**Lab Report. 09**

**Subject: Computer Graphics Lab**

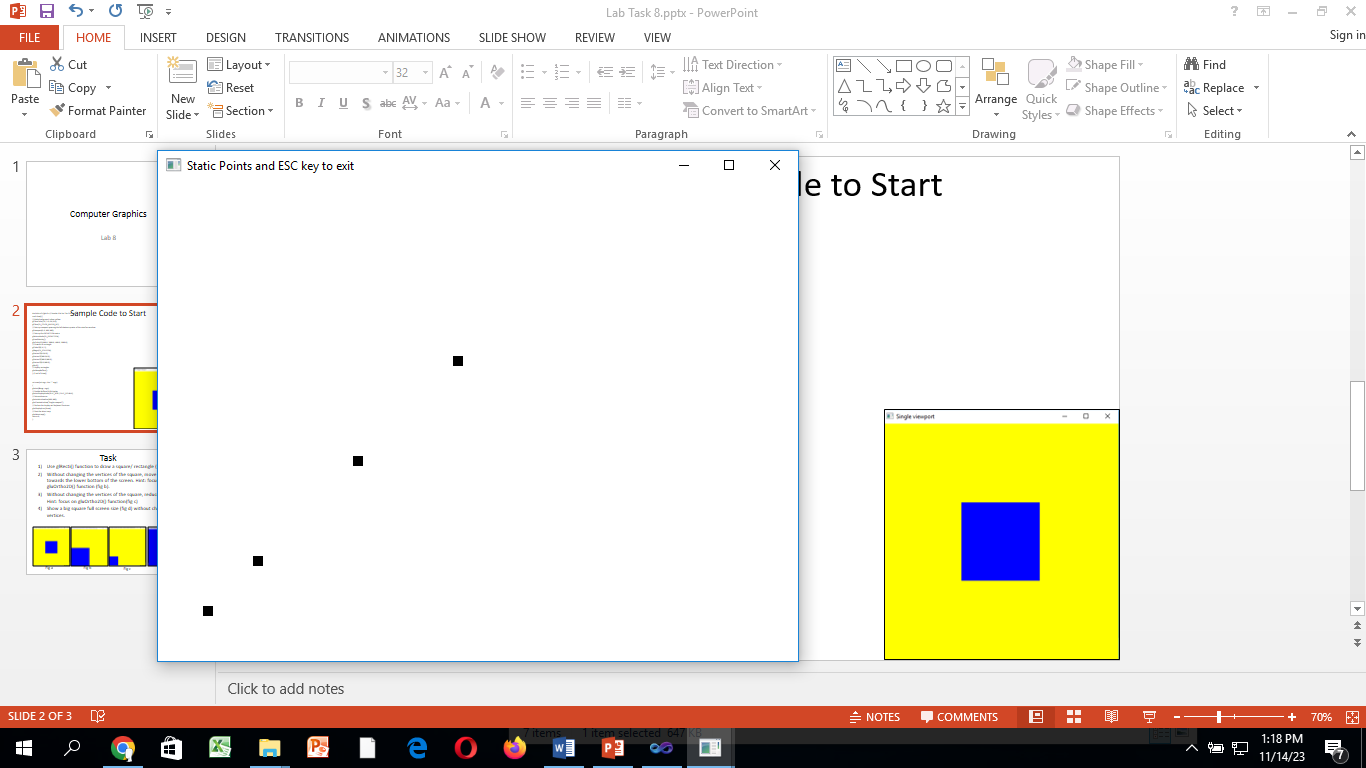
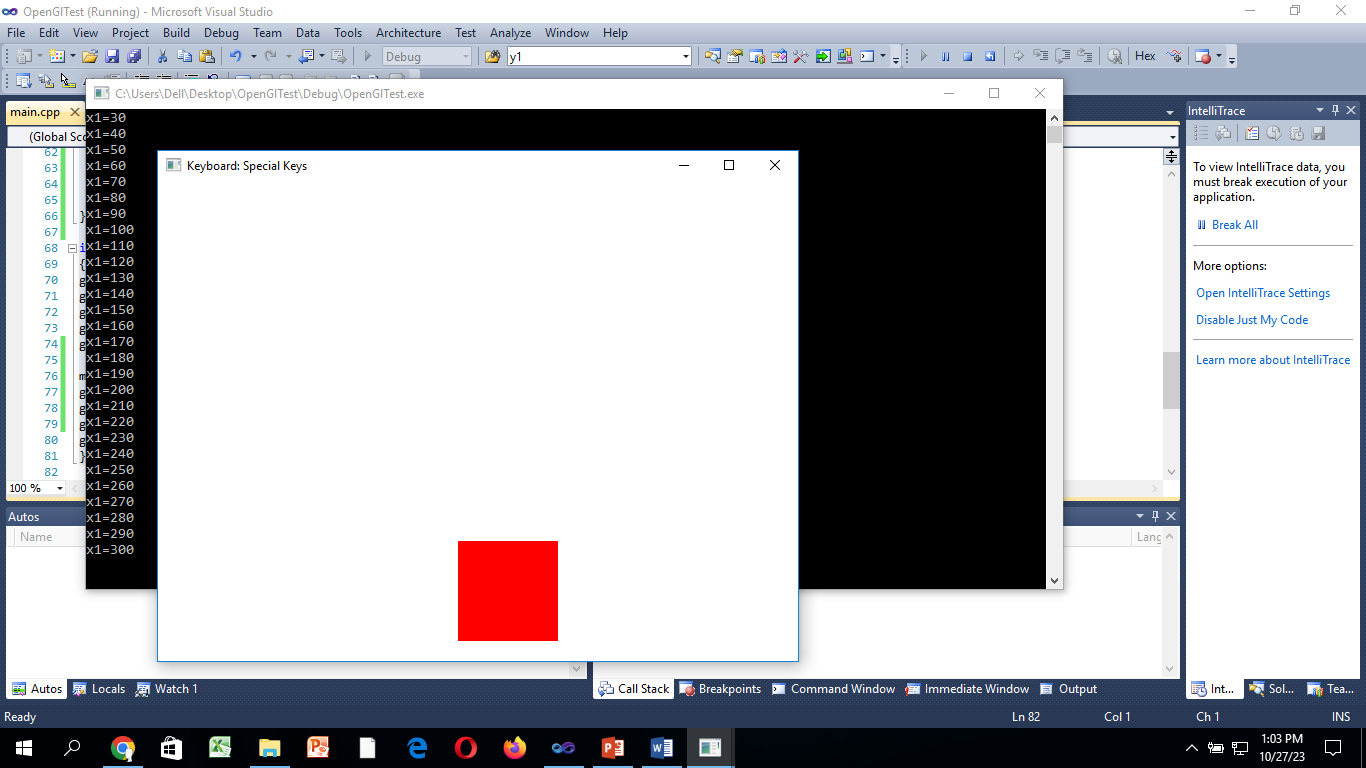


