Q-3-> Write and algorithm and program to implement Bresenham circle drawing algorithm.

Algorithm.->

If d<=0, then NX+1, YX+1, Y is to Be chosen as next pixel.

If d>0, then SX+1, Y-1X, Y-1 is to be chosen as the next pixel.

Step-1: Get the coordinats of the center of the circle and radius, and store them in X, y, and R respectively.

Set P=0 and Q=R.

Step-2: Set decision parameter D = 3-2R.

Step-3: Repeat through step-8 while P≤Q.

Step-4: Call Draw Circle X, Y, P, QX, Y, P, Q.

Step-5: Increment the value of P.

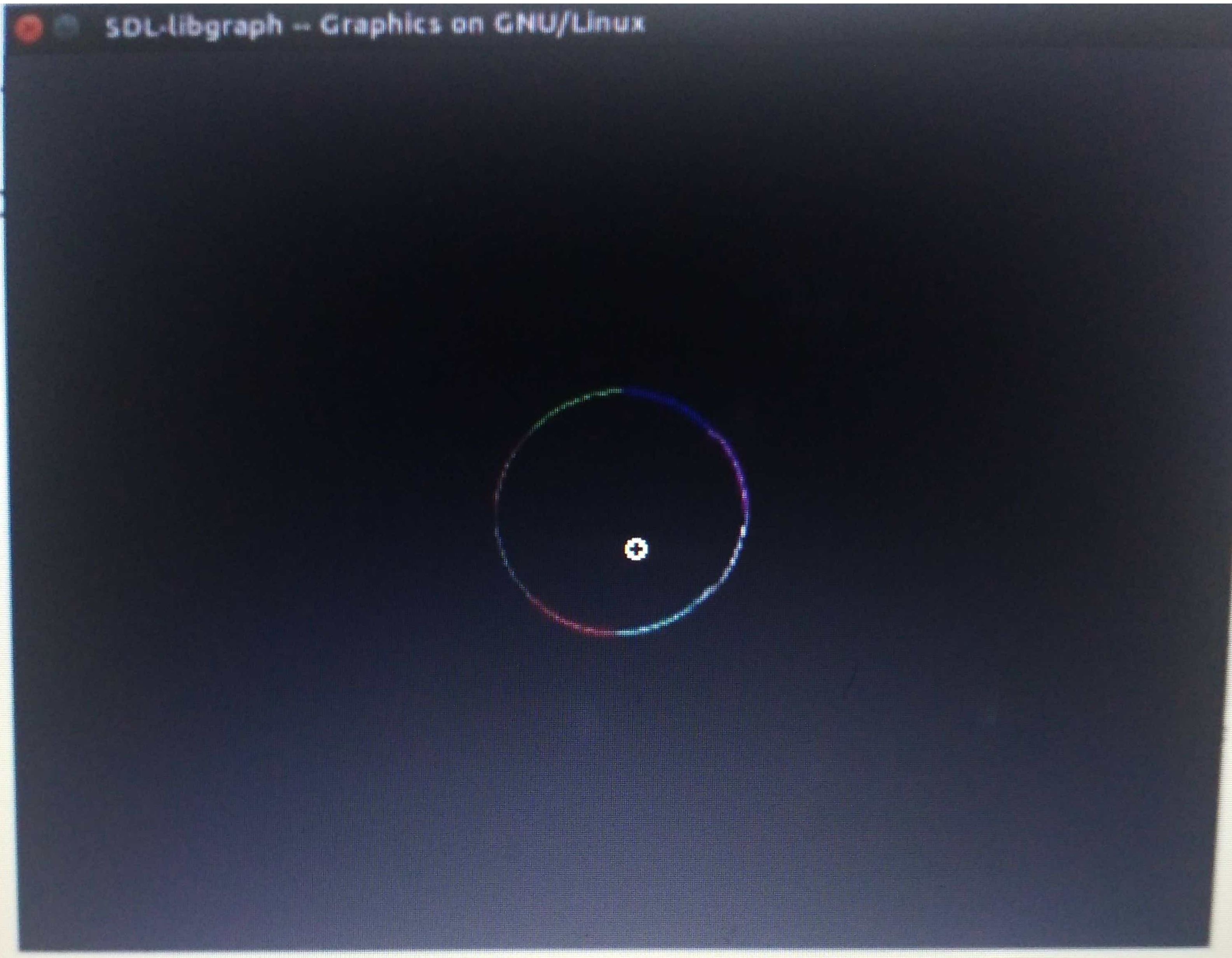
Step-6: If D<0 then D = D+4P+6.

Step-7: Else Set R = R-1, D = D+4P÷QP-Q+10.

Step-8: Call Draw Circle X, Y, P, QX, Y, P, Q

Program →

```
#include <stdio.h>
#include <graphics.h>
void main()
{
  int gd = DETECT, gm;
  int r,x,y,p,xc=320,yc=240;
  printf("Enter the radius");
  scanf("%d",&r);
  initgraph(&gd,&gm," ");
  x=0;
  y=r;
  putpixel(xc+x,yc-y,1);
  p=3-(2*r);
  for(x=0;x<=y;x++)
   {
    if(p<0)
     {
      y=y;
      p=(p+(4*x)+6);
     }
    else
     {
      y=y-1;
      p=p+((4*(x-y)+10));
     }
      putpixel(xc+x,yc-y,1);
      putpixel(xc-x,yc-y,2);
putpixel(xc+x, yc+y, 3);
putpixel(xc+x, yc+y, 4);
putpixel(xc+y, yc-x, 5);
putpixel(xc-y, yc-x, 6);
putpixel(xc+y, yc+x, 7);
putpixel(xc-y, yc+x, 8);
}
   getch();
   closegraph();
  }
```



SDL-libgraph -- Graphics on GNU/Linux