LAB REPORT-04

Submitted to

Hasin Rehana

Lecturer in Computer Science and Engineering

Daffodil International University

Submitted By

Md. Amirul Islam

Id No: 181-15-1748

Section: PC-A

Daffodil International University

Date: 26-07-2021

Experiment No: 04

Experiment Name: motion object with circle.

Lab Code:

```
#include <windows.h>

#include <GL/gl.h>

#include <GL/glut.h>

#include<math.h>

float j=0.0;

void car()

{


   //cars

   if(j<1.0)

   {

      j=j+0.001;

   }

   else

   {

      j=0.0;

   }

   glColor3f(1.0, 1.0, 1.0);


   glBegin(GL_QUADS);

   glVertex2d(0.0+j,0.12);

   glVertex2d(0.14+j,0.12);

   glVertex2d(0.14+j, 0.15);

   glVertex2d(0.0+j,0.15);
```

```
glEnd();


glColor3f(0.0, 0.0, 1.0);


glBegin(GL_QUADS);

glVertex2d(0.05+j,0.15);

glVertex2d(0.11+j,0.15);

glVertex2d(0.09+j, 0.18);

glVertex2d(0.05+j,0.18 );

glEnd();

glColor3f(0.0f,0.8f,0.0f);

glBegin(GL_TRIANGLES);

glVertex2d(0.02+j,0.15);

glVertex2d(0.05+j,0.15);

glVertex2d(0.05+j, 0.18);

glEnd();

glColor3f(1.0f,0.0f,0.5f);

glBegin(GL_QUADS);

glVertex2d(0.06+j,0.12);

glVertex2d(0.09+j,0.12);

glVertex2d(0.09+j, 0.15);

glVertex2d(0.06+j, 0.15);

glEnd();



glColor3f(0.0f,1.0f,0.0f);

glBegin(GL_POLYGON);


int theta;

for(int i=0; i<2880; i++)

{
        theta=i*(3.1416/180);
        glVertex2f(0.04+j+0.02*sin(theta),0.12+0.02*cos(theta));
    }
    glEnd();
    glColor3f(0.0f,1.0f,0.0f);
    glBegin(GL_POLYGON);

    for(int i=0; i<2880; i++)
    {
        theta=i*(3.1416/180);
        glVertex2f(0.11+j+0.02*sin(theta),0.12+0.02*cos(theta));
    }

    glEnd();
    glutPostRedisplay();

}

void display(void)
{
    glClear (GL_COLOR_BUFFER_BIT);



    glBegin(GL_QUADS);
    glColor3f(0.2,0.2,0.2);
    glVertex3f(0.0, 0.0, 0.0); //background
    glVertex3f(1.0, 0.0, 0.0);
    glVertex3f(1.0, 1.0, 0.0);
    glVertex3f(0.0, 1.0, 0.0);

    glColor3f (0.3, 0.2,0);
```

```
glVertex3f(0.1, 0.3, 0.0); //field
glVertex3f(0.5, 0.3, 0.0);
glVertex3f(0.5, 0.6, 0.0);
glVertex3f(0.1, 0.6, 0.0);

glColor3f (0.1f, 0.2f, 0.0f);  //debox
glVertex3f(0.1, 0.35, 0.0);
glVertex3f(0.18, 0.35, 0.0);
glVertex3f(0.18, 0.55, 0.0);
glVertex3f(0.1, 0.55, 0.0);

glColor3f (0.1f, 0.3f, 0.0f);  //penalty box
glVertex3f(0.1, 0.4, 0.0);
glVertex3f(0.14, 0.4, 0.0);
glVertex3f(0.14, 0.5, 0.0);
glVertex3f(0.1, 0.5, 0.0);
glColor3f (0.1f, 0.2f, 0.0f);  //debox 2
glVertex3f(0.5, 0.35, 0.0);
glVertex3f(0.42, 0.35, 0.0);
glVertex3f(0.42, 0.55, 0.0);
glVertex3f(0.5, 0.55, 0.0);

glColor3f (0.1f, 0.3f, 0.0f);  //penalty box 2
glVertex3f(0.5, 0.4, 0.0);
glVertex3f(0.46, 0.4, 0.0);
glVertex3f(0.46, 0.5, 0.0);
glVertex3f(0.5, 0.5, 0.0);

glColor3f (0.0f, 0.1f, 0.0f);  //house
glVertex3f(0.7, 0.3, 0.0);
glVertex3f(0.85, 0.3, 0.0);
```

```
glVertex3f(0.85, 0.4, 0.0);
glVertex3f(0.7, 0.4, 0.0);

glColor3f (0.0f, 0.1f, 0.1f);  //path
glVertex3f(0.73, 0.2, 0.0);
glVertex3f(0.78, 0.2, 0.0);
glVertex3f(0.8, 0.3, 0.0);
