## PROGRAM NO-1

## //Program to draw a line using graphics function.

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

void main()

{

//request auto detection

int gdriver=DETECT,gmode;

int x1,y1,x2,y2;

//initialize graphics and local variables

initgraph(&gdriver,&gmode,"x:\\bgi");

setcolor(BLUE);

setbkcolor(RED);

printf("\t\tWelcome to the c program.\n");

printf("Enter the first x and y coordinate:\n");

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

printf("Enter the second x and y coordinate:\n");

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

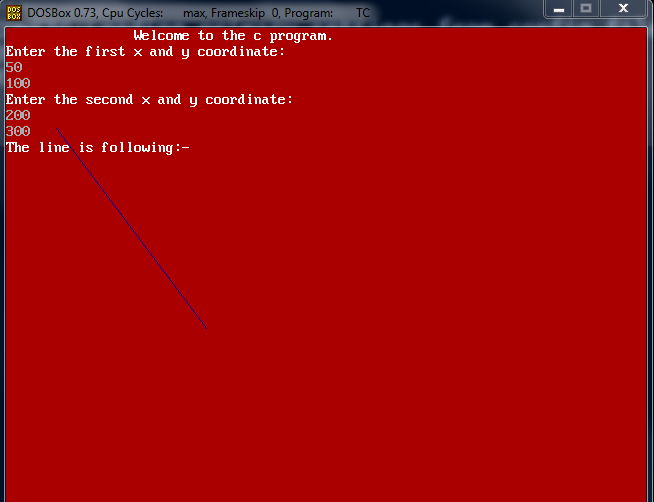
printf("The line is following:-");

line(x1,y1,x2,y2);

getch();

}

## OUTPUT:-



## PROGRAM NO-2

## //Program to draw a circle using graphics function.

#include<stdio.h>

#include <graphics.h>

#include <conio.h>

void main()

{

//request auto detection

int gdriver = DETECT, gmode;

int x,y,r;

// initialize graphics and local variables

initgraph(&gdriver,&gmode,"x:\\bgi");

setcolor(BLUE);

setbkcolor(RED);

printf("\t\tWelcome to the c program.\n");

printf("Enter the coordinates(x,y):-\n");

scanf("%d%d",&x,&y);

printf("Enter the radius:-\n");

scanf("%d",&r);

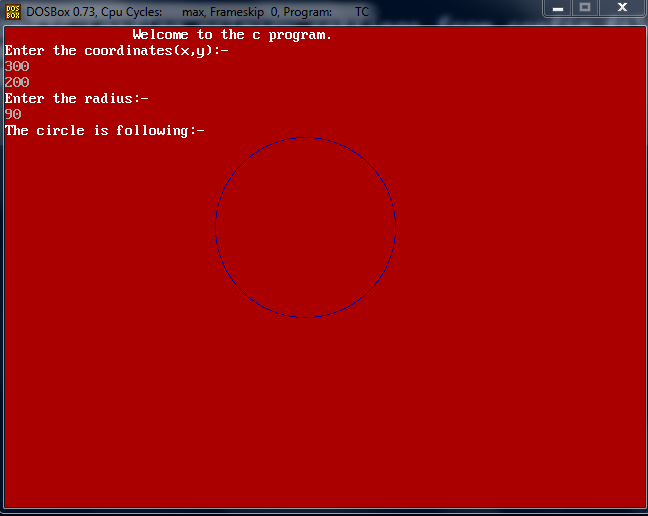
printf("The circle is following:-\n");

circle(x,y,r);

getch();

}

## OUTPUT:-



## PROGRAM NO-3

## //Program to draw a tringle using graphics function.

#include <graphics.h>

#include <stdio.h>

#include <conio.h>

void main()

{

// request auto detection

int gdriver = DETECT, gmode;

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

// initialize graphics and local variables

initgraph(&gdriver,&gmode, "x:\\bgi");

setcolor(BLUE);

setbkcolor(RED);

printf("\t\tWelcome to c the program.\n");

printf("Enter the first coordinate(x1,y1):-");

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

printf("Enter the second coordinate(x2,y2):-");

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

printf("Enter the third coordinate(x3,y3):-");

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

printf("The tringle is following:-\n");

line(x1,y1,x2,y2);

