

**Assignment Report of Computer Graphics Lab**

Course title: Computer Graphics Lab

Course code: CSE422

Submitted by,

Name: Rashidul Hasan Hridoy

ID: 171-15-8596

Section: O3

Department of Computer Science & Engineering,

Daffodil International University

Submitted to,

[S](https://sites.google.com/view/syedatanjilaatik)yeda Tanjila Atik

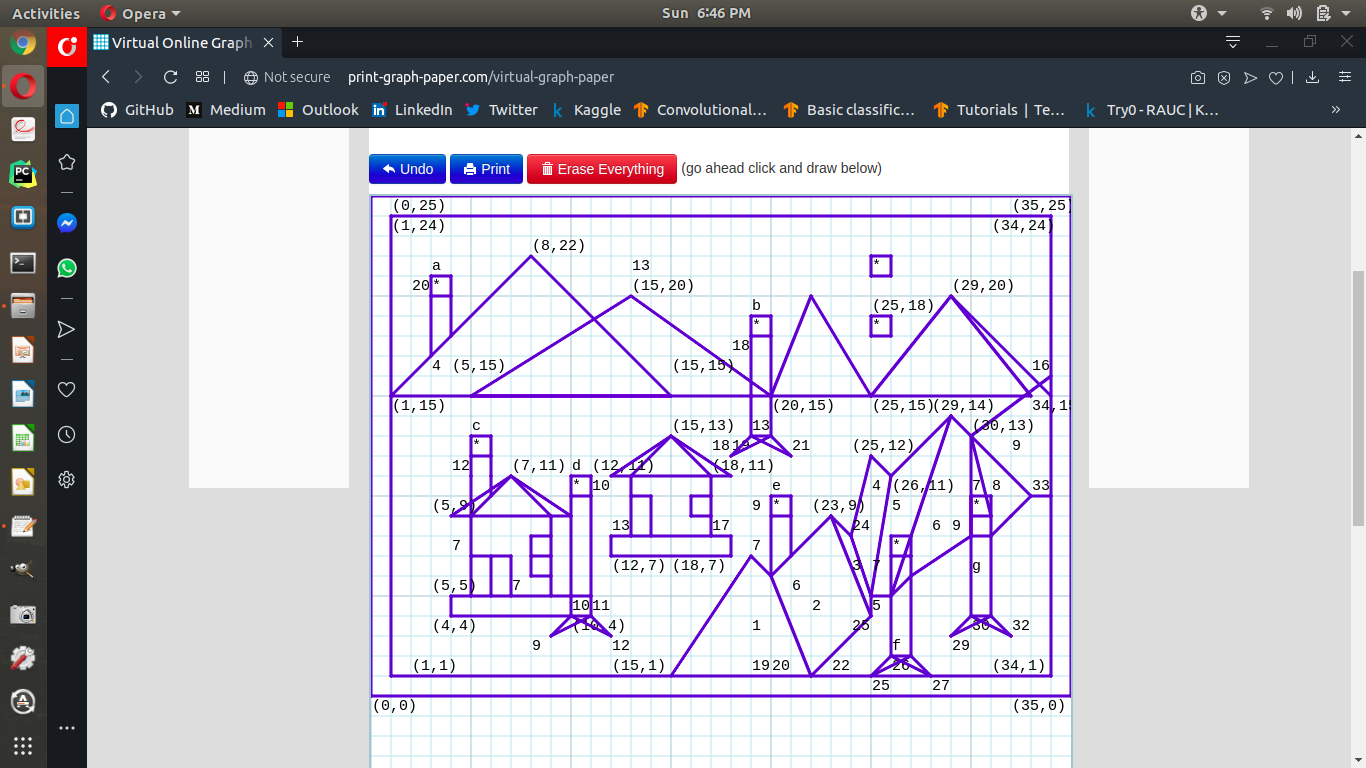
Lecturer

Department of Computer Science and Engineering,

Daffodil International University

Date of submission: 26-07-20

**Graph**

****

After the first try I adjust some points again, but can’t remove these from the graph. I’m sorry for that.

**Code**

#include <GL/glut.h>

#include <math.h>

void init(void)

{

// Set display window colour to white

glClearColor(0.0, 0.0, 0.0, 0.0);

// Set projection parameters

glMatrixMode(GL\_PROJECTION);

gluOrtho2D(0.0, 35.0, 0.0, 25.0);

}

void rashidul(void)

{

float theta;

int i;

// Clear display window

glClear(GL\_COLOR\_BUFFER\_BIT);

//first\_box

glColor3f(1,0.5,0);

glBegin(GL\_QUADS);

glVertex2i(0, 0);

glVertex2i(0, 25);

glVertex2i(35, 25);

glVertex2i(35, 0);

glEnd();

//second\_box

glColor3f(0,1,0.5);

glBegin(GL\_QUADS);

glVertex2i(1, 1);

glVertex2i(1, 24);

glVertex2i(34, 24);

glVertex2i(34, 1);

glEnd();

