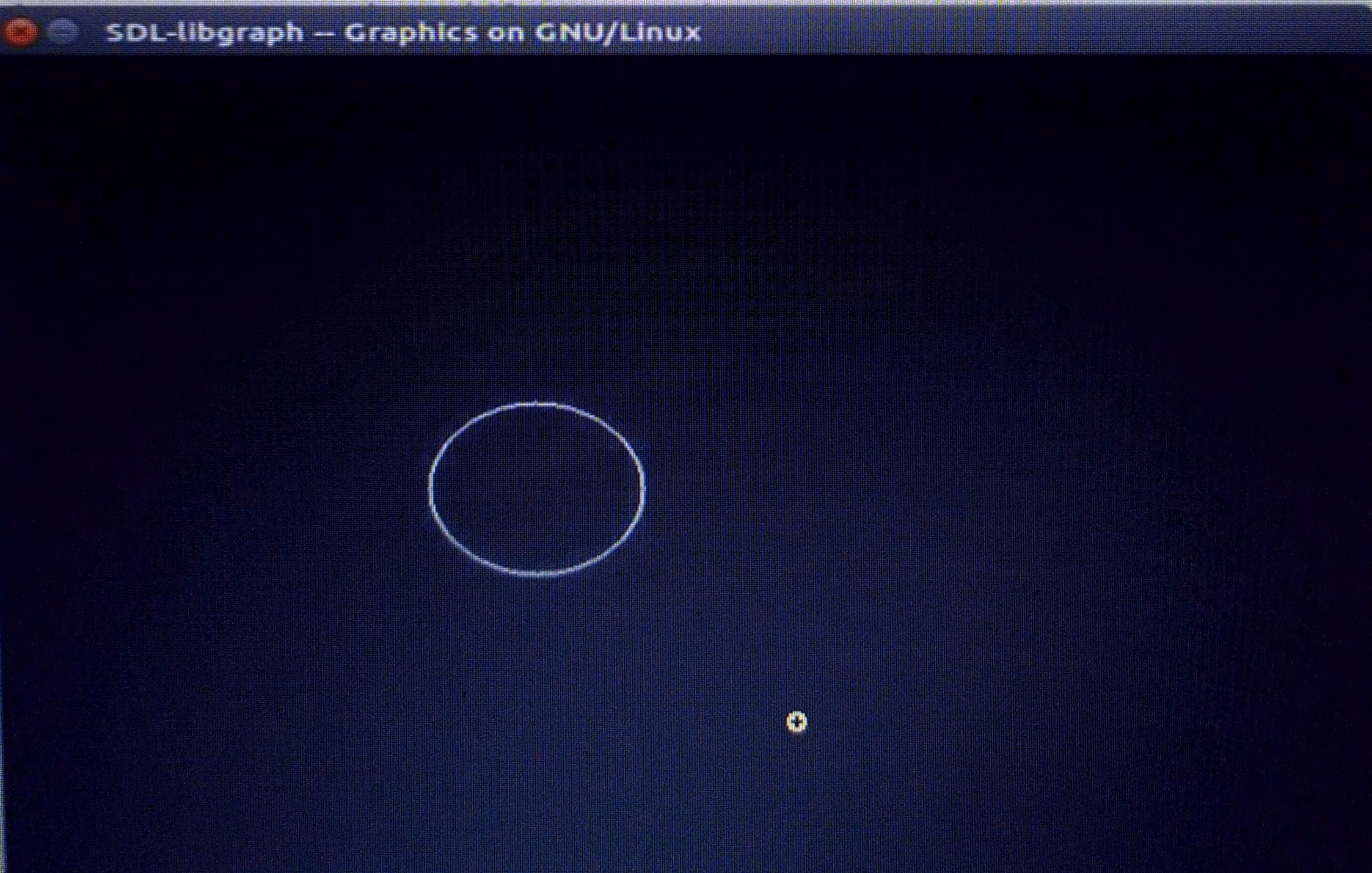
Ans 2:-

```
#include<stdio.h>
#include<graphics.h>
void drawcircle(int xo, int yo, int r)
{
 int x=r;
 int y=0;
 int err=0;
while(x>=y)
{
putpixel(xo+x, yo+y, 7);
putpixel(xo+y, yo+x, 7);
putpixel(xo-y, yo+x, 7);
putpixel(xo-x, yo+y, 7);
putpixel(xo-x, yo-y, 7);
putpixel(xo-y, yo-x, 7);
putpixel(xo+y, yo-x, 7);
putpixel(xo+x, yo-y, 7);
 if(err<=0)
 {
 y+=1;
 err+=2*y+1;
 }
 if(err>0)
 {
 x-=1;
 err-=2*x+1;
 }
}
}
```

```
int main()
{
int gdriver = DETECT, gmode, error, x, y, r;
printf("Enter radius of circle :");
scanf("%d", &r);
printf("Enter co-ordinates of center (x and y) :");
scanf("%d %d", &x, &y);
initgraph(&gdriver, &gmode, "");
drawcircle(x, y, r);
delay(9999999);
return 0;
}
```