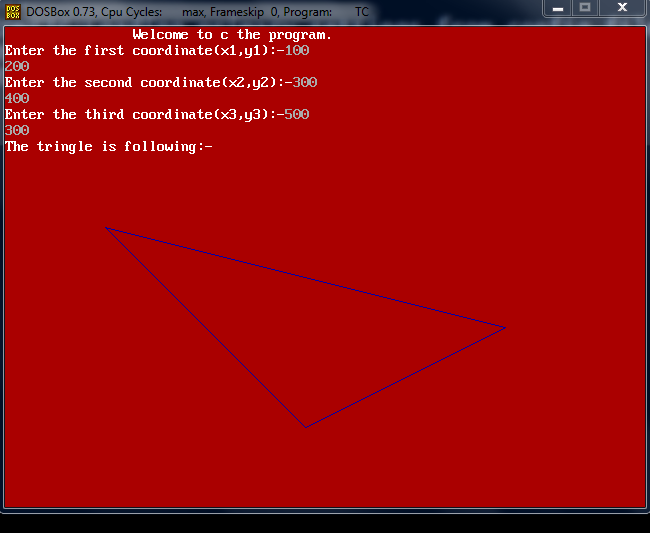
line(x2,y2,x3,y3);

line(x3,y3,x1,y1);

getch();

}

## OUTPUT:-



## PROGRAM NO-4

## //Program to draw a tringle within tringle using graphics function.

#include <graphics.h>

#include <stdio.h>

#include <conio.h>

void main()

{

// request auto detection

int gdriver = DETECT, gmode;

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

// initialize graphics and local variables

initgraph(&gdriver, &gmode, "x:\\bgi");

setcolor(BLUE);

setbkcolor(RED);

printf("\t\tWelcome to the c program.\n");

printf("The tringle within the tringle is following:-\n");

line(0,200,500,500);

line(500,500,500,0);

line(500,0,0,200);

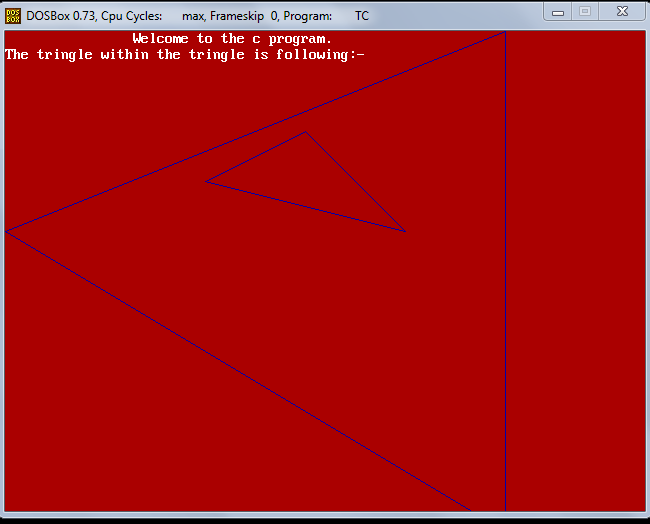
line(200,150,400,200);

line(400,200,300,100);

line(300,100,200,150);

getch(); }

## OUTPUT:-



## PROGRAM NO-5

## //Program to draw a line using DDA algorithm.

#include<stdio.h>

#include<graphics.h>

#include<conio.h>

#include<math.h>

void main()

{

// request auto detection

int gdriver=DETECT,gmode;

int x1,x2,y1,y2,length,dx,dy,i,x,y;

// initialize graphics and local variables

initgraph(&gdriver,&gmode,"x:\\bgi");

//setcolor(BLUE);

setbkcolor(RED);

printf("Welcome to c program.\n\n");

printf("Enter the first coordinate(x1,y1):-");

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

printf("Enter the second coordinate(x2,y2):-");

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

if(abs(x2-x1)>(y2-y1))

length=(abs(x2-x1));

else

length=(abs(y2-y1));

dx=(x2-x1)/length;

dy=(y2-y1)/length;

x=x1,y=y1;

printf("You are seeing line:-");

for(i=1;i<length;i++)

{

x=x+dx;

y=y+dy;

