Ans 2 →

Mid Point Circle Drawing Algorithm :

Step 1 → Start

Step 2 → Assign the starting point coordinates (x_0, y_0) as →

- $x_0 = 0$
- $y_0 = R$

Step 3 → Calculate the value of initial decision parameter P_0 as →

$$P_0 = 1 - R$$

Step 4 → Suppose the current point is (x_k, y_k) and the next point is (x_{k+1}, y_{k+1})

Case 1 → $(P_k < 0)$

$$x_{k+1} = x_k + 1$$

$$y_{k+1} = y_k$$

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

Case 2 → $(P_k >= 0)$

$$x_{k+1} = x_k + 1$$

$$y_{k+1} = y_k - 1$$

$$P_{k+1} = P_k - 2 * y_{k+1} + 2 * x_{k+1} + 1$$

Step 5 → If the given centre point (x_0, y_0) is not $(0,0)$ then do the following & plot the point →

- $X_{plot} = x_c + x_0$
- $Y_{plot} = y_c + y_0$

Step 6 → keep repeating Step 4 and Step 5 until

$X_{plot} >= Y_{plot}$

Step 7 → Step 6 generates all the points for one octant.

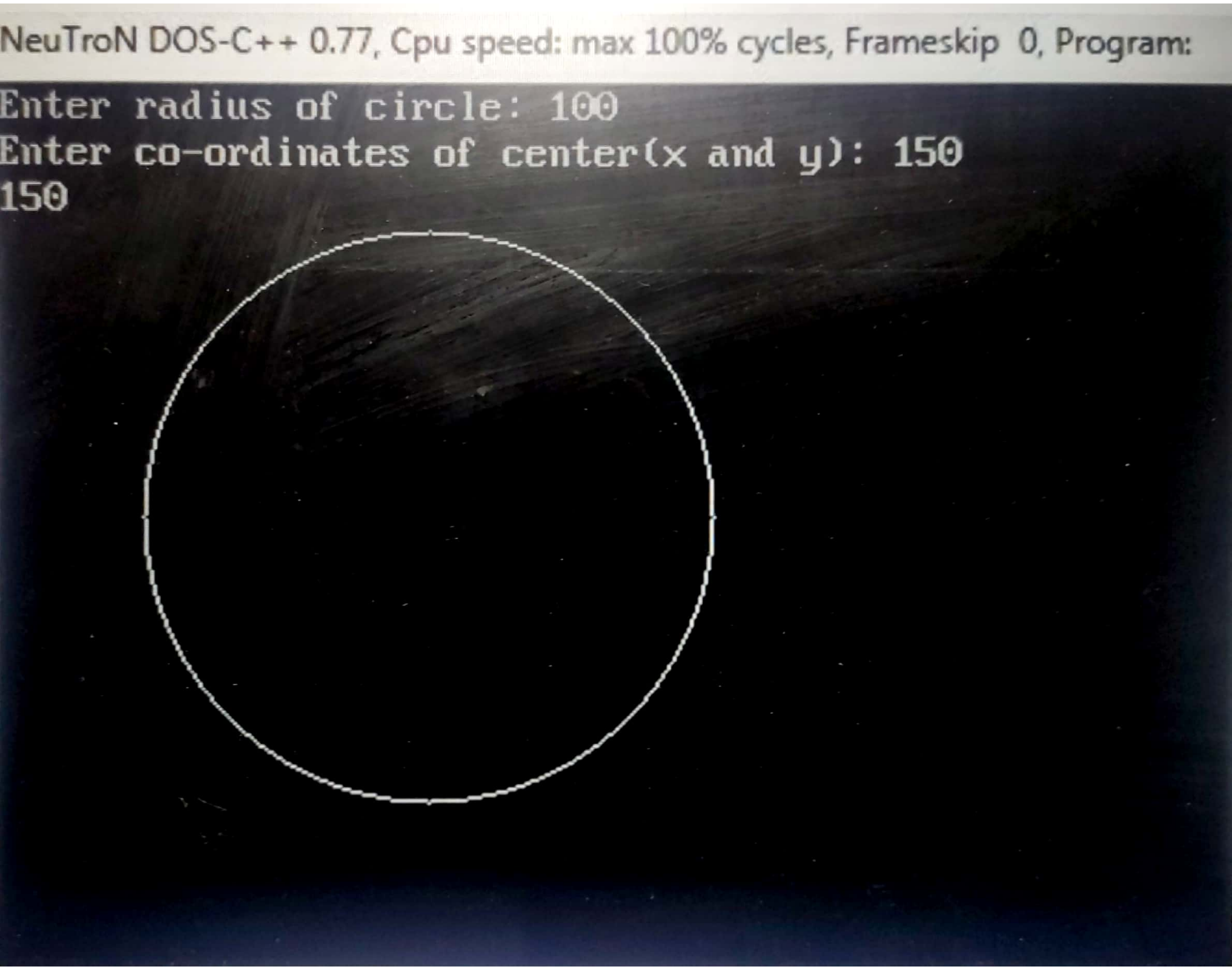
To find the point for other seven octants follow the eight symmetry property of circle.

Step 8 → End

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- = 1 ;
      err - = 2*x + 1 ;
 }
    }
}

int main ()
 {   int gd = DETECT, gm , error, x, y , r;
     initgraph ( &gd , &gm , "" );
     printf ( " Enter radius of circle : " );
     scanf ( "%d", &r ) ;

     printf (" Enter coordinates of center (x,y) : " );
     scanf ( "%d %d" , & x , &y );
     drawcircle ( x,y,r );
     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
```