

Name - Vaibhav Khatu

University Roll.No. - 1121158

Class Roll.No. - 43

Subject - Computer Graphics Lab

Subject Code - PBC 602

Section - C

Q2 \Rightarrow Algorithm

Step 1 :- Start

Step 2 :- Input center (x_c, y_c) of circle,
radius r .

Step 3 :- Assign starting coordinate
 $x_0 = 0$ and $y = r$

Step 4 :- Calculate initial parameter
 $P_0 = 1.25 - r$

Step 5 :- In first octant, coordinate will be

if $P_k < 0$ then

$$x_{new} = x_{old} + 1$$

$$y_{new} = y$$

$$P_{k+1} = P_k + 2 * x + 1$$

else

$$x_{new} = x_{old} + 1$$

$$y_{new} = y - 1$$

$$P_{k+1} = P_k + 2 * (x - y) + 1$$

Step 6 :- If center (x_c, y_c) is not $(0, 0)$ then

$$x_{plot} = x_c + x$$

$$y_{plot} = y_c + y$$

Step 7 :- Keep repeating step 5 and step 6 until
 $x_{plot} \geq y_{plot}$. Also plot points in other
octants using eight symmetry property of circle.

Step 8 :- Stop.

Program

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

void main ()
{
    int gd = DETECT, gm ;

    int x, y, xc, yc, r ;

    printf ("Enter center of circle :");
    scanf ("%d %d", &xc, &yc);

    printf ("Enter radius ");
    scanf ("%d", &r);

    x = 0 ;
    y = r ;
    r = 1.25 * r ;

    initgraph (&gd, &gm, "") ;

    do {
        putpixel (xc+x, yc+y, 2) ;
        putpixel (xc+y, yc+x, 2) ;
        putpixel (xc+y, yc-x, 2) ;
        putpixel (xc+x, yc-y, 2) ;
        putpixel (xc-x, yc-y, 2) ;
        putpixel (xc-y, yc-x, 2) ;
        putpixel (xc-y, yc+x, 2) ;
        putpixel (xc-x, yc+y, 2) ;
    }
```

if ($p < 0$)

{

$x = x + 1$;

$p = p + 2 * x + 1$;

}

else {

$x = x + 1$;

$y = y - 1$;

$p = p + 2 * (x - y) + 1$;

}

delay(1000) ;

{

while ($x \leq y$) ;

getch() ;

closegraph() ;

}