv);
    glutInitDisplayMode(GLUT_DOUBLE|GLUT_RGB);
    glutInitWindowSize(2000,1600);
    glutInitWindowPosition(0,0);
    glutCreateWindow("Lift BRIDGE");
    myinit();
    glutDisplayFunc(display);

    glClearColor (1.0, 1.0, 0.0, 1.0);
    glutKeyboardFunc(keyboard);
    glutMainLoop();
    return 0;

}
```