**Program for translation scaling and rotation along with shearing & reflection**

#include<stdio.h>

#include<graphics.h>

#include<stdlib.h>

#include<math.h>

#include<conio.h>

int x1,y1,x2,y2,midx,midy;

void axis();

void translation()

{

int tx,ty,xn1,yn1,xn2,yn2;

printf("\n Enter the translation:\n");

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

cleardevice();

outtextxy(400,100,"TRANSLATION");

xn1=x1+tx;

yn1=y1+ty;

xn2=x2+tx;

yn2=y2+ty;

axis();

rectangle(xn1,yn1,xn2,yn2);

getch();

}

void scaling()

{

float xn1,yn1,xn2,yn2;

float sx,sy;

printf("Enter the scaling factor");

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

cleardevice();

outtextxy(300,200,"SCALING");

xn1=x1\*sx;

yn1=y1\*sy;

xn2=x2\*sx;

yn2=y2\*sy;

axis();

rectangle(xn1,yn1,xn2,yn2);

getch();

}

void rotation()

{

int ang;

float rx,xn1,yn1,xn2,yn2,x1n1,y1n1,x2n2,y2n2;

printf("\n Enter the angle for rotation:\n");

scanf("%d",&ang);

cleardevice();

outtextxy(500,200,"ROTATION");

rx=(ang\*3.14)/180;

xn1=x1\*cos(rx)-y1\*sin(rx);

yn1=y1\*cos(rx)+x1\*sin(rx);

xn2=x2\*cos(rx)-y2\*sin(rx);

yn2=y2\*cos(rx)+x2\*sin(rx);

x1n1=x2\*cos(rx)-y1\*sin(rx);

y1n1=y1\*cos(rx)+x2\*sin(rx);

x2n2=x1\*cos(rx)-y2\*sin(rx);

y2n2=y2\*cos(rx)+x1\*sin(rx);

axis();

line(xn1,yn1,x1n1,y1n1);

line(x1n1,y1n1,xn2,yn2);

line(xn2,yn2,x2n2,y2n2);

line(x2n2,y2n2,xn1,yn1);

getch();

}

void shearing()

{

float sh;

float xn1,yn1,xn2,yn2,x1n1,y1n1,x2n2,y2n2;

printf("\n Enter the value for shearing:\n");

scanf("%f",&sh);

cleardevice();

outtextxy(500,100,"SHEARING");

xn1=x1+sh\*y1;

yn1=y1;

xn2=x2+sh\*y2;

yn2=y2;

x1n1=x2+sh\*y1;

y1n1=y1;

x2n2=x1+sh\*y2;

y2n2=y2;

axis();

line(xn1,yn1,x1n1,y1n1);

line(x1n1,y1n1,xn2,yn2);

line(xn2,yn2,x2n2,y2n2);

line(x2n2,y2n2,xn1,yn1);

getch();

}

void reflection()

{

int xn1,yn1,xn2,yn2;

cleardevice();

outtextxy(300,100,"REFLECTION");

if((x1<y1)^(x1<y1))

{

xn1=x1+100;

xn2=x2+100;

yn1=y1;

yn2=y2;

}

else

{

xn1=x1;

xn2=x2;

yn1=y1+100;

yn2=y2+100;

}

axis();

rectangle(xn1,yn1,xn2,yn2);

getch();

}

void get()

{

printf("\n Enter the coordinates x1,y1,x2,y2");

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

outtextxy(200,100,"ORIGINAL OBJECT");

x1= getmaxx() / 2-x1;

y1= getmaxy() / 2-y1;

x2 = getmaxx() / 2+x2;

y2 = getmaxy() / 2+y2;

axis();

getch();

}

void axis()

{

midx=getmaxx() / 2;

midy=getmaxy() / 2;

line(0,midy,midx\*2,midy);

line(midx,0,midx,midy\*2);

rectangle(x1,y1,x2,y2);

}

void main()

{

int ch,gd=DETECT,gm;

initgraph(&gd,&gm,"c:\\TC\\BGI");

get();

do

{

cleardevice();

outtextxy(10,10,"1)TRANSLATION");

outtextxy(10,20,"2)SCALING");

outtextxy(10,30,"3)ROTATION");

outtextxy(10,40,"4)SHEARING");

outtextxy(10,50,"5)REFLECTION");

outtextxy(10,60,"6)EXIT");

outtextxy(10,70,"ENTER UR CHOICE:");

scanf("%d",&ch);

switch(ch)

{

case 1:

translation();

break;

case 2:

scaling();

break;

case 3:

rotation();

break;

case 4:

shearing();

break;

case 5:

reflection();

break;

case 6:

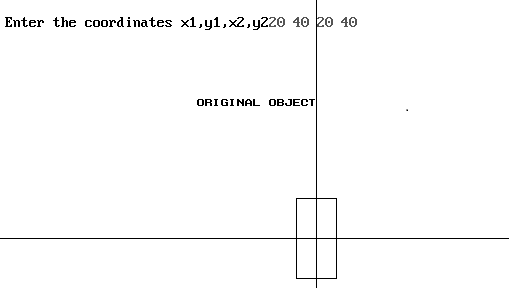
exit(0);

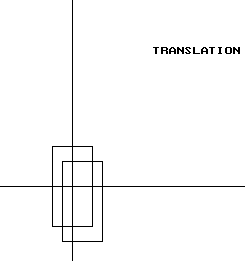
}

}while(ch<6);

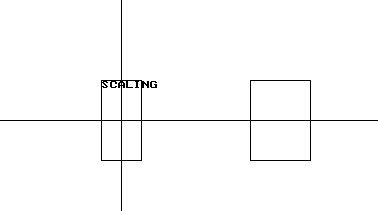
}

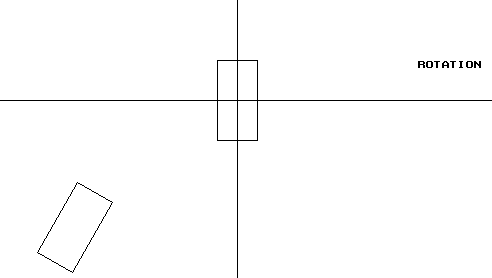
**Output:-**

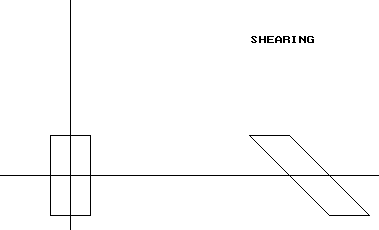
Input:-

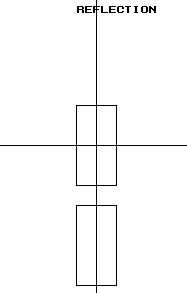


Translation output:-x-vlaue:-10 & y-value:-15

Scaling output:- with x-value:2 & y-value:-2

Rotation output:-Rotating angle:-30-degree

Shearing output:-shearx-value:-2 & sheary-value:-2

Reflection output:-