**Q28. Write a C graphics program to rotate a triangle by a specific angle defined by user.**

**Code:**

#include<stdio.h>

#include<graphics.h>

#include<conio.h>

#include<math.h>

#include<process.h>

void triangle(int x1,int y1,int x2,int y2,int x3,int y3);

void rotate(int x1,int y1,int x2,int y2,int x3,int y3);

void main()

{

int gd=DETECT,gm;

int x1,x2,x3,y1,y2,y3;

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

printf("Enter the first side of the triangle:");

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

printf("Enter the second side of the triangle:");

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

printf("Enter the third side of the triangle:");

scanf("%d%d",&x3,&y3);

triangle(x1,y1,x2,y2,x3,y3);

getch();

cleardevice();

rotate(x1,y1,x2,y2,x3,y3);

setcolor(1);

triangle(x1,y1,x2,y2,x3,y3);

getch();

}

void triangle(int x1,int y1,int x2,int y2,int x3,int y3)

{

line(x1,y1,x2,y2);

line(x2,y2,x3,y3);

line(x3,y3,x1,y1);

}

void rotate(int x1,int y1,int x2,int y2,int x3,int y3)

{

int x,y,a1,b1,a2,b2,a3,b3,p=x2,q=y2;

float angle;

printf("Enter angle of rotation:");

scanf("%d",&angle);

cleardevice();

angle=(angle\*3.14)/180;

a1=p+(x1-p)\*cos(angle)-(y1-q)\*sin(angle);

b1=q+(x1-p)\*sin(angle)-(y1-q)\*cos(angle);

a2=p+(x2-p)\*cos(angle)-(y2-q)\*sin(angle);

b2=q+(x2-p)\*sin(angle)-(y2-q)\*cos(angle);

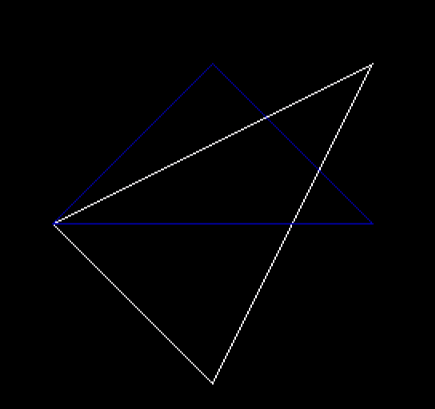
a3=p+(x3-p)\*cos(angle)-(y3-q)\*sin(angle);

b3=p+(x3-p)\*sin(angle)-(y3-q)\*cos(angle);

triangle(a1,b1,a2,b2,a3,b3);

}

**Output:**



**Q29. Write a C graphics program to rotate a wheel.**

**Code:**

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

#include<math.h>

#include<dos.h>

#include<stdlib.h>

#define PI 3.14159

void rotate\_wheel(int xc,int yc,int t)

{

int x,y;

for(t=t;t<180;t=t+60)

{

x=50\*cos(t\*PI/180);

y=50\*sin(t\*PI/180);

line(xc+x,yc+y,xc-x,yc-y);

}

circle(xc,yc,50);

circle(xc,yc,52);

}

void main()

{

int gd=DETECT,gm,x;

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

for(x=0;x<640;x++)

{

setcolor(RED);

rotate\_wheel(x,140,x%60);

delay(5);

cleardevice();

rotate\_wheel(x,140,x%60);

}

getch();

closegraph();

}

**Output:**

