Project Report

Members:

Anik Das ---------- (201-15-3685) ----- (55_PC-D2)

Mahmudul Hasan Shihab ------- (201-15-3409) ----- (55_PC-D1)

Shihad Mamun ---------- (201-15-3539) ----- (55_PC-D2)

Naimul Islam Noman ---------- (201-15-3266) ----- (55_PC-D2)

Mst Naima Sultana ---------- (201-15-3441) ----- (55_PC-D1)

Project Title

(City View)

Code:

```
#include<windows.h>

#include <GL/glut.h>

#include <stdio.h>

#include <GL/gl.h>


float xb1Pos, yb1Pos, xcar1Pos,ycar1Pos,xcar2Pos,ycar2Pos, xcloudPos,ycloudPos = 0.0;

float boat1boundaryRight = 1.0, boat1boundaryLeft = -1.0, car1boundaryRight = 1.0, car1boundaryLeft =
-1.0, car2boundaryRight = 1.0, car2boundaryLeft = -0.9, cloudboundaryRight = 1.0, cloudboundaryLeft = -
1.0;


void init(void)

{

  glClearColor(0.0,0.0,0.0,0.0); //GLfloat red,green,blue,alpha initial value 0 alpha values used by glclear
to clear the color buffers

  glMatrixMode(GL_PROJECTION);  // To specify which matrix is the current matrix & projection applies
subsequent matrix to projecton matrix stack

  glLoadIdentity();
```

```
  glOrtho(0.0, 1.0, 0.0, 1.0, -1.0, 1.0);
  //gluOrtho2D(0.0,300.0,0.0,300.0); // Orthographic representation; multiply the current matrix by an orthographic matrix 2D= left right,bottom,top equivalent near=-1,far=1
}

void Draw()
{
glClear(GL_COLOR_BUFFER_BIT);
///SKY
glLoadIdentity();
glBegin(GL_QUADS);
glColor3d(0.44, 0.80, 0.87);

glVertex2d(0.0,0.7125);
glVertex2d(0.25,0.5875);
glVertex2d(0.25,1.0);
glVertex2d(0.0,1.0);
glEnd();

glBegin(GL_QUADS);
glColor3d(0.44, 0.80, 0.87);

glVertex2d(0.25,0.5875);
glVertex2d(1.0,0.5875);
glVertex2d(1.0,1.0);
glVertex2d(0.25,1.0);
glEnd();
```

```
///Back_Ground
glBegin(GL_QUADS);
glColor3d(0.20, 0.73, 0.80);
glVertex2d(0.0,0.4625);
glVertex2d(0.25,0.4625);
glVertex2d(0.25,0.5875);
glVertex2d(0.0,0.7125);
glEnd();

glBegin(GL_QUADS);
glColor3d(0.20, 0.73, 0.80);

glVertex2d(0.25,0.4625);
glVertex2d(1.0,0.4625);
glVertex2d(1.0,0.5875);
glVertex2d(0.25,0.5875);
glEnd();


///BG_Building-1
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.32, 0.7625);
glVertex2d(0.45, 0.7625);
glVertex2d(0.45, 0.875);
glVertex2d(0.32, 0.875);

glEnd();
```

```
///BG_Building-1_Top-1
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.33, 0.875);
glVertex2d(0.37, 0.875);
glVertex2d(0.37, 0.9);
glVertex2d(0.33, 0.9);

glEnd();

///BG_Building-1_Top-2
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.4, 0.875);
glVertex2d(0.44, 0.875);
glVertex2d(0.44, 0.9);
glVertex2d(0.4, 0.9);

glEnd();

///BG_Building-2
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.52, 0.5875);
glVertex2d(0.67, 0.5875);
```

```
glVertex2d(0.67, 0.875);
glVertex2d(0.52, 0.875);

glEnd();

///BG_Building-2_Piller-1
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.54, 0.875);
glVertex2d(0.55, 0.875);
glVertex2d(0.55, 0.9);
glVertex2d(0.54, 0.9);

glEnd();

///BG_Building-2_Piller-2
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.64, 0.875);
glVertex2d(0.65, 0.875);
glVertex2d(0.65, 0.9);
glVertex2d(0.64, 0.9);

glEnd();


///BG_Building-2_Top
```

```
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.52, 0.9);
glVertex2d(0.67, 0.9);
glVertex2d(0.67, 0.9125);
glVertex2d(0.52, 0.9125);

glEnd();

///BG_Building-3
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.71, 0.7);
glVertex2d(0.83, 0.7);
