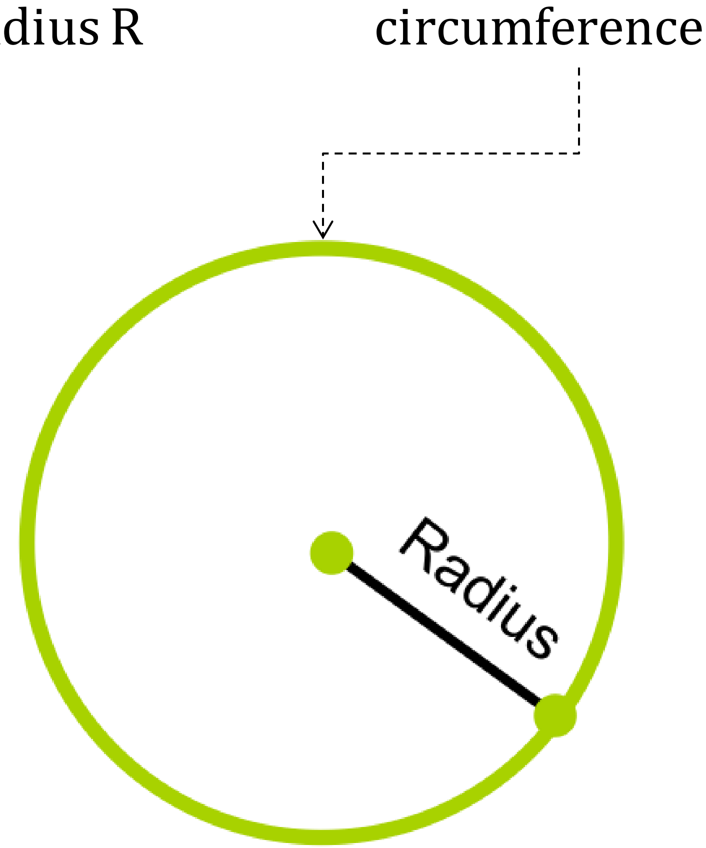
Bresenham's Circle Drawing Algorithm

- Mohammad Imrul Jubair

The Scenario

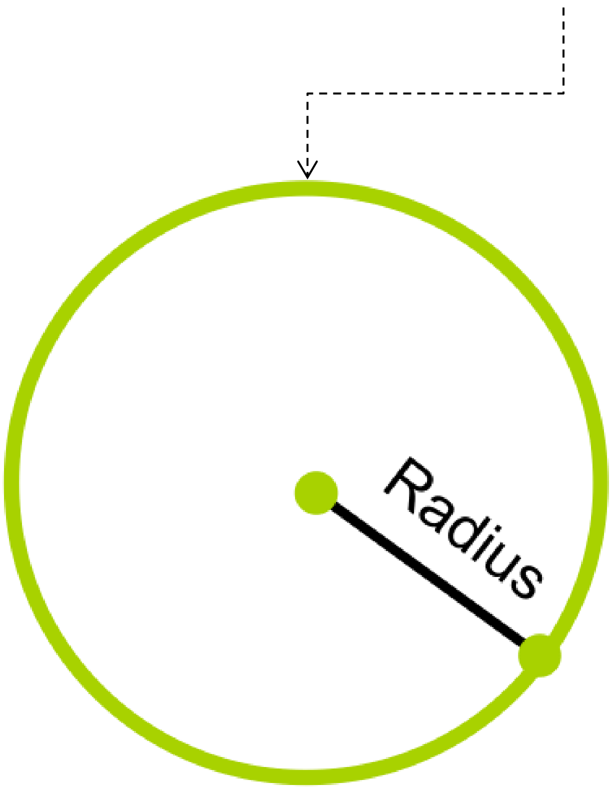
Given,
Radius R

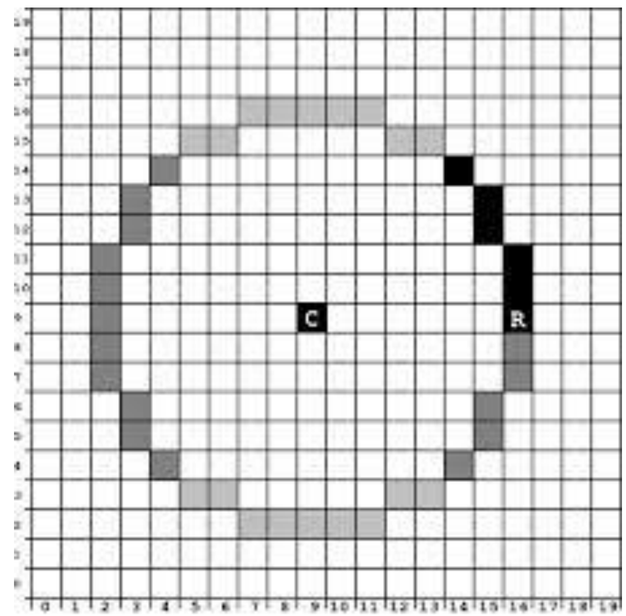


The Scenario

Given,
Radius R

circumference

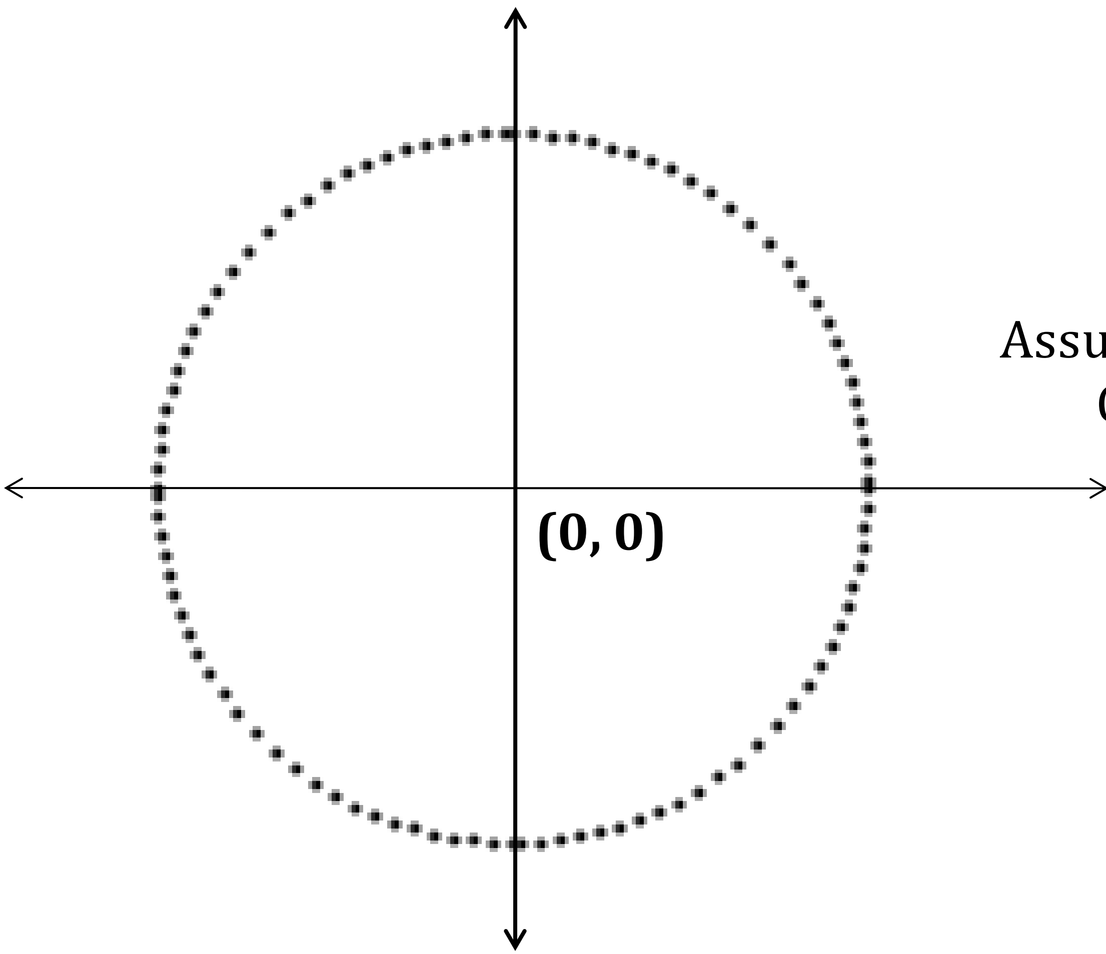




We have to develop an algorithm that generates this circumference

Assumptions

Given,
Radius R

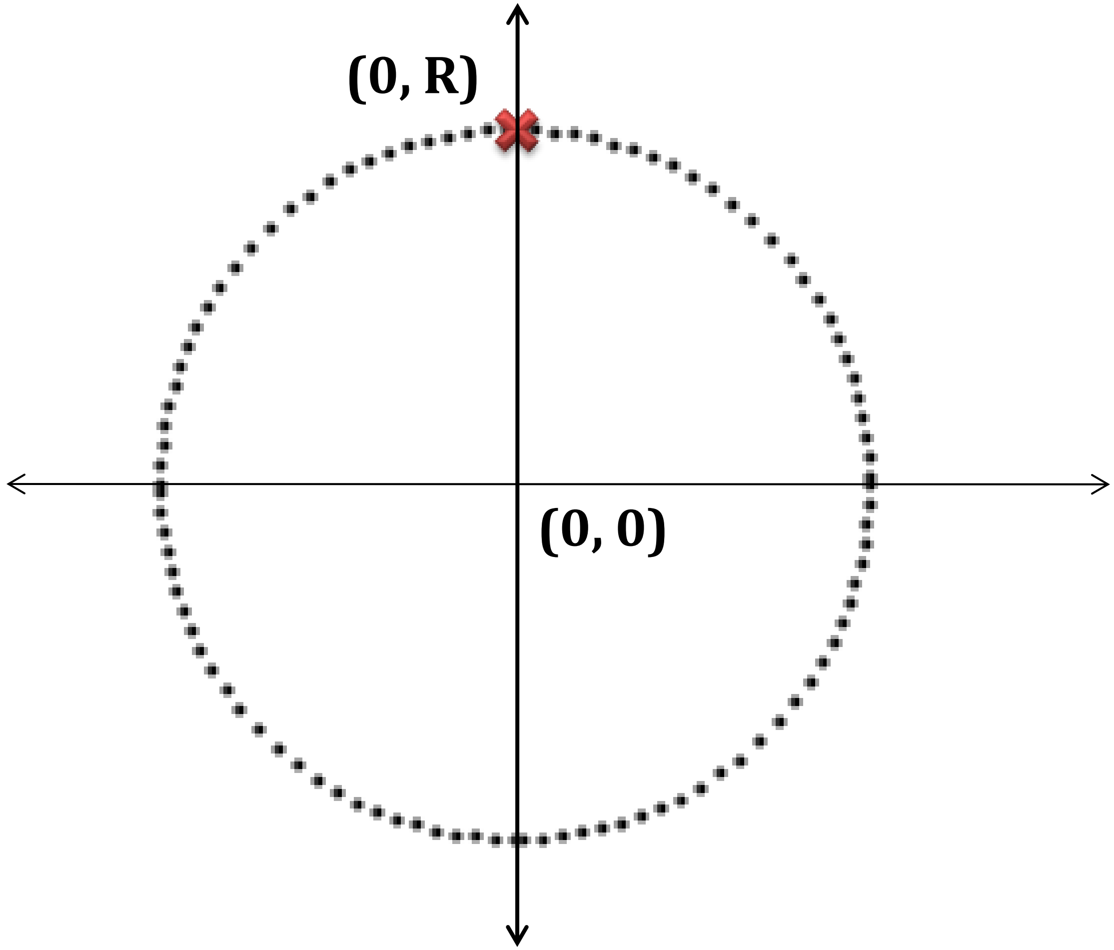


Assume,
Center is at (0,0)

