//assign 8 transformation

#include <graphics.h>

#include <stdlib.h>

#include<dos.h>

#include <iostream.h>

#include<conio.h>

#include<math.h>

//using namespace std;

class triangle

{

 int x1,x2,x3,y1,y2,y3,nx1,nx2,nx3,ny1,ny2,ny3,c;

 public:

 void read();

 void draw();

 triangle operator+(int tvector[]);

 triangle operator/(float theta);

 triangle operator\*(int svector[2]);

 };

 void triangle::read()

 {

   cout<<"\t Program for basic transformations";

   cout<<"\n\t Enter the coordinates of triangle";

   setcolor(11);

   cin>>x1>>y1>>x2>>y2>>x3>>y3;

  // draw(x1,y1,x2,y2,x3,y3);

 }

 void triangle::draw()

 {

  line(x1,y1,x2,y2);

  line(x2,y2,x3,y3);

  line(x3,y3,x1,y1);

  getch();

 }

triangle triangle:: operator+(int tvector[])

 { int tx,ty;

 tx=tvector[0];

 ty=tvector[1];

    triangle t;

   t.x1=x1+tx;

   t.y1=y1+ty;

   t.x2=x2+tx;

   t.y2=y2+ty;

   t.x3=x3+tx;

   t.y3=y3+ty;

   return(t);

 }

 triangle triangle:: operator\*(int svector[2])

 {   float sx,sy;

     triangle t;

    sx= svector[0];

    sy=svector[1];

   t.x1=x1\*sx;

   t.y1=y1\*sy;

   t.x2=x2\*sx;

   t.y2=y2\*sy;

   t.x3=x3\*sx;

   t.y3=y3\*sy;

   return(t);

 }

 triangle triangle ::operator/(float theta)

 { triangle t;

   t.x1=abs(x1\*cos(theta)-y1\*sin(theta));

   t.y1=abs(x1\*sin(theta)+y1\*cos(theta));

   t.x2=abs(x2\*cos(theta)-y2\*sin(theta));

   t.y2=abs(x2\*sin(theta)+y2\*cos(theta));

   t.x3=abs(x3\*cos(theta)-y3\*sin(theta));

   t.y3=abs(x3\*sin(theta)+y3\*cos(theta));

   return(t);

 }

int main()

{

      int gm,c,degree;

      float theta;

      int gd=DETECT;

       int tvector[2],svector[2];

      initgraph(&gd,&gm,"c:\\turboc3\\bgi");

      triangle t1,T, S,R,t2;

      t1.read();

      t1.draw();

      cout<<"\n 1.Translation\n 2.Rotation\n 3.Scalling\n 4.exit";

      cout<<"Enter your choice:";

       cin>>c;

      switch(c)

      {

    case 1: cout<<"\Enter translation factors(tx and ty)";

      cin>>tvector[0]>>tvector[1];

           T=t1+tvector;  //t1++

           T.draw();

           break;

    case 2: cout<<"\n Enter the angle of rotation";

      cin>>degree;

      theta=3.14\*degree/180;

          R=t1/theta;

          R.draw();

          break;

    case 3:

    cout<<"\n Enter the scalling factor(Sx and Sy)";

       cin>>svector[0]>>svector[1];

          S=t1\*svector;

          S.draw();

         }         // cleardevice();

   getch();

   closegraph();

   return 0;

   }