glVertex2d(0.83, 0.7875);
glVertex2d(0.71, 0.7875);

glEnd();

///BG_Building-3_Top
glBegin(GL_TRIANGLES);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.71, 0.7875);
glVertex2d(0.83, 0.7875);
glVertex2d(0.77, 0.85);
```

```
glEnd();



///Building-1
glBegin(GL_QUADS);
glColor3d(1.0,0.71,0.31);

glVertex2d(0.1, 0.4625);
glVertex2d(0.25, 0.4625);
glVertex2d(0.25, 0.75);
glVertex2d(0.1, 0.75);

glEnd();

///Building-1-Roof
glBegin(GL_QUADS);
glColor3d(1.0,0.98,0.91);

glVertex2d(0.09, 0.75);
glVertex2d(0.26, 0.75);
glVertex2d(0.26, 0.7625);
glVertex2d(0.09, 0.7625);

glEnd();

///Building-1_Top
glBegin(GL_QUADS);
glColor3d(0.87, 0.87, 0.87);
```

```
glVertex2d(0.2, 0.762);
glVertex2d(0.24, 0.762);
glVertex2d(0.24, 0.786);
glVertex2d(0.2, 0.786);

glEnd();

///Building-1_Top_Roof
glBegin(GL_QUADS);
glColor3d(0.74, 0.75, 0.74);

glVertex2d(0.195, 0.786);
glVertex2d(0.245, 0.786);
glVertex2d(0.245, 0.8);
glVertex2d(0.195, 0.8);

glEnd();

///Building-1_Door_Shadow
glBegin(GL_QUADS);
glColor3d(0.75, 0.75, 0.75);

glVertex2d(0.115, 0.462);
glVertex2d(0.175, 0.462);
glVertex2d(0.175, 0.55875);
glVertex2d(0.115, 0.55875);

glEnd();
```

```
///Building-1_Door
glBegin(GL_QUADS);
glColor3d(0.80, 0.80, 0.80);

glVertex2d(0.12, 0.4625);
glVertex2d(0.17, 0.4625);
glVertex2d(0.17, 0.5525);
glVertex2d(0.12, 0.5525);

glEnd();

///Building-1_Win-1
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.11, 0.7);
glVertex2d(0.14, 0.7);
glVertex2d(0.14, 0.7375);
glVertex2d(0.11, 0.7375);

glEnd();

///Building-1_Win-2
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.16, 0.7);
glVertex2d(0.19, 0.7);
```

```
glVertex2d(0.19, 0.7375);
glVertex2d(0.16, 0.7375);

glEnd();

///Building-1_Win-3
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.21, 0.7);
glVertex2d(0.24, 0.7);
glVertex2d(0.24, 0.7375);
glVertex2d(0.21, 0.7375);

glEnd();


///Building-1_Win-4
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.11, 0.6375);
glVertex2d(0.14, 0.6375);
glVertex2d(0.14, 0.675);
glVertex2d(0.11, 0.675);

glEnd();

///Building-1_Win-5
```

```
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.16, 0.6375);
glVertex2d(0.19, 0.6375);
glVertex2d(0.19, 0.675);
glVertex2d(0.16, 0.675);

glEnd();

///Building-1_Win-6
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.21, 0.6375);
glVertex2d(0.24, 0.6375);
glVertex2d(0.24, 0.675);
glVertex2d(0.21, 0.675);

glEnd();

///Building-1_Win-7
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.11, 0.575);
glVertex2d(0.15, 0.575);
glVertex2d(0.15, 0.6125);
glVertex2d(0.11, 0.6125);
```

```
glEnd();

///Building-1_Win-8
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.20, 0.575);
glVertex2d(0.24, 0.575);
glVertex2d(0.24, 0.6125);
glVertex2d(0.20, 0.6125);

glEnd();

///Building-1_Win-9
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.19, 0.486);
glVertex2d(0.24, 0.486);
glVertex2d(0.24, 0.55);
glVertex2d(0.19, 0.55);

glEnd();


///Building-2
glBegin(GL_QUADS);
glColor3d(0.88, 0.88, 0.88);
```

```
glVertex2d(0.3, 0.4625);
glVertex2d(0.45, 0.4625);
glVertex2d(0.45, 0.75);
glVertex2d(0.3, 0.75);

glEnd();

///Building-2_Roof
glBegin(GL_QUADS);
glColor3d(1.0,1.0,1.0);

glVertex2d(0.29, 0.75);
glVertex2d(0.46, 0.75);
glVertex2d(0.46, 0.7625);
glVertex2d(0.29, 0.7625);

glEnd();

///Building-2_Top
glBegin(GL_QUADS);
glColor3d(0.87, 0.87, 0.87);

glVertex2d(0.31, 0.762);
glVertex2d(0.35, 0.762);
glVertex2d(0.35, 0.786);
glVertex2d(0.31, 0.786);

glEnd();
```

```
///Building-2_Top_Roof
glBegin(GL_QUADS);
glColor3d(0.74, 0.75, 0.74);

glVertex2d(0.305, 0.786);
glVertex2d(0.355, 0.786);
glVertex2d(0.355, 0.8);
glVertex2d(0.305, 0.8);

glEnd();

///Building-2_Door_Shadow
glBegin(GL_QUADS);
glColor3d(0.75, 0.75, 0.75);

glVertex2d(0.335, 0.462);
glVertex2d(0.414, 0.462);
glVertex2d(0.414, 0.55875);
glVertex2d(0.335, 0.55875);

glEnd();

///Building-2_Door Part-1
glBegin(GL_QUADS);
glColor3d(0.50, 0.79, 0.76);

glVertex2d(0.34, 0.4625);
glVertex2d(0.374, 0.4625);
```

```
glVertex2d(0.374, 0.5525);
glVertex2d(0.34, 0.5525);

glEnd();

///Building-2_Door Part-2
glBegin(GL_QUADS);
glColor3d(0.50, 0.79, 0.76);

glVertex2d(0.375, 0.4625);
glVertex2d(0.41, 0.4625);
glVertex2d(0.41, 0.5525);
glVertex2d(0.375, 0.5525);

glEnd();

///Building-2_Win-1
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.31, 0.7);
glVertex2d(0.34, 0.7);
glVertex2d(0.34, 0.7375);
glVertex2d(0.31, 0.7375);

glEnd();

///Building-2_Win-2
glBegin(GL_QUADS);
```

```
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.36, 0.7);
glVertex2d(0.39, 0.7);
glVertex2d(0.39, 0.7375);
glVertex2d(0.36, 0.7375);

glEnd();

///Building-2_Win-3
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.41, 0.7);
glVertex2d(0.44, 0.7);
glVertex2d(0.44, 0.7375);
glVertex2d(0.41, 0.7375);

glEnd();


///Building-2_Win-4
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.31, 0.6375);
glVertex2d(0.34, 0.6375);
glVertex2d(0.34, 0.675);
glVertex2d(0.31, 0.675);
```

```
glEnd();

///Building-2_Win-5
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.36, 0.6375);
glVertex2d(0.39, 0.6375);
glVertex2d(0.39, 0.675);
glVertex2d(0.36, 0.675);

glEnd();

///Building-2_Win-6
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.41, 0.6375);
glVertex2d(0.44, 0.6375);
glVertex2d(0.44, 0.675);
glVertex2d(0.41, 0.675);

glEnd();

///Building-2_Win-7
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);
```

```
glVertex2d(0.31, 0.575);
glVertex2d(0.35, 0.575);
glVertex2d(0.35, 0.6125);
glVertex2d(0.31, 0.6125);

glEnd();

///Building-2_Win-8
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.40, 0.575);
glVertex2d(0.44, 0.575);
glVertex2d(0.44, 0.6125);
glVertex2d(0.40, 0.6125);

glEnd();



///Building-3
glBegin(GL_QUADS);
glColor3d(0.99, 0.80, 0.49);

glVertex2d(0.5, 0.4625);
glVertex2d(0.64, 0.4625);
glVertex2d(0.64, 0.6125);
glVertex2d(0.5, 0.6125);
```

```
glEnd();

///Building-3_Roof
glBegin(GL_QUADS);
glColor3d(1.0,1.0,1.0);

glVertex2d(0.49, 0.6125);
glVertex2d(0.65, 0.6125);
glVertex2d(0.65, 0.625);
glVertex2d(0.49, 0.625);

glEnd();

///Building-3_2nd Floor
glBegin(GL_QUADS);
glColor3d(0.99, 0.80, 0.49);

glVertex2d(0.55, 0.625);
glVertex2d(0.64, 0.625);
glVertex2d(0.64, 0.7);
glVertex2d(0.55, 0.7);

glEnd();

///Building-3_2nd Floor_Roof
glBegin(GL_QUADS);
glColor3d(1.0,1.0,1.0);

glVertex2d(0.545, 0.7);
```

```
glVertex2d(0.645, 0.7);
glVertex2d(0.645, 0.712);
glVertex2d(0.545, 0.712);

glEnd();

///Building-3_Top
glBegin(GL_QUADS);
glColor3d(0.87, 0.87, 0.87);

glVertex2d(0.60, 0.7125);
glVertex2d(0.64, 0.7125);
glVertex2d(0.64, 0.7375);
glVertex2d(0.60, 0.7375);

glEnd();

///Building-3_Top_Roof
glBegin(GL_QUADS);
glColor3d(0.74, 0.75, 0.74);

glVertex2d(0.595, 0.7375);
glVertex2d(0.645, 0.7375);
glVertex2d(0.645, 0.75275);
glVertex2d(0.595, 0.75275);

glEnd();

///Building-3_Door_Shadow
```

```
glBegin(GL_QUADS);
glColor3d(0.75, 0.75, 0.75);

glVertex2d(0.515, 0.462);
glVertex2d(0.575, 0.462);
glVertex2d(0.575, 0.55875);
glVertex2d(0.515, 0.55875);

glEnd();



///Building-3_Door
glBegin(GL_QUADS);
glColor3d(0.80, 0.80, 0.80);

glVertex2d(0.52, 0.4625);
glVertex2d(0.57, 0.4625);
glVertex2d(0.57, 0.5525);
glVertex2d(0.52, 0.5525);

glEnd();

///Building-3_Win-5
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.56, 0.6375);
glVertex2d(0.59, 0.6375);
glVertex2d(0.59, 0.675);
```

```
glVertex2d(0.56, 0.675);

glEnd();

///Building-3_Win-6
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.60, 0.6375);
glVertex2d(0.63, 0.6375);
glVertex2d(0.63, 0.675);
glVertex2d(0.60, 0.675);

glEnd();

///Building-3_Win-7
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.59, 0.4875);
glVertex2d(0.63, 0.4875);
glVertex2d(0.63, 0.5375);
glVertex2d(0.59, 0.5375);

glEnd();


///Building-4
glBegin(GL_QUADS);
```

```
glColor3d(1.00, 0.80, 0.51);

glVertex2d(0.7, 0.462);
glVertex2d(0.85, 0.462);
glVertex2d(0.85, 0.6875);
glVertex2d(0.7, 0.6875);

glEnd();

///Building-4_Roof
glBegin(GL_QUADS);
glColor3d(1.0,1.0,1.0);

glVertex2d(0.69, 0.6875);
glVertex2d(0.86, 0.6875);
glVertex2d(0.86, 0.7);
glVertex2d(0.69, 0.7);

glEnd();

///Building-4_Top
glBegin(GL_QUADS);
glColor3d(0.87, 0.87, 0.87);

glVertex2d(0.8, 0.7);
glVertex2d(0.85, 0.7);
glVertex2d(0.85, 0.725);
glVertex2d(0.8, 0.725);
```

```
glEnd();

///Building-4_Top_Roof
glBegin(GL_QUADS);
glColor3d(0.74, 0.75, 0.74);

glVertex2d(0.795, 0.725);
glVertex2d(0.855, 0.725);
glVertex2d(0.855, 0.7375);
glVertex2d(0.795, 0.7375);

glEnd();

///Building-2_Door_Shadow
glBegin(GL_QUADS);
glColor3d(0.75, 0.75, 0.75);

glVertex2d(0.715, 0.462);
glVertex2d(0.775, 0.462);
glVertex2d(0.775, 0.55875);
glVertex2d(0.715, 0.55875);

glEnd();


///Building-4_Door
glBegin(GL_QUADS);
glColor3d(0.80, 0.80, 0.80);
```

```
glVertex2d(0.72, 0.4625);
glVertex2d(0.77, 0.4625);
glVertex2d(0.77, 0.5525);
glVertex2d(0.72, 0.5525);

glEnd();

///Building-4_Win-4
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.71, 0.6375);
glVertex2d(0.74, 0.6375);
glVertex2d(0.74, 0.675);
glVertex2d(0.71, 0.675);

glEnd();

///Building-4_Win-5
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.76, 0.6375);
glVertex2d(0.79, 0.6375);
glVertex2d(0.79, 0.675);
glVertex2d(0.76, 0.675);

glEnd();
```

```
///Building-4_Win-6
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.81, 0.6375);
glVertex2d(0.84, 0.6375);
glVertex2d(0.84, 0.675);
glVertex2d(0.81, 0.675);

glEnd();

///Building-4_Win-7
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.71, 0.575);
glVertex2d(0.75, 0.575);
glVertex2d(0.75, 0.6125);
glVertex2d(0.71, 0.6125);

glEnd();

///Building-4_Win-8
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.80, 0.575);
glVertex2d(0.84, 0.575);
glVertex2d(0.84, 0.6125);
```

```
glVertex2d(0.80, 0.6125);

glEnd();

///Building-4_Win-9
glBegin(GL_QUADS);
glColor3d(0.43, 0.74, 0.74);

glVertex2d(0.79, 0.486);
glVertex2d(0.84, 0.486);
glVertex2d(0.84, 0.55);
glVertex2d(0.79, 0.55);

glEnd();



///Sun
   glPushMatrix();
   glColor3d(255,255,0);
   glTranslatef(0.90, 0.92, 0);
   glutSolidSphere(0.05, 32, 7);
   glPopMatrix();

///cloud 1///

glLoadIdentity();
glTranslatef(xcloudPos,ycloudPos,0.0);
   glPushMatrix();
```

```
//glTranslatef(_moveC,0.0f,0.0f);
glPushMatrix();
glColor3ub(220,220,220);
glTranslatef(0.10, 0.90, 0);
glutSolidSphere(0.03, 16, 8);
glPopMatrix();

glPushMatrix();
glColor3ub(220,220,220);
glTranslatef(0.13, 0.92, 0);
glutSolidSphere(0.03, 16, 8);
glPopMatrix();

glPushMatrix();
glColor3ub(220,220,220);
glTranslatef(0.13, 0.88, 0);
glutSolidSphere(0.03, 16, 8);
glPopMatrix();

glPushMatrix();
glColor3ub(220,220,220);
glTranslatef(0.16, 0.92, 0);
glutSolidSphere(0.03, 16, 8);
glPopMatrix();

glPushMatrix();
glColor3ub(220,220,220);
glTranslatef(0.16, 0.88, 0);
glutSolidSphere(0.03, 16, 8);
```

```
    glPopMatrix();

    glPushMatrix();
    glColor3ub(220,220,220);
    glTranslatef(0.19, 0.90, 0);
    glutSolidSphere(0.03, 16, 8);
    glPopMatrix();


///cloud 2///
    glPushMatrix();

    //glTranslatef(_moveC,0.0f,0.0f);
    glPushMatrix();
    glColor3ub(220,220,220);
    glTranslatef(0.45, 0.85, 0);
    glutSolidSphere(0.03, 16, 8);
    glPopMatrix();

     glPushMatrix();
    glColor3ub(220,220,220);
    glTranslatef(0.48, 0.87, 0);
    glutSolidSphere(0.03, 16, 8);
    glPopMatrix();

     glPushMatrix();
    glColor3ub(220,220,220);
    glTranslatef(0.48, 0.83, 0);
    glutSolidSphere(0.03, 16, 8);
```

```
    glPopMatrix();

    glPushMatrix();
    glColor3ub(220,220,220);
    glTranslatef(0.51, 0.87, 0);
    glutSolidSphere(0.03, 16, 8);
    glPopMatrix();

     glPushMatrix();
    glColor3ub(220,220,220);
    glTranslatef(0.51, 0.83, 0);
    glutSolidSphere(0.03, 16, 8);
    glPopMatrix();

    glPushMatrix();
    glColor3ub(220,220,220);
    glTranslatef(0.54, 0.85, 0);
    glutSolidSphere(0.03, 16, 8);
    glPopMatrix();

glLoadIdentity();



///Soil_Beside Road
glBegin(GL_QUADS);
glColor3d(0.45, 0.49, 0.44);

glVertex2d(0.0, 0.4625);
glVertex2d(1.0, 0.4625);
```

```
glVertex2d(1.0, 0.4375);
glVertex2d(0.0, 0.4375);

glEnd();

///Road_Border-1
glBegin(GL_QUADS);
glColor3d(0.0,0.0,0.0);

glVertex2d(0.0, 0.4375);
glVertex2d(1.0, 0.4375);
glVertex2d(1.0, 0.425);
glVertex2d(0.0, 0.425);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.1, 0.425);
glVertex2d(0.2, 0.425);
glVertex2d(0.2, 0.4375);
glVertex2d(0.1, 0.4375);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);
```

```
glVertex2d(0.3, 0.425);
glVertex2d(0.4, 0.425);
glVertex2d(0.4, 0.4375);
glVertex2d(0.3, 0.4375);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.5, 0.425);
glVertex2d(0.6, 0.425);
glVertex2d(0.6, 0.4375);
glVertex2d(0.5, 0.4375);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.7, 0.425);
glVertex2d(0.8, 0.425);
glVertex2d(0.8, 0.4375);
glVertex2d(0.7, 0.4375);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);
```

```
glVertex2d(0.9, 0.425);
glVertex2d(1.0, 0.425);
glVertex2d(1.0, 0.4375);
glVertex2d(0.9, 0.4375);

glEnd();

///Road
glBegin(GL_QUADS);
glColor3d(0.26, 0.26, 0.26);

glVertex2d(0.0, 0.425);
glVertex2d(1.0, 0.425);
glVertex2d(1.0, 0.325);
glVertex2d(0.0, 0.325);

glEnd();

///Road_Devider-Line
glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.08, 0.375);
glVertex2d(0.13, 0.375);
glVertex2d(0.13, 0.3875);
glVertex2d(0.08, 0.3875);

glEnd();
```

```
glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.2, 0.375);