adius of circle: 50
o-ordinates of center(x and y): 250 250



Algorithm :-

Procedure :-

Centre of circle = (x_0, y_0)

Radius of circle = R

Step1 :- Assign the starting point coordinates (x_0, y_0) as -

$$x_0 = 0$$

$$y_0 = R$$

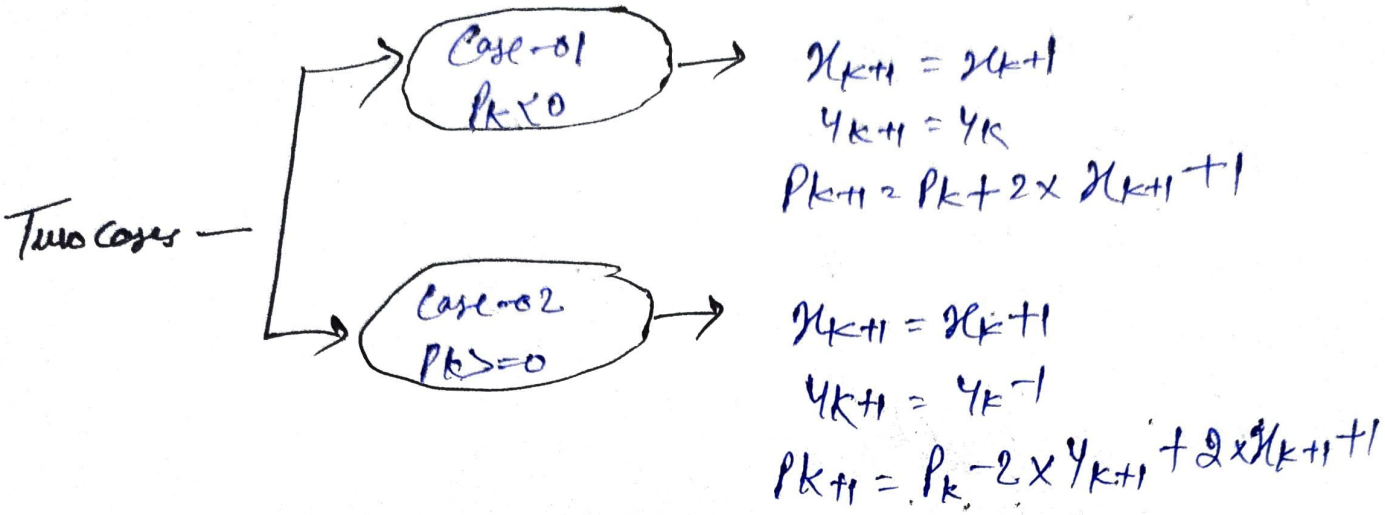
Step2 :- Calculate the value of initial decision parameter P_0 as -

$$P_0 = 1 - R$$

Step3 :- Suppose the current point is (x_k, y_k) and the next point is (x_{k+1}, y_{k+1})

find the next point the first octant depending on the value of decision parameter P_k.

follow the below two cases -



Step 4 :-

If the given centre point (x_0, y_0) is not $(0,0)$ then do the following and plot the point -

$x_{plot} = x_c + x_0$

$y_{plot} = y_c + y_0$

Step 5 :- Keep repeating step-03 and step-04 until $X_{plot} >= Y_{plot}$

Step 6 :- Step 5 generates all the points for one octant.