Scan Conversion Of Circle

Course Title: Computer Graphics
Course Code: CSE - 413

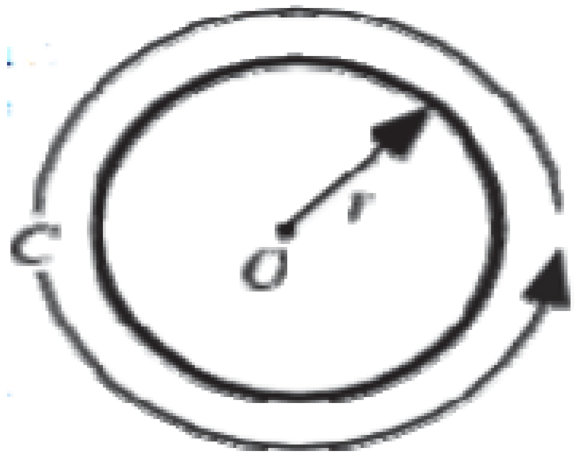
Outlines:-

- Circle
- 8-way symmetry of a point in a circle
- Defining a circle using Polynomial Method
- Defining a circle using Polar Co-ordinates
- Bresenham circle drawing algorithm
- Mid-point circle drawing algorithm

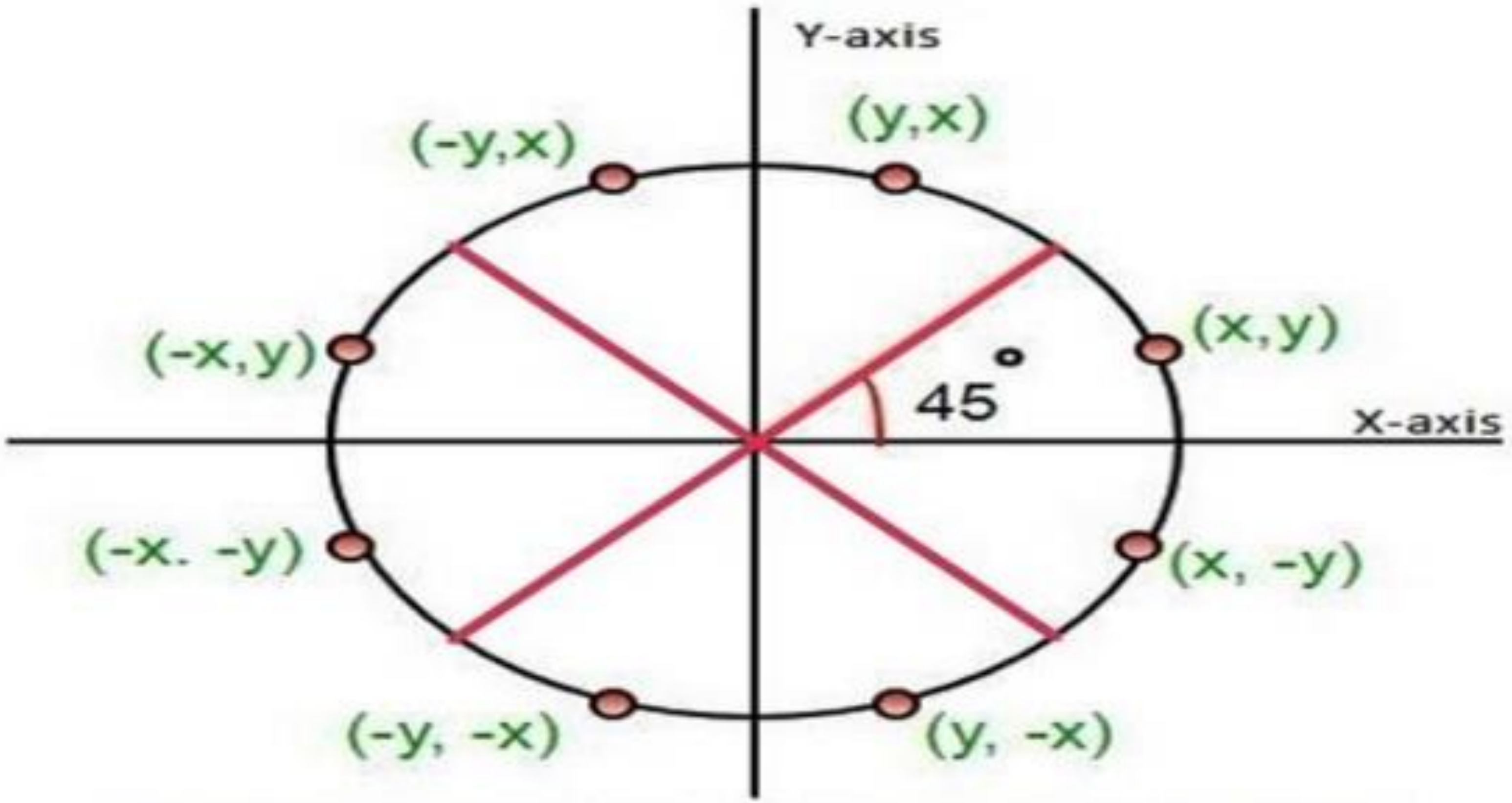
Circle

A circle is the set of points in a plane that are equidistant from a given point O. The distance r from the center is called the radius, and the point O is called the center. Twice the radius is known as the diameter **d=2r**. The angle a circle subtends from its center is a full angle, equal to 360° or 2π radians.

A circle has the maximum possible area for a given perimeter, and the minimum possible perimeter for a given area. The perimeter C of a circle is called the circumference, and is given by $\mathbf{C = 2 \pi r}$.

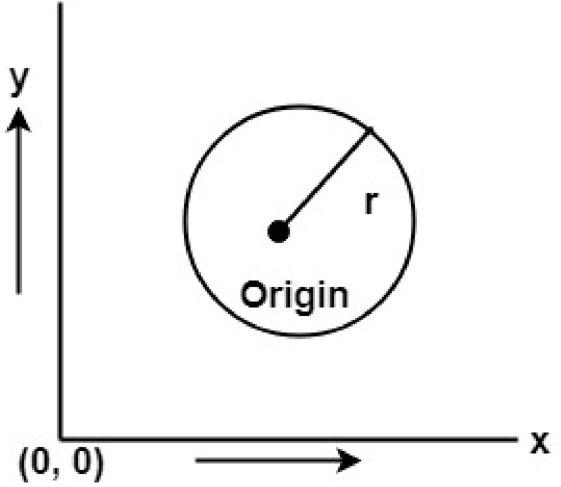


It is an eight-way symmetric figure which can be divided into four quadrants and each quadrant has two octants. This symmetry helps in drawing a circle on a computer by knowing only one point of any octant.



For each pixel (x,y) all possible pixels in 8 octants

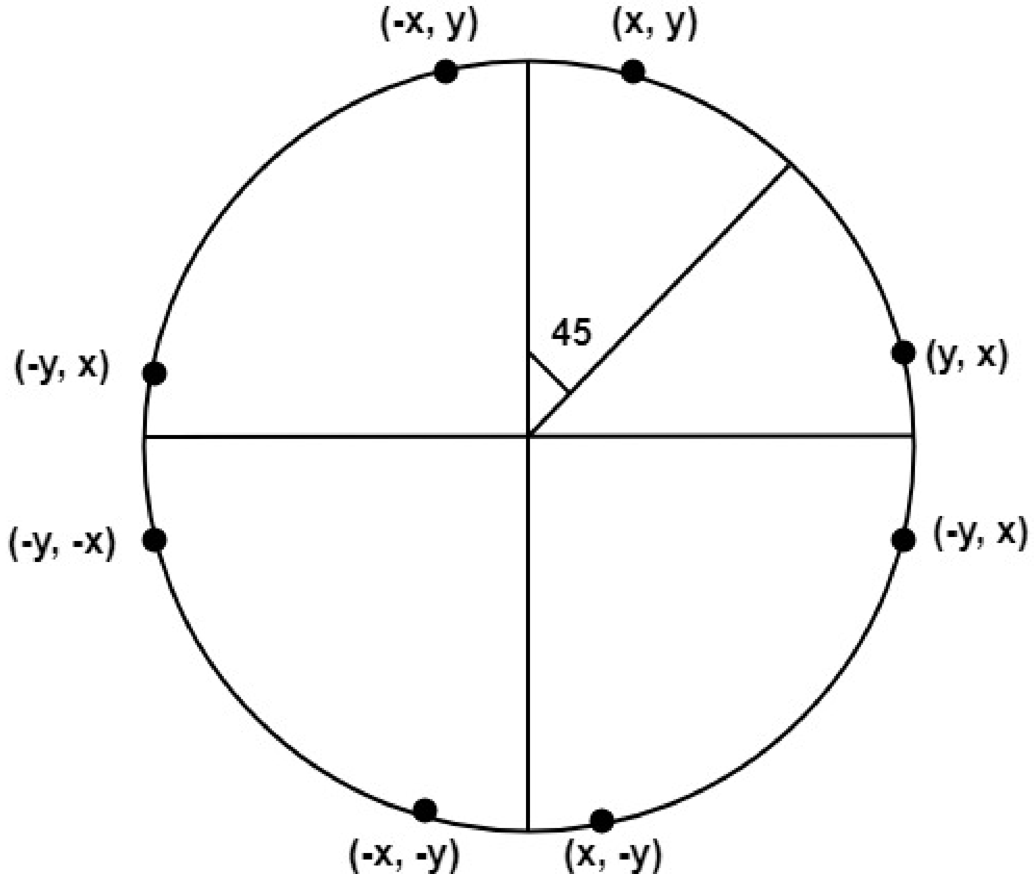
For drawing, circle considers it at the origin.
If a point is $P_1(x, y)$, then the other seven points will be
So we will calculate only 45°arc. From which the whole circle can be determined easily.