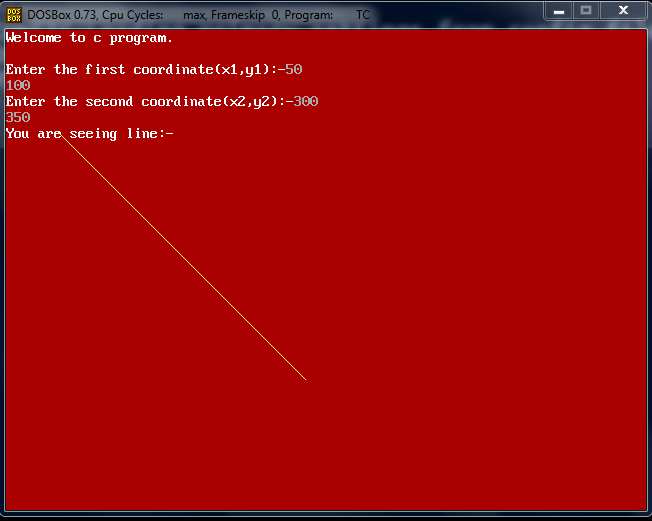
putpixel(x,y,YELLOW);

}

getch();

}

## OUTPUT:-



## PROGRAM NO-6

## //Program to draw a line using Bresenhams algorithm.

#include<graphics.h>

#include<stdio.h>

#include<conio.h>

#include<math.h>

void main()

{

//request auto detection.

int gdriver=DETECT,gmode;

int x1,x2,y1,y2,length,dx,dy,i,x=0,y=0,inc1,inc2;

int xend,d,a,b;

//initialize graphics and local variables.

initgraph(&gdriver,&gmode,"x:\\bgi");

setbkcolor(RED);

printf("\t\tWelcome to the c program.\n\n");

printf("Enter the first coordinate(x1,y1):-");

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

printf("Enter the second coordinate(x2,y2):-");

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

a=x1,b=y1;

if(x1&&y1>x2&&y2)

{

x1=x2;

y1=y2;

x2=a;

y2=b;

}

dx=x2-x1;

dy=y2-y1;

inc1=2\*dy;

d=inc1-dx;

inc2=2\*(dy-dx);

if(d<0)

{

x=x2;

y=y2;

xend=x1;

}

else

{

x=x1;

y=y1;

xend=x2;

}

printf("You are seeing line:-");

for(i=1;i<=xend;i++)

{

if(d<0)

d=d+inc1;

else

{

d=d+inc2;

y=y+1;

}

putpixel(x,y,BLUE);

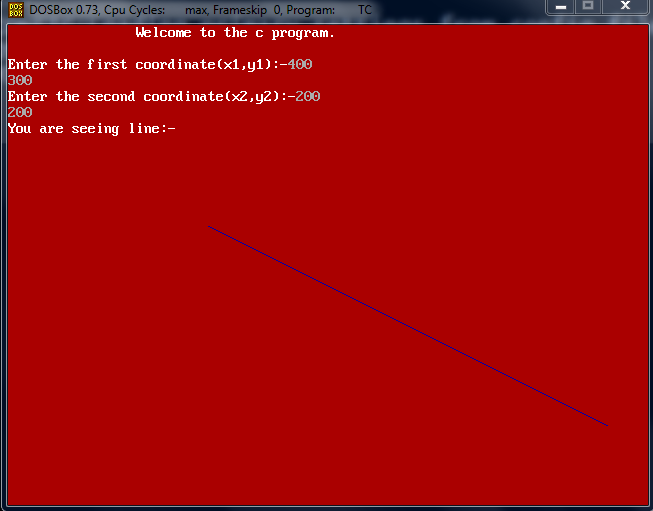
x=x+1;

}

getch();

}

## OUTPUT:-



## PROGRAM NO-7

## //Program to draw two circles coming from two end and after colliding go to other end.

#include<stdio.h>

#include<conio.h>

#include<graphics.h>

void main()

{

//request auto detection.

int gdriver=DETECT,gmode;

int i,k;

//initialize graphics and local variables.

initgraph(&gdriver,&gmode, "x:\\bgi");

k=600;

for(i=0;i<=250;i++)

{

circle(i,50,50);

setcolor(BLUE);

circle(k,50,50);

sleep(1);

setcolor(RED);

k--;

if(i==k)

break;

clrscr();

}

k=350;

for(i=250;i>=0;i--)

{

circle(i,50,50);

setcolor(BLUE);

circle(k,50,50);

setcolor(RED);

sleep(1);

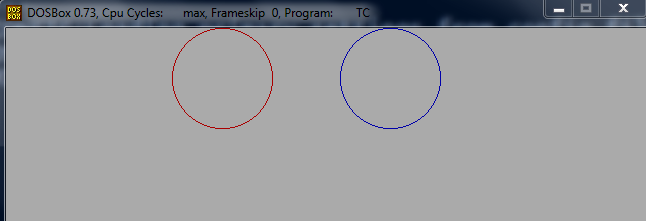
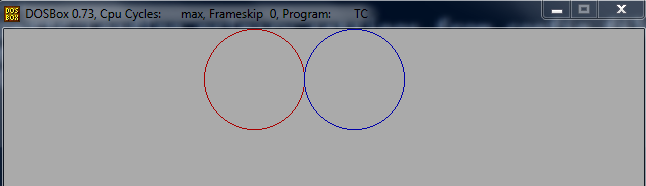
k++;

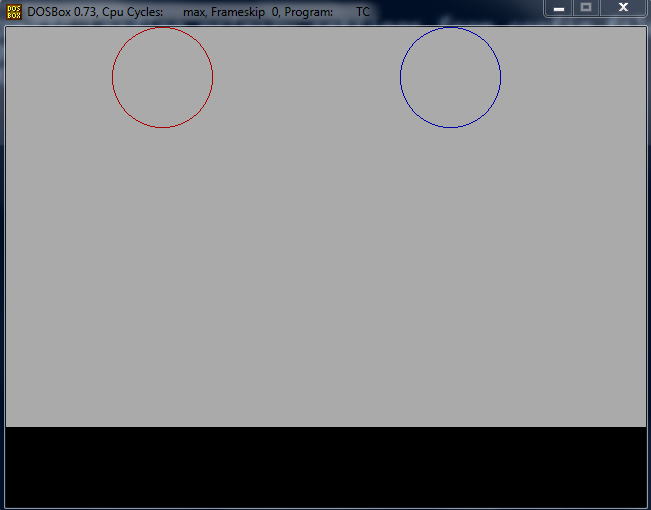
clrscr();

}

}

## OUTPUT:-





## PROGRAM NO-8

## //Program to draw a semicircle using graphics function.

#include <graphics.h>

#include <stdio.h>

#include <conio.h>

void main()

{

//request auto detection.

int gdriver=DETECT,gmode;

int x,y,radius;

//initialize graphics and local variables.

initgraph(&gdriver, &gmode, "x:\\bgi");

printf("\t\tWelcome to the c program.\n");

setbkcolor(BLUE);

setcolor(YELLOW);

printf("Enter the x and y coordinate:\n");

scanf("%d%d",&x,&y);

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

scanf("%d",&radius);

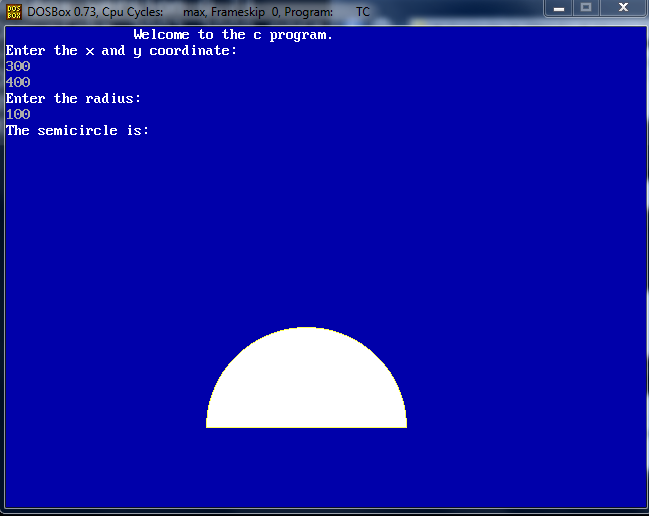
printf("The semicircle is:\n");

pieslice(x,y,0,180,radius);

getch();

}

## OUTPUT:-



## PROGRAM NO-9

## //Program to rotate a tringle by an given angle.

#include <graphics.h>

#include <stdio.h>

#include <conio.h>

#include<math.h>

void main()

{

// request auto detection

int gdriver=DETECT,gmode;

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

int i,j,k;

float tri[3][3];

float R[3][3],newtri[3][3],theta;

// initialize graphics and local variables

initgraph(&gdriver,&gmode,"x:\\bgi");

printf("\t\tWelcome to c the program.\n");

step1:

printf("Enter the first coordinate(x1,y1):-");

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

printf("Enter the second coordinate(x2,y2):-");

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

printf("Enter the third coordinate(x3,y3):-");

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

if(x1&&y1&&x2&&y2&&x3&&y3<0)

{

printf("Sorry! You can't see the tringle.");

goto step1;

}

printf("\nEnter the angle: \n");

scanf("%f",&theta);

theta=theta\*(3.14)/180;

printf("The angle about which tringle is rotated %f\n",theta);

//Initialization of tringle point.

tri[0][0]=x1;

tri[1][0]=y1;

tri[0][1]=x2;

tri[1][1]=y2;

tri[0][2]=x3;

tri[1][2]=y3;

tri[2][0]=1;

tri[2][1]=1;

tri[2][2]=1;

printf("The rotation is:\n");

line(x1,y1,x2,y2);

line(x2,y2,x3,y3);

line(x3,y3,x1,y1);

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

printf("%.2f\t",tri[i][j]);

printf("\n");

}

R[0][0]=cos(theta);

R[0][1]=-sin(theta);

R[1][0]=sin(theta);

R[1][1]=cos(theta);

R[0][2]=0;

R[1][2]=0;

R[2][0]=0;

R[2][1]=0;

R[2][2]=1;

printf("The rotation matrix\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

printf("%.2f\t",R[i][j]);

printf("\n");

}

//Rotation of the tringle.

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

{

newtri[i][j]=0;

for(k=0;k<3;k++)

{

newtri[i][j]+=(R[i][k]\*tri[k][j]);

}

}

}

printf("Rotated tringle matrix\n");

for(i=0;i<3;i++)

{

for(j=0;j<3;j++)

printf("%.2f\t",newtri[i][j]);

printf("\n");

}

line(newtri[0][0],newtri[1][0],newtri[0][1],newtri[1][1]);

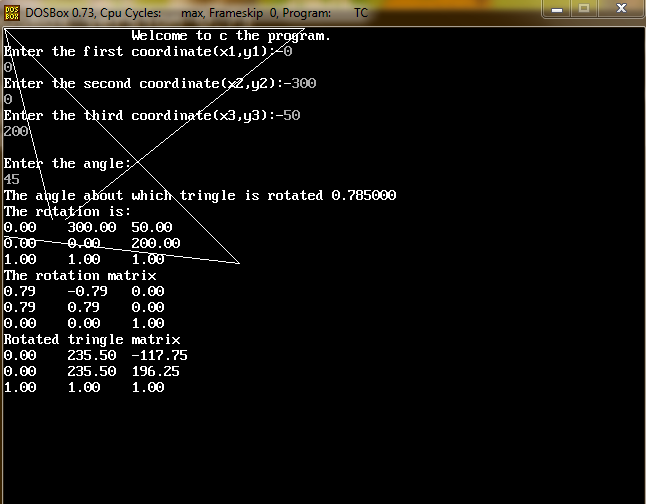
line(newtri[0][1],newtri[1][1],newtri[0][2],newtri[1][2]);

line(newtri[0][2],newtri[1][2],newtri[0][0],newtri[1][0]);

getch();

}

## OUTPUT:-



## PROGRAM NO-10

## //Program to draw an ellipse using graphics function.

#include <graphics.h>

#include <stdio.h>

#include <conio.h>

void main()

{

// request auto detection.

int gdriver=DETECT,gmode;

int x,y,stangle,endangle,xradius,yradius;

//initialize graphics,localvariables.

initgraph(&gdriver,&gmode,"x:\\bgi");

printf("\t\tWelcome to the c program.\n");

step1:

printf("Enter the centre of the ellipse.\n");

scanf("%d%d",&x,&y);

if(x<0||y<0)

{

printf("Sorry invalid point.");

printf("Try again.\n");

goto step1;

}

printf("Enter the x radius:-\n");

scanf("%d",&xradius);

printf("Enter the y radius:-\n");

scanf("%d",&yradius);

if(xradius==0&&yradius==0)

printf("It is a point ellipse.");

else

printf("The ellipse is:-");

ellipse(x,y,-180,180,xradius,yradius);

getch();

}

## OUTPUT:-