glVertex3f(0.75, 0.3, 0.0);

glColor3f (0.5f, 0.5f, 0.5f);  //siri
glVertex3f(0.69, 0.29, 0.0);
glVertex3f(0.86, 0.29, 0.0);
glVertex3f(0.86, 0.3, 0.0);
glVertex3f(0.69, 0.3, 0.0);
glColor3f (0.5f, 0.5f, 0.5f);  //door
glVertex3f(0.75, 0.3, 0.0);
glVertex3f(0.8, 0.3, 0.0);
glVertex3f(0.8, 0.37, 0.0);
glVertex3f(0.75, 0.37, 0.0);


glColor3f (0.5f, 0.0f, 0.0f);  //door kopat
glVertex3f(0.75, 0.3, 0.0);
glVertex3f(0.78, 0.32, 0.0);
glVertex3f(0.78, 0.35, 0.0);
glVertex3f(0.75, 0.37, 0.0);

glColor3f (0.5f, 0.5f, 0.5f);  //windows1
glVertex3f(0.71, 0.33, 0.0);
glVertex3f(0.74, 0.33, 0.0);
glVertex3f(0.74, 0.37, 0.0);
```

```
glVertex3f(0.71, 0.37, 0.0);

glColor3f (0.5f, 0.5f, 0.5f);  //windows2
glVertex3f(0.81, 0.33, 0.0);
glVertex3f(0.84, 0.33, 0.0);
glVertex3f(0.84, 0.37, 0.0);
glVertex3f(0.81, 0.37, 0.0);
glColor3f (0.0f, 0.1f, 0.1f);  //road 1
glVertex3f(0.0, 0.0, 0.0);
glVertex3f(1.0, 0.0, 0.0);
glVertex3f(1.0, 0.2, 0.0);
glVertex3f(0.0, 0.2, 0.0);

glColor3f (0.0f, 0.1f, 0.1f);  //road 2
glVertex3f(0.54, 0.0, 0.0);
glVertex3f(0.64, 0.0, 0.0);
glVertex3f(0.64,1.0,0.0);
glVertex3f(0.54, 1.0, 0.0);

/* glColor3f (0.0f, 0.0f, 0.4f);  //river
 glVertex3f(0.0,0.7, 0.0);
 glVertex3f(1.0, 0.7, 0.0);
 glVertex3f(1.0,1.0,0.0);
 glVertex3f(0.0, 1.0, 0.0);*/

glColor3f (1.0f, 1.0f, 1.0f);  //dag 1
glVertex3f(0.02, 0.09, 0.0);
glVertex3f(0.07, 0.09, 0.0);
glVertex3f(0.07,0.11, 0.0);
glVertex3f(0.02,0.11,0.0);
```

```
glColor3f (1.0f, 1.0f, 1.0f);  //dag 2
glVertex3f(0.3, 0.09, 0.0);
glVertex3f(0.37, 0.09, 0.0);
glVertex3f(0.37,0.11, 0.0);
glVertex3f(0.3,0.11,0.0);

glColor3f (1.0f, 1.0f, 1.0f);  //dag 3
glVertex3f(0.6, 0.09, 0.0);
glVertex3f(0.68, 0.09, 0.0);
glVertex3f(0.68,0.11, 0.0);
glVertex3f(0.6,0.11,0.0);

glColor3f (1.0f, 1.0f, 1.0f);  //dag 4
glVertex3f(0.82, 0.09, 0.0);
glVertex3f(0.92, 0.09, 0.0);
glVertex3f(0.92,0.11, 0.0);
glVertex3f(0.82,0.11,0.0);


glColor3f (0.3, 0.2,0.0);  //tree1
glVertex3f(0.04, 0.25, 0.0);
glVertex3f(0.06, 0.25, 0.0);
glVertex3f(0.06,0.35, 0.0);
glVertex3f(0.04,0.35,0.0);

glColor3f (0.3, 0.2, 0.0f); //tree2
glVertex3f(0.93, 0.3, 0.0);
glVertex3f(0.95, 0.3, 0.0);
glVertex3f(0.95,0.4, 0.0);
glVertex3f(0.93,0.4,0.0);
glEnd();
```

```
glBegin(GL_TRIANGLES);
glColor3f(0.5,0.5,0.5);   //roof
glVertex3f(0.65,0.395,0.0);
glVertex3f(0.895,0.395,0.0);
glVertex3f(0.775,0.47,0.0);

glColor3f(0.5,0.3,0.1);   //shade
glVertex3f(0.67,0.4,0.0);
glVertex3f(0.88,0.4,0.0);
glVertex3f(0.775,0.46,0.0);

glColor3f(0.1f, 0.2f, 0.0f);   //t1 triangle
glVertex3f(0.02,0.35,0.0);
glVertex3f(0.08,0.35,0.0);
glVertex3f(0.05,0.4,0.0);



glColor3f(0.1f, 0.2f, 0.0f);
glVertex3f(0.02,0.37,0.0);
glVertex3f(0.08,0.37,0.0);
glVertex3f(0.05,0.42,0.0);

glColor3f(0.1f, 0.2f, 0.0f);
glVertex3f(0.02,0.39,0.0);
glVertex3f(0.08,0.39,0.0);
glVertex3f(0.05,0.44,0.0);
glColor3f(0.1f, 0.2f, 0.0f);   //t2 triangle
glVertex3f(0.91,0.4,0.0);
glVertex3f(0.97,0.4,0.0);
glVertex3f(0.94,0.46,0.0);
```