P2 (x, -y)
P3(-x, -y)
P4 (-x, y)
P5 (y, x)
P6 (y, -x)
P7 (-y, -x)
P8 (-y, x)

If we want to display circle on screen then the putpixel function is used for eight points as shown below:

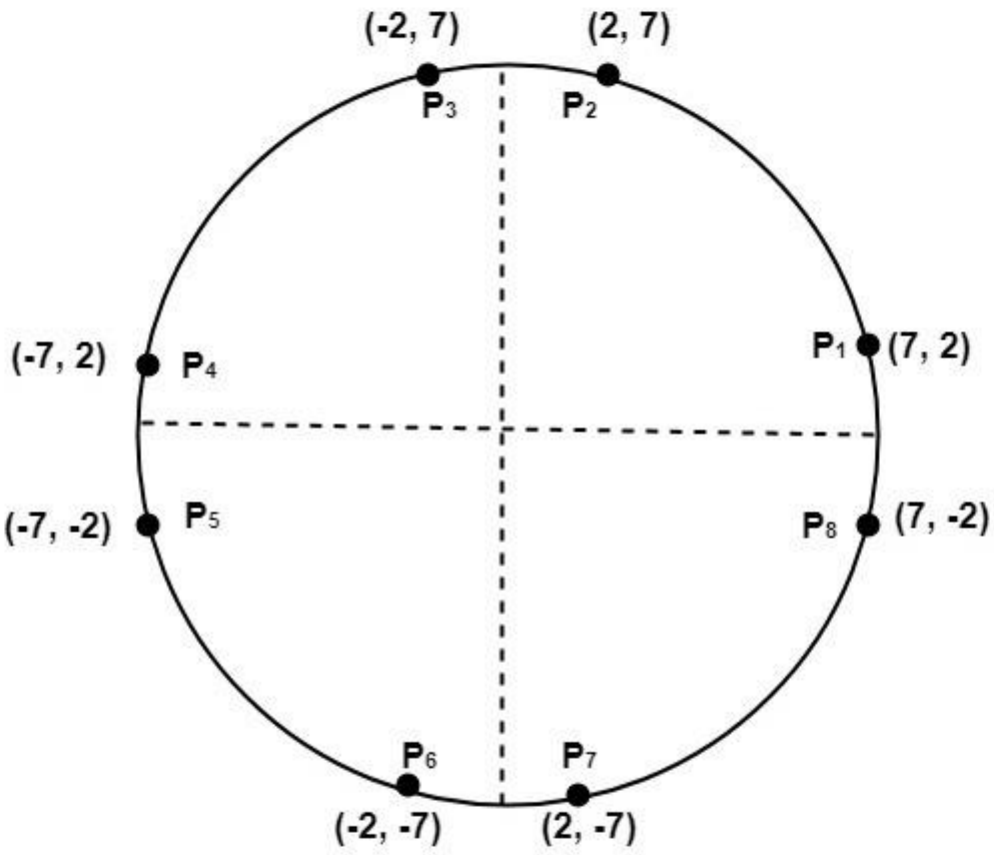
putpixel (x, y, color)
putpixel (x, -y, color)
putpixel (-x, y, color)
putpixel (-x, -y, color)
putpixel (y, x, color)
putpixel (y, -x, color)
putpixel (-y, x, color)
putpixel (-y, -x, color)



Example: Let we determine a point (2, 7) of the circle then other points will be = (2, -7), (-2, -7), (-2, 7), (7, 2), (-7, 2), (-7, -2), (7, -2)

These seven points are calculated by using the property of reflection. The reflection is accomplished in the following way:

The reflection is accomplished by reversing x, y co-ordinates.



Eight way symmetry of a Circle

There are two standards methods of mathematically defining a circle centered at the origin.

1. Defining a circle using Polynomial Method
2. Defining a circle using Polar Co-ordinates

1. Direct or Polynomial Method

This technique uses the equation for a circle on radius r centered at (0, 0) given as:

$x 2+ y2 = r2$,

an obvious choice is to plot

$y = \pm \sqrt{}(r2-x2)$

Obviously in most of the cases the circle is not centered at (0, 0), rather there is a center point (h, k); other than (0, 0). Therefore, the equation of the circle having center at point (h, k):

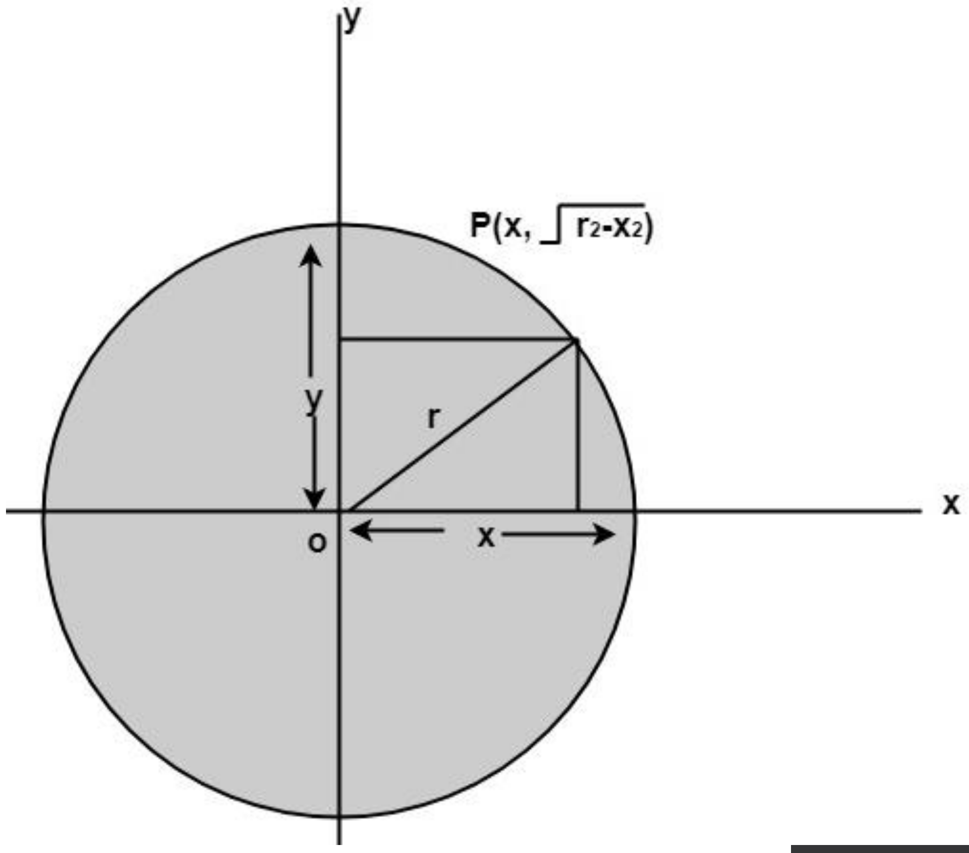
$(x - h) 2 + (y - k) 2 = r2$

For each value of x, value of y can be calculated using,

$y = k \pm \sqrt{}r2 – (x - h) 2$

The initial points will be: x = h – r, and y = k

This is a very ineffective method because for each point value of h, x and r are squared and then subtracted and then the square root is calculated, which leads to high time complexity. It also creates large gaps in the circle for values of x close to r as shown in the following figure.



Algorithm: Direct or Polynomial Method

Step1: Set the initial variables

r = circle radius

(h, k) = coordinates of circle center

x=o

I = step size

x_{end}= r / √2

Step2: Test to determine whether the entire circle has been scan-converted.

If x > x_{end} then stop.

Step3: Compute y = ± √(r2-x2)

Step4: Plot the eight points found by symmetry concerning the center (h, k) at the current (x, y) coordinates.

Plot (x + h, y +k)	Plot (-x + h, -y + k)
Plot (y + h, x + k)	Plot (-y + h, -x + k)
Plot (-y + h, x + k)	Plot (y + h, -x + k)
Plot (-x + h, y + k)	Plot (x + h, -y + k)

Step5: Increment x = x + i

Step6: Go to step (ii).

2. **Polar coordinates Method** A better approach, to eliminate unequal spacing as shown in above figure is to calculate points along the circular boundary using polar coordinates r and θ.

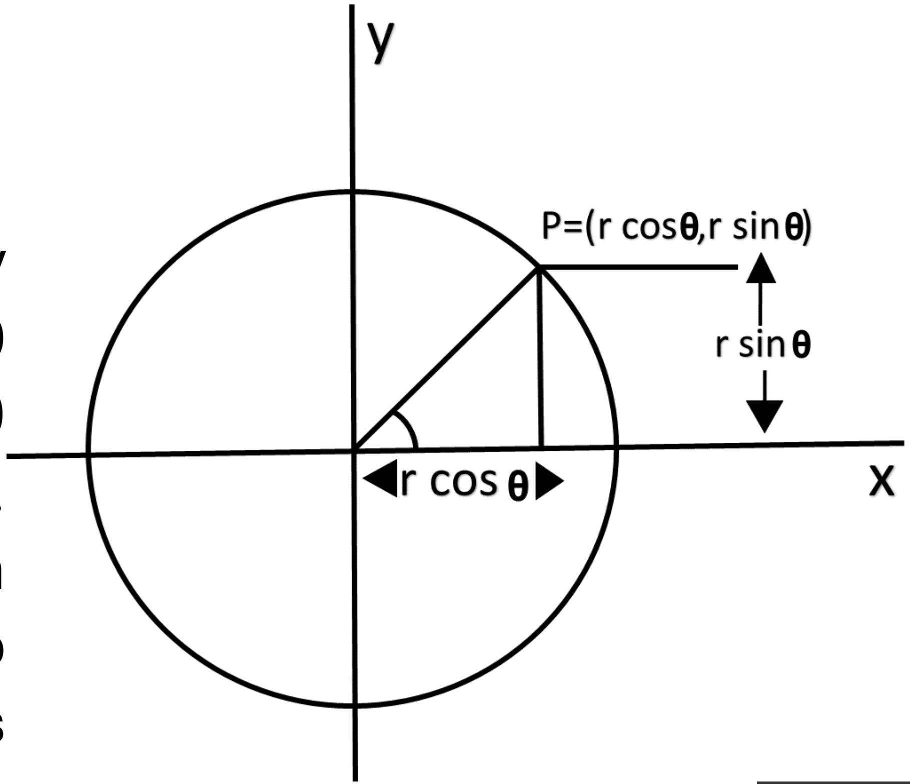
Expressing the circle equation in parametric polar form yields the pair of equations

$x = r \cos\theta$ ...(1)

$y = r \sin\theta$ ...(2)

X2+y2 = r2

Using above equation circle can be plotted by calculating x and y coordinates as θ takes values from 0 to 360 degrees or 0 to 2π. The step size chosen for θ depends on the application and the display device. Larger angular separations along the circumference can be connected with straight-line segments to approximate the circular path. For a more continuous boundary on a raster display, we can set the step size at 1/r. This plots pixel positions that are approximately one unit apart.



Now let us see how this technique can be sum up in algorithmic form.