Assumptions

The first pixel of the circumference is plotted on (0, R)

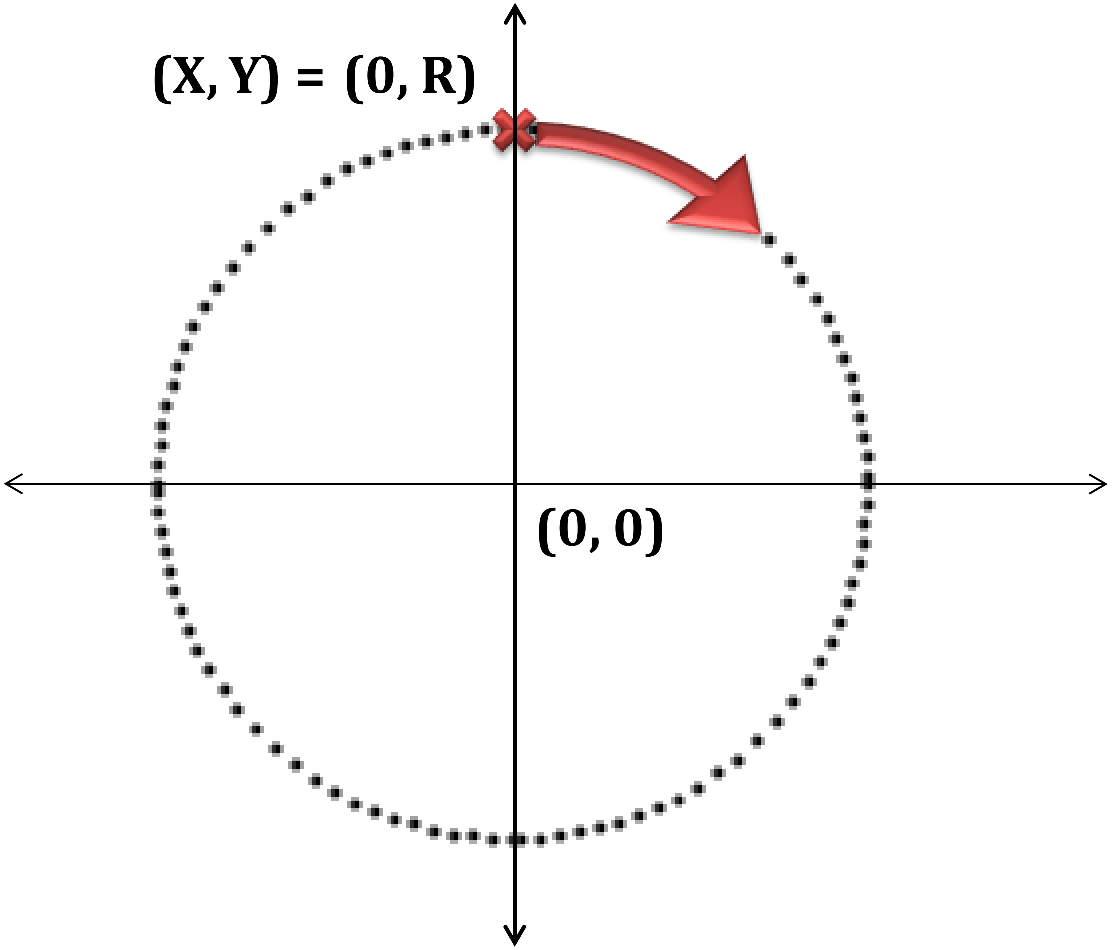
Given,
Radius R



Assumptions

The first pixel of the circumference is plotted on (0, R)

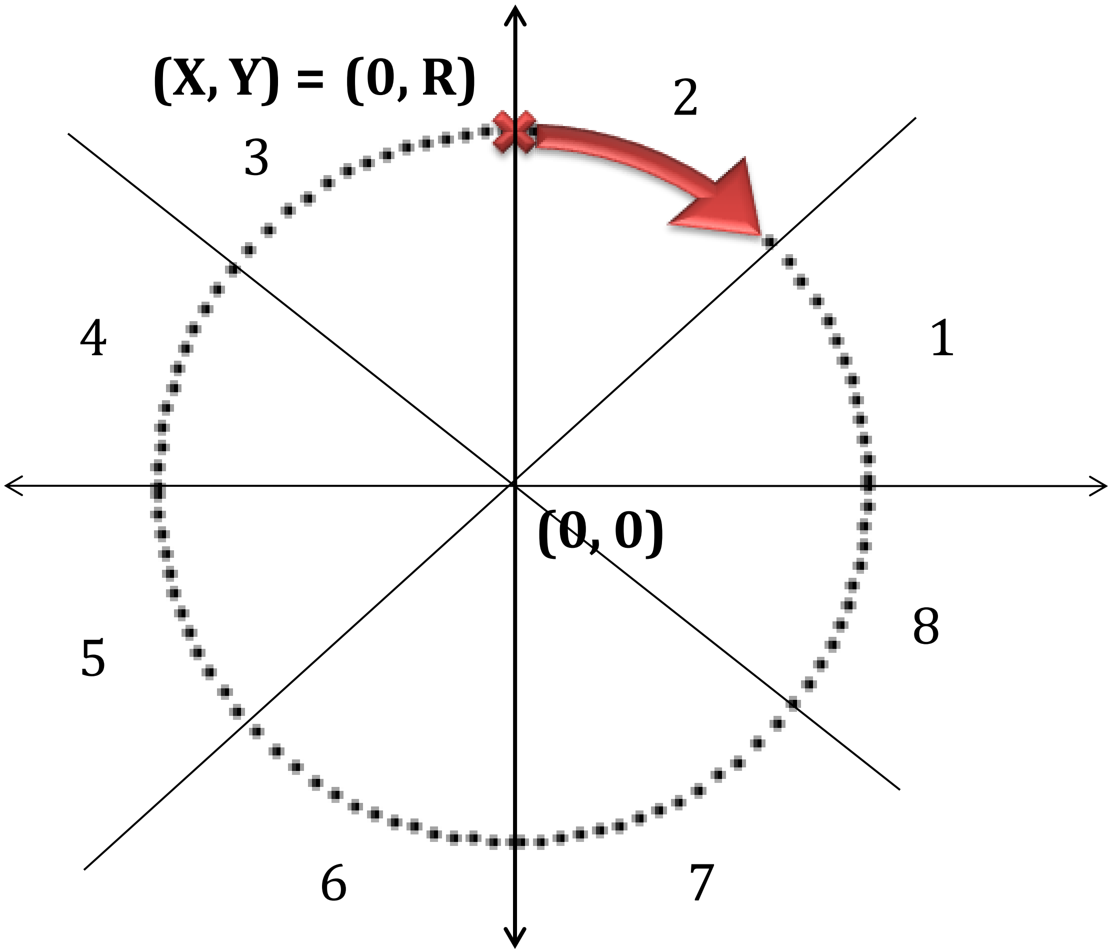
Then the plotting of next pixels starts clock-wise….



Observation

The first pixel of the circumference is plotted on (0, R)

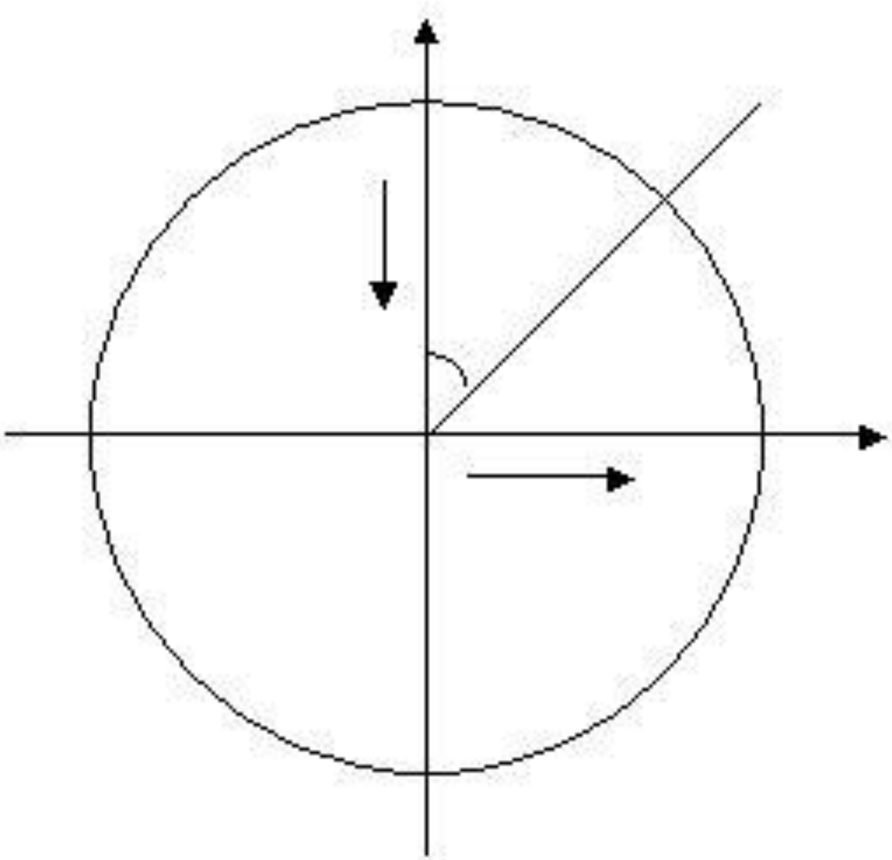
Then the plotting of next pixels starts clock-wise....



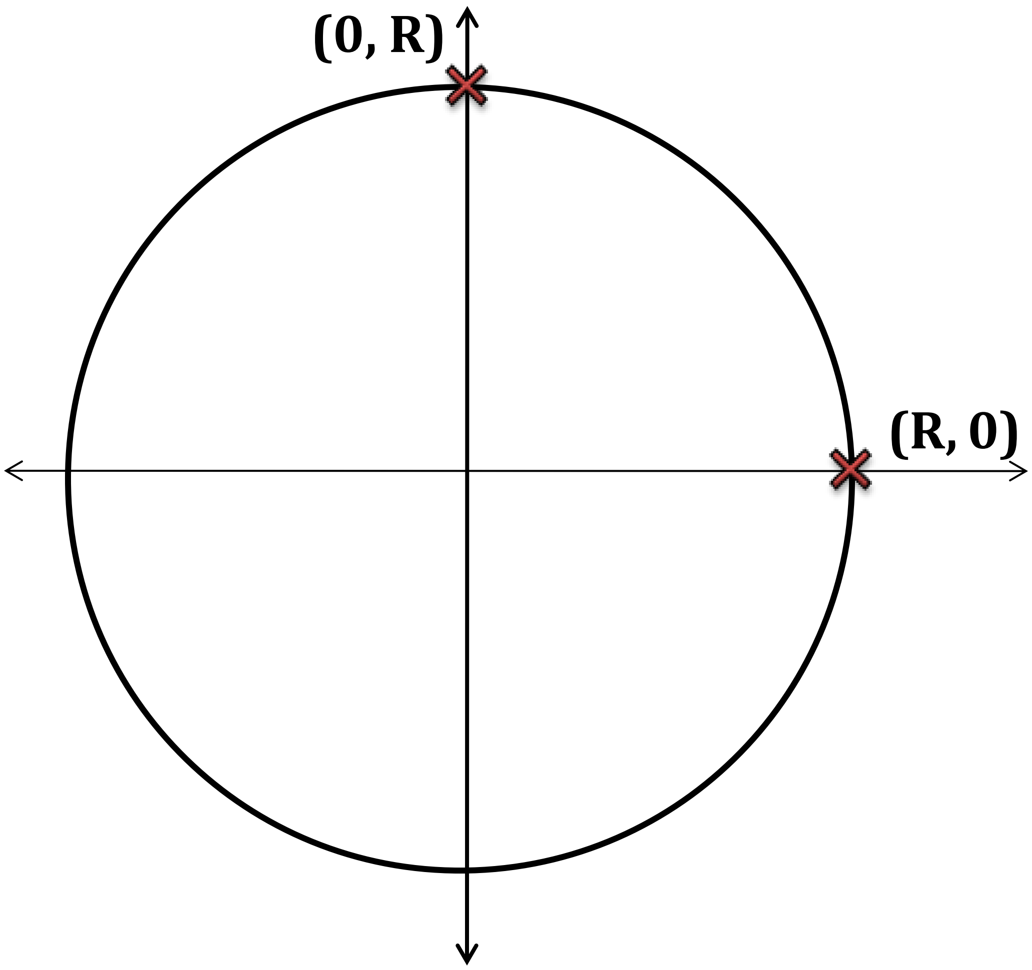
That means the plotting starts from (0, R) and moving into the 2^{nd} Octant

Observation

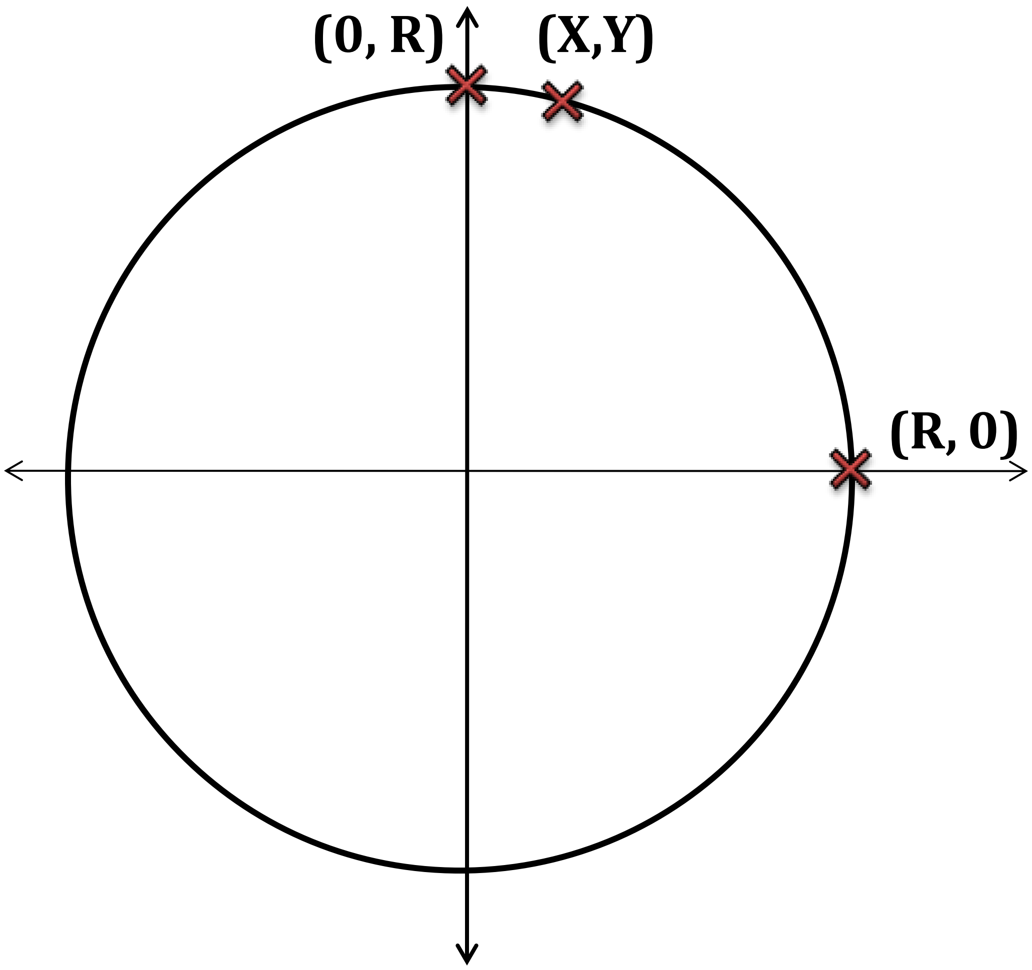
while moving through the 2nd octant, the X value is increasing and Y value is decreasing



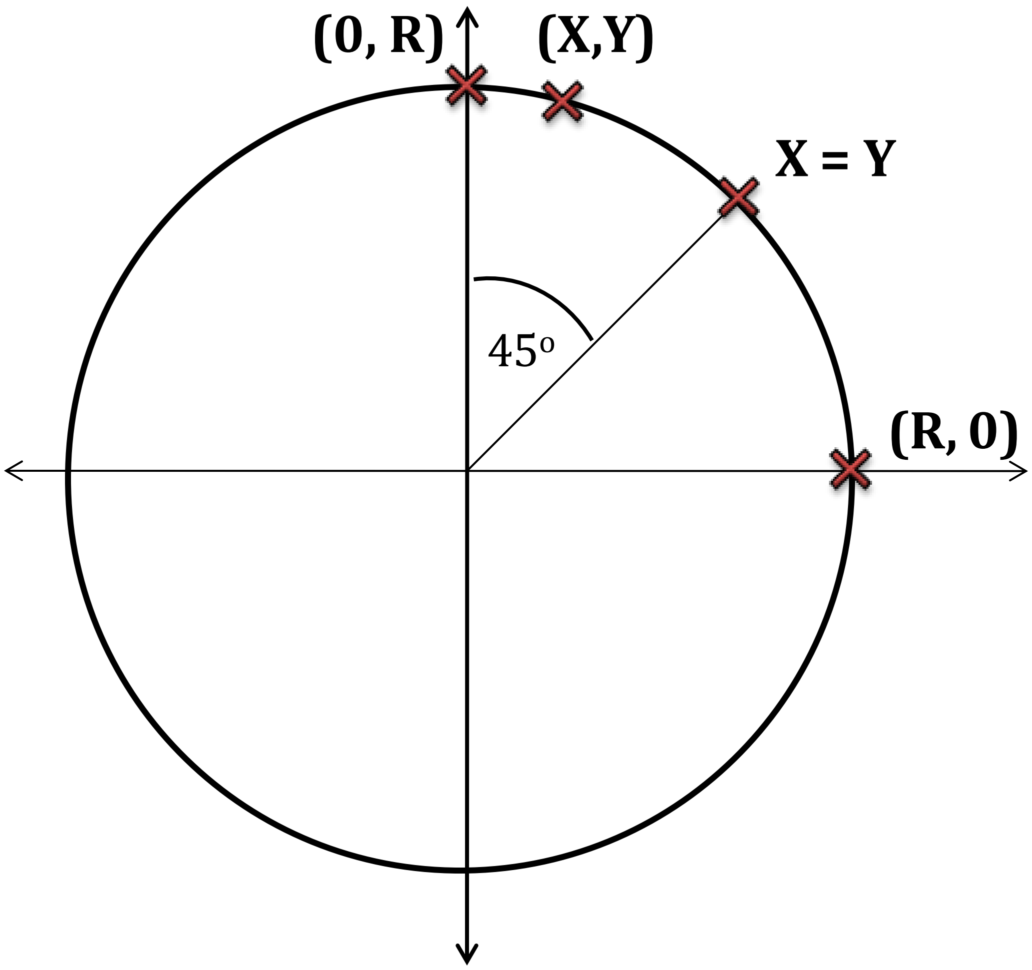
Observation



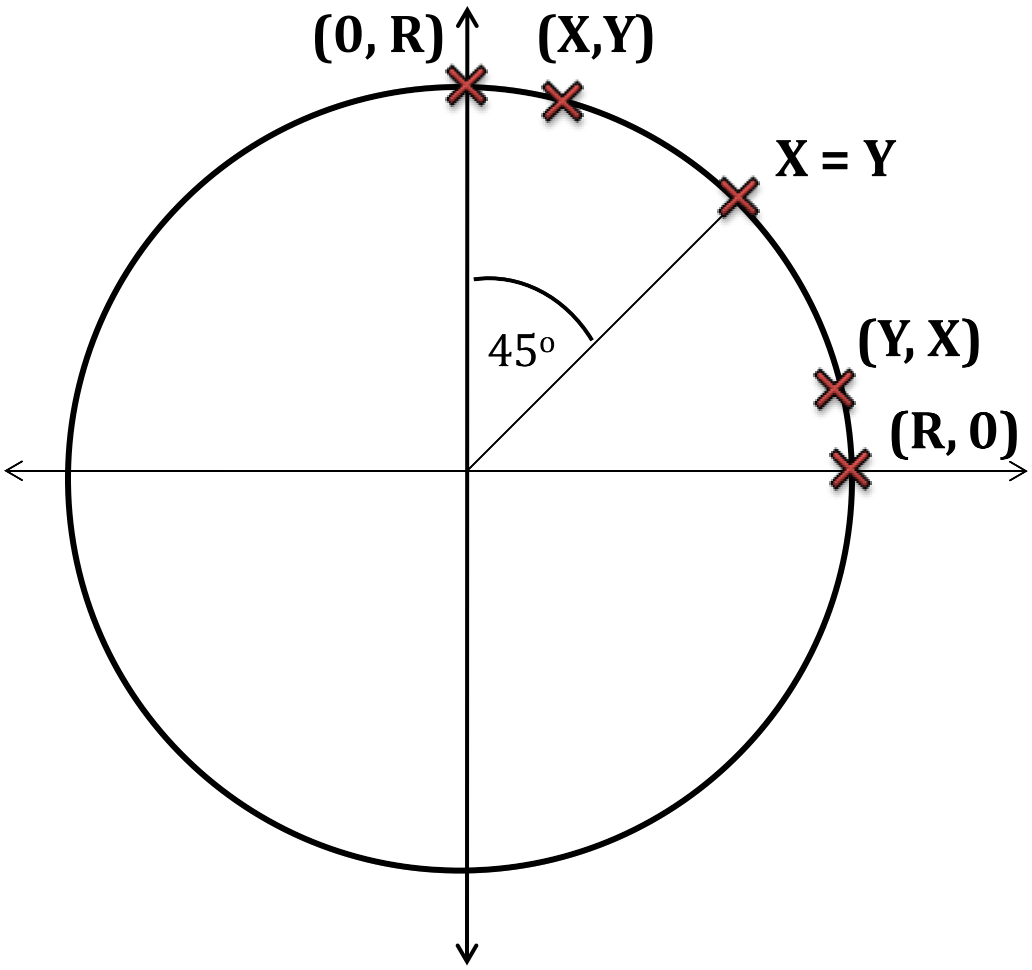
Observation



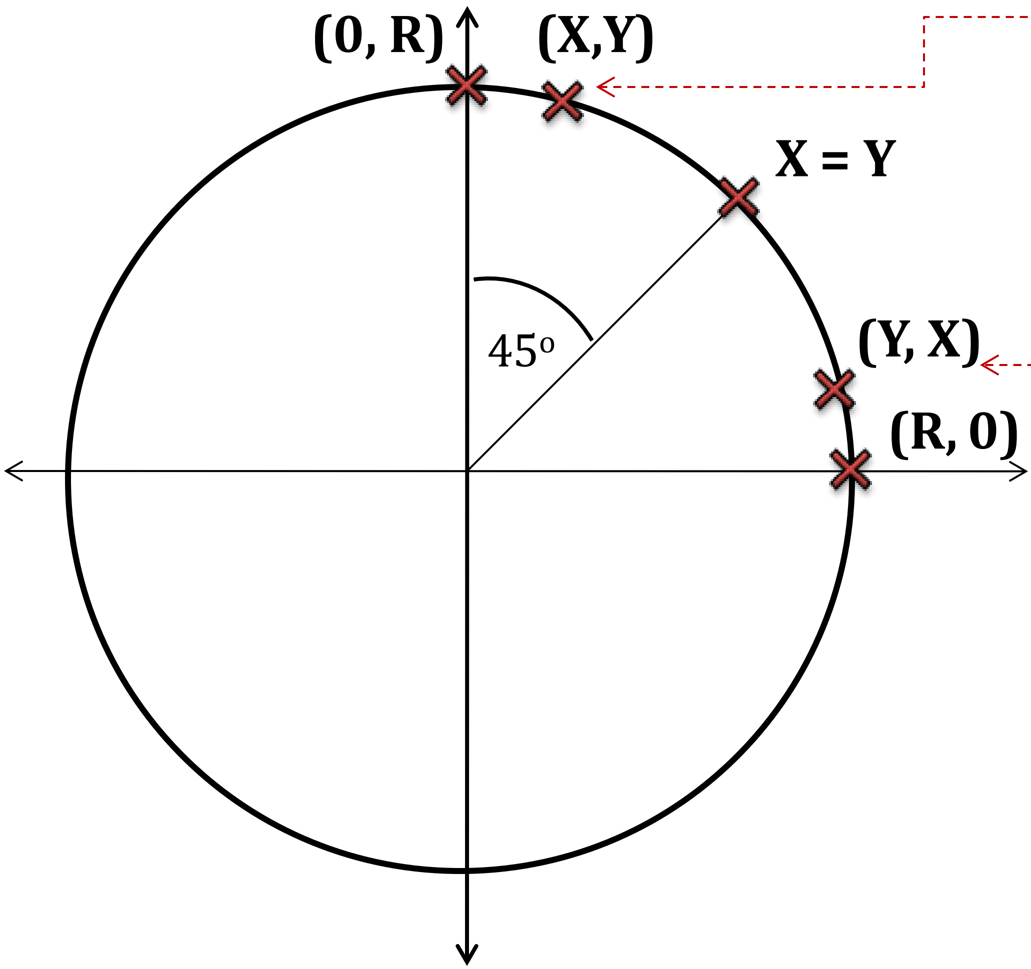
Observation



Observation



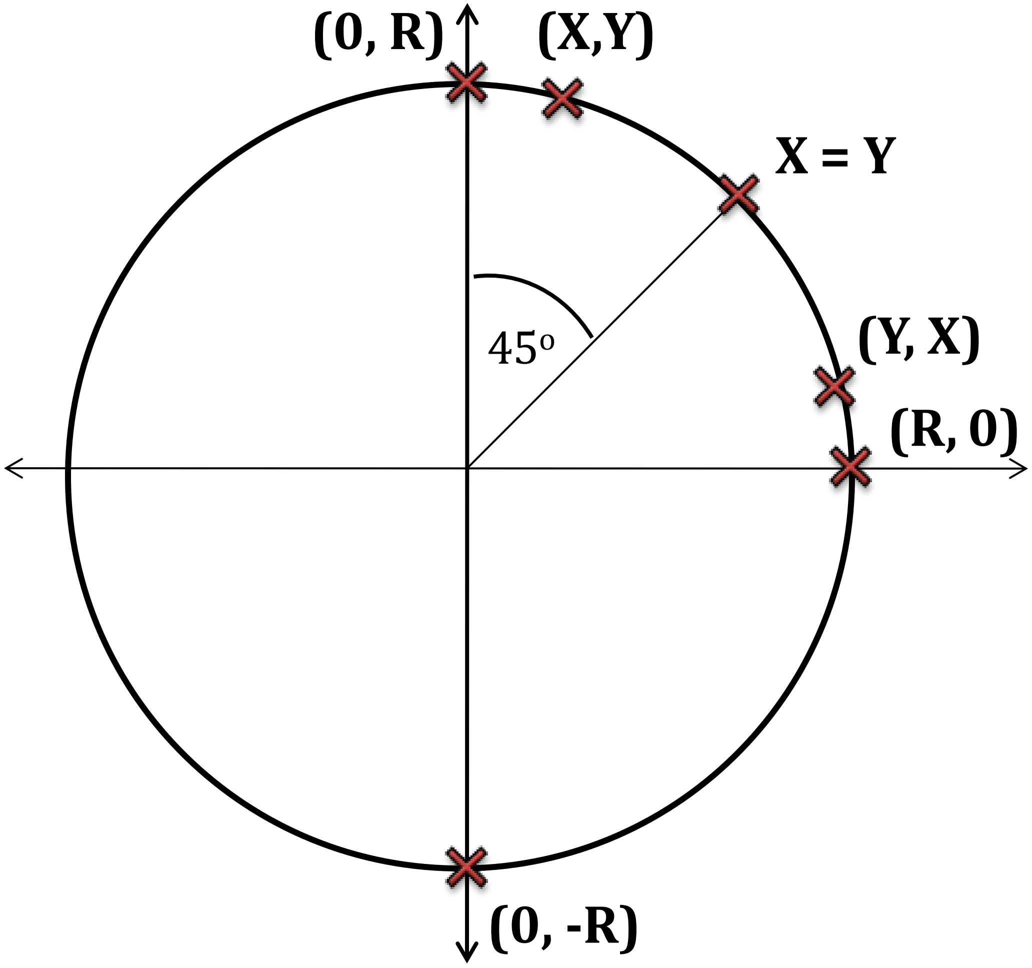
Observation



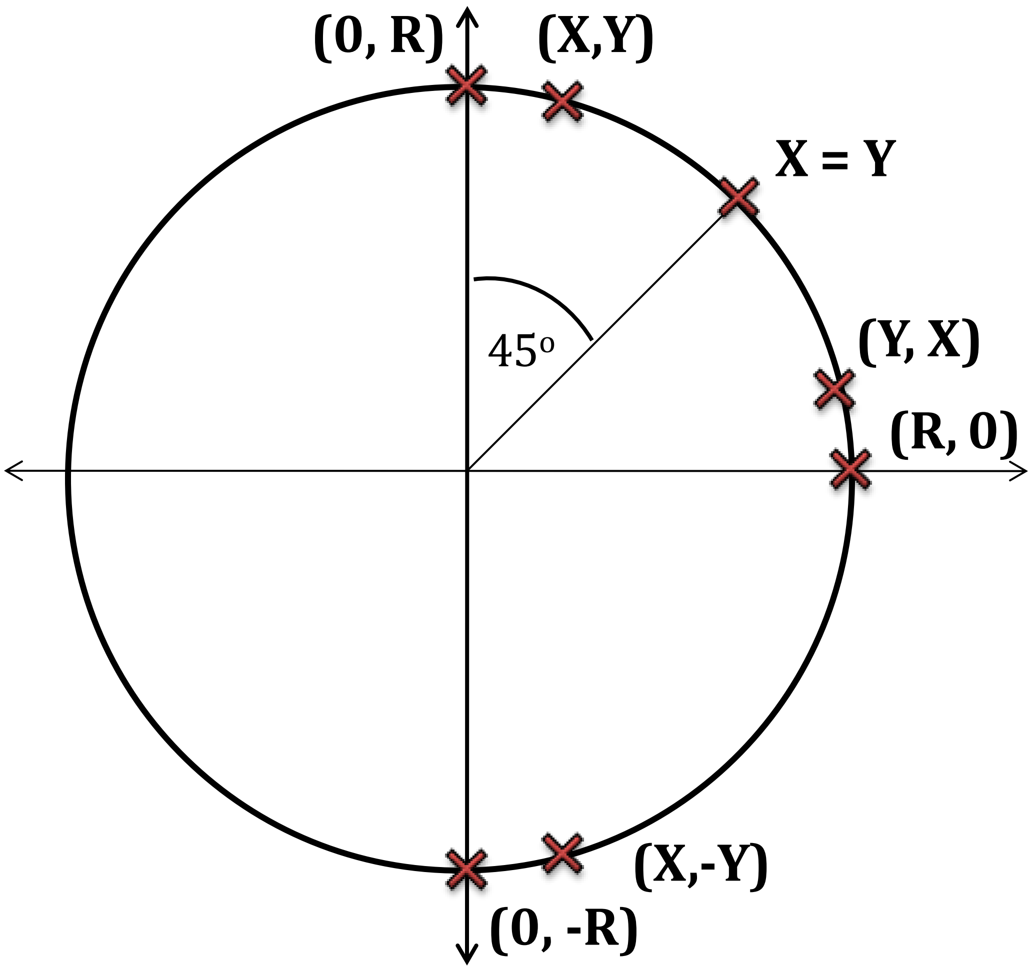
In 2^{nd} Octant : **X < Y**

But, in 1^{st} Octant : **X > Y**

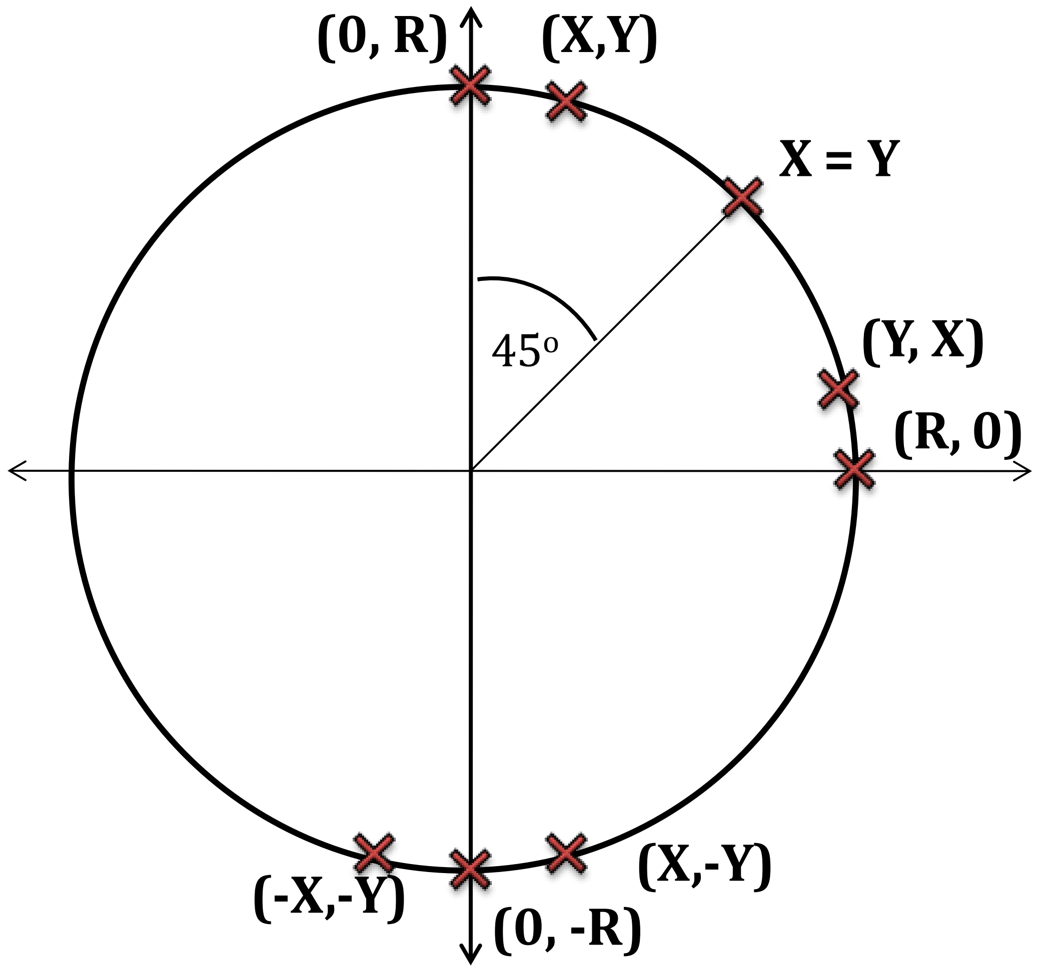
Observation



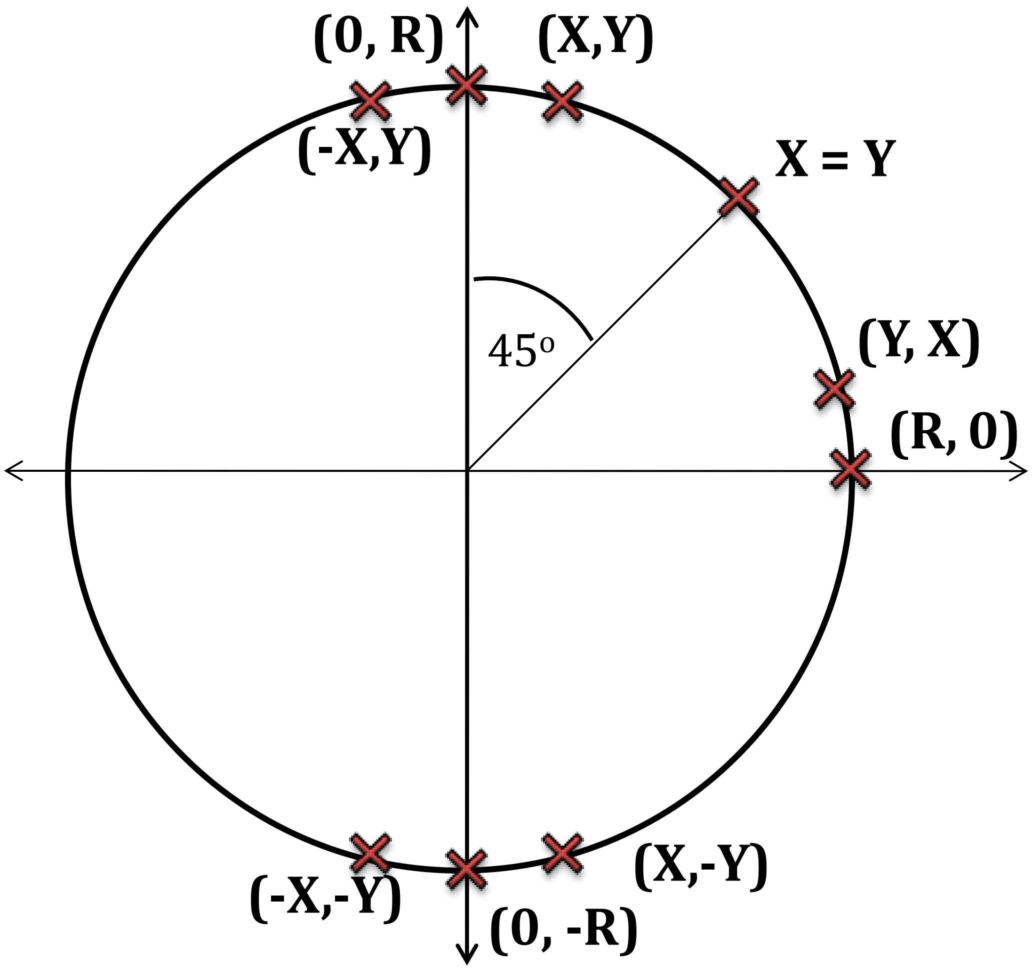
Observation



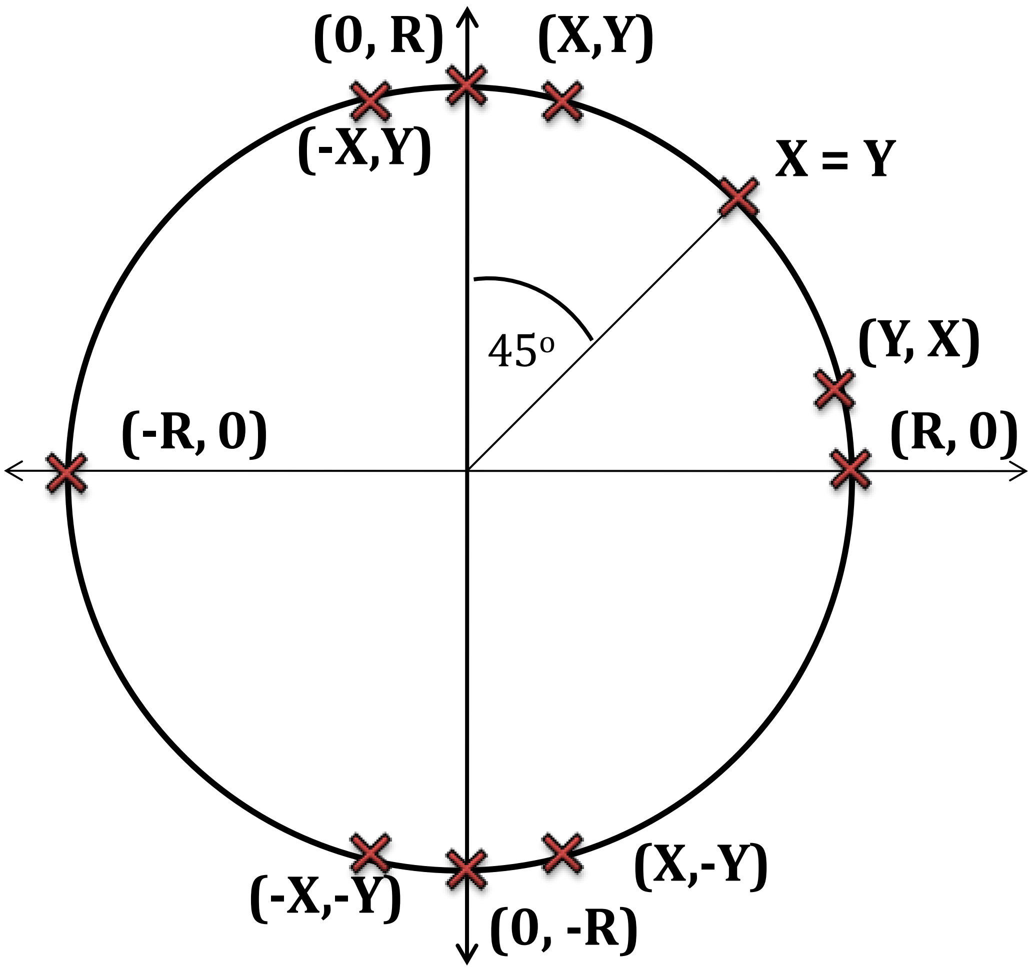
Observation



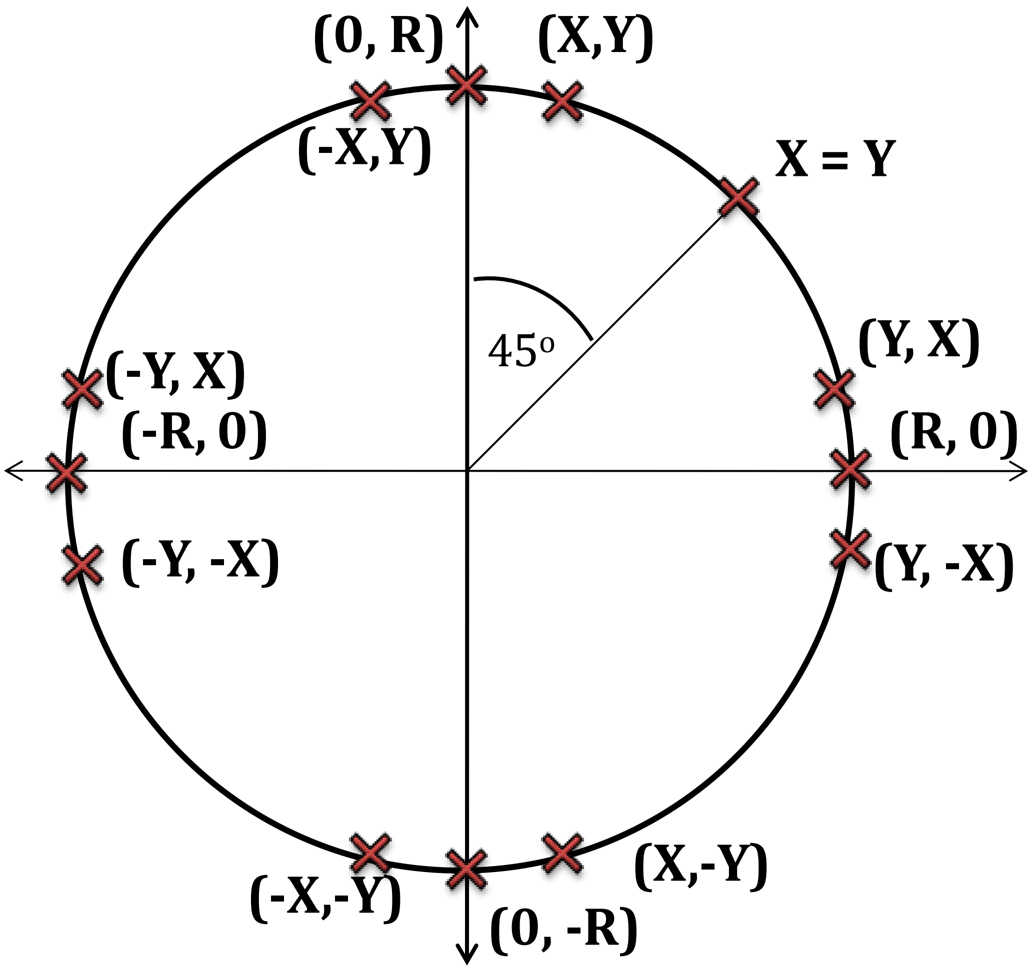
Observation



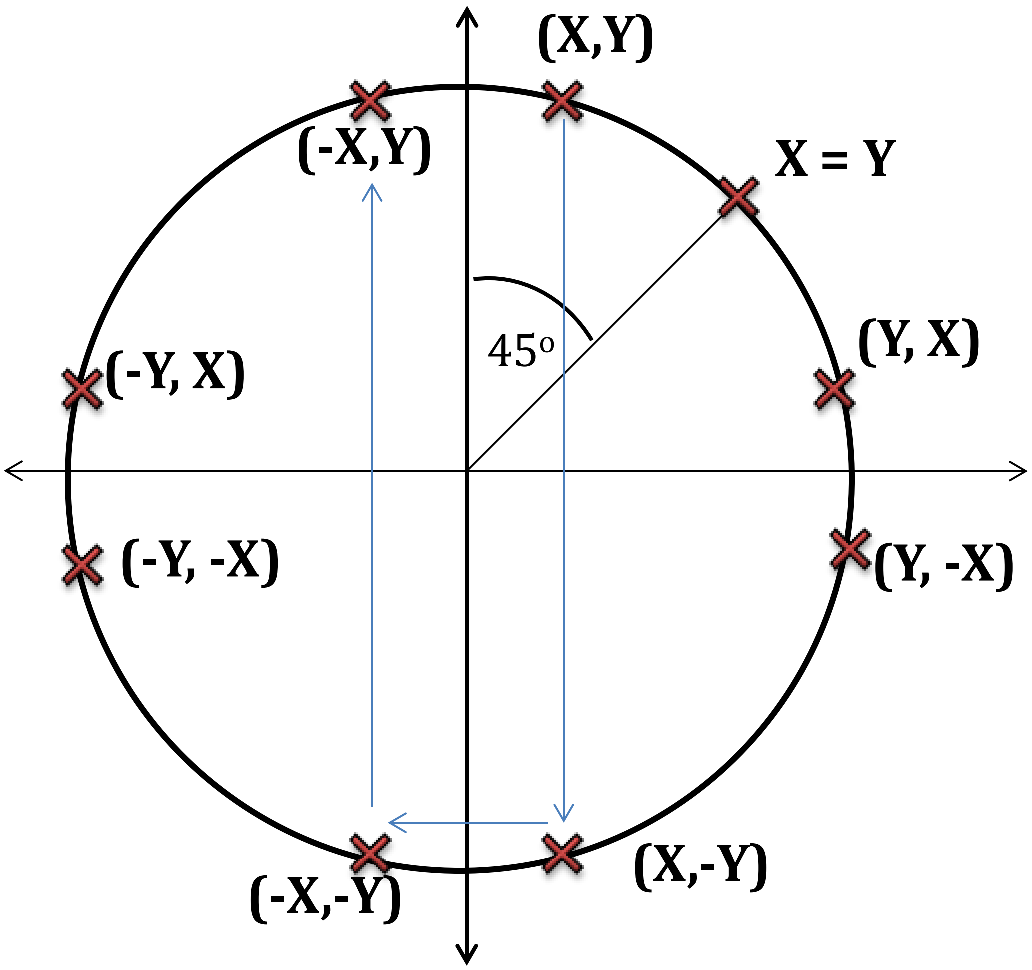
Observation



Observation



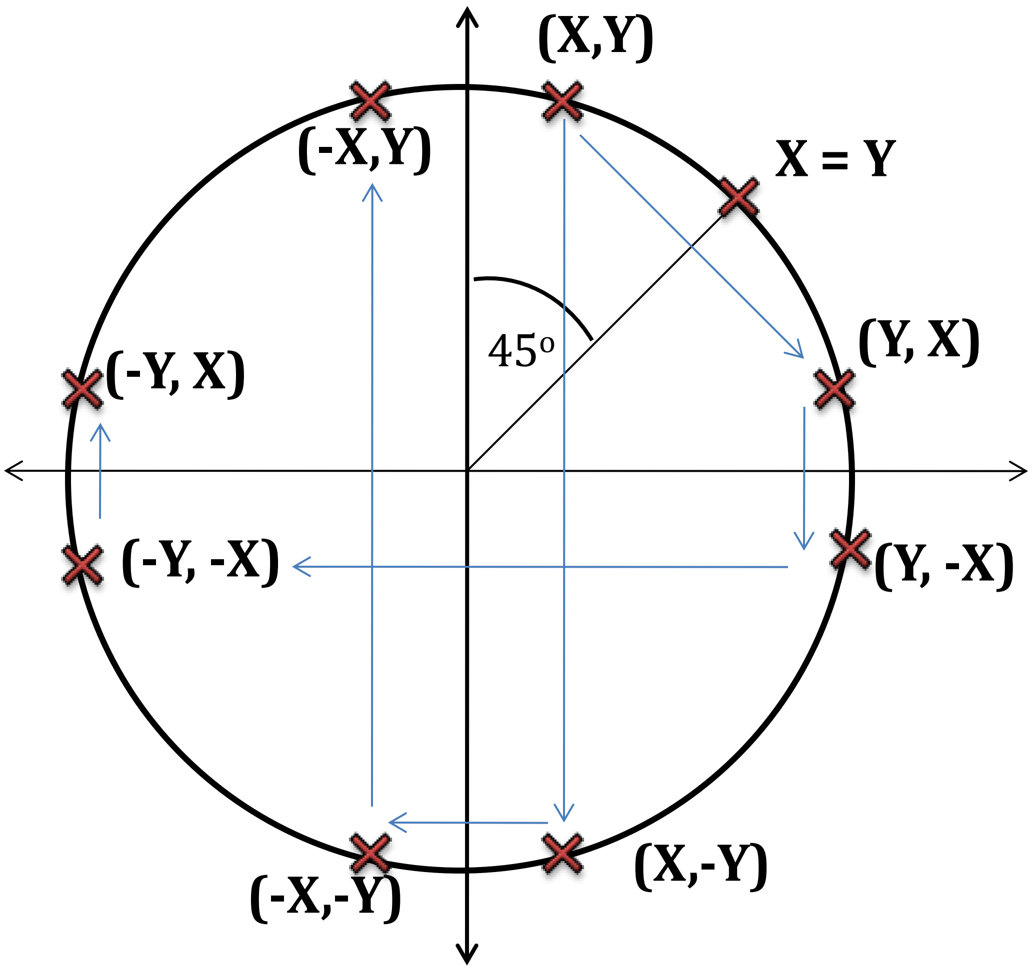
Observation



So, if we can obtain (X,Y) in 2nd octant, we can calculate the points-

- 7th Octant : (X,-Y)
- 6th Octant : (-X, -Y)
- 3rd Octant : (-X, Y)