Submitted to: Dr. Junaid

Submitted by: Nayab Ashraf (RC-299)

**Date of Submission: 20 November, 2024**

National University of Modern Languages, Department of Computer Science, Rawalpindi Campus.



**Fig a Fig b**

**Task 01:**

**Write a program to draw dot on (x,y) mouse position when ‘d’ is pressed from the keyboard and exit on ‘E’. (fig a)**

#include<Windows.h>

#include<GL\GL.h>

#include<GL\glut.h>

#include<math.h>

void myInit()

{

glClearColor(1.0, 1.0, 1.0, 0.0);

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

glPointSize(10.0);

glMatrixMode(GL\_PROJECTION);

gluOrtho2D(0.0, 640.0, 0.0, 480.0);

}

void display(int x, int y)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glPointSize(10);

glBegin(GL\_POINTS);

glVertex2i(x, y);

glEnd();

glFlush();

}

// Keyboard method to allow ESC key to quit

void keyboard(unsigned char key, int x, int y)

{

if (key == 'e')

exit(0);

else if (key == 'd')

{

display(x, 480 - y);

}

}

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

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE);

glutInitWindowSize(640, 480);

glutInitWindowPosition(150, 150);

glutCreateWindow("Static Points and ESC key to exit");

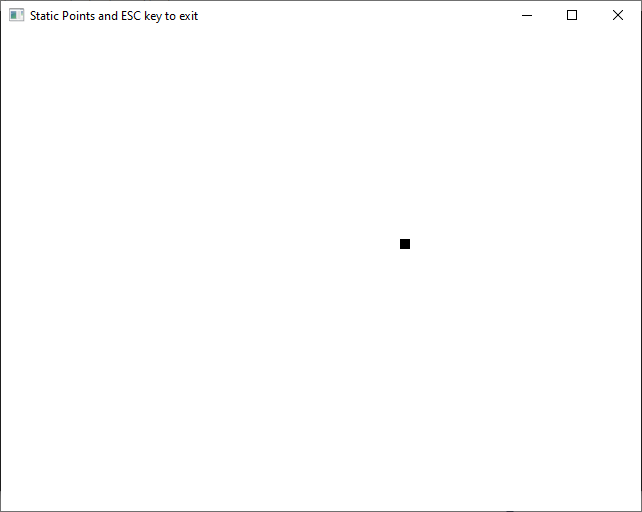
myInit();

//glutDisplayFunc(display);

glutKeyboardFunc(keyboard);

glutMainLoop();

}

****

**Task 02:**

**Write a program to move square left and right using arrow keys on the keyboard. (fig b)**

#include<Windows.h>

#include<GL\GL.h>

#include<GL\glut.h>

#include<math.h>

int dotX;

int dotY;

void myInit()

{

glClearColor(1.0, 1.0, 1.0, 0.0);

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

glPointSize(50.0);

glMatrixMode(GL\_PROJECTION);

gluOrtho2D(0.0, 640.0, 0.0, 480.0);

}

void display()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glFlush();

}

// Keyboard method to allow ESC key to quit

void keyboard(unsigned char key, int x, int y)

{

if (key == 'e')

exit(0);

else if (key == 'a')

{

dotX = x;

dotY = 480 - y;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'l')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotX = dotX - 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'r')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotX = dotX + 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'u')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotY = dotY + 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'd')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotY = dotY - 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

}

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

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE);

glutInitWindowSize(640, 480);

glutInitWindowPosition(150, 150);

glutCreateWindow("Static Points and ESC key to exit");

myInit();

glutDisplayFunc(display);

glutKeyboardFunc(keyboard);

glutMainLoop();

}

****

**Task 03:**

**Write a program to move circle up and down using arrow keys on the keyboard.**

#include<Windows.h>

#include<GL\GL.h>

#include<GL\glut.h>

#include<math.h>

int dotX;

int dotY;

void myInit()

{

glClearColor(1.0, 1.0, 1.0, 0.0);

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

glPointSize(50.0);

glMatrixMode(GL\_PROJECTION);

gluOrtho2D(0.0, 640.0, 0.0, 480.0);

}

void display()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glFlush();

}

// Keyboard method to allow ESC key to quit

void handleSpecialKeys(int key, int x, int y)

{

if (key == GLUT\_KEY\_F2)

exit(0);

else if (key == GLUT\_KEY\_F1)

{

dotX = x;

dotY = 480 - y;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == GLUT\_KEY\_LEFT)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotX = dotX - 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == GLUT\_KEY\_RIGHT)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotX = dotX + 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == GLUT\_KEY\_UP)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotY = dotY + 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == GLUT\_KEY\_DOWN)

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotY = dotY - 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

}

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

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE);

glutInitWindowSize(640, 480);

glutInitWindowPosition(150, 150);

glutCreateWindow("Static Points and ESC key to exit");

