CGA

Practical 1

graphics.h .....header file graphics library
initgraph .....method to initialize graphics drivers
initgraph(&gd,&gm,"C:\\TC\\BGI") .....gd-graphics driver, gm-graphics mode, "path"

Sample program:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT, gm;  //We use DETECT macro of graphics.h library
initgraph(&gd,&gm,"C:\\TC\\BGI");
……. //drawing code
getch();
closegraph();  //to come out of graphic mode
}
```

A) Explain different functions

putpixel()

It displays a pixel specified by co-ordinate locations.

Syntax:
putpixel(int x, int y, COLOR);

example:
putpixel(100, 200, WHITE);

line()

It is a predefined function to draw from a point (x1, y1) to (x2, y2)

Syntax:
line(x1, y1, x2, y2);

Example:
line(100, 200, 300, 300);

rectangle()

It is used to draw rectangle. Coordinates of left-top and right-bottom corner is required.

Syntax:
rectangle(left,top,right, bottom);

Example:
rectangle(100, 100, 300, 300);

bar()

Bar function is used to draw 2D rectanglular filled in bar.

Syntax:
bar(left,top,right, bottom);

Example:
bar(100, 100, 300, 300);

circle()

It is use to draw circle.

Syntax:
Circle (x, y, radius); …….. x,y are centre coordinates

Example:
circle(200, 200, 50);

arc()

It is used to draw an arc.
Syntax:
arc (x, y, stangle, endangle, radius); …….. x,y are centre coordinates, stangle= starting angle, endangle

Example:

arc(200, 200, 0, 90, 50);

ellipse()

It is used to draw ellipse.

Syntax:

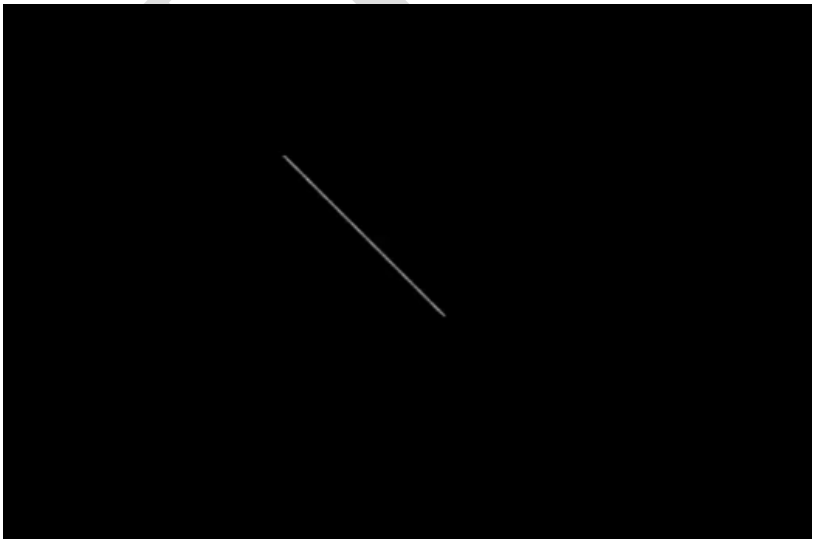
ellipse(x, y, stangle, endangle, xradius, yradius); …….. x,y are centre coordinates, stangle= starting angle, endangle

Example:

ellipse(200, 200, 0, 360, 50, 25);

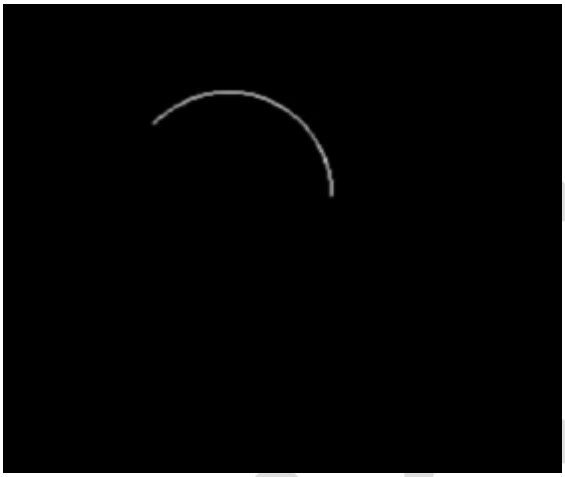
CODE 1:

```
#include <graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"C:\\TC\\BGI");
line(100,100,200,200);
getch();
closegraph();
}
```



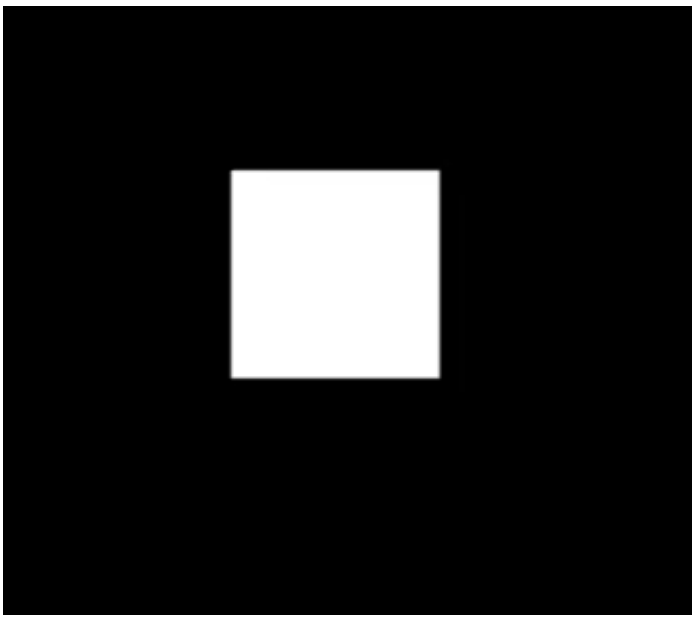
CODE 2:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"C:\\TC\\BGI");
arc(100,100,0,135,50);
getch();
closegraph();
}
```



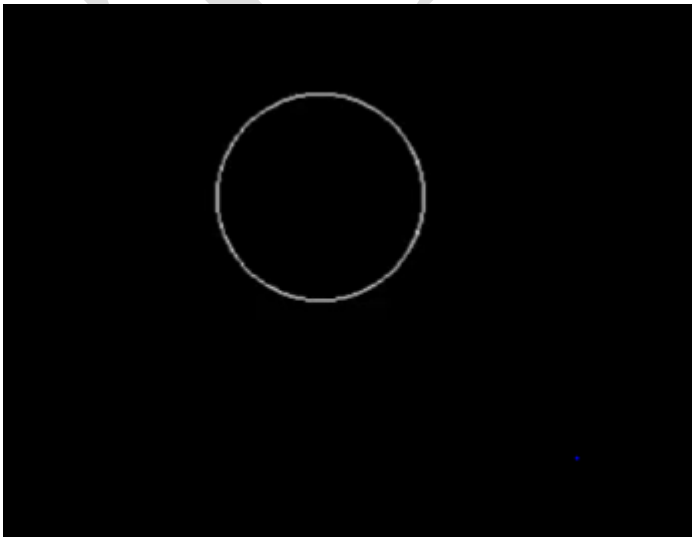
CODE 3:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"C:\\TC\\BGI");
bar(100,100,200,200);
getch();
closegraph();
}
```



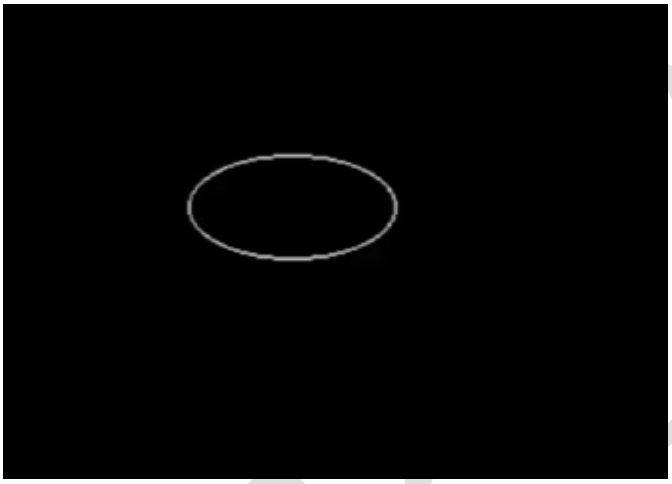
CODE 4:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"C:\\TC\\BGI");
circle(100,100,50);
getch();
closegraph();
}
```