Circle (h, k, radius)

for $\theta = 0$ to 2π; step $1/r$

$x = h + r * \cos\theta$

$y = k + r * \sin\theta$

drawPixel (x, y)

Again, this is very simple technique and also solves problem of unequal space but unfortunately this technique is still inefficient in terms of calculations involves especially floating-point calculations.Calculations can be reduced by considering the symmetry of circles.

We know that the shape of circle is similar in each quadrant.

For the first octant i.e., from $\theta = 0^0$ to $\theta = 45^0$

The values of x and y are calculated using equations (1) and (2). Simultaneously, all the eight points for the rest of the octants can be calculated using the eightway symmetry property of the circle. Then, the plotting of the circle is done.

Hence, we have reduced half the calculations by considering symmetric octants of the circle but as we discussed earlier inefficiency is still there and that is due to the use of floating-point calculations. In next algorithm we will try to remove this problem.

Algorithm for Polar coordinates Method

Step1: Set the initial variables:

r = circle radius
(h, k) = coordinates of the circle center

Step2: for $\theta = 0$ to $\pi / 4$;

Compute $x = h + r * \cos\theta$

$y = k + r * \sin\theta$

DrawSymmetricPoints (h, k, x, y)

Step3: DrawSymmetricPoints (h, k, x, y)

(i.e., the center (h, k), at the current (x, y) coordinates)

Plot (x + h, y + k) Plot (-x + h, -y + k)

Plot (y + h, x + k) Plot (-y + h, -x + k)

Plot (-y + h, x + k) Plot (y + h, -x + k)

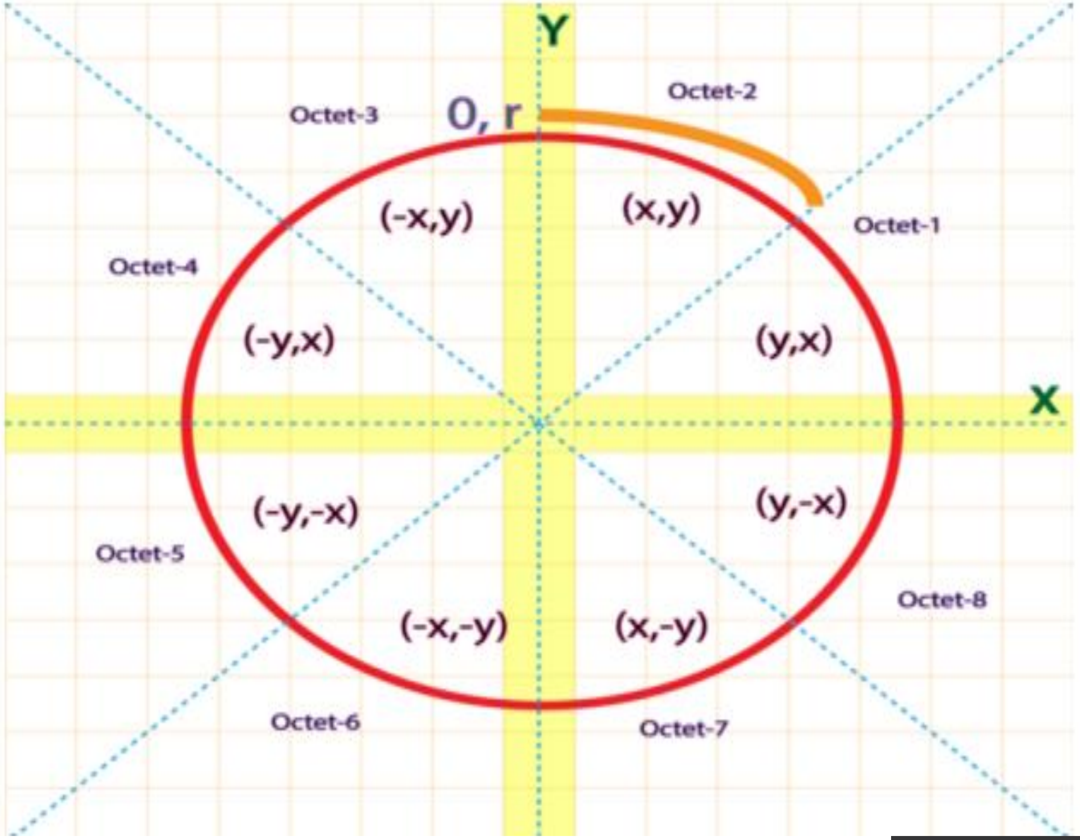
Plot (-x + h, y + k) Plot (x + h, -y + k)

Step4: Go to step2.

Step5: Stop

Bresenham Circle Drawing Algorithm

The Bresenham's circle drawing algorithm is a circle drawing algorithm which calculates all the nearest points nearest to the circle boundary. It is an incremental method (i.e., we increment one of the coordinates of the point and calculate the other coordinate according to it. In this manner we find all the points of that particular polygon). It only uses integer arithmetic which makes it's working faster as well as less complex. The strategy that is followed in this algorithm is to select the pixel which has the least distance with the true circle boundary and with then keep calculating the successive points on the circle.



As we know that the circle follows 8 symmetry properties, i.e., if we know the boundary coordinates of the first octant, the rest 7 octant's value can be easily calculated by changing their magnitudes or by interchanging the coordinate values according to the respective octants. Scan-Converting a circle using Bresenham's algorithm works as follows: Points are generated from 90° to 45°, moves will be made only in the +x & -y directions. This can be well illustrated by the following diagram:

This Algorithm calculates the location of the pixels similarly. Here, we calculate the values only for the first octant and the values for the rest octants can be calculated by extending these values to the other 7 octants, using the eight-way symmetry property of the circle.

Working of Bresenham's Circle Drawing Algorithm

Let us assume we have a point **p (x, y)** on the boundary of the circle and with **r** radius satisfying the equation **f_c (x, y) = 0**

As we know the equation of the circle is – **$f_c (x, y) = x^2 + y^2 = r^2$**

$$if\ \text{fc}\ (x, y) = \begin{cases} < 0 & \text{The point } (x, y) \text{ is inside the circle boundary.} \\ = 0 & \text{The point } (x, y) \text{ is on the circle boundary.} \\ > & \text{The point } (x, y) \text{ is outside the circle boundary.} \end{cases}$$

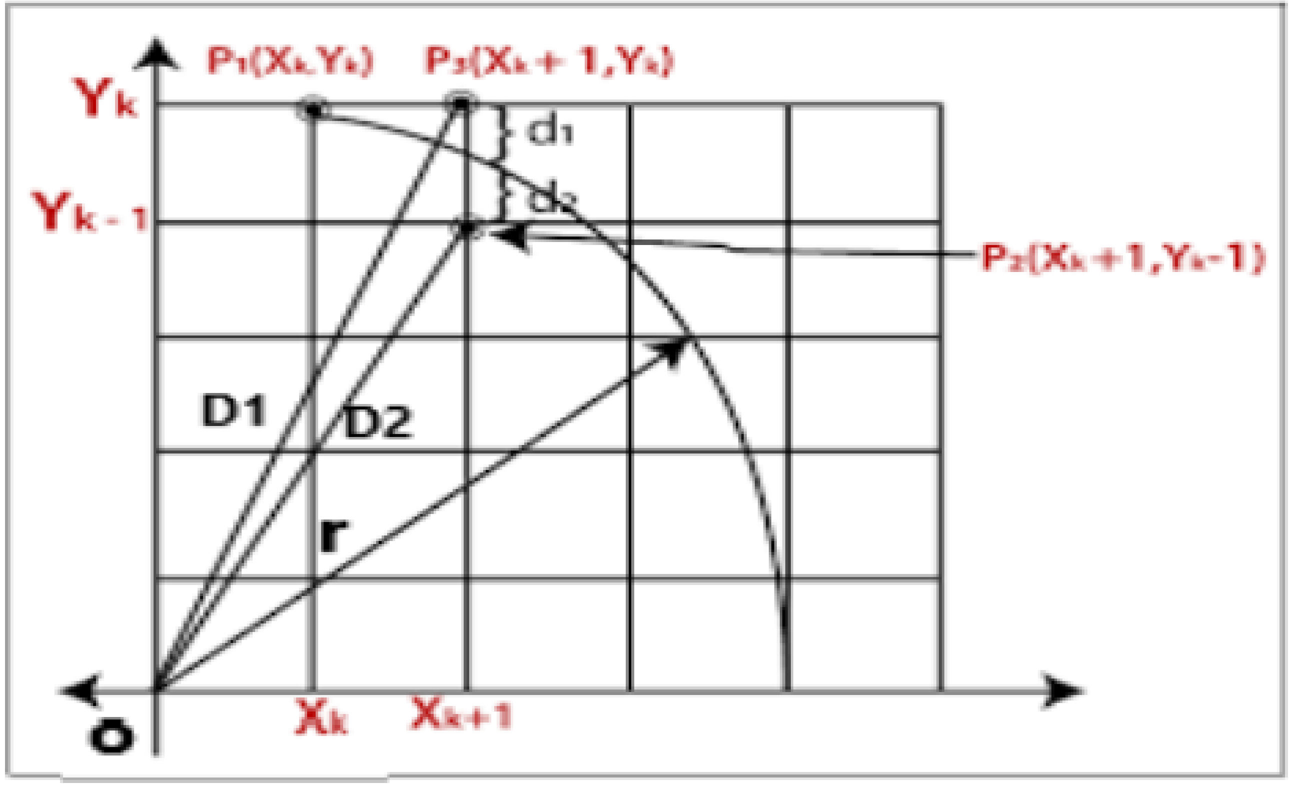
The Algorithm works in the following way:

Let us consider a point **(x_k + 1, y)** on true circle boundary having a radius **'r'**. When the circle passes through two pixels simultaneously then the one to chosen will be decided on the basis of their least distance with the circle.

There can be two positions from which the circle passes:

a. to the top, say point P3

b. to the bottom, say point P2



The coordinates for point **P3** will be $(x_k + 1, y_k)$.

The coordinates for point **P2** will be $(x_k + 1, y_k - 1)$.

