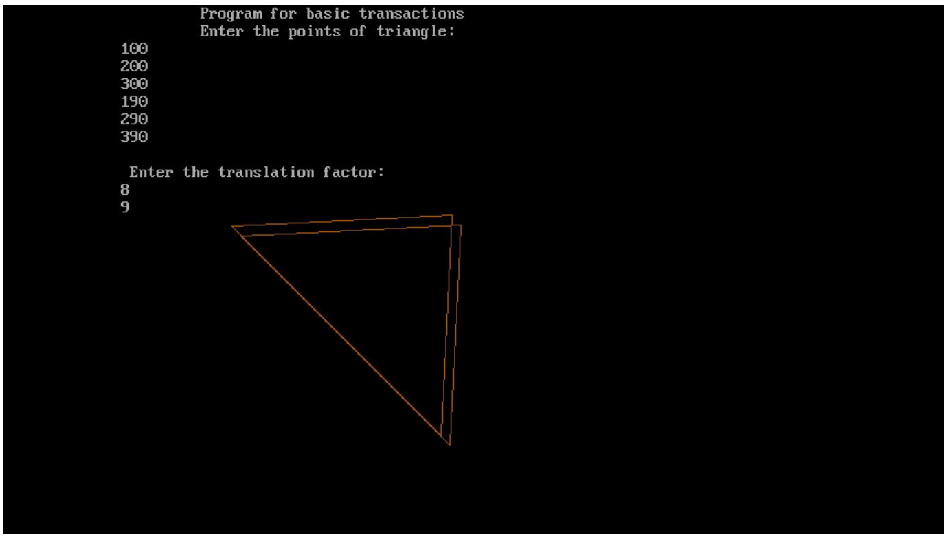
Implement 2D Transformations: Translation, Scaling & Rotation

1) Translation

```
#include <graphics.h>

#include <stdlib.h>

#include <stdio.h>

#include <conio.h>

#include<math.h>

int main()

{

                int gm;

                int gd=DETECT;

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

                int sx,sy,xt,yt,r;

                float t;

                initgraph(&gd,&gm," ");

                printf("¥t Program for basic transactions");

                printf("¥n¥t Enter the points of triangle");

                setcolor(1);

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

                line(x1,y1,x2,y2);

                line(x2,y2,x3,y3);

                line(x3,y3,x1,y1);

                printf("¥n Enter the translation factor");
```

```
                scanf("%d%d",&xt,&yt);
                 nx1=x1+xt;
                 ny1=y1+yt;
                 nx2=x2+xt;
                 ny2=y2+yt;
                 nx3=x3+xt;
                 ny3=y3+yt;
                 line(nx1,ny1,nx2,ny2);
                 line(nx2,ny2,nx3,ny3);
                 line(nx3,ny3,nx1,ny1);
  getch();
  closegraph();
}
```

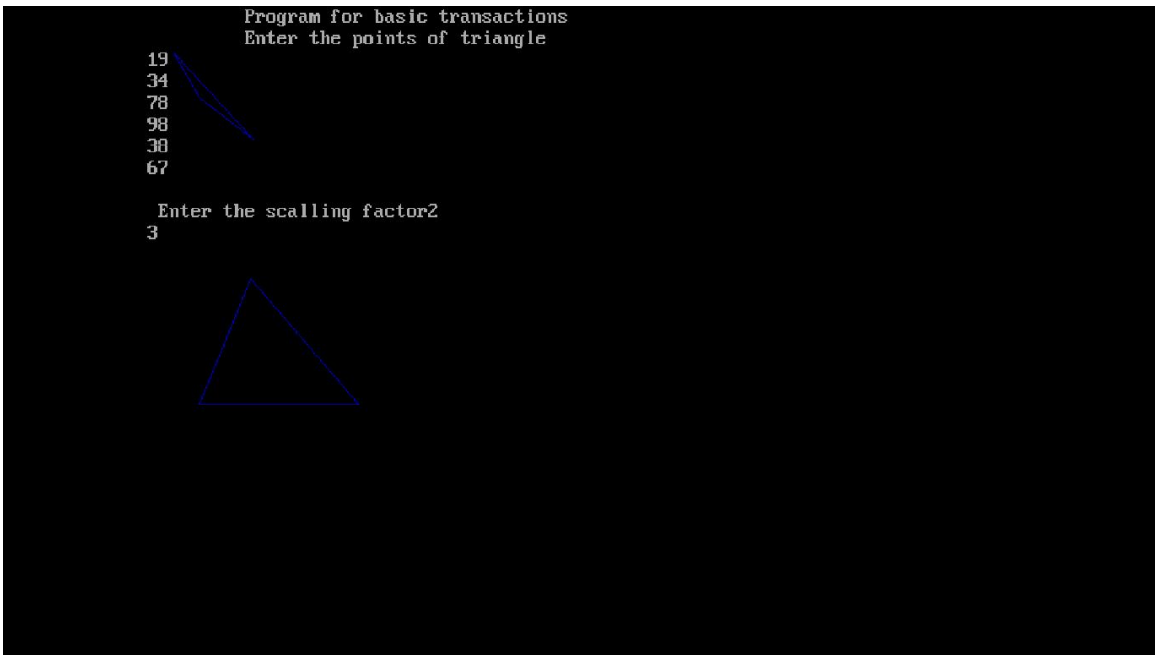


2) Scaling

```
#include <graphics.h>
#include <stdlib.h>
#include <stdio.h>
#include <conio.h>
#include<math.h>
int main()
{
        int gm;
        int gd=DETECT;
        int x1,x2,x3,y1,y2,y3,nx1,nx2,nx3,ny1,ny2,ny3,c;
        int sx,sy,xt,yt,r;
        float t;
        initgraph(&gd,&gm,"C:¥¥TURBOC3¥¥BGI");
        printf("¥t Program for basic transactions");
        printf("¥n¥t Enter the points of triangle¥n");
        setcolor(1);
        scanf("%d%d%d%d%d%d",&x1,&y1,&x2,&y2,&x3,&y3);
        line(x1,y1,x2,y2);
        line(x2,y2,x3,y3);
        line(x3,y3,x1,y1);
printf("¥n Enter the scalling factor");
                scanf("%d%d",&sx,&sy);
                nx1=x1*sx;
                ny1=y2*sy;
```

```
nx2=x2*sx;

ny2=y2*sy;

nx3=x3*sx;

ny3=y3*sy;

line(nx1,ny1,nx2,ny2);

line(nx2,ny2,nx3,ny3);

line(nx3,ny3,nx1,ny1);

getch();
```

```
closegraph();

}
```



3) Rotation

```
#include <graphics.h>

#include <stdlib.h>
```

```
#include <stdio.h>
#include <conio.h>
#include<math.h>
int main()
{
        int gm;
        int gd=DETECT;
        int x1,x2,x3,y1,y2,y3,nx1,nx2,nx3,ny1,ny2,ny3,c;
        int sx,sy,xt,yt,r;
        float t;
        initgraph(&gd,&gm,"C://TURBOC3//BGI");
        printf("¥t Program for basic transactions");
        printf("¥n¥t Enter the points of triangle");
        setcolor(1);
        scanf("%d%d%d%d%d%d",&x1,&y1,&x2,&y2,&x3,&y3);
        line(x1,y1,x2,y2);
        line(x2,y2,x3,y3);
        line(x3,y3,x1,y1);
  printf("¥n Enter the angle of rotation");
                        scanf("%d",&r);
                        t=3.14*r/180;
                        nx1=abs(x1*cos(t)-y1*sin(t));
                        ny1=abs(x1*sin(t)+y1*cos(t));
                        nx2=abs(x2*cos(t)-y2*sin(t));
```

```
                ny2=abs(x2*sin(t)+y2*cos(t));
                nx3=abs(x3*cos(t)-y3*sin(t));
                ny3=abs(x3*sin(t)+y3*cos(t));
                line(nx1,ny1,nx2,ny2);
                line(nx2,ny2,nx3,ny3);
                line(nx3,ny3,nx1,ny1);
    getch();
    closegraph();
    return 0;
}
```