//sky

glColor3f(0.196078, 0.6, 0.8);

glBegin(GL\_QUADS);

glVertex2i(1,15);

glVertex2i(1,24);

glVertex2i(34,24);

glVertex2i(34,15);

glEnd();

//sun

glColor3f(1,0,0);

glBegin(GL\_POLYGON);

for(i=0;i<360;i++)

{

theta=i\*3.142/10;

glVertex2f(25.5+3.5\*cos(theta),18.50+3.5\*sin(theta));

}

glEnd();

glColor3f(1,1,0);

glBegin(GL\_POLYGON);

for(i=0;i<360;i++)

{

theta=i\*3.142/10;

glVertex2f(25.5+3.25\*cos(theta),18.50+3.25\*sin(theta));

}

glEnd();

//mountrain\_last

glColor3f(0.36, 0.25, 0.20);

glBegin(GL\_TRIANGLES);

glVertex2i(20,15);

glVertex2i(29,20);

glVertex2i(34,15);

glEnd();

//river\_started

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(15, 1);

glVertex2i(19, 7);

glVertex2i(20, 6);

glVertex2i(22, 1);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(20, 6);

glVertex2i(23, 9);

glVertex2i(25, 4);

glVertex2i(22, 1);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(24, 8);

glVertex2i(25, 12);

glVertex2i(26, 11);

glVertex2i(25, 5);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(23, 9);

glVertex2i(24, 8);

glVertex2i(25, 5);

glVertex2i(25, 4);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(24, 8);

glVertex2i(25, 12);

glVertex2i(26, 11);

glVertex2i(25, 5);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(26, 11);

glVertex2i(29, 14);

glVertex2i(26, 5);

glVertex2i(25, 5);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(29, 14);

glVertex2i(30, 13);

glVertex2i(30, 8);

glVertex2i(26, 5);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_TRIANGLES);

glVertex2i(30, 13);

glVertex2i(31, 8);

glVertex2i(30, 8);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_TRIANGLES);

glVertex2i(30, 13);

glVertex2i(33, 10);

glVertex2i(31, 8);

glEnd();

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_QUADS);

glVertex2i(30, 13);

glVertex2i(34, 16);

glVertex2i(34, 10);

glVertex2i(33, 10);

glEnd();

//tree\_c

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_QUADS);

glVertex2i(5,9);

glVertex2i(5,12);

glVertex2i(6,12);

glVertex2i(6,10);

glEnd();

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_QUADS);

glVertex2i(10,4);

glVertex2i(10,11);

glVertex2i(11,11);

glVertex2i(11,4);

glEnd();

//round

glColor3f(0.137255, 0.556863, 0.137255);

glBegin(GL\_POLYGON);

for(i=0;i<360;i++)

{

theta=i\*3.142/10;

glVertex2f(10.5+2.6\*cos(theta),10.5+2.6\*sin(theta));

}

glEnd();

//house\_1

glColor3f(0.309804,0.309804,0.184314);

glBegin(GL\_QUADS);

glVertex2i(4,4);

glVertex2i(4,5);

glVertex2i(10,5);

glVertex2i(10,4);

glEnd();

glColor3f(1.0, 1.0, 0.0);

glBegin(GL\_QUADS);

glVertex2i(5,9);

glVertex2i(9,9);

glVertex2i(9,5);

glVertex2i(5,5);

glEnd();

glColor3f(1,0,0);

glBegin(GL\_TRIANGLES);

glVertex2i(4,9);

glVertex2i(10,9);

glVertex2i(7,11);

glEnd();

glColor3f(1.0, 0.0, 0.0);

glBegin(GL\_QUADS);

glVertex2i(5,5);

glVertex2i(5,7);

glVertex2i(6,7);

glVertex2i(6,5);

glEnd();

glColor3f(1.0, 0.0, 0.0);

glBegin(GL\_QUADS);

glVertex2i(8,6);

glVertex2i(8,7);

glVertex2i(9,7);

glVertex2i(9,6);

glEnd();

//house\_2

glColor3f(0.309804,0.309804,0.184314);

glBegin(GL\_QUADS);

glVertex2i(12,7);

glVertex2i(12,8);

glVertex2i(18,8);

glVertex2i(18,7);

glEnd();

glColor3f(1.0, 1.0, 0.0);

glBegin(GL\_QUADS);

glVertex2i(13,8);

glVertex2i(13, 11);

glVertex2i(17, 11);

glVertex2i(17,8);

glEnd();

glColor3f(1,0,0);

glBegin(GL\_TRIANGLES);

glVertex2i(12,11);

glVertex2i(15,13);

glVertex2i(18,11);

glEnd();

glColor3f(1.0, 0.0, 0.0);

glBegin(GL\_QUADS);

glVertex2i(16,9);

glVertex2i(16,10);

glVertex2i(17,10);

glVertex2i(17, 9);

glEnd();

glColor3f(1.0, 0.0, 0.0);

glBegin(GL\_QUADS);

glVertex2i(13,8);

glVertex2i(13,10);

glVertex2i(14,10);

glVertex2i(14,8);

glEnd();