Let, the distance from origin O to point P3= D_1

And the distance from origin O to point P2 = D_2

According to Pythagoras theorem we can write:

$D_1 = \sqrt{(x_k + 1)^2 + y_k^2}$

$D_2 = \sqrt{(x_k + 1)^2 + (y_k - 1)^2}$

We also assume that,

The distance between point P_3 and circle boundary = d_1

The distance between point P_2 and circle boundary = d_2

To simply the algorithm, we have to avoid the squared root calculations:

$d_1 = D_1^2 - r^2 = (x_k + 1)^2 + (y_k)^2 - r^2$ [always +ve because P_3 is outside of the circle boundary]

$d_2 = D_2^2 - r^2 = (x_k + 1)^2 + (y_k - 1)^2 - r^2$ [always –ve because P_2 is inside of the circle boundary]

Let us introduce the decision parameter D_k which will help in the selection process of the successive points of the circle (Since $\mathbf{d_1}$ will always be +ve & $\mathbf{d_2}$ will always be -ve). So, we can write D_k as follows:

$D_k = \mathbf{d_1} + \mathbf{d_2}$

$$= (x_k + 1)^2 + (y_k)^2 - r^2 + (x_k + 1)^2 + (y_k - 1)^2 - r^2$$

$$= 2(x_k + 1)^2 + (y_k)^2 - 2r^2 + (y_k - 1)^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(1)}$$

For next decision variable:

$$D_{k+1} = 2(x_{k+1} + 1)^2 + (y_{k+1})^2 - 2r^2 + (y_{k+1} - 1)^2 \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \text{(2)}$$

If $\mathbf{d_k < 0}$, then **point P_3 is closer to circle boundary, and the final coordinates are-**

$$(x_{k+1}, y_{k+1}) = (x_k + 1, y_k)$$

If $\mathbf{d_k >= 0}$, then **point P_2 is closer to circle boundary, and the final coordinates are-**

$$(x_{k+1}, y_{k+1}) = (x_k + 1, y_k - 1)$$

Now, we find the difference between decision parameter equation (2) – equation (1)

$d_{k+1} - d_k = 2(x_{k+1}+1)^2 + (y_{k+1})^2 - 2r^2 + (y_{k+1} - 1)^2 - 2(x_k + 1)^2 - (y_k)^2 + 2r^2 - (y_k - 1)^2$

For both points, put $x_{k+1} = x_k + 1$

$d_{k+1} = d_k + 2(x_k + 1+1)^2 + (y_{k+1})^2 - 2r^2 + (y_{k+1} - 1)^2 - 2(x_k + 1)^2 - (y_k)^2 + 2r^2 - (y_k - 1)^2$

$= d_k + 2(x_k + 2)^2 + (y_{k+1})^2 - 2r^2 + (y_{k+1} - 1)^2 - 2(x_k + 1)^2 - y_k^2 + 2r^2 - (y_k - 1)^2$

$= d_k + 2(x_k^2 + 2.\ x_k.2 + 4) + y_{k+1}^2 + (y_{k+1}^2 - 2.\ y_{k+1}.1 + 1) - 2(x_k^2 + 2.\ x_k.1 + 1) - y_k^2 - (y_k^2 - 2.$

$y_k.1 + 1)$

$= d_k + 2x_k^2 + 8x_k + 8 + y_{k+1}^2 + y_{k+1}^2 - 2y_{k+1} + 1 - 2x_k^2 - 4x_k - 2 - y_k^2 - y_k^2 + 2y_k - 1$

$= d_k + 4x_k + 6 + 2y_{k+1}^2 - 2y_{k+1} - 2y_k^2 + 2y_k$

$d_{k+1} = d_k + 4x_k + 2(y_{k+1}^2 - y_k^2) - 2(y_{k+1} - y_k) + 6$

Now, we check two conditions for decision parameter-

Condition 1: If $d_k < 0$, **then** $y_{k+1} = y_k$ (We select point P_3)

So, we get

$d_{k+1} = d_k + 4x_k + 2(y_k^2 - y_k^2) - 2(y_k - y_k) + 6$

$= d_k + 4x_k + 6$

Condition 2: If $\mathbf{d_k >= 0}$, **then** $\mathbf{y_{k+1} = y_k\text{-}1}$ (We select point P_2)

So, we get

$d_{k+1} = d_k + 4x_k + 2[(y_k\text{-}1)^2 - y_k^2)] - 2(y_k\text{-}1 - y_k) + 6$

$= d_k + 4x_k + 2[(y_k{}^2 - 2.y_k.1+1) - y_k^2)] + 2+6$

$= d_k + 4x_k + 2y_k{}^2 - 4y_k + 2 - 2y_k^2+8$

$= d_k + 4x_k - 4y_k + 10$

$= d_k + 4(x_k - y_k) + 10$

Now, we calculate initial decision parameter $\mathbf{d_0}$, the initial points will be (0, r) [it is assumed that the circle is centered at the origin, then at the first step $x_0 = 0$ & $y_0 = r$].

From equation (1):

$D_k = 2\,(x_k + 1)^2 + (y_k)^2 - 2r^2 + (y_k - 1)^2$

So, $D_0 = 2\,(x_0 + 1)^2 + (y_0)^2 - 2r^2 + (y_0 - 1)^2$

$D_0 = 2\,(0 + 1)^2 + (r)^2 - 2r^2 + (r - 1)^2$

$D_0 = 2 + r^2 - 2r^2 + r^2 - 2r + 1$

$D_0 = 3 - 2r$

Algorithm for Bresenham's Circle Drawing Algorithm

Step 1: Declare x, y, r, h, k and d as variables, where (h, k) are coordinates of the center.
Step 2: Calculate the decision parameter as follows: D = 3 - 2r
Step 3: Initialize x = 0, y = r
Step 4: If x >= y Stop.

Step 5: Plot eight points by using concepts of eight-way symmetry. The center is at (h, k). Current active pixel is (x, y).

putpixel (x+h, y+k)
putpixel (y+h, x+k)
putpixel (-y+h, x+k)
putpixel (-x+h, y+k)
putpixel (-x+h, -y+k)
putpixel (-y+h, -x+k)
putpixel (y+h, -x+k)
putpixel (x+h, -y-k)

Step 6: **If D < 0 → then D = D + 4x + 6**
x = x + 1
y = y
If D ≥ 0 → then D = D+ 4 (x - y) + 10
x = x + 1
y = y – 1
Step 7: Go to step 5.

Advantages of Bresenham's Circle Drawing Algorithm

1. The Bresenhem’s circle drawing algorithm uses integer arithmetic which makes the implementation less complex.
2. Due to its integer arithmetic, it is less time-consuming.
3. This algorithm is more accurate than any other circle drawing algorithm as it avoids the use of round off function.

Disadvantages of Bresenham's Circle Drawing Algorithm

1. This algorithm does not produce smooth results due to its integer arithmetic as it fails to diminish the zigzags completely.
2. The Bresenhem’s circle drawing algorithm is not accurate in the case of drawing of complex graphical images.

Example-01: The radius of a circle is 7, and center point coordinates are (0, 0). Apply bresenham's circle drawing algorithm to plot all points of the circle.

Solution: Given radius r= 7 And initial center coordinates (h, k): (0, 0)

Initialize x = 0, and y = r=7 Calculate D= 3 – 2r = 3 – 2.7= -11

Step	X	Y	(x,y)	Plot (x+h, y+k) = (x + 0, y +0)	Previous Decision parameter(D)	D<0 : D = D + 4x + 6 [x = x+1, y=y] D>= 0 : D = D + 4(x – y) + 10 [x = x+1, y=y-1]
1	0	7	(0, 7)	= (0, 7)	- 11	D = D + 4x + 6 = -11 + 4*0 + 6 = -5 x = x+1 = 0+1 = 1 y = 7 new points = (1, 7)
2	1	7	(1, 7)	= (1, 7)	-5	D = D + 4x + 6 = -5 + 4*1 + 6 = 5 x = x + 1 = 1+1 =2 y = 7 new points = (2, 7)
3	2	7	(2, 7)	= (2, 7)	5	D = D + 4(x – y) + 10 =5 + 4(2-7) + 10 = -5 x = x+1 = 2+1 = 3 Y = y-1 = 7-1 = 6 new points = (3, 6)
4	3	6	(3, 6)	= (3, 6)	-5	D = D + 4x + 6 = -5 + 4*3 + 6 = 13 x = x + 1 = 3+1 =4 y = 6 new points = (4, 6)
5	4	6	(4, 6)	= (4, 6)	13	D = D + 4(x – y) + 10 =13 + 4(4-6) + 10 = 15 x = x+1 = 4+1 = 5 Y = y-1 = 6-1 = 5 new points = (5, 5)
6	5	5	(5, 5)	= (5, 5)	Since x>=y i.e., 5=5, so stops here.	
These are all the points of octant -1.						

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(0, 7)	(5, 5)
(1, 7)	(6, 4)
(2, 7)	(6, 3)
(3, 6)	(7, 2)
(4, 6)	(7, 1)
(5, 5)	(7, 0)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants. The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(0, 7)	(0, 7)	(0, -7)	(0, -7)
(1, 7)	(-1, 7)	(-1, -7)	(1, -7)
(2, 7)	(-2, 7)	(-2, -7)	(2, -7)
(3, 6)	(-3, 6)	(-3, -6)	(3, -6)
(4, 6)	(-4, 6)	(-4, -6)	(4, -6)
(5, 5)	(-5, 5)	(-5, -5)	(5, -5)
(6, 4)	(-6, 4)	(-6, -4)	(6, -4)
(6, 3)	(-6, 3)	(-6, -3)	(6, -3)
(7, 2)	(-7, 2)	(-7, -2)	(7, -2)
(7, 1)	(-7, 1)	(-7, -1)	(7, -1)
(7, 0)	(-7, 0)	(-7, 0)	(7, 0)
These are all points of the Circle.			

Example-02: The radius of a circle is 8, and center point coordinates are (0, 0). Apply bresenham's circle drawing algorithm to plot all points of the circle.

Solution: Given radius r= 8 And initial center coordinates (h, k): (0, 0)

Initialize x = 0, and y = r=8 Calculate D= 3 – 2r = 3 – 2.8= -13

Step	X	Y	(x,y)	Plot (x+h, y+k) = (x + 0, y +0)	Previous Decision parameter(D)	D<0 : D = D + 4x + 6 [x = x+1, y=y] D>= 0 : D = D + 4(x – y) + 10 [x = x+1, y=y-1]
1	0	8	(0, 8)	= (0, 8)	-13	D = D + 4x + 6 = -13 + 4*0 + 6 = -7 x = x+1 = 0+1 = 1 y = 8 new points = (1, 8)
2	1	8	(1, 8)	= (1, 8)	-7	D = D + 4x + 6 = -7 + 4*1 + 6 = 3 x = x+1 = 1+1 = 2 y = 8 new points = (2, 8)
3	2	8	(2, 8)	= (2, 8)	3	D = D + 4(x-y) + 10 = 3 + 4*(2-8) + 10 = 13 – 24 = - 11 x = x + 1 = 2+1 = 3 y = y – 1 = 8-1 = 7 new points = (3, 7)
4	3	7	(3, 7)	= (3, 7)	-11	D = D + 4x + 6 = -11 + 4*3 + 6 = 7 x = x+1 =3+1=4 y = 7 new points = (4, 7)
5	4	7	(4, 7)	= (4, 7)	7	D = D + 4(x-y) + 10 = 7 + 4*(4-7) + 10 = 17 – 12 = 5 x = x + 1 = 4+1 = 5 y = y – 1 = 7-1 = 6 new points = (5, 6)

Step	X	Y	(x,y)	Plot (x+h, y+k) = (x + 0, y +0)	Previous Decision parameter(D)	D<0 : D = D + 4x + 6 [x = x+1, y=y] D>= 0 : D = D + 4(x – y) + 10 [x = x+1, y=y-1]
6	5	6	(5, 6)	= (5, 6)	5	D = D + 4(x-y) + 10 =5 + 4*(5-6)+10 = 15 – 4 = 11 x = x + 1 = 5+1=6 y = y – 1 = 6 -1=5 new points = (6, 5)
7	6	5	(6, 5)	= (6, 5)	Here, x >= y . So it stops further execution.	
These are all the points of octant -1.						

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(0, 8)	(5, 5)
(1, 8)	(6, 4)
(2, 8)	(7, 3)
(3, 7)	(8, 2)
(4, 6)	(8, 1)
(5, 5)	(8, 0)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants. The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(0, 8)	(0, 8)	(0, -8)	(0, -8)
(1, 8)	(-1, 8)	(-1, -8)	(1, -8)
(2, 8)	(-2, 8)	(-2, -8)	(2, -8)
(3, 7)	(-3, 7)	(-3, -7)	(3, -7)
(4, 6)	(-4, 6)	(-4, -6)	(4, -6)
(5, 5)	(-5, 5)	(-5, -5)	(5, -5)
(6, 4)	(-6, 4)	(-6, -4)	(6, -4)
(7, 3)	(-7, 3)	(-7, -3)	(7, -3)
(8, 2)	(-8, 2)	(-8, -2)	(8, -2)
(8, 1)	(-8, 1)	(-8, -1)	(8, -1)
(8, 0)	(-8, 0)	(-8, 0)	(8, 0)
These are all points of the Circle.			

Example-03: Plot all points of circle using Bresenham Algorithm. When radius of circle is 10 units. The circle has center (0, 0).

Solution: Given radius r= 10 And initial center coordinates (h, k): (0, 0)

Initialize x = 0, and y = r=10 Calculate D= 3 – 2r = 3 – 2.10= -17

Step	x	y	(x, y)	plot (x+h, y+k)	Check D<0 or D>=0
01	0	10	(0, 10)	(0, 10)	Initially we get, D=-17 i.e., D<0 So, D= D +4x+6= -17+4.0+6=-11 Hence, next x=x+1=0+1=1 and y=10 (same as previous)
02	1	10	(1, 10)	(1, 10)	Now D=-11 i.e., D<0 So, D= D +4x+6= -11+4.1+6=-1 Hence, next x=x+1=1+1=2 and y=10 (same as previous)
03	2	10	(2, 10)	(2, 10)	Now D=-1 i.e., D<0 So, D= D +4x+6= -1+4.2+6=13 Hence, next x=x+1=2+1=3 and y=10 (same as previous)
04	3	10	(3, 10)	(3, 10)	Now D=13 i.e., D>=0 So, D=D+4(x-y)+10=13+4(3-10)+10=-5 Hence, x=x+1=3+1=4 and y=y-1=10-1=9

05	4	9	(4, 9)	(4, 9)	Now D=-5 i.e., D<0 So, D= D +4x+6= -5+4.4+6=17 Hence, next x=x+1=4+1=5 and y=9 (same as previous)
06	5	9	(5, 9)	(5, 9)	Now D=17 i.e., D>=0 So, D=D+4(x-y)+10=17+4(5-9)+10=11 Hence, x=x+1=5+1=6 and y=y-1=9-1=8
07	6	8	(6, 8)	(6, 8)	Now D=11 i.e., D>=0 So, D=D+4(x-y)+10=11+4(6-8)+10=13 Hence, x=x+1=6+1=7 and y=y-1=8-1=7
08	7	7	(7, 7)	(7, 7)	Since x>=y i.e., 7=7, so stops here.
These are all the points of octant -1.					

Octant-1 Points	Octant-2 Points
(0, 10)	(7, 7)
(1, 10)	(8, 6)
(2, 10)	(9, 5)
(3, 10)	(9, 4)
(4, 9)	(10, 3)
(5, 9)	(10, 2)
(6, 8)	(10, 1)
(7, 7)	(10, 0)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(0, 10)	(0, 10)	(0, -10)	(0, -10)
(1, 10)	(-1, 10)	(-1, -10)	(1, -10)
(2, 10)	(-2, 10)	(-2, -10)	(2, -10)
(3, 10)	(-3, 10)	(-3, -10)	(3, -10)
(4, 9)	(-4, 9)	(-4, -9)	(4, -9)
(5, 9)	(-5, 9)	(-5, -9)	(5, -9)
(6, 8)	(-6, 8)	(-6, -8)	(6, -8)
(7, 7)	(-7, 7)	(-7, -7)	(7, -7)
(8, 6)	(-8, 6)	(-8, -6)	(8, -6)
(9, 5)	(-9, 5)	(-9, -5)	(9, -5)
(9, 4)	(-9, 4)	(-9, -4)	(9, -4)
(10, 3)	(-10, 3)	(-10, -3)	(10, -3)
(10, 2)	(-10, 2)	(-10, -2)	(10, -2)
(10, 1)	(-10, 1)	(-10, -1)	(10, -1)
(10, 0)	(-10, 0)	(-10, 0)	(10, 0)
These are all points of the Circle.			

Example-04: Plot all points of circle using Bresenham Algorithm. When radius of circle is 10 units. The circle has center (50, 50).

Solution: Given radius r= 10 And initial center coordinates (h, k): (50, 50)

Initialize x = 0, and y = r=10 Calculate D= 3 – 2r = 3 – 2.10= -17

Step	x	y	(x, y)	plot (x+h, y+k)	Check D<0 or D>=0
01	0	10	(0, 10)	(50, 60)	Initially we get, D=-17 i.e., D<0 So, D= D +4x+6= -17+4.0+6=-11 Hence, next x=x+1=0+1=1 and y=10 (same as previous)
02	1	10	(1, 10)	(51, 60)	Now D=-11 i.e., D<0 So, D= D +4x+6= -11+4.1+6=-1 Hence, next x=x+1=1+1=2 and y=10 (same as previous)
03	2	10	(2, 10)	(52, 60)	Now D=-1 i.e., D<0 So, D= D +4x+6= -1+4.2+6=13 Hence, next x=x+1=2+1=3 and y=10 (same as previous)
04	3	10	(3, 10)	(53, 60)	Now D=13 i.e., D>=0 So, D=D+4(x-y)+10=13+4(3-10)+10=-5 Hence, x=x+1=3+1=4 and y=y-1=10-1=9

05	4	9	(4, 9)	(54, 59)	Now D=-5 i.e., D<0 So, D= D +4x+6= -5+4.4+6=17 Hence, next x=x+1=4+1=5 and y=9 (same as previous)
06	5	9	(5, 9)	(55, 59)	Now D=17 i.e., D>=0 So, D=D+4(x-y)+10=17+4(5-9)+10=11 Hence, x=x+1=5+1=6 and y=y-1=9-1=8
07	6	8	(6, 8)	(56, 58)	Now D=11 i.e., D>=0 So, D=D+4(x-y)+10=11+4(6-8)+10=13 Hence, x=x+1=6+1=7 and y=y-1=8-1=7
08	7	7	(7, 7)	(57, 57)	Since x>=y i.e., 7=7, so stops here.
These are all the points of octant -1.					

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(50, 60)	(57, 57)
(51, 60)	(58, 56)
(52, 60)	(59, 55)
(53, 60)	(59, 54)
(54, 59)	(60, 53)
(55, 59)	(60, 52)
(56, 58)	(60, 51)
(57, 57)	(60, 50)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(50, 60)	(-50, 60)	(-50, -60)	(50, -60)
(51, 60)	(-51, 60)	(-51, -60)	(51, -60)
(52, 60)	(-52, 60)	(-52, -60)	(52, -60)
(53, 60)	(-53, 60)	(-53, -60)	(53, -60)
(54, 59)	(-54, 59)	(-54, -59)	(54, -59)
(55, 59)	(-55, 59)	(-55, -59)	(55, -59)
(56, 58)	(-56, 58)	(-56, -58)	(56, -58)
(57, 57)	(-57, 57)	(-57, -57)	(57, -57)
(58, 56)	(-58, 56)	(-58, -56)	(58, -56)
(59, 55)	(-59, 55)	(-59, -55)	(59, -55)
(59, 54)	(-59, 54)	(-59, -54)	(59, -54)
(60, 53)	(-60, 53)	(-60, -53)	(60, -53)
(60, 52)	(-60, 52)	(-60, -52)	(60, -52)
(60, 51)	(-60, 51)	(-60, -51)	(60, -51)
(60, 50)	(-60, 50)	(-60, -50)	(60, -50)
These are all points of the Circle.			

Example-05: Given the center point coordinates (10, 10) and radius as 10, generate all the points to form a circle using Bresenham Algorithm.

Solution: Given radius r= 10

And initial center coordinates (h, k): (10, 10)

Initialize x = 0, and y = r=10

Calculate D= 3 – 2r = 3 – 2.10= -17

Step	x	y	(x, y)	plot (x+h, y+k)	Check D<0 or D>=0
01	0	10	(0, 10)	(10, 20)	Initially we get, D=-17 i.e., D<0 So, D= D +4x+6= -17+4.0+6=-11 Hence, next x=x+1=0+1=1 and y=10 (same as previous)
02	1	10	(1, 10)	(11, 20)	Now D=-11 i.e., D<0 So, D= D +4x+6= -11+4.1+6=-1 Hence, next x=x+1=1+1=2 and y=10 (same as previous)
03	2	10	(2, 10)	(12, 20)	Now D=-1 i.e., D<0 So, D= D +4x+6= -1+4.2+6=13 Hence, next x=x+1=2+1=3 and y=10 (same as previous)

04	3	10	(3, 10)	(13, 20)	Now D=13 i.e., D>=0 So, D=D+4(x-y)+10=13+4(3-10)+10=-5 Hence, x=x+1=3+1=4 and y=y-1=10-1=9
05	4	9	(4, 9)	(14, 19)	Now D=-5 i.e., D<0 So, D= D +4x+6= -5+4.4+6=17 Hence, next x=x+1=4+1=5 and y=9 (same as previous)
06	5	9	(5, 9)	(15, 19)	Now D=17 i.e., D>=0 So, D=D+4(x-y)+10=17+4(5-9)+10=11 Hence, x=x+1=5+1=6 and y=y-1=9-1=8
07	6	8	(6, 8)	(16, 18)	Now D=11 i.e., D>=0 So, D=D+4(x-y)+10=11+4(6-8)+10=13 Hence, x=x+1=6+1=7 and y=y-1=8-1=7
08	7	7	(7, 7)	(17, 17)	Since x>=y i.e., 7=7, so stops here.
These are all the points of octant -1.					

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(10, 20)	(17, 17)
(11, 20)	(18, 16)
(12, 20)	(19, 15)
(13, 20)	(19, 14)
(14, 19)	(20, 13)
(15, 19)	(20, 12)
(16, 18)	(20, 11)
(17, 17)	(20, 10)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(10, 20)	(-10, 20)	(-10, -20)	(10, -20)
(11, 20)	(-11, 20)	(-11, -20)	(11, -20)
(12, 20)	(-12, 20)	(-12, -20)	(12, -20)
(13, 20)	(-13, 20)	(-13, -20)	(13, -20)
(14, 19)	(-14, 19)	(-14, -19)	(14, -19)
(15, 19)	(-15, 19)	(-15, -19)	(15, -19)
(16, 18)	(-16, 18)	(-16, -18)	(16, -18)
(17, 17)	(-17, 17)	(-17, -17)	(17, -17)
(18, 16)	(-18, 16)	(-18, -16)	(18, -16)
(19, 15)	(-19, 15)	(-19, -15)	(19, -15)
(19, 14)	(-19, 14)	(-19, -14)	(19, -14)
(20, 13)	(-20, 13)	(-20, -13)	(20, -13)
(20, 12)	(-20, 12)	(-20, -12)	(20, -12)
(20, 11)	(-20, 11)	(-20, -11)	(20, -11)
(20, 10)	(-20, 10)	(-20, -10)	(20, -10)
These are all points of the Circle.			

Assignment (Solve these problems using Bresenham's Circle Algorithm)

Problem-01: The radius of a circle is 7, and center point coordinates are (0, 0). Apply bresenham's circle drawing algorithm to plot all points of the circle.

Problem-02: The radius of a circle is 8, and center point coordinates are (0, 0). Apply bresenham's circle drawing algorithm to plot all points of the circle.

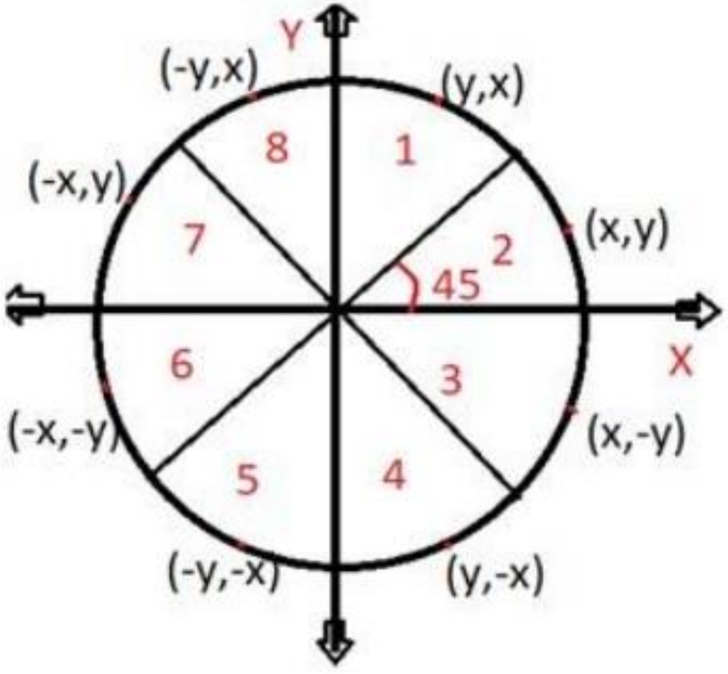
Problem-03: Plot all points of circle using Bresenham Algorithm. When radius of circle is 10 units. The circle has center (0, 0).

Problem-04: Plot all points of circle using Bresenham Algorithm. When radius of circle is 10 units. The circle has center (50, 50).

Problem-05: Given the center point coordinates (10, 10) and radius as 10, generate all the points to form a circle using Bresenham Algorithm.

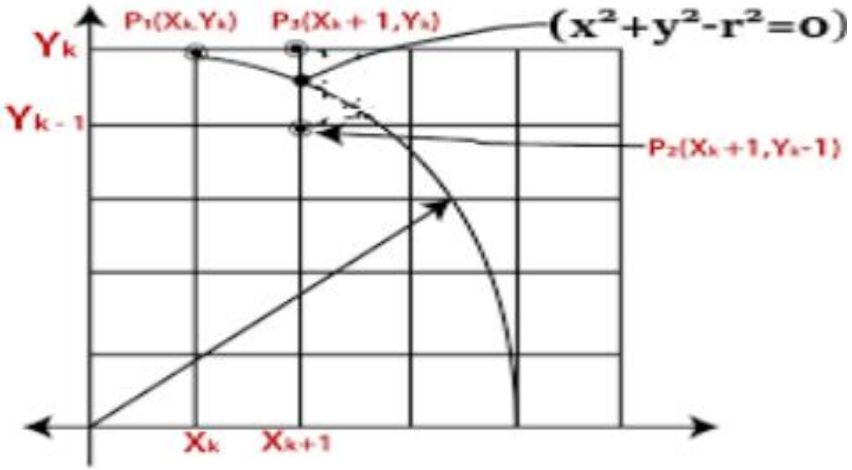
Midpoint Circle Drawing Algorithm

We present another incremental circle algorithm that is very similar Bresenham's circle drawing algorithm. The midpoint circle drawing algorithm helps us to calculate the complete perimeter points of a circle for the first octant so that the points of the other octant can be taken easily as they are mirror points. That is the midpoint circle drawing algorithm also uses the eight-way symmetry of the circle to generate it. It plots 1/8 part of the circle, i.e., from 90° to 45°, as shown in the figure below.



As circle is drawn from 90° to 45°, the x moves in the positive direction and y moves in the negative direction. To draw a 1/8 part of a circle we take unit steps in the positive x direction and make use of decision parameter to determine which of the two possible y positions is closer to the circle path at each step.

Working of Bresenham's Circle Drawing Algorithm



Let us consider a point **($x_k + 1$, y)** on true circle boundary having a radius **'r'**. Assume:

$P_1 = (\mathbf{x_k}, \mathbf{y_k})$ is the current pixel.

$P_3 = (\mathbf{x_k + 1}, \mathbf{y_k})$ is the pixel top.

$P_2 = (\mathbf{x_k + 1}, \mathbf{y_k - 1})$ is the pixel bottom.

In this algorithm ***decision parameter*** is based on a circle equation. As we know the equation of the circle is – $\mathbf{f_c (x, y) = x^2 + y^2 = r^2}$ when the center is (0, 0).

Let us assume we have a point **p (x, y)** on the boundary of the circle and with **r** radius satisfying the equation $\mathbf{f_c (x, y) = 0}$

$$if \ \text{fc}\,(x, y) = \begin{cases} < 0 & \text{The point } (x, y) \text{ is inside the circle boundary.} \\ = 0 & \text{The point } (x, y) \text{ is on the circle boundary.} \\ > & \text{The point } (x, y) \text{ is outside the circle boundary.} \end{cases}$$

The circle function tests given above are performed for the midpoints between pixels near the circle path at each sampling step. Thus, the circle function is the decision parameter in the midpoint algorithm, and we can set up incremental calculations for this function as we did in the line algorithm.

Figure given above shows the midpoint between the two candidate pixels at sampling position x_k+1. Assuming we have just plotted the pixel at $(\mathbf{x_k}, \mathbf{y_k})$, we next need to determine whether the pixel at position $(\mathbf{x_k + 1, y_k})$, we next need to determine whether the pixel at position $(\mathbf{x_k + 1, y_k})$ or the one at position $(\mathbf{x_k + 1, y_k - 1})$ is closer to the circle.

Our decision parameter is the circle function evaluated at the midpoint between these two pixels (in this ½ is taken because it is a midpoint value through which it is easy to calculate value of $\mathbf{y_k}$ and $\mathbf{y_k - 1}$):

$$P_k = f_c(x_k + 1, y_k - \tfrac{1}{2})$$

$$P_k = (x_k + 1)^2 + (y_k - \tfrac{1}{2})^2 - r^2 \quad \text{..............................} \quad (1)$$

If $P_k < 0$, this midpoint is inside the circle and the pixel on scan line y_k is closer to the circle boundary. Otherwise, the mid position is outside or on the circle boundary, and we select the pixel on scan-line $y_k - 1$.

Successive decision parameters are obtained using incremental calculations. We obtain a recursive expression for the next decision parameter by evaluating the circle function at sampling position $x_k + 2 = x_{k+1} + 1 = x_k + 1 + 1 = x_k + 2$:

$P_{k+1} = f_c(x_{k+1} + 1, y_{k+1} - ½)$

$P_{k+1} = [(x_k + 1) + 1]^2 + (y_{k+1} - ½)^2 - r^2$.............................. (2)

Subtracting (1) from (2), we get

$P_{k+1} - P_k = [(x_k + 1) + 1]^2 + (y_{k+1} - ½)^2 - r^2 - (x_k + 1)^2 - (y_k - ½)^2 + r^2$

or $P_{k+1} = P_k + (x_k + 1)^2 + 2(x_k + 1) + 1 + (y_{k+1})^2 - y_{k+1} + (½)^2 - (x_k + 1)^2 - y_k^2 + y_k - (½)^2$

or $P_{k+1} = P_k + 2(x_k + 1) + (y_{k+1}^2 - y_k^2) - (y_{k+1} - y_k) + 1$

Where $\mathbf{y_{k+1}}$ is either $\mathbf{y_k}$ or $\mathbf{y_k - 1}$, depending on the sign of $\mathbf{P_k}$.

Therefore, if $\mathbf{P_k} < 0$ or negative then $\mathbf{y_{k+1}}$ will be $\mathbf{y_k}$ and the formula to calculate $\mathbf{P_{k+1}}$ will be:

$$P_{k+1} = P_k + 2(x_k + 1) + (y_k^2 - y_k^2) - (y_k - y_k) + 1$$

$$P_{k+1} = P_k + 2(x_k + 1) + 1$$

$$P_{k+1} = P_k + 2x_k + 3$$

Otherwise, if $\mathbf{P_k >= 0}$ or (zero or positive) then $\mathbf{y_{k+1}}$ will be $\mathbf{y_k - 1}$ and the formula to calculate $\mathbf{P_{k+1}}$ will be:

$$P_{k+1} = P_k + 2(x_k + 1) + [(y_k - 1)^2 - y_k^2)] - (y_k - 1 - y_k) + 1$$

$$P_{k+1} = P_k + 2(x_k + 1) + (y_k^2 - 2y_k + 1 - y_k^2) + 1 + 1$$

$$P_{k+1} = P_k + 2x_k + 2 - 2y_k + 1 + 2$$

$$P_{k+1} = P_k + 2x_k + 2 - 2y_k + 3$$

$$P_{k+1} = P_k + 2(x_k - y_k) + 5$$

Now a similar case that we observe in line algorithm is that how would be starting $\mathbf{P_k}$ be evaluated. For this at the start pixel position will be $\mathbf{(x_o, y_o) = (0, r)}$. Therefore, putting this value in equation (1), we get

$$P_k = (x_k + 1)^2 + (y_k - 1/2)^2 - r^2$$

$$P_o = (x_o + 1)^2 + (y_o - 1/2)^2 - r^2$$

$$P_o = (0 + 1)^2 + (r - 1/2)^2 - r^2$$

$$P_o = 1 + r^2 - r + 1/4 - r^2$$

$$P_o = 5/4 - r$$

If radius r is specified as an integer, we can simply round $\mathbf{P_o}$ to:

$$P_o = 1 - r$$

Step 1: Start.

Step 2: Declare x, y, r, h, k and p as variables, where (h, k) are coordinates of the center.

Step 3: Calculate the decision parameter as follows: $p = 1 - r$;

Step 4: Initialize $x = 0$, $y = r$;

Step 5: If $x >= y$

Stop.

Step 6: Plot eight points by using concepts of eight-way symmetry. The center is at (h, k). Current active pixel is (x, y).

putpixel (x+h, y+k)

putpixel (y+h, x+k)

putpixel (-y+h, x+k)

putpixel (-x+h, y+k)

putpixel (-x+h, -y+k)

putpixel (-y+h, -x+k)

putpixel (y+h, -x+k)

putpixel (x+h, -y-k)

Step 7: if $p < 0$

then

$p = p + 2x + 3$

$x = x + 1$

$y = y$

else

$p = p + 2(x - y) + 5$

$x = x + 1$

$y = y - 1$

Step 8: Go to step 5.

Step 9: Stop algorithm.

Advantages of Midpoint Circle Drawing Algorithm

- It is a powerful and efficient algorithm.
- The midpoint circle drawing algorithm is easy to implement.
- It is also an algorithm based on a simple circle equation ($x2 + y\ 2 = r\ 2$).
- This algorithm helps to create curves on a raster display.

Disadvantages of Midpoint Circle Drawing Algorithm

- It is a time-consuming algorithm.
- Sometimes the points of the circle are not accurate.

Example-01: The radius of a circle is 7, and center point coordinates are (0, 0). Apply Midpoint circle drawing algorithm to plot all points of the circle.

Solution:

Given radius r= 7
And initial center coordinates
(h, k)= (0, 0)
Initialize x = 0, and y = r=7

Calculate P= 1 – r = 1 – 7= -6

Step	x	y	(x, y)	Plot(x+h, y+k)	Check P<0 or P>=0
01	0	7	(0, 7)	(0, 7)	Initially we get, P=-6 i.e., P<0 So, P= P +2x+3= -6+2.0+3=-3 Hence, next x=x+1=0+1=1 and y=7 (same as previous)
02	1	7	(1, 7)	(1, 7)	Now P=-3 i.e., P<0 So, P= P +2x+3= -3+2.1+3=2 Hence, next x=x+1=1+1=2 and y=7 (same as previous)
03	2	7	(2, 7)	(2, 7)	Now P=2 i.e., P>=0 So, P=P+2(x-y)+5=2+2(2-7)+5=-3 Hence, next x=x+1=2+1=3 and y=y-1=7-1=6
04	3	6	(3, 6)	(3, 6)	Now P=-3 i.e., P<0 So, P= P+2x+3= -3+2.3+3=6 Hence, next x=x+1=3+1=4 and y=6 (same as previous)
05	4	6	(4, 6)	(4, 6)	Now P=6 i.e., P>=0 So, P=P+2(x-y)+5=6+2(4-6)+5=7 Hence, next x=x+1=4+1=5 and y=y-1=6-1=5
06	5	5	(5, 5)	(5, 5)	Since x>=y i.e., 5=5, so stops here.
These are all the points of octant -1.					

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(0, 7)	(5, 5)
(1, 7)	(6, 4)
(2, 7)	(6, 3)
(3, 6)	(7, 2)
(4, 6)	(7, 1)
(5, 5)	(7, 0)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(0, 7)	(0, 7)	(0, -7)	(0, -7)
(1, 7)	(-1, 7)	(-1, -7)	(1, -7)
(2, 7)	(-2, 7)	(-2, -7)	(2, -7)
(3, 6)	(-3, 6)	(-3, -6)	(3, -6)
(4, 6)	(-4, 6)	(-4, -6)	(4, -6)
(5, 5)	(-5, 5)	(-5, -5)	(5, -5)
(6, 4)	(-6, 4)	(-6, -4)	(6, -4)
(6, 3)	(-6, 3)	(-6, -3)	(6, -3)
(7, 2)	(-7, 2)	(-7, -2)	(7, -2)
(7, 1)	(-7, 1)	(-7, -1)	(7, -1)
(7, 0)	(-7, 0)	(-7, 0)	(7, 0)
These are all points of the Circle.			

Example-02: The radius of a circle is 8, and center point coordinates are (0, 0). Apply Midpoint circle drawing algorithm to plot all points of the circle.

Solution:

Given radius r= 8

initial center coordinates (h, k): (0, 0)

Initialize x = 0, and y = r=8

Calculate P= 1 – r = 1 – 8= -7

Step	x	y	(x, y)	Plot (x+h, y+k)	Check P<0 or P>=0
01	0	8	(0, 8)	(0, 8)	Initially we get, P=-7 i.e., P<0 So, P= P +2x+3= -7+2.0+3=-4 Hence, next x=x+1=0+1=1 and y=8 (same as previous)
02	1	8	(1, 8)	(1, 8)	Now P=-4 i.e., P<0 So, P= P +2x+3= -4+2.1+3=1 Hence, next x=x+1=1+1=2 and y=8 (same as previous)
03	2	8	(2, 8)	(2, 8)	Now P=1 i.e., P>=0 So, P=P+2(x-y)+5=1+2(2-8)+5=-6 Hence, next x=x+1=2+1=3 and y=y-1=8-1=7
04	3	7	(3, 7)	(3, 7)	Now P=-6 i.e., P<0 So, P= P +2x+3= -6+2.3+3=3 Hence, next x=x+1=3+1=4 and y=7 (same as previous)
05	4	6	(4, 6)	(4, 6)	Now P=3 i.e., P>=0 So, P=P+2(x-y)+5=3+2(4-6)+5=4 Hence, next x=x+1=4+1=5 and y=y-1=6-1=5
06	5	5	(5, 5)	(5, 5)	Since x>=y i.e., 5=5, so stops here.
These are all the points of octant -1.					

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(0, 8)	(5, 5)
(1, 8)	(6, 4)
(2, 8)	(7, 3)
(3, 7)	(8, 2)
(4, 6)	(8, 1)
(5, 5)	(8, 0)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(0, 8)	(0, 8)	(0, -8)	(0, -8)
(1, 8)	(-1, 8)	(-1, -8)	(1, -8)
(2, 8)	(-2, 8)	(-2, -8)	(2, -8)
(3, 7)	(-3, 7)	(-3, -7)	(3, -7)
(4, 6)	(-4, 6)	(-4, -6)	(4, -6)
(5, 5)	(-5, 5)	(-5, -5)	(5, -5)
(6, 4)	(-6, 4)	(-6, -4)	(6, -4)
(7, 3)	(-7, 3)	(-7, -3)	(7, -3)
(8, 2)	(-8, 2)	(-8, -2)	(8, -2)
(8, 1)	(-8, 1)	(-8, -1)	(8, -1)
(8, 0)	(-8, 0)	(-8, 0)	(8, 0)
These are all points of the Circle.			

Example-03: Plot all points of circle using Midpoint Algorithm. When radius of circle is 10 units. The circle has center (0, 0).

Solution:

Given radius r= 10
initial center coordinates (h, k): (0, 0)

Initialize x = 0, and y = r=10

Calculate P= 1 – r = 1 – 10= -9

Step	x	y	(x, y)	plot (x+h, y+k)	Check P<0 or P>=0
01	0	10	(0, 10)	(0, 10)	Initially we get, P=-9 i.e., P<0 So, P= P +2x+3= -9+2.0+3=-6 Hence, next x=x+1=0+1=1 and y=10 (same as previous)
02	1	10	(1, 10)	(1, 10)	Now P=-6 i.e., P<0 So, P= P +2x+3= -6+2.1+3=-1 Hence, next x=x+1=1+1=2 and y=10 (same as previous)
03	2	10	(2, 10)	(2, 10)	Now P=-1 i.e., P<0 So, P= P +2x+3= -1+2.2+3=6 Hence, next x=x+1=2+1=3 and y=10 (same as previous)
04	3	10	(3, 10)	(3, 10)	Now P=6 i.e., P>=0 So, P=P+2(x-y)+5=6+2(3-10)+5=-3 Hence, x=x+1=3+1=4 and y=y-1=10-1=9
05	4	9	(4, 9)	(4, 9)	Now P=-3 i.e., P<0 So, P= P +2x+3= -3+2.4+3=8 Hence, next x=x+1=4+1=5 and y=9 (same as previous)
06	5	9	(5, 9)	(5, 9)	Now P=8 i.e., P>=0 So, P=P+2(x-y)+5=8+2(5-9)+5=5 Hence, x=x+1=5+1=6 and y=y-1=9-1=8
07	6	8	(6, 8)	(6, 8)	Now P=5 i.e., P>=0 So, P=P+2(x-y)+5=5+2(6-8)+5=6 Hence, x=x+1=6+1=7 and y=y-1=8-1=7
08	7	7	(7, 7)	(7, 7)	Since x>=y i.e., 7=7, so stops here.
These are all the points of octant -1.					

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	**Octant-2 Points**
(0, 10)	(7, 7)
(1, 10)	(8, 6)
(2, 10)	(9, 5)
(3, 10)	(9, 4)
(4, 9)	(10, 3)
(5, 9)	(10, 2)
(6, 8)	(10, 1)
(7, 7)	(10, 0)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(0, 10)	(0, 10)	(0, -10)	(0, -10)
(1, 10)	(-1, 10)	(-1, -10)	(1, -10)
(2, 10)	(-2, 10)	(-2, -10)	(2, -10)
(3, 10)	(-3, 10)	(-3, -10)	(3, -10)
(4, 9)	(-4, 9)	(-4, -9)	(4, -9)
(5, 9)	(-5, 9)	(-5, -9)	(5, -9)
(6, 8)	(-6, 8)	(-6, -8)	(6, -8)
(7, 7)	(-7, 7)	(-7, -7)	(7, -7)
(8, 6)	(-8, 6)	(-8, -6)	(8, -6)
(9, 5)	(-9, 5)	(-9, -5)	(9, -5)
(9, 4)	(-9, 4)	(-9, -4)	(9, -4)
(10, 3)	(-10, 3)	(-10, -3)	(10, -3)
(10, 2)	(-10, 2)	(-10, -2)	(10, -2)
(10, 1)	(-10, 1)	(-10, -1)	(10, -1)
(10, 0)	(-10, 0)	(-10, 0)	(10, 0)
These are all points of the Circle.			

Example-04: Plot all points of circle using Midpoint Algorithm. When radius of circle is 10 units. The circle has center (50, 50).

Solution:

Given radius r= 10

And initial center coordinates (h, k): (50, 50)

Initialize x = 0, and y = r=10

Calculate P= 1 – r = 1 – 10= -9

Step	x	y	(x, y)	plot (x+h, y+k)	Check $P<0$ or $P>=0$
01	0	10	(0, 10)	(50, 60)	Initially we get, P=-9 i.e., $P<0$ So, P= P +2x+3= -9+2.0+3=-6 Hence, next x=x+1=0+1=1 and y=10 (same as previous)
02	1	10	(1, 10)	(51, 60)	Now P=-6 i.e., $P<0$ So, P= P +2x+3= -6+2.1+3=-1 Hence, next x=x+1=1+1=2 and y=10 (same as previous)
03	2	10	(2, 10)	(52, 60)	Now P=-1 i.e., $P<0$ So, P= P +2x+3= -1+2.2+3=6 Hence, next x=x+1=2+1=3 and y=10 (same as previous)
04	3	10	(3, 10)	(53, 60)	Now P=6 i.e., $P>=0$ So, P=P+2(x-y)+5=6+2(3-10)+5=-3 Hence, x=x+1=3+1=4 and y=y-1=10-1=9
05	4	9	(4, 9)	(54, 59)	Now P=-3 i.e., $P<0$ So, P= P +2x+3= -3+2.4+3=8 Hence, next x=x+1=4+1=5 and y=9 (same as previous)
06	5	9	(5, 9)	(55, 59)	Now P=8 i.e., $P>=0$ So, P=P+2(x-y)+5=8+2(5-9)+5=5 Hence, x=x+1=5+1=6 and y=y-1=9-1=8
07	6	8	(6, 8)	(56, 58)	Now P=5 i.e., $P>=0$ So, P=P+2(x-y)+5=5+2(6-8)+5=6 Hence, x=x+1=6+1=7 and y=y-1=8-1=7
08	7	7	(7, 7)	(57, 57)	Since $x>=y$ i.e., 7=7, so stops here.
These are all the points of octant -1.					

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(50, 60)	(57, 57)
(51, 60)	(58, 56)
(52, 60)	(59, 55)
(53, 60)	(59, 54)
(54, 59)	(60, 53)
(55, 59)	(60, 52)
(56, 58)	(60, 51)
(57, 57)	(60, 50)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(50, 60)	(-50, 60)	(-50, -60)	(50, -60)
(51, 60)	(-51, 60)	(-51, -60)	(51, -60)
(52, 60)	(-52, 60)	(-52, -60)	(52, -60)
(53, 60)	(-53, 60)	(-53, -60)	(53, -60)
(54, 59)	(-54, 59)	(-54, -59)	(54, -59)
(55, 59)	(-55, 59)	(-55, -59)	(55, -59)
(56, 58)	(-56, 58)	(-56, -58)	(56, -58)
(57, 57)	(-57, 57)	(-57, -57)	(57, -57)
(58, 56)	(-58, 56)	(-58, -56)	(58, -56)
(59, 55)	(-59, 55)	(-59, -55)	(59, -55)
(59, 54)	(-59, 54)	(-59, -54)	(59, -54)
(60, 53)	(-60, 53)	(-60, -53)	(60, -53)
(60, 52)	(-60, 52)	(-60, -52)	(60, -52)
(60, 51)	(-60, 51)	(-60, -51)	(60, -51)
(60, 50)	(-60, 50)	(-60, -50)	(60, -50)
These are all points of the Circle.			

Example-05: Given the center point coordinates (10, 10) and radius as 10, generate all the points to form a circle using Bresenham Algorithm.

Solution:
Given radius r= 10

initial center coordinates
(h, k): (10, 10)

Initialize x = 0, and y = r=10

Calculate P= 1 – r = 1 – 10= -9

Step	x	y	(x, y)	plot (x+h, y+k)	Check P<0 or P>=0
01	0	10	(0, 10)	(10, 20)	Initially we get, P=-9 i.e., P<0 So, P= P +2x+3= -9+2.0+3=-6 Hence, next x=x+1=0+1=1 and y=10 (same as previous)
02	1	10	(1, 10)	(11, 20)	Now P=-6 i.e., P<0 So, P= P +2x+3= -6+2.1+3=-1 Hence, next x=x+1=1+1=2 and y=10 (same as previous)
03	2	10	(2, 10)	(12, 20)	Now P=-1 i.e., P<0 So, P= P +2x+3= -1+2.2+3=6 Hence, next x=x+1=2+1=3 and y=10 (same as previous)
04	3	10	(3, 10)	(13, 20)	Now P=6 i.e., P>=0 So, P=P+2(x-y)+5=6+2(3-10)+5=-3 Hence, x=x+1=3+1=4 and y=y-1=10-1=9
05	4	9	(4, 9)	(14, 19)	Now P=-3 i.e., P<0 So, P= P +2x+3= -3+2.4+3=8 Hence, next x=x+1=4+1=5 and y=9 (same as previous)
06	5	9	(5, 9)	(15, 19)	Now P=8 i.e., P>=0 So, P=P+2(x-y)+5=8+2(5-9)+5=5 Hence, x=x+1=5+1=6 and y=y-1=9-1=8
07	6	8	(6, 8)	(16, 18)	Now P=5 i.e., P>=0 So, P=P+2(x-y)+5=5+2(6-8)+5=6 Hence, x=x+1=6+1=7 and y=y-1=8-1=7
08	7	7	(7, 7)	(17, 17)	Since x>=y i.e., 7=7, so stops here.
These are all the points of octant -1.					

Now, the points of octant-2 are obtained using the mirror effect by swapping x and y coordinates.

Octant-1 Points	Octant-2 Points
(10, 20)	(17, 17)
(11, 20)	(18, 16)
(12, 20)	(19, 15)
(13, 20)	(19, 14)
(14, 19)	(20, 13)
(15, 19)	(20, 12)
(16, 18)	(20, 11)
(17, 17)	(20, 10)
These are all points for Quadrant-1.	

Now, the points for rest of the part are generated by following the signs of other quadrants.

The other points can also be generated by calculating each octant separately.

Here, all the points have been generated with respect to quadrant-1.

Quadrant-1 (x,y)	Quadrant-2 (-x,y)	Quadrant-3 (-x,-y)	Quadrant-4 (x,-y)
(10, 20)	(-10, 20)	(-10, -20)	(10, -20)
(11, 20)	(-11, 20)	(-11, -20)	(11, -20)
(12, 20)	(-12, 20)	(-12, -20)	(12, -20)
(13, 20)	(-13, 20)	(-13, -20)	(13, -20)
(14, 19)	(-14, 19)	(-14, -19)	(14, -19)
(15, 19)	(-15, 19)	(-15, -19)	(15, -19)
(16, 18)	(-16, 18)	(-16, -18)	(16, -18)
(17, 17)	(-17, 17)	(-17, -17)	(17, -17)
(18, 16)	(-18, 16)	(-18, -16)	(18, -16)
(19, 15)	(-19, 15)	(-19, -15)	(19, -15)
(19, 14)	(-19, 14)	(-19, -14)	(19, -14)
(20, 13)	(-20, 13)	(-20, -13)	(20, -13)
(20, 12)	(-20, 12)	(-20, -12)	(20, -12)
(20, 11)	(-20, 11)	(-20, -11)	(20, -11)
(20, 10)	(-20, 10)	(-20, -10)	(20, -10)
These are all points of the Circle.			

Assignment (Solve these problems using Midpoint Circle Algorithm)

Problem-01: The radius of a circle is 7, and center point coordinates are (0, 0). Apply Midpoint circle drawing algorithm to plot all points of the circle.

Problem-02: The radius of a circle is 8, and center point coordinates are (0, 0). Apply Midpoint circle drawing algorithm to plot all points of the circle.

Problem-03: Plot all points of circle using Midpoint Algorithm. When radius of circle is 10 units. The circle has center (0, 0).

Problem-04: Plot all points of circle using Midpoint Algorithm. When radius of circle is 10 units. The circle has center (50, 50).

Problem-05: Given the center point coordinates (10, 10) and radius as 10, generate all the points to form a circle using Midpoint Algorithm.