CG_4

1 message

C43-Mayur Rajgude <mrajgude2003@gmail.com> Wed, 6 Dec, 2023 at 11:45 pm
To: C43-Mayur Rajgude <mrajgude2003@gmail.com>

```
#include<iostream>
#include<math.h>
#include<graphics.h>
using namespace std;

class matrix {
public:
    int n, i, j, tx, ty, k, sum, sx, sy;
    double a[6][3], b[6][3], mult[6][3], mat3[6][3];
    double p, q, r;
    double ang = 0, angle = 0;

public:
    void get() {
        cout << "\nEnter the number of vertices of the polygon: ";
        cin >> n;

        cout << "\nEnter the x and y coordinates:\n";
        for (i = 0; i < n; i++) {
            cout << "Enter coordinates for vertex " << i + 1 << ": ";
            cin >> b[i][0] >> b[i][1];
            b[i][2] = 1;
        }

        // Display original object matrix
        cout << "\nOriginal coordinates are:\n";
        for (i = 0; i < n; i++) {
            for (j = 0; j < 3; j++) {
                cout << b[i][j] << "\t";
            }
            cout << "\n";
        }
    }

    void identitymat() {
        for (i = 0; i < n; i++) {
            for (j = 0; j < 3; j++) {
                if (i == j) {
                    a[i][j] = 1;
                } else {
                    a[i][j] = 0;
                }
            }
        }
    }

    void trans() {
        cout << "Enter values of tx and ty: ";
        cin >> tx >> ty;
        a[2][0] = tx;
        a[2][1] = ty;
        cout << "Matrix is:\n";
        for (i = 0; i < n; i++) {
            for (j = 0; j < 3; j++) {
                cout << a[i][j] << "\t";
            }
            cout << "\n";
        }
```

```
}

void scale() {
    cout << "\nEnter the values of sx and sy: ";
    cin >> sx >> sy;
    a[0][0] = sx;
    a[1][1] = sy;
    cout << "\nMatrix is:\n";
    // To display scaling matrix
    for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
            cout << a[i][j] << "\t";
        }
        cout << "\n";
    }
}

void rot() {
    cout << "Enter the angle: ";
    cin >> ang;
    angle = (ang * 3.142) / 180;
    q = sin(angle);
    p = cos(angle);
    r = -sin(angle);
    a[0][0] = p;
    a[0][1] = q;
    a[1][0] = r;
    a[1][1] = p;
    cout << "Transformation matrix is:\n";
    for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
            cout << a[i][j] << "\t";
        }
        cout << "\n";
    }
}

void multi() {
    cout << "\nMultiplying two matrices...\n";
    for (i = 0; i < n; i++) {
        for (j = 0; j < 3; j++) {
            sum = 0;
            for (k = 0; k < 3; k++) {
                sum = sum + b[i][k] * a[k][j];
            }
            mat3[i][j] = sum;
        }
    }
}

void display() {
    cout << "\nMultiplication of two matrices:\n";
    for (i = 0; i < n; i++) {
        for (j = 0; j < 3; j++) {
            cout << mat3[i][j] << " ";
        }
        cout << "\n";
    }

    int gd = DETECT, gm;
    initgraph(&gd, &gm, NULL);
    for (int i = 0; i < n - 1; i++) {
        line(b[i][0], b[i][1], b[i + 1][0], b[i + 1][1]);
    }
    line(b[n - 1][0], b[n - 1][1], b[0][0], b[0][1]);

    for (int i = 0; i < n - 1; i++) {
        line(mat3[i][0], mat3[i][1], mat3[i + 1][0], mat3[i + 1][1]);
    }
    line(mat3[n - 1][0], mat3[n - 1][1], mat3[0][0], mat3[0][1]);
```

```
        delay(5000);
        closegraph();
    }
};

int main() {
    matrix g;
    int ch;
    char ans;
    g.get();
    g.identitymat();
    do {
        cout << "Menu\n1. Translation\n2. Scaling\n3. Rotation\n";
        cin >> ch;
        switch (ch) {
            case 1:
                g.trans();
                g.multi();
                g.display();
                break;
            case 2:
                g.scale();
                g.multi();
                g.display();
                break;
            case 3:
                g.rot();
                g.multi();
                g.display();
                break;
        }
        cout << "Do you want to continue? (Y/N): ";
        cin >> ans;
    } while (ans == 'Y' || ans == 'y');

    return 0;
}
```