Ans (3) Bresenham's Circle Algo =>

Step 1: Start

S2: Declare p, q, r, x, y, d variables
p, q are coordinates of centre of the circle.
r is radius of circle.

S3: Enter value of r

S4: Calculate $d = 3 - 2r$

S5: initialize
$x = 0$
$y = r$

S6: Check for circle scan converted
if $x >= y$
Stop.

S7: Plot 8 points. Centre is at (p, q)
putpixel (x+p, y+q)
putpixel (y+p, x+q)
putpixel (-y+p, x+q)
putpixel (-x+p, y+q)
putpixel (-x+p, -y+q)
putpixel (-y+p, -x+q)
putpixel (y+p, -x+q)
putpixel (x+p, -y-q)

S8: Find next pixel to be scanned
if $d < 0$
then $d = d + 4x + 6$
increment $x = x + 1$
if $d \geq 0$
then $d = d + 4(x-y) + 10$
increment $x = x + 1$
increment $y = y - 1$

S9: Go to Step 6

S10: Stop.

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

Void points (int q, int n ; int x, int y)
{
   putpixel (x+q ; y+n ,RED);
   putpixel (x+q , -y+n , YELLOW);
   putpixel (-x+q , -y+n , GREEN);
   putpixel (-x+q , y+n , YELLOW);
   putpixel (y+q , y+n , 12);
   putpixel (y+q , -x+n , 14);
   putpixel (-y+q , -x+n, 15);
   putpixel (-y+q, y+n, 6);
}

Void Bresh (int q , int n , s).
{
   int x=0, y=s; d = 3-(2*s);
   points (q, n, x, y);
   while (x<=y)
   {
      int (d<=0)
        {
          d = d+(4*x)+6;
        }
      else
        {
          d = d+(4*x)-(4*y)+10;
          y = y+1;
        }
      x = x+1;
   Points (q, n, x, y);
   }
}
```

```
int main(void)
{
   int q, r, s, gdriver = DETECT, gmode, errorcode;
   initgraph (&gdriver, &gmode, " C ");
   errorcode = graphresult ();
   if (errorcode != grOK)
   {
     printf (" Error ", grapherrormsg (errorcode));
     printf (" Press any Key :");
     getch ();
     exit (1);
   }

   printf(" Enter value of q & r :");
   Scanf (" %d %d ", &q, &r);
   Print f ( " Enter the values of radius :");
   scanf (" %d ", &s);
   Bresh (q, r, s);
   getch ();
   closegraph ();
   return 0;
}
```