

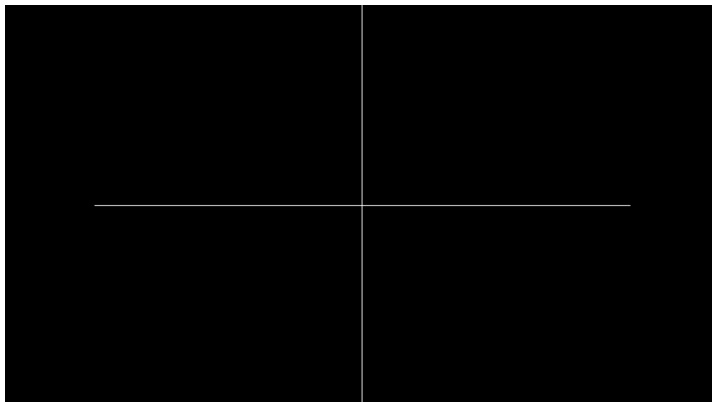
## **Practical No. 1**

**Aim:** Draw a co-ordinate axis at the center of the screen.

### **Input:**

```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
#include<stdlib.h>
int main(void){
int gdriver=DETECT, gmode;
int xmax, ymax;
initgraph(&gdriver, &gmode, "C:\\\\turbo3\\\\bgi");
setcolor(getmaxcolor());
xmax = getmaxx();
ymax = getmaxy();
line(xmax/2, 0, xmax/2, ymax);
line(0, ymax/2, xmax, ymax/2);
getch();
closegraph();
return 0;
}
```

### **Output:**



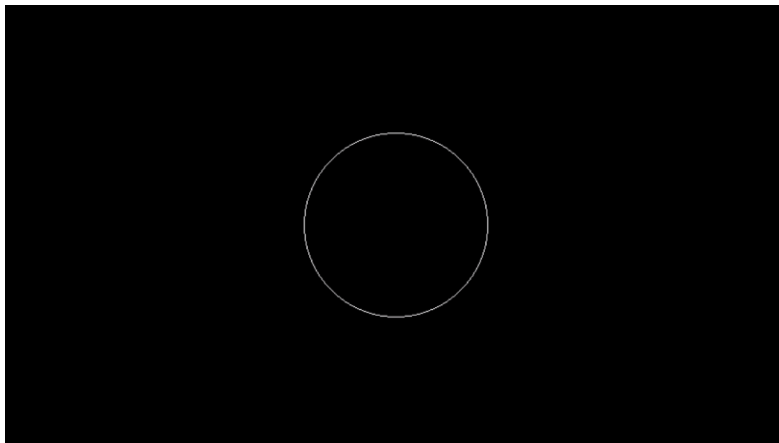
## **Practical No. 2**

**Aim:** Develop the program for the mid-point circle drawing algorithm.

**Input:**

```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
#include<stdlib.h>
int main(void){
int gdriver=DETECT, gmode, errorcode;
int midx, midy;
int radius = 100;
initgraph(&gdriver, &gmode, "C:\\\\turbo3\\\\bgi");
setcolor(getmaxcolor());
midx=getmaxx()/2;
midy=getmaxy()/2;
circle(midx, midy, radius);
getch();
return 0;
}
```

**Output:**



## **Practical No. 3**

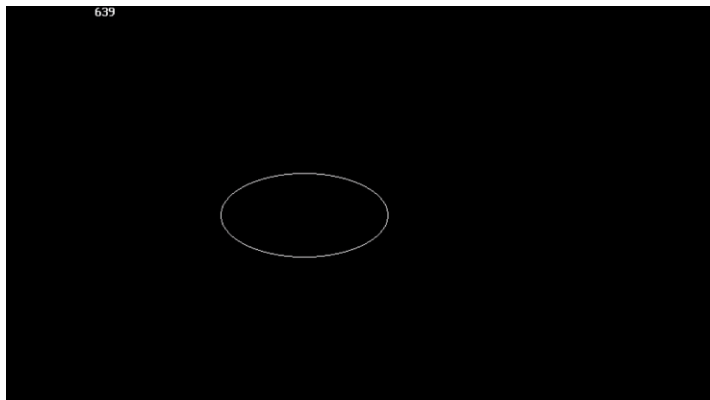
**Aim:** Develop the program for the mid-point ellipse drawing algorithm

### **Input:**

```
#include<graphics.h>
#include<conio.h>
#include<stdio.h>
#include<stdlib.h>

int main(void){
int gdriver=DETECT, gmode, errorcode;
int midx, midy;
initgraph(&gdriver, &gmode, "C:\\\\turbo3\\\\bgi");
setcolor(getmaxcolor());
printf("%d", getmaxx());
midx=500/2;
midy=500/2;
ellipse(midx, midy, 0, 360, 100, 50);
getch();
closegraph();
return 0;
}
```