glVertex2d(0.25, 0.375);
glVertex2d(0.25, 0.3875);
glVertex2d(0.2, 0.3875);

glEnd();

glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.33, 0.375);
glVertex2d(0.38, 0.375);
glVertex2d(0.38, 0.3875);
glVertex2d(0.33, 0.3875);

glEnd();

glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.46, 0.375);
glVertex2d(0.51, 0.375);
glVertex2d(0.51, 0.3875);
glVertex2d(0.46, 0.3875);
```

```
glEnd();

glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.59, 0.375);
glVertex2d(0.64, 0.375);
glVertex2d(0.64, 0.3875);
glVertex2d(0.59, 0.3875);

glEnd();

glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.72, 0.375);
glVertex2d(0.77, 0.375);
glVertex2d(0.77, 0.3875);
glVertex2d(0.72, 0.3875);

glEnd();

glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.85, 0.375);
glVertex2d(0.90, 0.375);
glVertex2d(0.90, 0.3875);
glVertex2d(0.85, 0.3875);
```

```
glEnd();

glBegin(GL_QUADS);
glColor3d(0.95, 0.95, 0.956);

glVertex2d(0.98, 0.375);
glVertex2d(1.0, 0.375);
glVertex2d(1.0, 0.3875);
glVertex2d(0.98, 0.3875);

glEnd();

///Road_Border-2
glBegin(GL_QUADS);
glColor3d(0.0,0.0,0.0);

glVertex2d(0.0, 0.325);
glVertex2d(1.0, 0.325);
glVertex2d(1.0, 0.3125);
glVertex2d(0.0, 0.3125);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.1, 0.3125);
glVertex2d(0.2, 0.3125);
```

```
glVertex2d(0.2, 0.325);
glVertex2d(0.1, 0.325);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.3, 0.3125);
glVertex2d(0.4, 0.3125);
glVertex2d(0.4, 0.325);
glVertex2d(0.3, 0.325);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.5, 0.3125);
glVertex2d(0.6, 0.3125);
glVertex2d(0.6, 0.325);
glVertex2d(0.5, 0.325);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.7, 0.3125);
```

```
glVertex2d(0.8, 0.3125);
glVertex2d(0.8, 0.325);
glVertex2d(0.7, 0.325);

glEnd();

glBegin(GL_QUADS);
glColor3d(1.0,0.0,0.0);

glVertex2d(0.9, 0.3125);
glVertex2d(1.0, 0.3125);
glVertex2d(1.0, 0.325);
glVertex2d(0.9, 0.325);

glEnd();



///Garden
glBegin(GL_QUADS);
glColor3d(0.40, 0.73, 0.42);

glVertex2d(0.0, 0.1875);
glVertex2d(1.0, 0.1875);
glVertex2d(1.0, 0.3125);
glVertex2d(0.0, 0.3125);

glEnd();

///Tree-1
```

```
glBegin(GL_QUADS);
glColor3d(0.49, 0.33, 0.31);

glVertex2d(0.11, 0.2);
glVertex2d(0.12, 0.2);
glVertex2d(0.12, 0.25);
glVertex2d(0.11, 0.25);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.09, 0.25);
glVertex2d(0.14, 0.25);
glVertex2d(0.115, 0.3);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.09, 0.275);
glVertex2d(0.14, 0.275);
glVertex2d(0.115, 0.33);

glEnd();

///Tree-2
```

```
glBegin(GL_QUADS);
glColor3d(0.49, 0.33, 0.31);

glVertex2d(0.28, 0.225);
glVertex2d(0.29, 0.225);
glVertex2d(0.29, 0.275);
glVertex2d(0.28, 0.275);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.26, 0.275);
glVertex2d(0.31, 0.275);
glVertex2d(0.285, 0.325);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.26, 0.3);
glVertex2d(0.31, 0.3);
glVertex2d(0.285, 0.35);

glEnd();

///Tree-3
```

```
glBegin(GL_QUADS);
glColor3d(0.49, 0.33, 0.31);

glVertex2d(0.49, 0.2125);
glVertex2d(0.5, 0.2125);
glVertex2d(0.5, 0.2625);
glVertex2d(0.49, 0.2625);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.47, 0.2625);
glVertex2d(0.52, 0.2625);
glVertex2d(0.495, 0.3125);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.47, 0.2875);
glVertex2d(0.52, 0.2875);
glVertex2d(0.495, 0.3375);

glEnd();

///Tree-4
```

```
glBegin(GL_QUADS);
glColor3d(0.49, 0.33, 0.31);

glVertex2d(0.68, 0.2);
glVertex2d(0.69, 0.2);
glVertex2d(0.69, 0.25);
glVertex2d(0.68, 0.25);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.66, 0.25);
glVertex2d(0.71, 0.25);
glVertex2d(0.685, 0.2875);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.66, 0.275);
glVertex2d(0.71, 0.275);
glVertex2d(0.685, 0.3125);

glEnd();

glBegin(GL_TRIANGLES);
```

```
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.66, 0.3);
glVertex2d(0.71, 0.3);
glVertex2d(0.685, 0.35);

glEnd();

///Tree-5
glBegin(GL_QUADS);
glColor3d(0.49, 0.33, 0.31);

glVertex2d(0.87, 0.225);
glVertex2d(0.88, 0.225);
glVertex2d(0.88, 0.275);
glVertex2d(0.87, 0.275);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.85, 0.275);
glVertex2d(0.9, 0.275);
glVertex2d(0.875, 0.3125);

glEnd();

glBegin(GL_TRIANGLES);
```

```
glColor3d(0.18, 0.49, 0.24);

glVertex2d(0.85, 0.3);
glVertex2d(0.9, 0.3);
glVertex2d(0.875, 0.3375);

glEnd();



///Car-2 R2L
glLoadIdentity();
glTranslatef(xcar1Pos,ycar1Pos,0.0);
//wheel1
glPushMatrix();
   glColor3ub(105,105,105);
   glTranslatef(0.11, 0.343, 0);
   glutSolidSphere(0.010, 32, 32);
   glPopMatrix();
//wheel2
glPushMatrix();
   glColor3ub(105,105,105);
   glTranslatef(0.18, 0.343, 0);
   glutSolidSphere(0.010, 32, 32);
   glPopMatrix();

//body
glBegin(GL_POLYGON);
glColor3ub(0, 0, 102);
```

```
glVertex2d(0.075, 0.343);
glVertex2d(0.075, 0.362);
glVertex2d(0.12, 0.39);
glVertex2d(0.18, 0.39);
glVertex2d(0.21, 0.362);
glVertex2d(0.21, 0.343);

glEnd();

glBegin(GL_QUADS);
glColor3d(0.45, 0.49, 0.44);

glVertex2d(0.10, 0.37);
glVertex2d(0.12, 0.387);
glVertex2d(0.149, 0.387);
glVertex2d(0.149, 0.37);

glEnd();

glBegin(GL_QUADS);
glColor3d(0.45, 0.49, 0.44);

glVertex2d(0.151, 0.37);
glVertex2d(0.151, 0.387);
glVertex2d(0.18, 0.387);
glVertex2d(0.195, 0.37);

glEnd();
```

```
///Car-1 L2R
glLoadIdentity();
glTranslatef(xcar2Pos,ycar2Pos,0.0);
//wheel1
glPushMatrix();
   glColor3ub(105,105,105);
   glTranslatef(0.50, 0.4, 0);
   glutSolidSphere(0.010, 32, 32);
   glPopMatrix();
//wheel2
glPushMatrix();
   glColor3ub(105, 105, 105);
   glTranslatef(0.57, 0.4, 0);
   glutSolidSphere(0.010, 32, 32);
   glPopMatrix();

//body
glBegin(GL_POLYGON);
glColor3ub(139, 0, 0);

glVertex2d(0.47, 0.4);
glVertex2d(0.47, 0.4187);
glVertex2d(0.50, 0.4375);
glVertex2d(0.56, 0.4375);
glVertex2d(0.61, 0.4187);
glVertex2d(0.61, 0.4);

glEnd();
```

```
glBegin(GL_QUADS);
glColor3d(0.45, 0.49, 0.44);

glVertex2d(0.482, 0.4187);
glVertex2d(0.50, 0.435);
glVertex2d(0.529, 0.435);
glVertex2d(0.529, 0.4187);

glEnd();

glBegin(GL_QUADS);
glColor3d(0.45, 0.49, 0.44);

glVertex2d(0.532, 0.4187);
glVertex2d(0.532, 0.435);
glVertex2d(0.56, 0.435);
glVertex2d(0.59, 0.4187);

glEnd();
glLoadIdentity();

///River Side
glBegin(GL_QUADS);
glColor3d(0.45, 0.49, 0.44);

glVertex2d(0.0, 0.175);
glVertex2d(1.0, 0.175);
glVertex2d(1.0, 0.1875);
```

```
glVertex2d(0.0, 0.1875);

glEnd();

///River
glBegin(GL_QUADS);
glColor3d(0.21, 0.76, 0.84);

glVertex2d(0.0, 0.0);
glVertex2d(1.0, 0.0);
glVertex2d(1.0, 0.1775);
glVertex2d(0.0, 0.1775);

glEnd();

///Boat
//Boat Base
glLoadIdentity();
glTranslatef(xb1Pos,yb1Pos,0.0);
glBegin(GL_QUADS);
glColor3d(0.0,0.0,0.0);

glVertex2d(0.10,0.1);
glVertex2d(0.20,0.1);
glVertex2d(0.20,0.125);
glVertex2d(0.10,0.125);
glEnd();

glBegin(GL_TRIANGLES);
```

```
glColor3d(0.0,0.0,0.0);

glVertex2d(0.10,0.1);
glVertex2d(0.06,0.1375);
glVertex2d(0.10,0.125);

glEnd();

glBegin(GL_TRIANGLES);
glColor3d(0.0,0.0,0.0);

glVertex2d(0.20,0.1);
glVertex2d(0.24,0.1375);
glVertex2d(0.20,0.125);

glEnd();


glBegin(GL_TRIANGLES);
glColor3d(0.0,0.0,0.0);

glVertex2d(0.16,0.125);
glVertex2d(0.20,0.125);
glVertex2d(0.18,0.1625);

glEnd();

glBegin(GL_QUADS);
glColor3ub(139,69,19);
```

```
glVertex2d(0.10,0.125);
glVertex2d(0.16,0.125);
glVertex2d(0.18,0.1625);
glVertex2d(0.12,0.1625);
glEnd();

///End Boat



glutSwapBuffers();
}

///Movement

void timercloud(int value)
{
   xcloudPos += 0.01; // // Move the 1st cloud forward to the Right
   if (xcloudPos > cloudboundaryRight)
      xcloudPos = cloudboundaryLeft;

   glutPostRedisplay();
   glutTimerFunc(50, timercloud, 0);
}

void timercloud1(int value)
{
   xcloudPos -= 0.01; // // Move the 2nd cloud forward to the Right
   if (xcloudPos < cloudboundaryLeft)
```

```
        xcloudPos = cloudboundaryRight;

    glutPostRedisplay();
    glutTimerFunc(50, timercloud1, 0);
}



void timercar1(int value)
{
    xcar1Pos -= 0.01; // Move the 1st car forward to the right
    if (xcar1Pos < car1boundaryLeft)
        xcar1Pos = car1boundaryRight;

    glutPostRedisplay();
    glutTimerFunc(50, timercar1, 0);
}



void timercar2(int value)
{
    xcar2Pos += 0.01; // // Move the 2nd car forward to the right
    if (xcar2Pos > car2boundaryRight)
        xcar2Pos = car2boundaryLeft;

    glutPostRedisplay();
    glutTimerFunc(50, timercar2, 0);
}
```

```
void timerboat1(int value)
{
   xb1Pos += 0.01; // Move the large boat forward to the right
   if (xb1Pos > boat1boundaryRight)
      xb1Pos = boat1boundaryLeft;

   glutPostRedisplay();
   glutTimerFunc(50, timerboat1, 0);
}



void timerboat2(int value)
{
   xb1Pos -= 0.01; // Move the large boat forward to the right
   if (xb1Pos < boat1boundaryLeft)
      xb1Pos = boat1boundaryRight;

   glutPostRedisplay();
   glutTimerFunc(50, timerboat2, 0);
}

///For Boat
void keyboard(unsigned char key, int cx, int cy)
{
   switch (key)
   {
```

```
    case 'd':
        glutTimerFunc(50, timerboat1, 0);

        break;
    case 'a':

       glutTimerFunc(50, timerboat2, 0);

    default:
        break;
    }
}

///For Car
void handleMouse(int button, int state, int x, int y)
{
    if (button == GLUT_LEFT_BUTTON && state == GLUT_DOWN)
    {
        glutTimerFunc(50, timercar2, 0);
    }
    if (button == GLUT_RIGHT_BUTTON && state == GLUT_DOWN)
    {
        glutTimerFunc(50, timercar1, 0);
    }

}

///For Cloud
void handleSpecialKeypress(int key, int x, int y)
```

```
{
   switch(key)
   {
   case GLUT_KEY_LEFT:
   glutTimerFunc(50, timercloud1, 0);

      break;
   case GLUT_KEY_RIGHT:
       glutTimerFunc(50, timercloud, 0);
      break;
   }
   glutPostRedisplay();
}


int main(int argc,char **argv){


   glutInit(&argc,argv);
   glutInitDisplayMode ( GLUT_RGB | GLUT_DOUBLE );
   glutInitWindowPosition(0,0);
   glutInitWindowSize(1000,800);
   glutCreateWindow("AAKA-LAB");
   init();
      glMatrixMode(GL_MODELVIEW);

   glutDisplayFunc(Draw);
    glutKeyboardFunc(keyboard);
```

```
glutMouseFunc(handleMouse);
glutSpecialFunc(handleSpecialKeypress);


    glutMainLoop();
    return 0;
}
```