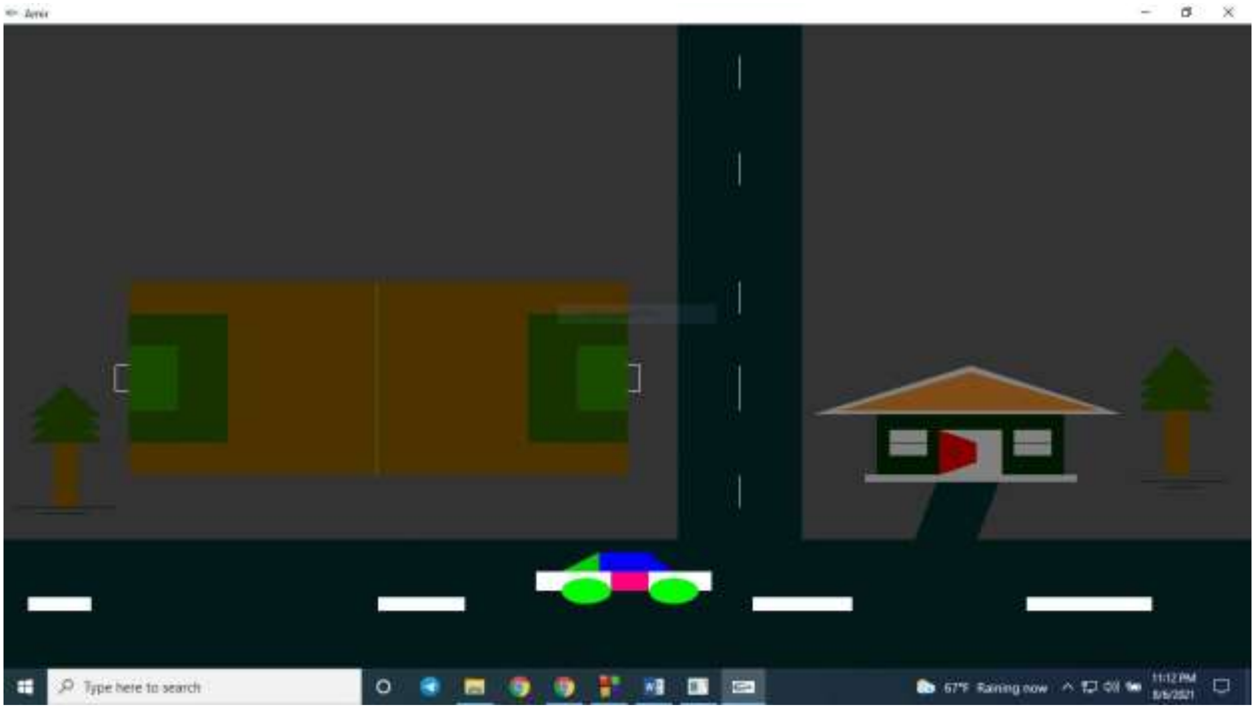
```
glColor3f(0.1f, 0.2f, 0.0f);
glVertex3f(0.91,0.42,0.0);
glVertex3f(0.97,0.42,0.0);
glVertex3f(0.94,0.48,0.0);

glColor3f(0.1f, 0.2f, 0.0f);
glVertex3f(0.91,0.44,0.0);
glVertex3f(0.97,0.44,0.0);
glVertex3f(0.94,0.5,0.0);
glEnd();
glBegin(GL_LINES);     //bar post
glColor3f(1,1,1);
glVertex3f(0.1,0.43,0.0);
glVertex3f(0.09,0.43,0.0);

glColor3f(1,1,1);
glVertex3f(0.09,0.43,0.0);
glVertex3f(0.09,0.47,0.0);

glColor3f(1,1,1);
glVertex3f(0.1,0.47,0.0);
glVertex3f(0.09,0.47,0.0);

glColor3f(1,1,1);      //bar post 2
glVertex3f(0.5,0.43,0.0);
glVertex3f(0.51,0.43,0.0);

glColor3f(1,1,1);
glVertex3f(0.51,0.43,0.0);
glVertex3f(0.51,0.47,0.0);
```

```
  glColor3f(1,1,1);
  glVertex3f(0.51,0.47,0.0);
  glVertex3f(0.5,0.47,0.0);

  glColor3f(0.2,0.5,0.0);    //middle line
  glVertex3f(0.3,0.3,0.0);
  glVertex3f(0.3,0.6,0.0);

  glColor3f(1,1,1);    //bar 1
  glVertex3f(0.59,0.25,0.0);
  glVertex3f(0.59,0.3,0.0);

  glColor3f(1,1,1);   //bar 2
  glVertex3f(0.59,0.4,0.0);
  glVertex3f(0.59,0.47,0.0);

  glColor3f(1,1,1);   //bar 3
  glVertex3f(0.59,0.55,0.0);
  glVertex3f(0.59,0.6,0.0);
 glColor3f(1,1,1);   //bar 4
  glVertex3f(0.59,0.75,0.0);
  glVertex3f(0.59,0.8,0.0);

  glColor3f(1,1,1);   //bar 5
  glVertex3f(0.59,0.9,0.0);
  glVertex3f(0.59,0.95,0.0);
  glColor3f(0,0.1,0);   //window 1 dag
  glVertex3f(0.71,0.35,0.0);
  glVertex3f(0.74,0.35,0.0);
```

```
    glColor3f(0,0.1,0);    //window 2 dag
    glVertex3f(0.81,0.35,0.0);
    glVertex3f(0.84,0.35,0.0);
    glColor3f(0,0.1,0);    //tree 1 dag