CODE 5:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"C:\\TC\\BGI");
ellipse(100,100,0,360,50,25);
getch();
closegraph();
}
```



setcolor()

Setcolor function is used to change the current drawing color.

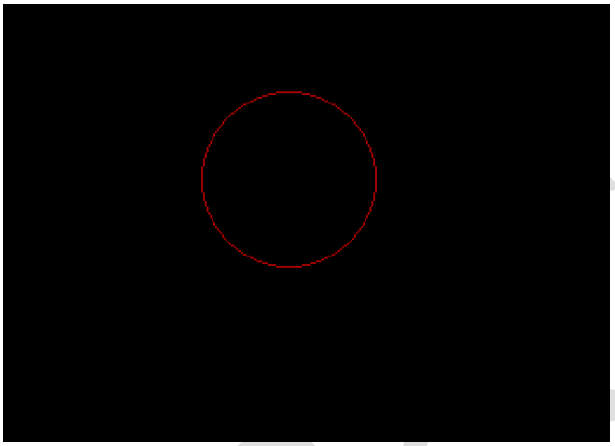
void setcolor(int color);

E.g. setcolor(RED); or setcolor(4);
changes the current drawing color to RED.

Name	Value
BLACK	0
BLUE	1
GREEN	2
CYAN	3
RED	4
MAGENTA	5
BROWN	6
LIGHTGRAY	7
DARKGRAY	8
LIGHTBLUE	9
LIGHTGREEN	10
LIGHTCYAN	11
LIGHTRED	12
LIGHTMAGENTA	13
YELLOW	14
WHITE	15

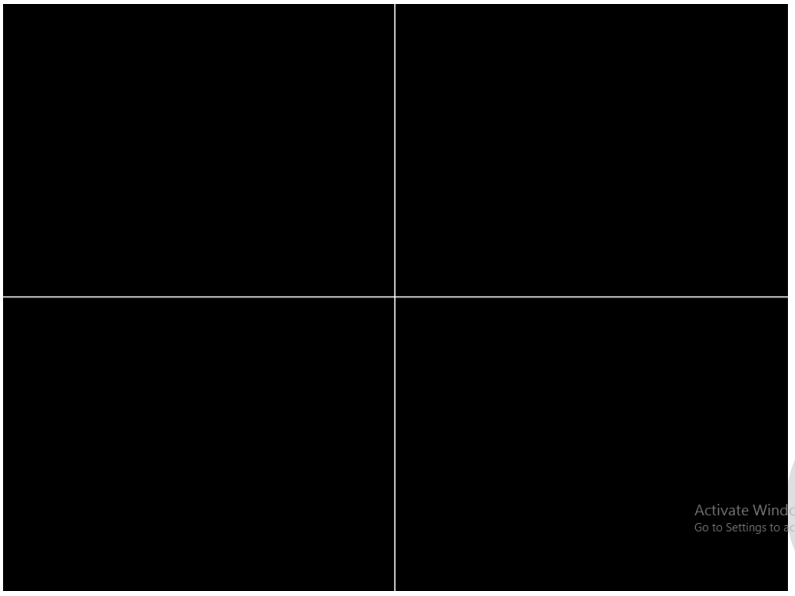
CODE 6:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm;
initgraph(&gd,&gm,"C:\\TC\\BGI");
setcolor(RED);
circle(100,100,50);
getch();
closegraph();
}
```



B) Draw co-ordinate axis at centre of screen

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
line(xcen,0,xcen,getmaxy());
line(0,ycen,getmaxx(),ycen);
getch();
closegraph();
}
```

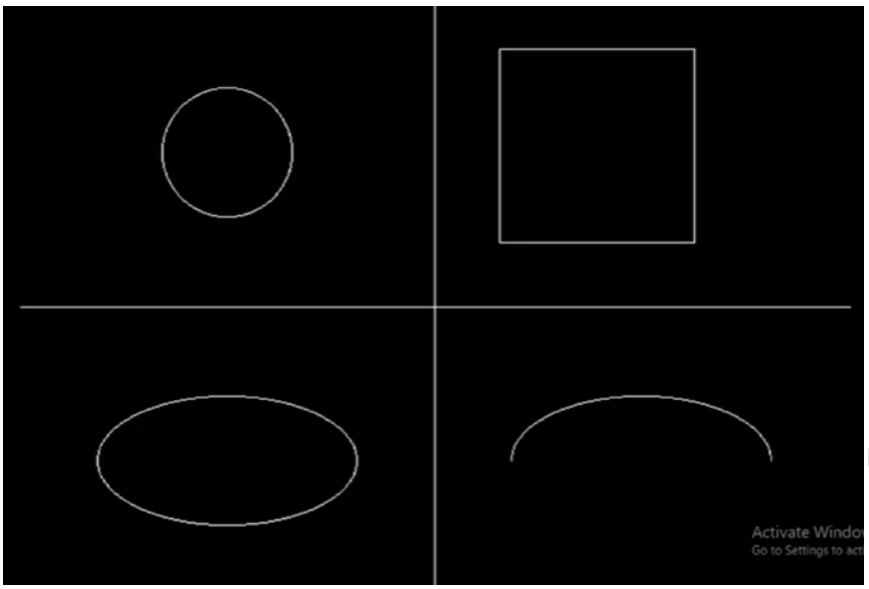


PRACTICAL 2

A. Divide your screen into four region, draw circle, rectangle, ellipse and half ellipse in each region

CODE:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
line(xcen,0,xcen,getmaxy());
line(0,ycen,getmaxx(),ycen);
rectangle(xcen+50,ycen-200,xcen+200,ycen-50);
ellipse(xcen-xcen/2,ycen+ycen/2,0,360,100,50);
ellipse(xcen+xcen/2,ycen+ycen/2,0,180,100,50);
getch();
closegraph();
}
```

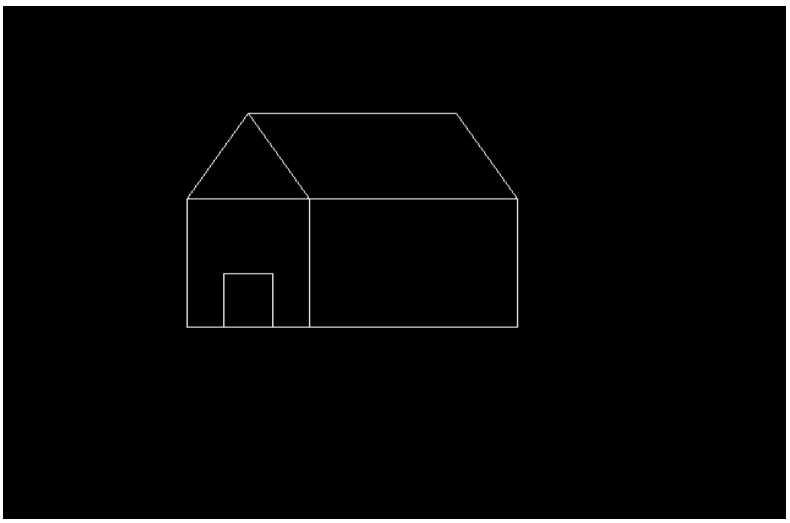


B. Draw a simple hut on the screen.

CODE:

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT, gm;
initgraph(&gd,&gm,"C:\\TC\\BGI");
rectangle(150,180,250,300);
rectangle(250,180,420,300);
rectangle(180,250,220,300);
line(200,100,150,180);
line(200,100,250,180);
line(200,100,370,100);
line(370,100,420,180);
getch();
closegraph();
}
```

OUTPUT:

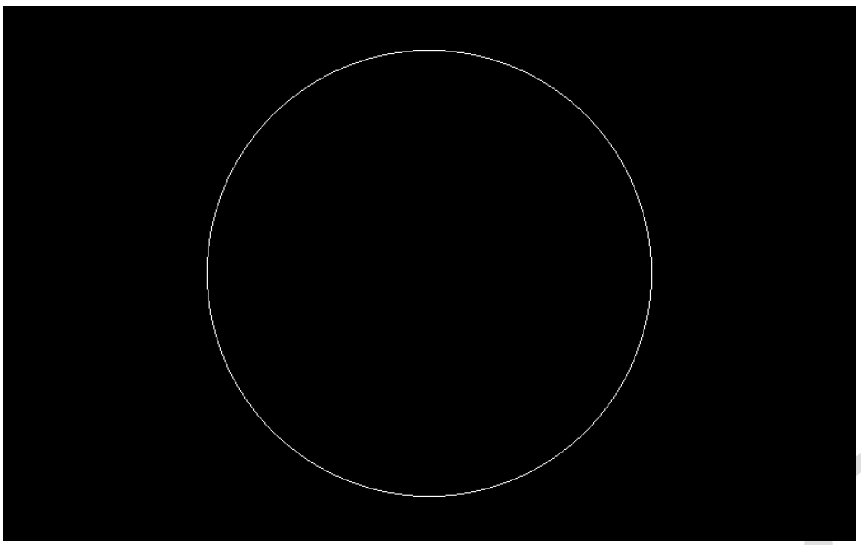


Practical 3

Draw the following basic shapes in the center of the screen :
i. Circle ii. Rectangle iii. Square iv. Concentric Circles v. Ellipse vi. Line

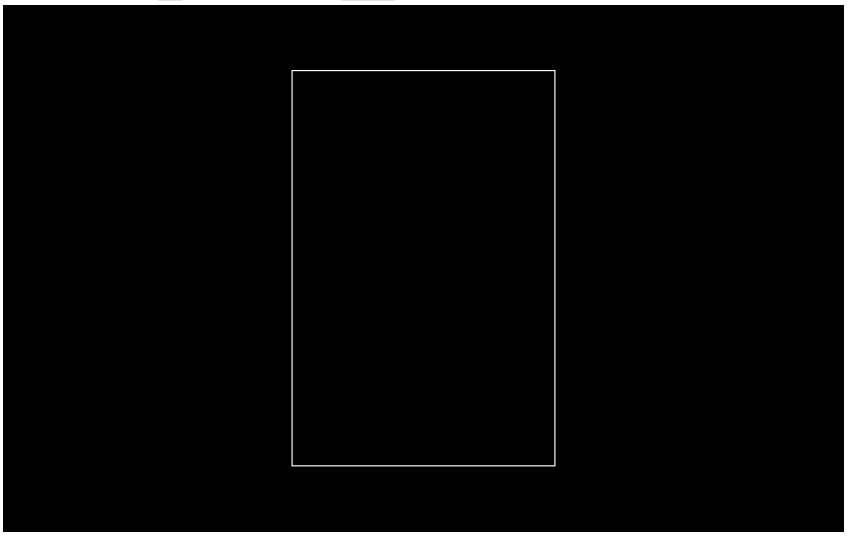
i. Circle

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
circle(xcen,ycen,200);
getch();
closegraph();
}
```



ii. Rectangle

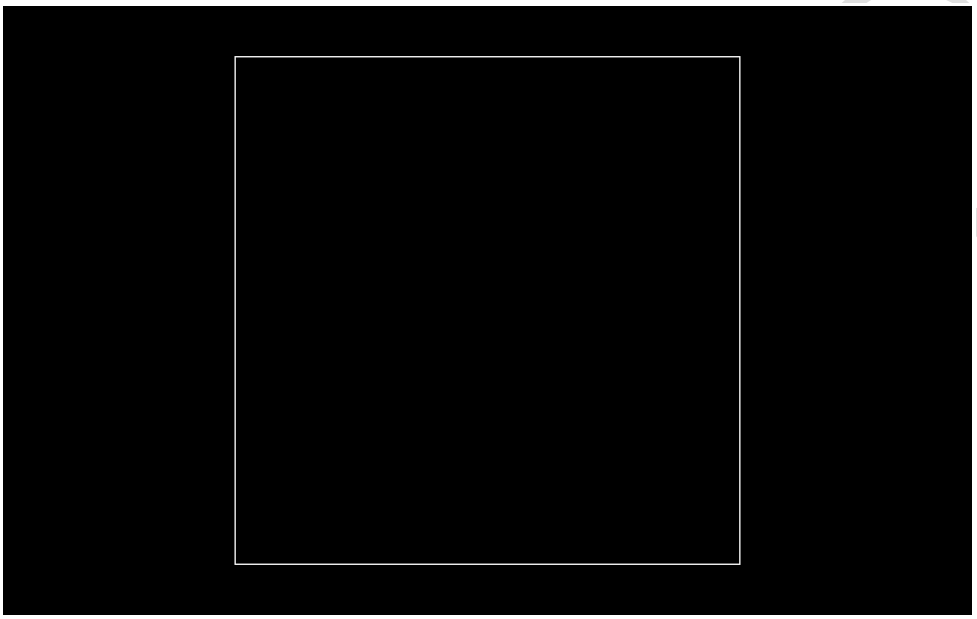
```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
rectangle(xcen-120,ycen-180,xcen+120,ycen+180);
getch();
closegraph();
}
```



iii. Square

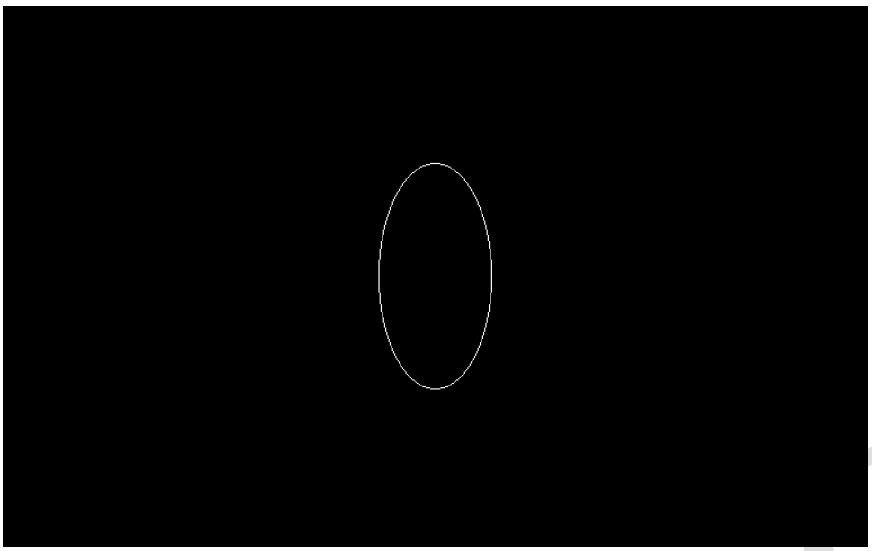
```
#include<graphics.h>
```

```
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
rectangle(xcen-200,ycen-200,xcen+200,ycen+200);
getch();
closegraph();
}
```



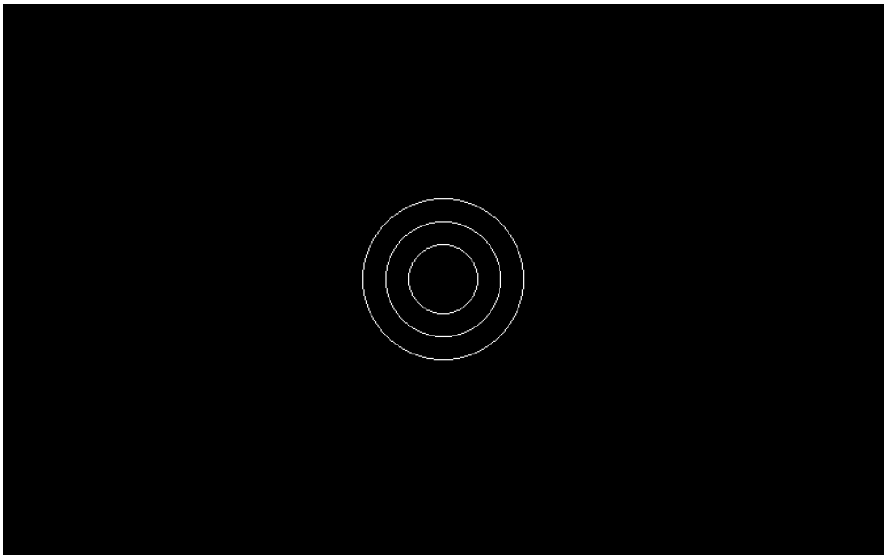
iv. Ellipse

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
ellipse(xcen,ycen,0,360,50,100);
getch();
closegraph();
}
```



v. Concentric Circles

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
circle(xcen,ycen,30);
circle(xcen,ycen,50);
circle(xcen,ycen,70);
getch();
closegraph();
}
```



vi. Line

SAME AS PRAC 1B

CIRCLE WITH COLOR

```
#include<graphics.h>
#include<conio.h>
void main()
{
int gd=DETECT,gm,xcen,ycen;
initgraph(&gd,&gm,"C:\\TC\\BGI");
xcen=getmaxx()/2;
ycen=getmaxy()/2;
setfillstyle(XHATCH_FILL,RED);
circle(xcen,ycen,70);
floodfill(xcen,ycen,WHITE);
getch();
closegraph();
}
```