### **Output:**



## **Practical No. 4**

**Aim: Draw the following basic shapes in the center of the screen:**

**i. Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line**

**Input:**

```
#include<graphics.h>

#include<conio.h>

#include<stdio.h>

#include<stdlib.h>

int main(void){

int gdriver=DETECT, gmode, errorcode;

int midx, midy, left, right, top, bottom, radius;

initgraph(&gdriver, &gmode, "C:\\\\turbo3\\\\bgi");

setcolor(getmaxcolor());

printf("%d", getmaxx());

printf(" %d ", getmaxy());

midx = getmaxx()/2;

midy = getmaxy()/2;

radius = 50;

circle(midx, midy, radius);

ellipse(midx, midy, 0, 360, 100, 50);

left=midx-200;

top=midy-100;

right=midx+200;

bottom=midy+100 ;

rectangle(left, top, right, bottom);

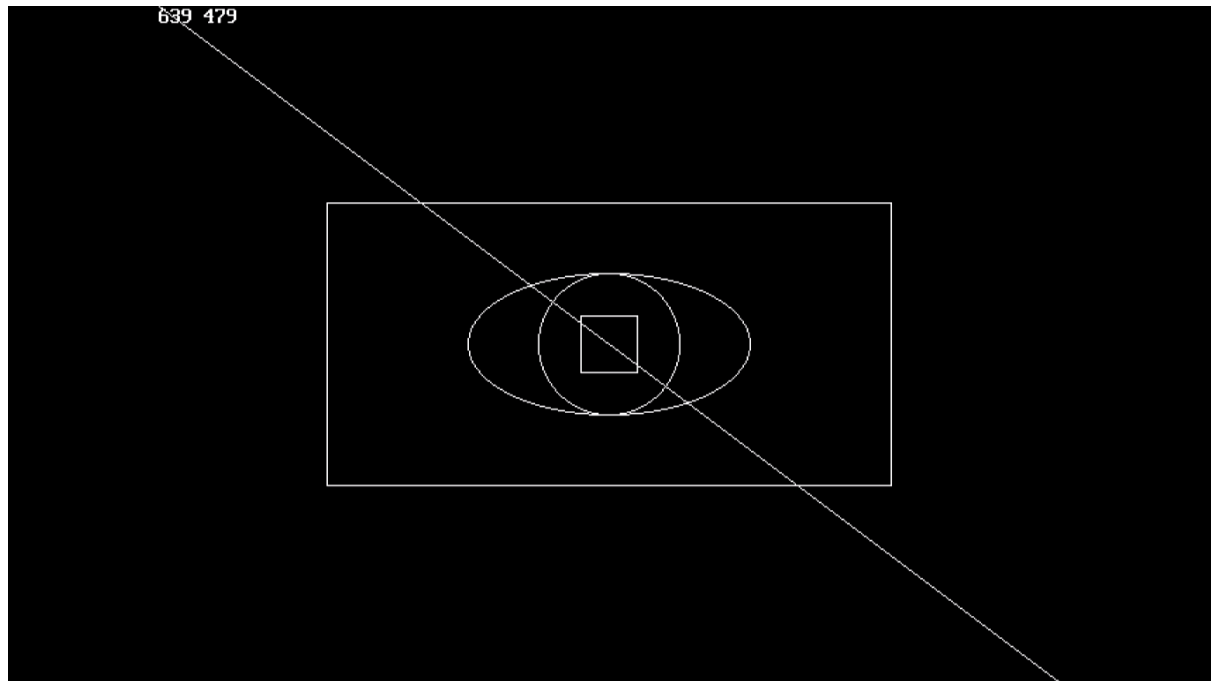
rectangle(midx-20, midy+20, midx+20, midy-20);

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

getch();
```

```
closegraph();  
return 0;  
}
```

**Output:**



## **Practical No. 5**

**Aim: Develop the program for DDA Line drawing algorithm.**

**Input:**

```
#include<graphics.h>
#include<stdio.h>
#include<conio.h>
#include<math.h>
#include<dos.h>

void main( ){
float x, y, x1, y1, x2, y2, dx, dy, step, xinc, yinc;
int i,gd=DETECT,gm;
initgraph(&gd,&gm,"c:\\turbo3\\bgi");
printf("Enter the value of x1 and y1 : ");
scanf("%f%f",&x1,&y1);
printf("Enter the value of x2 and y2: ");
scanf("%f%f",&x2,&y2);
dx=abs(x2-x1);
dy=abs(y2-y1);
if(dx>=dy)
step=dx;
else
step=dy;
xinc=dx/step;
yinc=dy/step;
x=x1;
y=y1;
putpixel(x,y,5);
```

```
i=1;
while(i<=step)
{
x=x+xinc;
y=y+yinc;
putpixel(x, y, WHITE);
i=i+1;
delay(50);
}
getch();
closegraph();
}
```

### Output:

