Computer Graphics : Transformation (Translate, Scale and Rotate)

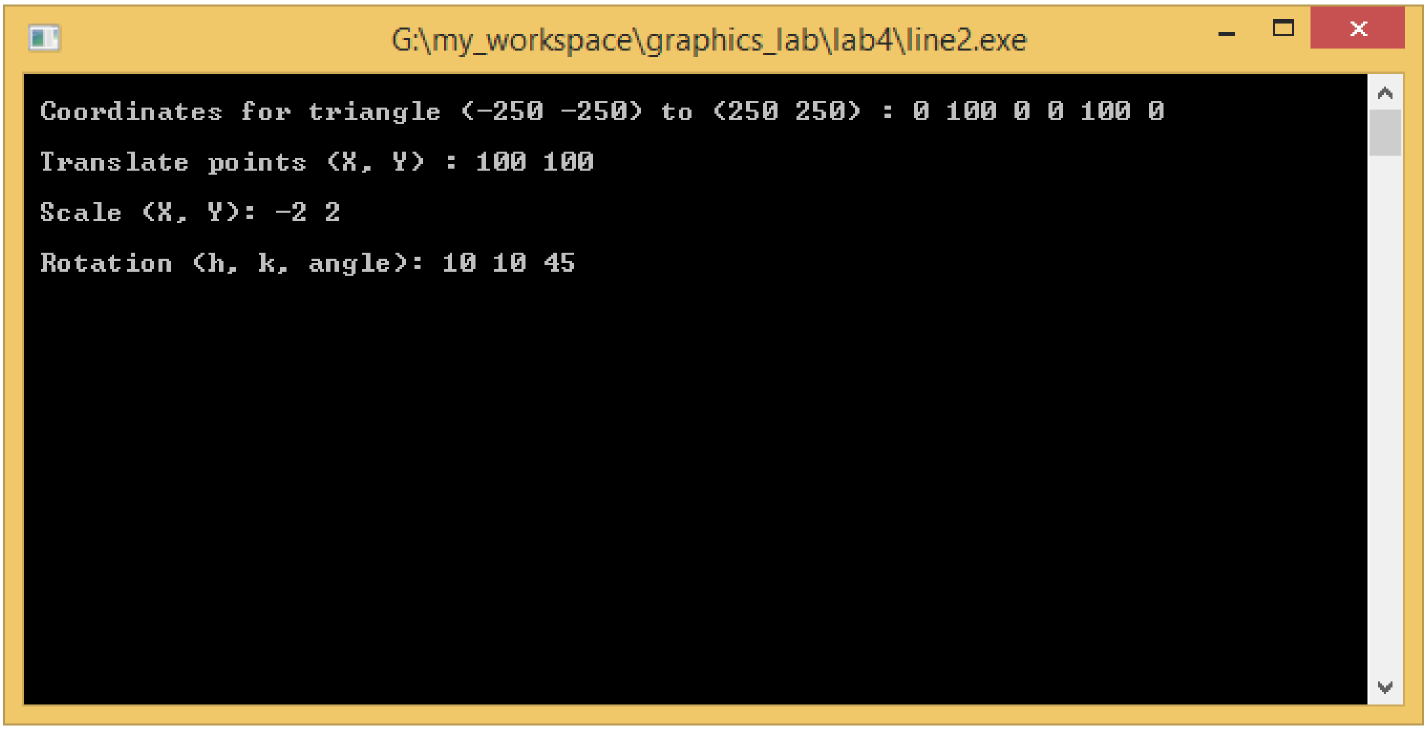
Name : Obaidur Rahamn

Reg : 12101006

Code

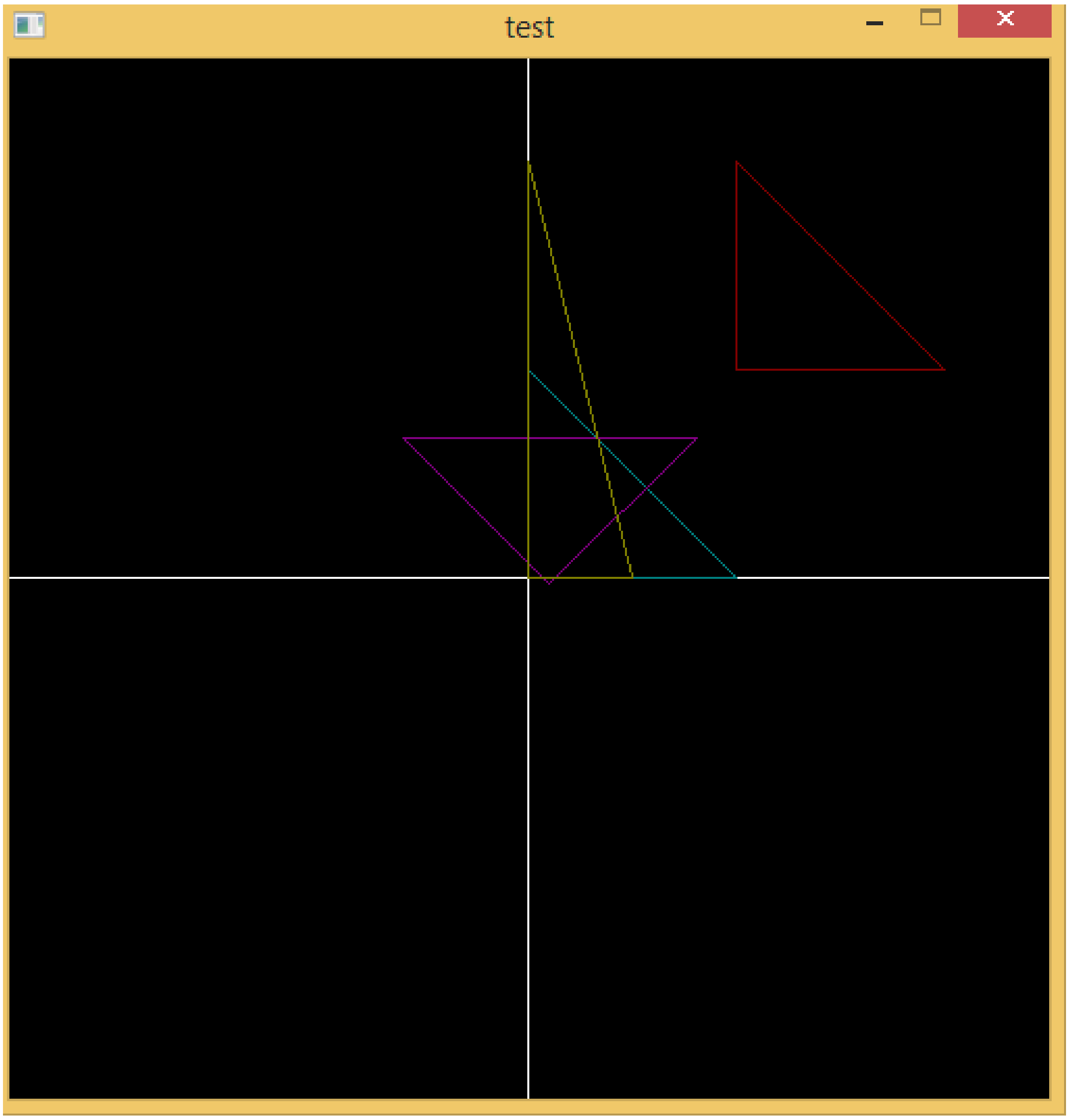
```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
#include<math.h>

void joinPoints(int x0, int y0,  int x1,int y1, int color){
    int x, y, dx, dy, e;
    dx = x1 - x0; dy = y1 - y0;
    int tmp;
    if (abs(dy) <= abs(dx))
    {
        if (x1 < x0)
        {
            tmp = x0; x0 = x1; x1 = tmp;
            tmp = y0; y0 = y1; y1 = tmp;
        }

        dx = x1 - x0; dy = y1 - y0;
        if (y0<=y1)
        {
            e = -dx;
            x = x0; y = y0;
            for (int i = 0; i <= dx; i++)
            {
                putpixel(x, y, color);
                x++; e = e + 2 * dy;
                if (e >= 0)
                {
                    y++;
                    e = e - 2 * dx;
                }
            }
        }
        else
        {
            e = dx;
            x = x0; y = y0;
            for (int i = 0; i <= dx; i++)
            {
                putpixel(x, y, color);
                x++; e = e + 2 * dy;
                if (e <= 0)
                {
                    --y;
                    e = e + 2 * dx;
                }
            }
        }
    }
    else
    {
        if (y1 < y0)
        {
            tmp = x0; x0 = x1; x1 = tmp;
            tmp = y0; y0 = y1; y1 = tmp;
        }

        dx = x1 - x0; dy = y1 - y0;
        if (x0 <= x1)
        {
            e = -dy;
            x = x0; y = y0;
            for (int i = 0; i <= dy; i++)
            {
                putpixel(x, y, color);
                y++; e = e + 2 * dx;
                if (e >= 0)
                {
                    x++;
                    e = e - 2 * dy;
                }
            }
        }
        else
        {
            e = dy;
            x = x0; y = y0;
            for (int i = 0; i <= dy; i++)
            {
                putpixel(x, y, color);
                y++; e = e + 2 * dx;
                if (e <= 0)
                {
                    --x;
                    e = e + 2 * dy;
                }
            }
        }
    }
}

void plotLine(int x0, int y0,  int x1,int y1, int color){
    int cx = getmaxx()/2;
    int cy = getmaxy()/2;

    x0 = cx+x0;
    x1 = cx+x1;

    y0 = cy-y0;
    y1 = cy-y1;

    joinPoints(x0, y0, x1, y1, color);
}

void drawAxis(){
    int x = getmaxx();
    int y = getmaxy();
    joinPoints(0, y/2, x, y/2, WHITE);
    joinPoints(x/2, 0, x/2, y, WHITE);
}

void plotTriangle(int x0, int y0, int x1, int y1, int x2, int y2, int color){
    plotLine(x0, y0, x1, y1, color);
    plotLine(x0, y0, x2, y2, color);
    plotLine(x1, y1, x2, y2, color);
}

int getXPrime(int x, int y, int h, int k, int t){
    double tmp1 = cos((t*3.1416)/180);
    double tmp2 = sin((t*3.1416)/180);
    double xP = (tmp1*(x-h))+(tmp2*(k-y))+h+.5;
    return (int)xP;
}

int getYPrime(int x, int y, int h, int k, int t){
    double tmp1 = cos((t*3.1416)/180);
    double tmp2 = sin((t*3.1416)/180);
    double yP = (tmp2*(x-h))+(tmp1*(y-k))+k+.5;
    return (int)yP;
}

int getXscaled(int x, int sx){
