#include<iostream.h>

#include<graphics.h>

#include<stdio.h>

#include<conio.h>

#include<math.h>

//using namespace std;

int main()

{

    int gd=DETECT,gm,s;

    initgraph(&gd,&gm,"C:\\TurboC3\\BGI");

    cout<<"1.Translation\n2.Rotation\n3.Scaling\n4.Reflection\n5.Shearing   "<<endl;

    cout<<"Selection:";

    cin>>s;

    switch(s)

    {

        case 1:

            {   int x1=200,y1=150,x2=300,y2=250;

                int tx=50,ty=50;

                cout<<"Rectangle before translation"<<endl;

                setcolor(3);

                rectangle(x1,y1,x2,y2);

                setcolor(4);

                cout<<"Rectangle after translation"<<endl;

                rectangle(x1+tx,y1+ty,x2+tx,y2+ty);

                getch();

                break;

            }

        case 2:

            {  long x1=200,y1=200,x2=300,y2=300;

                double a;

                cout<<"Rectangle with rotation"<<endl;

                setcolor(3);

                rectangle(x1,y1,x2,y2);

                cout<<"Angle of rotation:";

                cin>>a;

                a=(a\*3.14)/180;

                long xr=x1+((x2-x1)\*cos(a)-(y2-y1)\*sin(a));

                long yr=y1+((x2-x1)\*sin(a)+(y2-y1)\*cos(a));

                setcolor(2);

                rectangle(x1,y1,xr,yr);

                getch();

                break;}

        case 3:

            {

                int x1=30,y1=30,x2=70,y2=70,y=2,x=2;

                cout<<"Before scaling"<<endl;

                setcolor(3);

                rectangle(x1,y1,x2,y2);

                cout<<"After scaling"<<endl;

                setcolor(10);

                rectangle(x1\*x,y1\*y,x2\*x,y2\*y);

                getch();

                break;}

        case 4:

            {

                 int x1=200,y1=300,x2=500,y2=300,x3=350,y3=400;

                 cout<<"triangle before reflection"<<endl;

                 setcolor(3);

                 line(x1,y1,x2,y2);

                 line(x1,y1,x3,y3);

                 line(x2,y2,x3,y3);cout<<"triangle after reflection"<<endl;

                 setcolor(5);

                 line(x1,-y1+500,x2,-y2+500);

                 line(x1,-y1+500,x3,-y3+500);

                 line(x2,-y2+500,x3,-y3+500);

                 getch();

                 break;}

        case 5:

        {

  int x1=400,y1=100,x2=600,y2=100,x3=400,y3=200,x4=600,y4=200,shx=2;

             cout<<"Before shearing of rectangle"<<endl;

             setcolor(3);

             line(x1,y1,x2,y2);

             line(x1,y1,x3,y3);

             line(x3,y3,x4,y4);

             line(x2,y2,x4,y4);

             cout<<"After shearing of rectangle"<<endl;

             x1=x1+shx\*y1;

             x2=x2+shx\*y2;

             x3=x3+shx\*y3;

             x4=x4+shx\*y4;

             setcolor(13);

             line(x1,y1,x2,y2);

             line(x1,y1,x3,y3);

             line(x3,y3,x4,y4);

             line(x2,y2,x4,y4);getch();}default:

          {

            cout<<"Invalid Selection"<<endl;

            break;

          }

        }

    getch();

    closegraph();

    return 0;

}