|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  | | --- | | **An opengl program to Draw chess board** |         **SUBMITTED BY**   |  |  |  | | --- | --- | --- | | Name | ID | Section | | Kabid Yeiad | 202-15-14440 | 57\_A |   **SUBMITTED TO**  **Deawan Rakin Ahamed Remal,**  **Lecturer**  **Dept. of CSE**  **Daffodil International University**   |  | | --- | |  |   Submitted on November 1, 2023 |

**An opengl program to draw chess board**

**Code:**

#include <stdio.h>

#include <GL/glut.h>

int x = 50, y = 50;

bool isBlack = true;

void whiteBox(int x, int y)

{

    glBegin(GL\_LINE\_LOOP);

    glVertex2i(x, y);

    glVertex2i(x, y + 50);

    glVertex2i(x + 50, y + 50);

    glVertex2i(x + 50, y);

    glEnd();

}

void blackBox(int x, int y)

{

    glBegin(GL\_POLYGON);

    glVertex2i(x, y);

    glVertex2i(x, y + 50);

    glVertex2i(x + 50, y + 50);

    glVertex2i(x + 50, y);

    glEnd();

}

void myDisplay(void)

{

    glClear(GL\_COLOR\_BUFFER\_BIT);

    glColor3f(0.0, 0.0, 0.0);

    glPointSize(1.0);

    for (int i = 0; i < 8; i++)

    {

        if (i % 2 == 0)

        {

            isBlack = true;

        }

        else

        {

            isBlack = false;

        }

        for (int j = 0; j < 8; j++)

        {

            if (isBlack)

            {

                blackBox(x, y);

                isBlack = false;

            }

            else

            {

                whiteBox(x, y);

                isBlack = true;

            }

            x += 50;

        }

        y += 50;

        x = 50;

    }

    blackBox(100, 100);

    whiteBox(150, 100);

    glFlush();

}

void myInit(void)

{

    glClearColor(1.0, 1.0, 1.0, 0.0);

    glColor3f(0.0f, 0.0f, 0.0f);

    glPointSize(4.0);

    glMatrixMode(GL\_PROJECTION);

    glLoadIdentity();

    gluOrtho2D(0.0, 640.0, 0.0, 480.0);

}

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

{

    glutInit(&argc, argv);

    glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

    glutInitWindowSize(800, 600);

    glutInitWindowPosition(100, 150);

    glutCreateWindow("Draw an 8X8 chess board using loop");

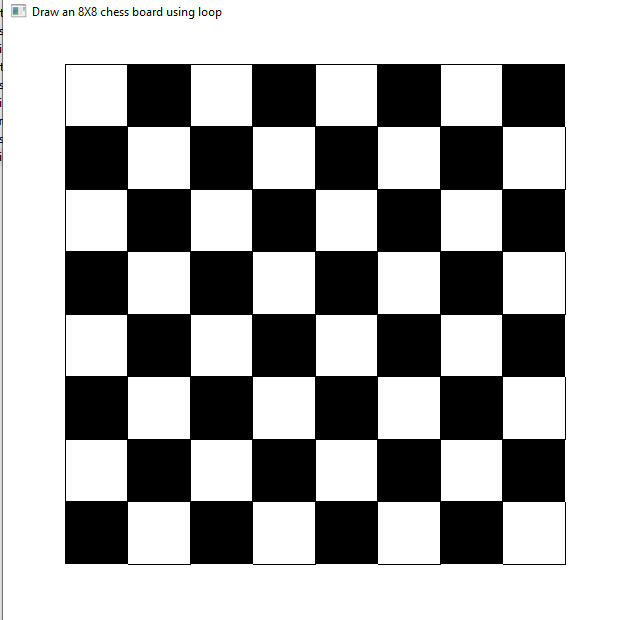
    glutDisplayFunc(myDisplay);

    myInit();

    glutMainLoop();

}

**Output**:



**Graph**:

**Discussion**:

In this experiment, we implemented an OpenGL program to draw an 8x8 chess board. The code provided utilizes OpenGL to render the board, consisting of alternating black and white squares.

We defined two functions, whiteBox and blackBox, to draw the white and black squares of the board, respectively. The graph provides a visual representation of the coordinates used to draw the chess board using OpenGL. The x-axis represents the horizontal position, while the y-axis represents the vertical position. Labels have been added to mark the positions of individual squares on the chess board. Each square is uniquely identified by its row and column coordinates. For example, the square at (0,0) represents the bottom-left corner, while the square at (8,8) represents the top-right corner.