

IMPLEMENT 2D TRANSFORMATIONS: TRANSLATION, SCALING, ROTATION, REFLECTION, SHEAR.

```
#include <stdio.h>

#include <conio.h>

#include <graphics.h>

#include <math.h>


// Function to draw the triangle

void drawTriangle(int x1, int y1, int x2, int y2, int x3, int y3) {

    line(x1, y1, x2, y2);

    line(x2, y2, x3, y3);

    line(x3, y3, x1, y1);

}


// Function for translation

void translate(int *x1, int *y1, int *x2, int *y2, int *x3, int *y3, int tx, int ty) {

    *x1 += tx;

    *y1 += ty;

    *x2 += tx;

    *y2 += ty;

    *x3 += tx;

    *y3 += ty;

}


// Function for rotation

void rotate(int *x1, int *y1, int *x2, int *y2, int *x3, int *y3, float angle) {

    float radians = angle * (3.14 / 180);

    int tempX, tempY;
```

```

// Rotate each vertex

tempX = *x1;

tempY = *y1;

*x1 = tempX * cos(radians) - tempY * sin(radians);

*y1 = tempX * sin(radians) + tempY * cos(radians);

tempX = *x2;

tempY = *y2;

*x2 = tempX * cos(radians) - tempY * sin(radians);

*y2 = tempX * sin(radians) + tempY * cos(radians);


tempX = *x3;

tempY = *y3;

*x3 = tempX * cos(radians) - tempY * sin(radians);

*y3 = tempX * sin(radians) + tempY * cos(radians);
}


// Function for scaling

void scale(int *x1, int *y1, int *x2, int *y2, int *x3, int *y3, float sx, float sy) {

    *x1 *= sx;

    *y1 *= sy;

    *x2 *= sx;

    *y2 *= sy;

    *x3 *= sx;

    *y3 *= sy;
}


// Function for reflection

void reflect(int *x1, int *y1, int *x2, int *y2, int *x3, int *y3, int axis) {

    // Reflection about Y-axis

```

```

if (axis == 1) {

    *x1 = -(*x1);

    *x2 = -(*x2);

    *x3 = -(*x3);

}

// Reflection about X-axis

else if (axis == 2) {

    *y1 = -(*y1);

    *y2 = -(*y2);

    *y3 = -(*y3);

}

}

// Function for shearing

void shear(int *x1, int *y1, int *x2, int *y2, int *x3, int *y3, float shx, float shy) {

    *x1 += *y1 * shx;

    *x2 += *y2 * shx;

    *x3 += *y3 * shx;

    *y1 += *x1 * shy;

    *y2 += *x2 * shy;

    *y3 += *x3 * shy;

}

int main() {

    int gd = DETECT, gm, choice;

    int x1, y1, x2, y2, x3, y3;

    initgraph(&gd, &gm, "C:\\TURBOC3\\bgi");

```

```
printf("Enter the coordinates of the triangle (x1 y1 x2 y2 x3 y3): ");
```

```
scanf("%d%d%d%d%d%d", &x1, &y1, &x2, &y2, &x3, &y3);
```

```
// Main menu loop
```

```
printf("\nMenu:\n");
```

```
printf("1. Translate Triangle\n");
```

```
printf("2. Rotate Triangle\n");
```

```
printf("3. Scale Triangle\n");
```

```
printf("4. Reflect Triangle\n");
```

```
printf("5. Shear Triangle\n");
```

```
printf("6. Exit\n");
```

```
printf("Choose a transformation (1-6): ");
```

```
scanf("%d", &choice);
```

```
switch (choice) {
```

```
case 1: {
```

```
    int tx, ty;
```

```
    clrscr();
```

```
    printf("Enter translation values (tx ty): ");
```

```
    scanf("%d%d", &tx, &ty);
```

```
    setcolor(WHITE);
```

```
    drawTriangle(x1, y1, x2, y2, x3, y3);
```

```
    translate(&x1, &y1, &x2, &y2, &x3, &y3, tx, ty);
```

```
    printf("Triangle after translation\n");
```

```
    delay(1000);
```

```
    setcolor(GREEN);
```

```
    drawTriangle(x1, y1, x2, y2, x3, y3);
```

```

        break;
    }

    case 2: {
        float angle;

        clrscr();

        printf("Enter angle of rotation: ");

        scanf("%f", &angle);

        setcolor(WHITE);

        drawTriangle(x1, y1, x2, y2, x3, y3);

        rotate(&x1, &y1, &x2, &y2, &x3, &y3, angle);

        setcolor(RED);

        delay(10000);

        drawTriangle(x1, y1, x2, y2, x3, y3);

        break;
    }

```

```

    case 3: {
        float sx, sy;

        clrscr();

        printf("Enter scaling factors (sx sy): ");

        scanf("%f%f", &sx, &sy);

        setcolor(WHITE);

        drawTriangle(x1, y1, x2, y2, x3, y3);

        scale(&x1, &y1, &x2, &y2, &x3, &y3, sx, sy);

        setcolor(BLUE);

        delay(1000);

        drawTriangle(x1, y1, x2, y2, x3, y3);

        break;
    }

```

```
}
```

```
case 4: {
```

```
    int axis;
```

```
    clrscr();
```

```
    printf("Enter 1 for reflection about Y-axis or 2 for reflection about X-axis: ");
```

```
    scanf("%d", &axis);
```

```
    printf("The current coordinates are :\n");
```

```
    printf("%d,%d %d,%d %d,%d\n",x1, y1, x2, y2, x3, y3);
```

```
    reflect(&x1, &y1, &x2, &y2, &x3, &y3, axis);
```

```
    printf("The coordinates after reflection are :\n");
```

```
    printf("%d,%d %d,%d %d,%d\n",x1, y1, x2, y2, x3, y3);
```

```
    break;
```

```
}
```

```
case 5: {
```

```
    float shx, shy;
```

```
    printf("Enter shear factors (shx shy): ");
```

```
    scanf("%f%f", &shx, &shy);
```

```
    setcolor(WHITE);
```

```
    drawTriangle(x1, y1, x2, y2, x3, y3);
```

```
    shear(&x1, &y1, &x2, &y2, &x3, &y3, shx, shy);
```

```
    setcolor(MAGENTA);
```

```
    drawTriangle(x1, y1, x2, y2, x3, y3);
```

```
    break;
```

```
}
```

```
case 6:
```

```
    printf("Exiting...\n");
```

```
    getch();
```

```
        break;

    default:

        printf("Invalid choice. Please try again.\n");

    }

    getch();

    closegraph();

    return 0;

}
```