    float tmp;
    if(sx>=0)return x*sx;
    tmp = x/abs(sx);
    return (int)(tmp+.5);
}

int getYscaled(int y, int sy){
    float tmp;
    if(sy>=0)return y*sy;
    tmp = y/abs(sy);
    return (int)(tmp+.5);
}

int main()
{
    int winW = 500;
    int winH = 500;

    int x1, y1,x2, y2,x3, y3,tx, ty,sx, sy,h, k, ang;

    printf("\n Coordinates for triangle (%d %d) to (%d %d) : ",
           -winW/2, -winH/2, winW/2, winH/2);
    scanf("%d %d %d %d %d %d", &x1, &y1, &x2, &y2, &x3, &y3);

    printf("\n Translate points (X, Y) : ");
    scanf("%d %d", &tx, &ty);

    printf("\n Scale (X, Y): ");
    scanf("%d %d", &sx, &sy);

    printf("\n Rotation (h, k, angle): ");
    scanf("%d %d %d", &h, &k, &ang);

    initwindow(winW,winH,"test");
    drawAxis();

    plotTriangle(x1, y1, x2, y2, x3, y3, 3);

    // translate
    plotTriangle(x1+tx, y1+ty, x2+tx, y2+ty, x3+tx, y3+ty, 4);

    // rotate
    plotTriangle(getXPrime(x1, y1, h, k, ang), getYPrime(x1, y1, h, k, ang),
                 getXPrime(x2, y2, h, k, ang), getYPrime(x2, y2, h, k, ang),
                 getXPrime(x3, y3, h, k, ang), getYPrime(x3, y3, h, k, ang), 5);

    // scaling
    plotTriangle(getXscaled(x1, sx), getYscaled(y1, sy),
                 getXscaled(x2, sx), getYscaled(y2, sy),
                 getXscaled(x3, sx), getYscaled(y3, sy), 6);

    getch();
    closegraph();
    return 0;
}
```

Input



```
Coordinates for triangle (-250 -250) to (250 250) : 0 100 0 0 100 0

Translate points (X, Y) : 100 100

Scale (X, Y): -2 2

Rotation (h, k, angle): 10 10 45
```

Output


test