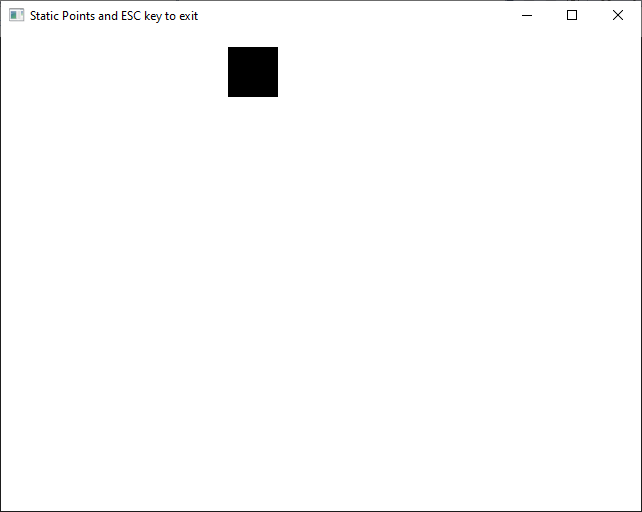
myInit();

glutDisplayFunc(display);

glutSpecialFunc(handleSpecialKeys);

glutMainLoop();

}

****

**Task 04:**

**Create any menu and map it to the right-click.**

#include<Windows.h>

#include<GL\GL.h>

#include<GL\glut.h>

#include<math.h>

int dotX;

int dotY;

void myInit()

{

glClearColor(1.0, 1.0, 1.0, 0.0);

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

glPointSize(50.0);

glMatrixMode(GL\_PROJECTION);

gluOrtho2D(0.0, 640.0, 0.0, 480.0);

}

void display()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glFlush();

}

// Keyboard method to allow ESC key to quit

void keyboard(unsigned char key, int x, int y)

{

if (key == 'e')

exit(0);

else if (key == 'a')

{

dotX = x;

dotY = 480 - y;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'l')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotX = dotX - 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'r')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotX = dotX + 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'u')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotY = dotY + 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

else if (key == 'd')

{

glClear(GL\_COLOR\_BUFFER\_BIT);

dotY = dotY - 5;

glBegin(GL\_POINTS);

glVertex2i(dotX, dotY);

glEnd();

glFlush();

}

}

void processMenuItems(int id)

{

if (id == 0)

exit(0);

else if (id == 1)

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

else if (id == 2)

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

else if (id == 3)

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

glutPostRedisplay();

}

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

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_SINGLE);

glutInitWindowSize(640, 480);

glutInitWindowPosition(150, 150);

glutCreateWindow("Static Points and ESC key to exit");

myInit();

glutDisplayFunc(display);

glutKeyboardFunc(keyboard);

//\_\_\_menu\_\_\_

int submenu = glutCreateMenu(processMenuItems);

glutAddMenuEntry("Red", 1);

glutAddMenuEntry("Green", 2);

glutAddMenuEntry("Blue", 3);

int main\_menu = glutCreateMenu(processMenuItems);

glutAddSubMenu("Color", submenu);

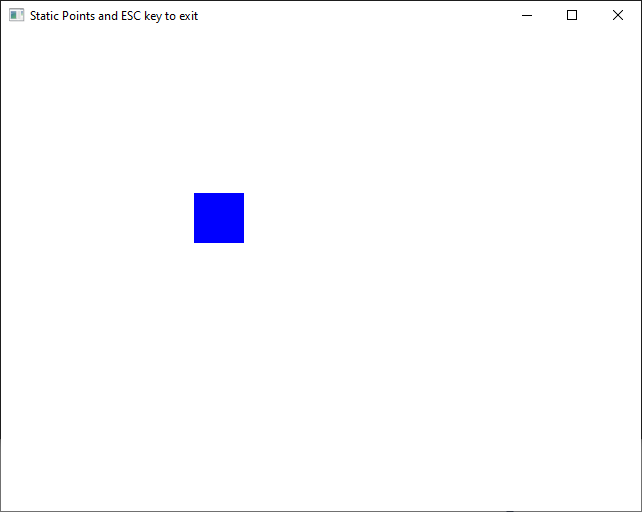
glutAddMenuEntry("Exit", 0);

glutAttachMenu(GLUT\_RIGHT\_BUTTON);

//\_\_\_menu End\_\_\_

glutMainLoop();

}

****