Observation

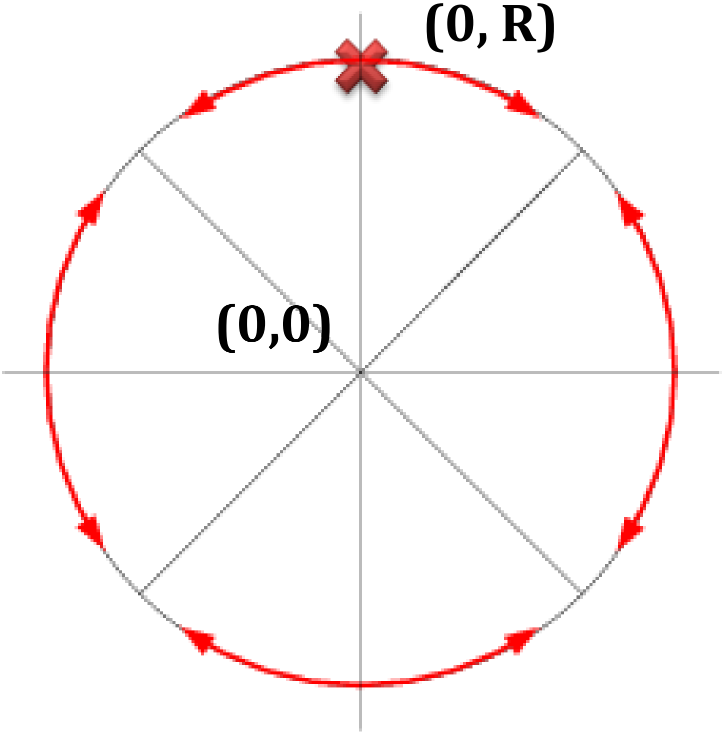


So, if we can obtain (X,Y) in 2nd octant, we can simply swap X and Y to get the points-

- 1st Octant : (Y, X)
- 8th Octant : (Y, -X)
- 5th Octant : (-Y, -X)
- 4th Octant : (-Y, X)

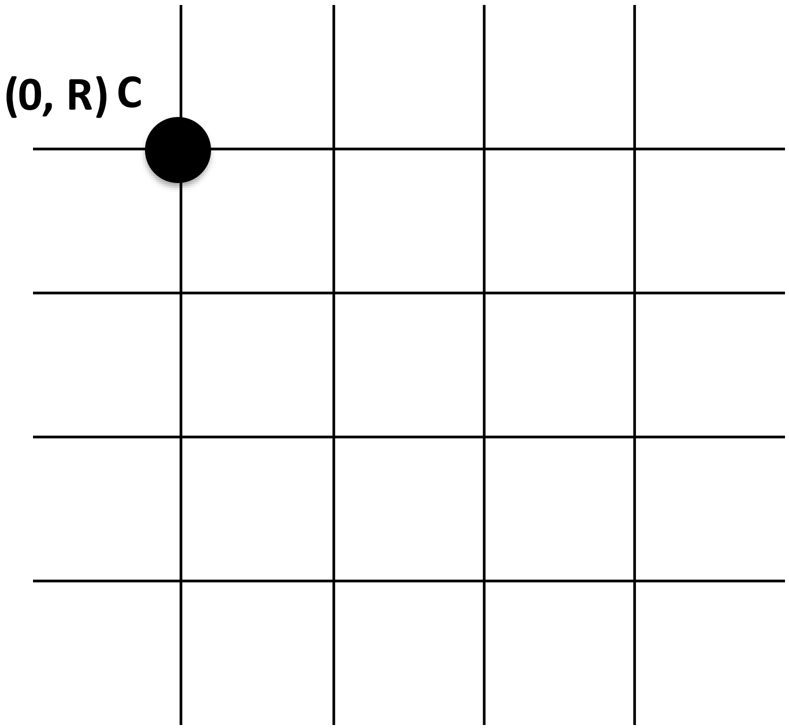
Using symmetric property of circle

So, if we can obtain (X,Y) in 2nd octant, we can calculate the points in other 7 octants

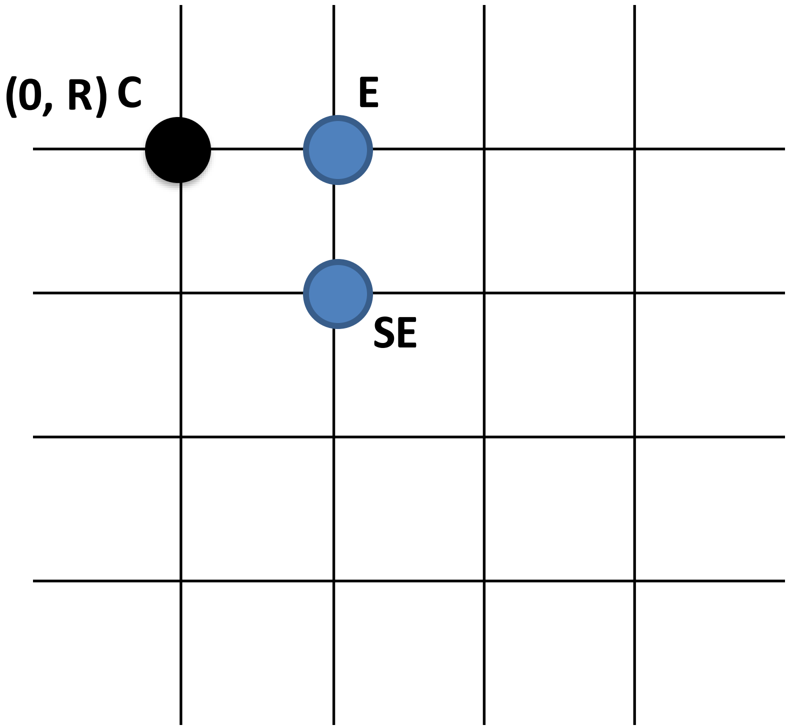


So, our target is to get the pixels of only 2nd octant of the circumference

Bresenham's Circle Drawing Algorithm: How it works

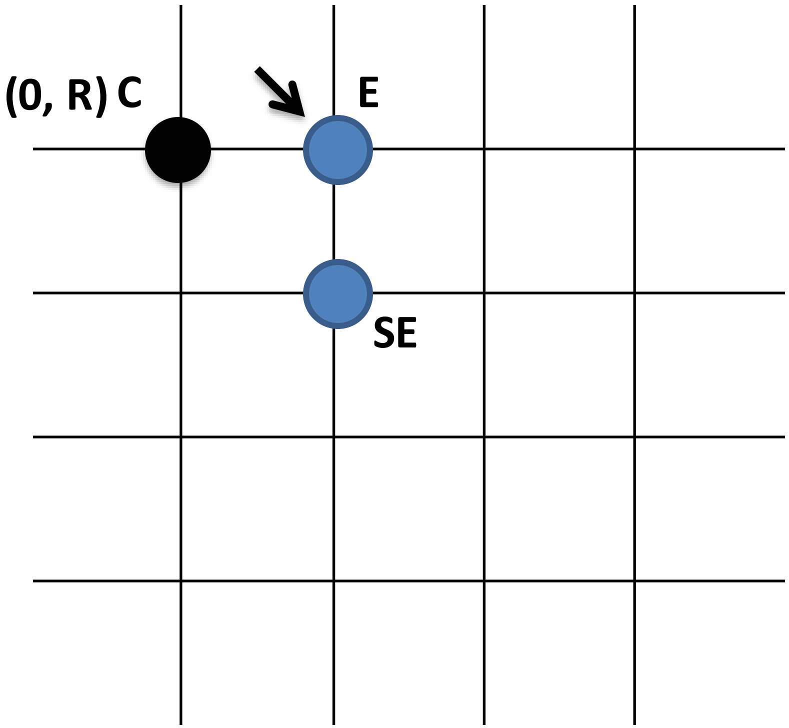


Bresenham's Circle Drawing Algorithm: How it works



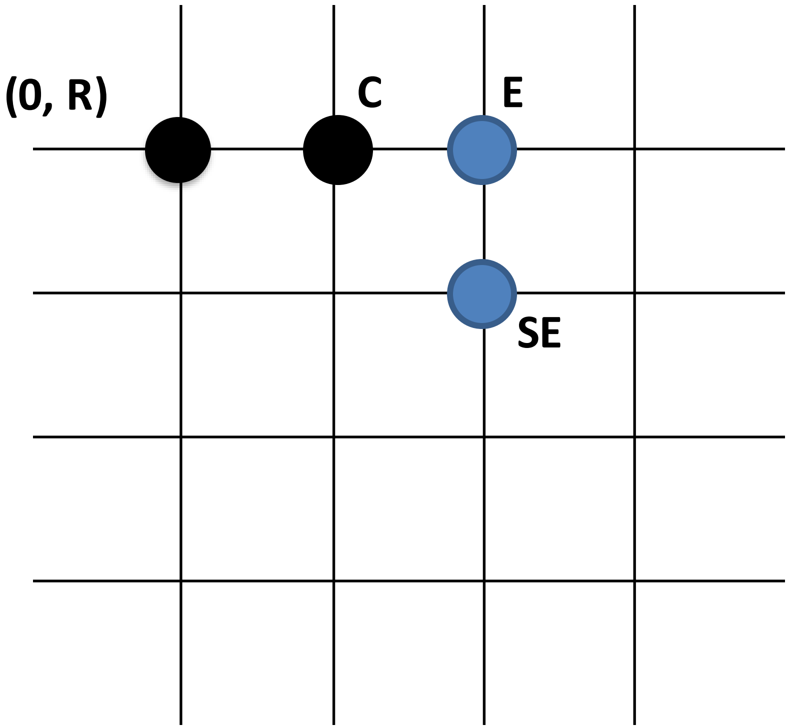
Next pixel is chosen (from E or SE) to build the line successively

Bresenham's Circle Drawing Algorithm: How it works



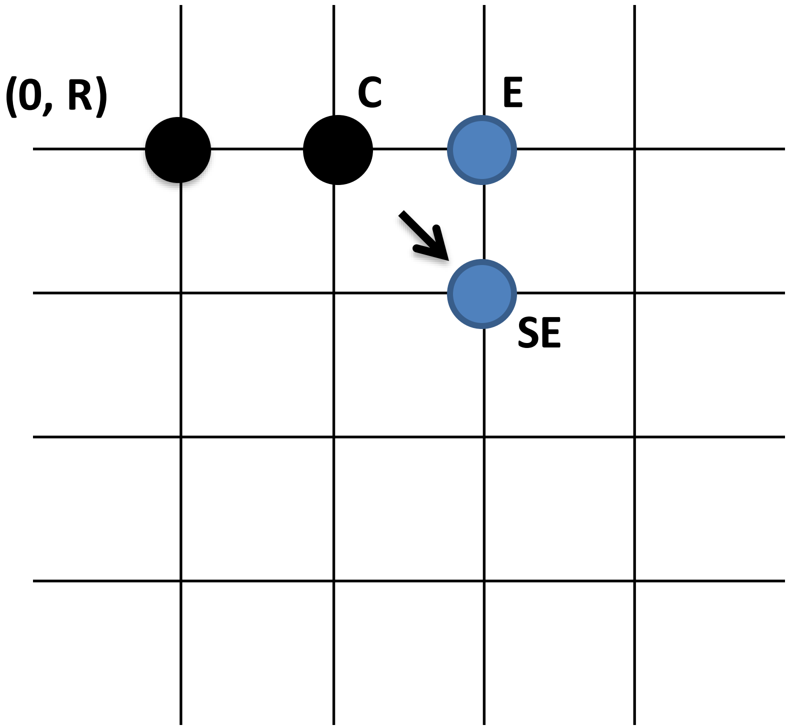
Next pixel is chosen (from E or SE) to build the line successively

