Name → Hrithik Rawal

Roll no. → 1121065

Subject → Computer Graphics Pratical

Sub code → PBC-602

Ques 2 → Mid point circle drawing Algorithm.

Algorithm →

Steps →

step 1 → Assign the starting point coordinates (X_0, Y_0) as-

$X_0 = 0$

$Y_0 = R$

step 2 → Calculate the value of initial decision parameter

P_0 as $P_0 = 1 - R$

step 3 → Suppose the current point is (X_K, Y_K) and the the next point is (X_{K+1}, Y_{K+1})

Find the next point of the first octant depending on the value of decision parameter P_K.

Follow the below cases:-

Case 1 → if $P_K < 0$ →

$X_{K+1} = X_K + 1$

$Y_{K+1} = Y_K$

$P_{K+1} = P_K + 2 * X_{K+1} + 1$

Case 2 → if $P_K >= 0$ →

$X_{K+1} = X_K + 1$

$Y_{K+1} = Y_K - 1$

$P_{K+1} = P_K - 2 * Y_{K+1} + 2 * X_{K-1} + 1$

Step 4 → If the given centre point (X_0, Y_0) is not $(0, 0)$ then do the following

$X_{plot} = X_c + Y_0$

$Y_{plot} = Y_c + X_0$

there (X_c, Y_c) denotes the current value of x and y coordinates

Step 5 → Keep repeating step-03 and step-04 untill $x_{plot} >= y_{plot}$

Step 6 → step 5 generates all the point of one octant to find the other seven octant follow the eight symmetry property of circle.

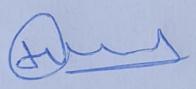
Program →

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

Void drawcircle (int xo, int yo, int radius)
  {
    int x = radius;
    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--;
    err -= 2*x+1;
  }
 }
}

int main()
  {
    int gd = DETECT, gm, errorcode, x, y, r;
    printf("Enter radius: ");
    scanf("%d", &r);
    Printf("Enter cordinates of center (x and y)");
    scanf("%d %d", &x, &y)
     initgraph(&gd, &gm, "");
     drawcircle(x, y, r);
     delay(999999);
     return 0;
   }
```



NeuTroN DOS-C++ 0.77, Cpu speed: max 100% cycles, Frameskip 0, Program:

```
Enter radius of circle: 100
Enter co-ordinates of center(x and y): 150
150
```