
# National University of Modern Languages

## LAB-Report No: 04

**Name:** Abdul Rafay

**Roll No:** Fl-23791

**Subject:** HCI

**Program:** BSSE 5

**Submitted To:** Mr Khateeb Khan

# Lab Tasks:

```
#include <Windows.h>
#include <GL/glut.h>
#include <vector>

const int sw = 640;
const int sh = 480;
std::vector<int> pointsX;
std::vector<int> pointsY;

std::vector<int> pointColor;

void myInit()
{
    glClearColor(1.0, 1.0, 1.0, 0.0);
    glPointSize(4.0);
    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    gluOrtho2D(0.0, sw, 0.0, sh);
}

void drawdot(int x, int y, int color)
{
    if (color == 0)
        glColor3f(0.8f, 0.0f, 0.0f);
    else if (color == 1)
        glColor3f(0.0f, 0.8f, 0.0f);

    glBegin(GL_POINTS);
    glVertex2i(x, y);
    glEnd();
}

void display()
{
    glClear(GL_COLOR_BUFFER_BIT);


    for (size_t i = 0; i < pointsX.size(); ++i)
        drawdot(pointsX[i], pointsY[i], pointColor[i]);

    glFlush();
}

void mymousefunc(int button, int state, int mousex, int mousey)
{
    if (state == GLUT_DOWN)
    {
        int y = sh - mousey;

        if (button == GLUT_LEFT_BUTTON)
        {
            pointsX.push_back(mousex);
            pointsY.push_back(y);
            pointColor.push_back(0);
```

```
        }
        else if (button == GLUT_RIGHT_BUTTON)
        {
            pointsX.push_back(mousex);
            pointsY.push_back(y);
            pointColor.push_back(1);
        }

        glutPostRedisplay();
    }
}

int main(int argc, char** argv)
{
    glutInit(&argc, argv);
    glutInitDisplayMode(GLUT_SINGLE);
    glutInitWindowSize(sw, sh);
    glutInitWindowPosition(150, 150);
    glutCreateWindow("Draw Red and Green Dots");

    myInit();
    glutDisplayFunc(display);
    glutMouseFunc(mymousefunc);
    glutMainLoop();

    return 0;
}
```

**OUTPUT:**