    glVertex3f(0.01,0.25,0.0);
    glVertex3f(0.09,0.25,0.0);

    glColor3f(0,0.1,0);    //tree 1 dag2
    glVertex3f(0.03,0.24,0.0);
    glVertex3f(0.07,0.24,0.0);

    glColor3f(0,0.1,0);    //tree 2 dag2
    glVertex3f(0.91,0.29,0.0);
    glVertex3f(0.98,0.29,0.0);
    glColor3f(0,0.1,0);    //tree 2 dag2
    glVertex3f(0.93,0.28,0.0);
    glVertex3f(0.96,0.28,0.0);

    glEnd();
    car();
    glFlush ();
}
void init (void)
{
    glClearColor (0.0, 0.0, 0.0, 0.0); //select clearing (background) color
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(0.0, 1.0, 0.0, 1.0, -10.0, 10.0);
}

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

```
{
   glutInit(&argc, argv);
   glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
   glutInitWindowSize (780, 600);
   glutInitWindowPosition (100, 100);
   glutCreateWindow ("Amir");
   init ();
   glutDisplayFunc(display);
   glutMainLoop();
   return 0;
}
```

output Screenshot:



Discussion:

In this project I designed a road and a car over the road. The road designed by quads and lines. And the car designed by quads, triangles and circles .The car is moving on the road. I used if else condition for moving the car. I also used a for loop to draw the circles which are the wheels of the car. I also designed a football field using by quads and lines. I also design a house using by quads, triangles and lines. I also design two trees using by quads triangles and lines. All of this design I will draw the graph paper before designing.

Thank You