//mountain

glColor3f(0.36, 0.25, 0.20);

glBegin(GL\_TRIANGLES);

glVertex2i(1,15);

glVertex2i(8,22);

glVertex2i(15,15);

glEnd();

glColor3f(0.36, 0.25, 0.20);

glBegin(GL\_TRIANGLES);

glVertex2i(5,15);

glVertex2i(13,20);

glVertex2i(20,15);

glEnd();

glColor3f(0.36, 0.25, 0.20);

glBegin(GL\_TRIANGLES);

glVertex2i(20,15);

glVertex2i(22.5,20);

glVertex2i(25,15);

glEnd();

//tree\_a

glColor3f(0.52, 0.37, 0.26);

glBegin(GL\_QUADS);

glVertex2i(3,17);

glVertex2i(3,20);

glVertex2i(4,20);

glVertex2i(4,18);

glEnd();

//round

glColor3f(0.137255, 0.556863, 0.137255);

glBegin(GL\_POLYGON);

for(i=0;i<60;i++)

{

theta=i\*3.142/10;

glVertex2f(3.5+2\*cos(theta),21+2\*sin(theta));

}

glEnd();

//tree\_b

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_TRIANGLES);

glVertex2i(18,12);

glVertex2i(19,13);

glVertex2i(20,13);

glEnd();

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_TRIANGLES);

glVertex2i(21,12);

glVertex2i(19,14);

glVertex2i(19,13);

glEnd();

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_QUADS);

glVertex2i(19,13);

glVertex2i(19,18);

glVertex2i(20,18);

glVertex2i(20,13);

glEnd();

//round

glColor3f(0.137255, 0.556863, 0.137255);

glBegin(GL\_POLYGON);

for(i=0;i<60;i++)

{

theta=i\*3.142/10;

glVertex2f(19.5+2.5\*cos(theta),18.5+2.5\*sin(theta));

}

glEnd();

//tree\_d

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_TRIANGLES);

glVertex2i(12,3);

glVertex2i(11,4);

glVertex2i(10,4);

glEnd();

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_TRIANGLES);

glVertex2i(9,3);

glVertex2i(10,4);

glVertex2i(11,4);

glEnd();

//round

glColor3f(0.137255, 0.556863, 0.137255);

glBegin(GL\_POLYGON);

for(i=0;i<60;i++)

{

theta=i\*3.142/10;

glVertex2f(5.5+2\*cos(theta),12.5+2\*sin(theta));

}

glEnd();

//tree\_e

glColor3f(0.52, 0.37, 0.26);

glBegin(GL\_QUADS);

glVertex2i(20,6);

glVertex2i(20,9);

glVertex2i(21,9);

glVertex2i(21,7);

glEnd();

//round

glColor3f(0.137255, 0.556863, 0.137255);

glBegin(GL\_POLYGON);

for(i=0;i<360;i++)

{

theta=i\*3.142/10;

glVertex2f(20.5+1.8\*cos(theta),10.5+1.8\*sin(theta));

}

glEnd();

//tree\_f

glColor3f(0.52, 0.37, 0.26);

glBegin(GL\_TRIANGLES);

glVertex2i(25,1);

glVertex2i(26,2);

glVertex2i(27,2);

glEnd();

glColor3f(0.52, 0.37, 0.26);

glBegin(GL\_TRIANGLES);

glVertex2i(28,1);

glVertex2i(27,2);

glVertex2i(26,2);

glEnd();

glColor3f(0.52, 0.37, 0.26);

glBegin(GL\_QUADS);

glVertex2i(26,2);

glVertex2i(26,7);

glVertex2i(27,7);

glVertex2i(27,2);

glEnd();

//round

glColor3f(0.137255, 0.556863, 0.137255);

glBegin(GL\_POLYGON);

for(i=0;i<360;i++)

{

theta=i\*3.142/10;

glVertex2f(26.5+2\*cos(theta),5.5+2\*sin(theta));

}

glEnd();

//tree\_g

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_TRIANGLES);

glVertex2i(31,4);

glVertex2i(30,4);

glVertex2i(29,3);

glEnd();

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_TRIANGLES);

glVertex2i(30,4);

glVertex2i(31,4);

glVertex2i(32,3);

glEnd();

glColor3f(0.6, 0.8, 0.196078);

glBegin(GL\_QUADS);

glVertex2i(30,4);

glVertex2i(30,9);

glVertex2i(31,9);

glVertex2i(31,4);

glEnd();

//round

glColor3f(0.137255, 0.556863, 0.137255);

glBegin(GL\_POLYGON);

for(i=0;i<360;i++)

{

theta=i\*3.142/10;

glVertex2f(30.5+2.6\*cos(theta),7.5+2.6\*sin(theta));

}

glEnd();

glFlush();

}

int main(int argc, char\* argv[])

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE|GLUT\_RGB);

glutInitWindowPosition(100, 100);

glutInitWindowSize(1366, 768);

glutCreateWindow("171-15-8596");

init();

glutDisplayFunc(rashidul);

glutMainLoop();

return 0;

}

**Screenshot of the output**