Bresenham's Circle Drawing Algorithm: How it works



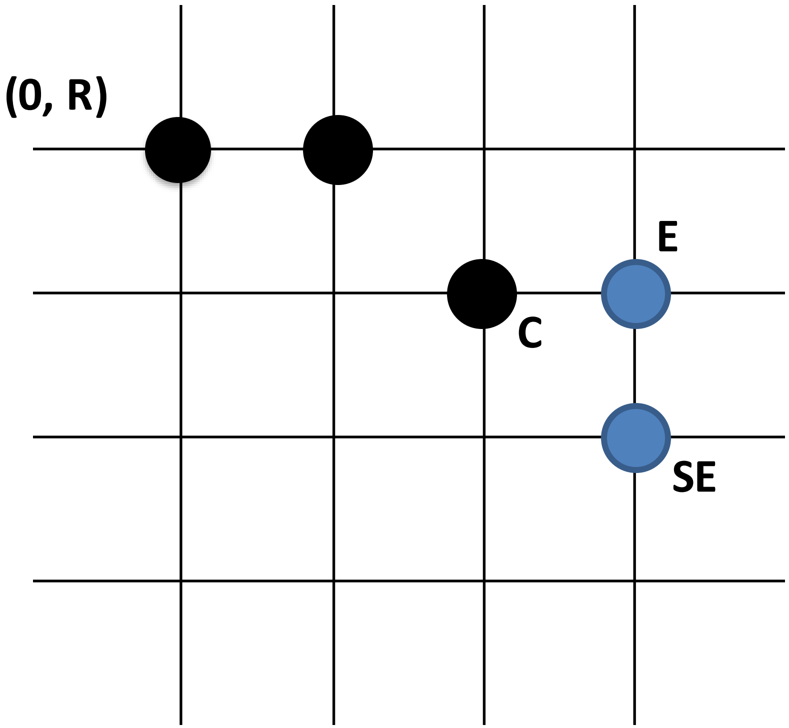
Next pixel is chosen (from E or SE) to build the line successively

Bresenham's Circle Drawing Algorithm: How it works



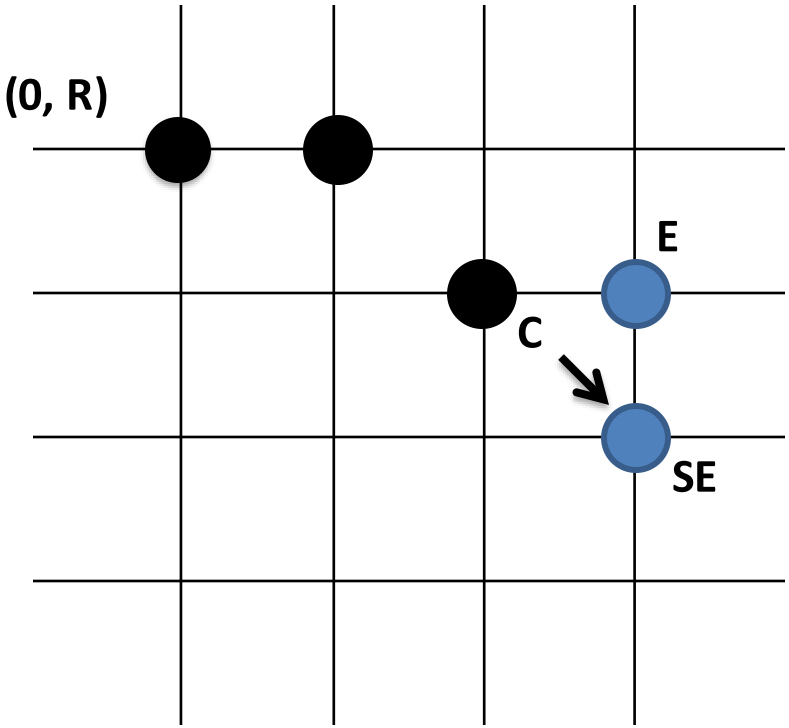
Next pixel is chosen (from E or SE) to build the line successively

Bresenham's Circle Drawing Algorithm: How it works



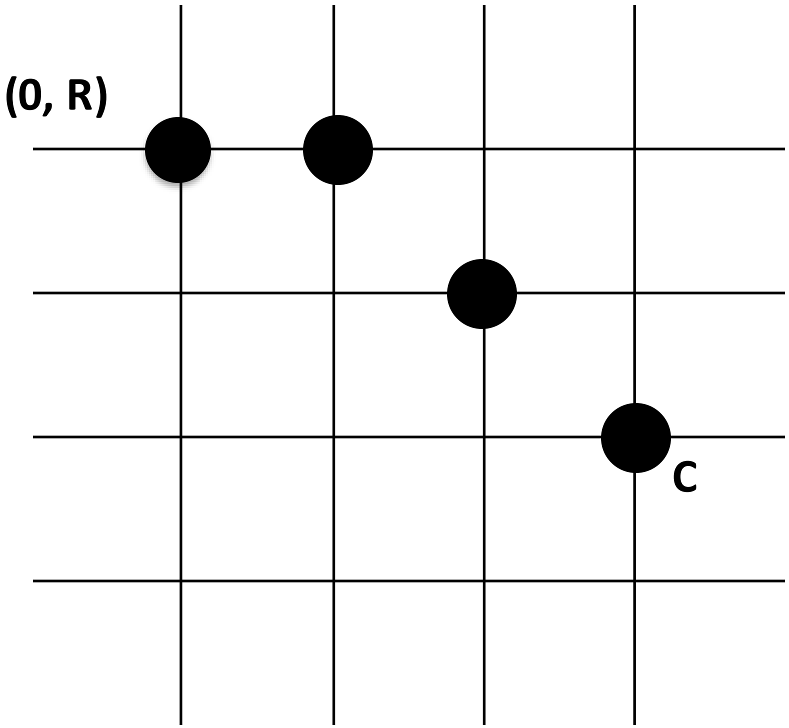
Next pixel is chosen (from E or SE) to build the line successively

Bresenham's Circle Drawing Algorithm: How it works



Next pixel is chosen (from E or SE) to build the line successively

Bresenham's Circle Drawing Algorithm: How it works



As we know that,
In 2nd Octant : **X < Y**
in 1st Octant : **X > Y**

We will stop selecting E or SE when X > Y, that means when 2nd octant is completed

Equation of Circle and its function representation

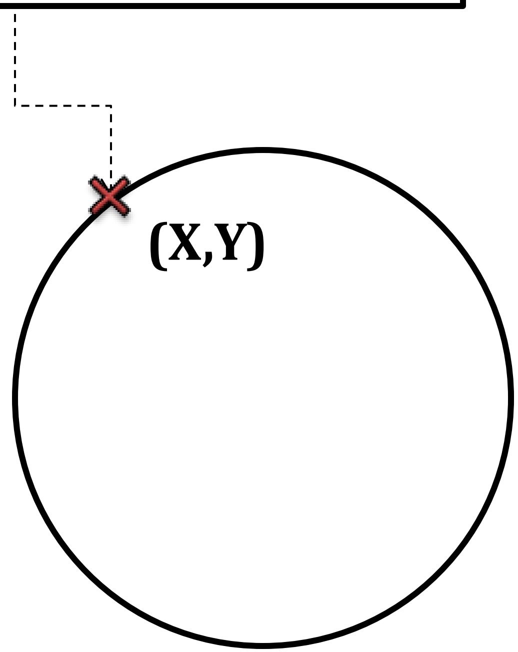
$$x^2 + y^2 = R^2$$

$$F(x, y) = x^2 + y^2 - R^2 = 0$$

Equation of Circle and its function representation

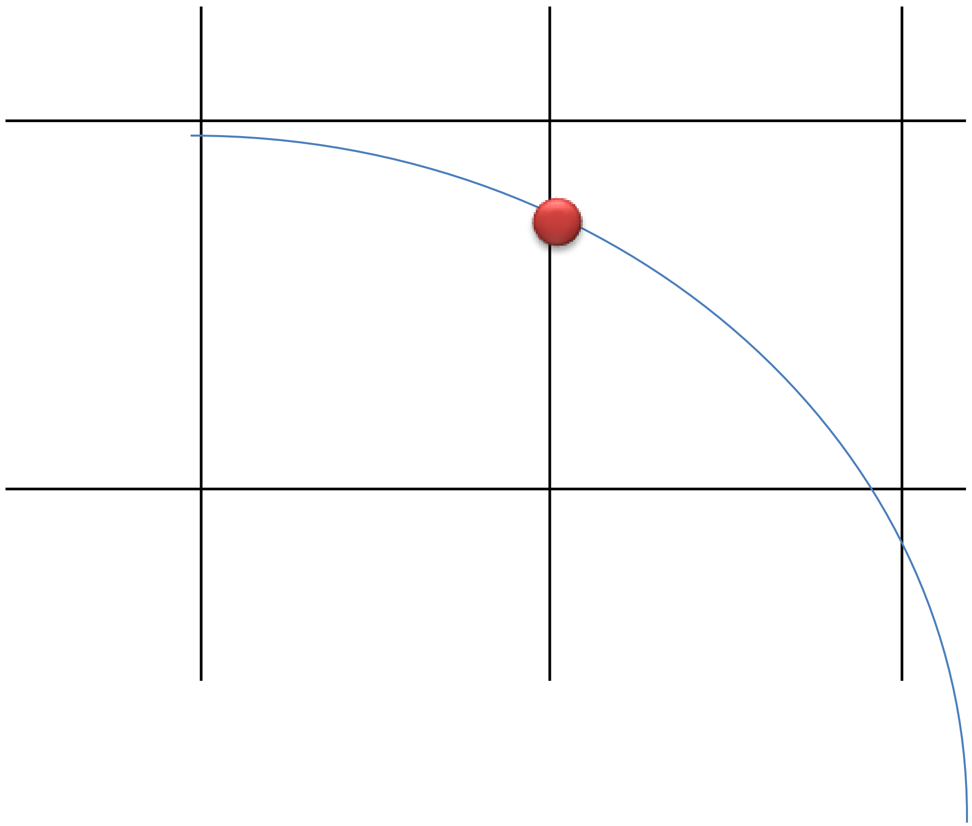
$$x^2 + y^2 = R^2$$

$$\boxed{F(x,y) = x^2 + y^2 - R^2 = 0}$$



Equation of Circle and its function representation

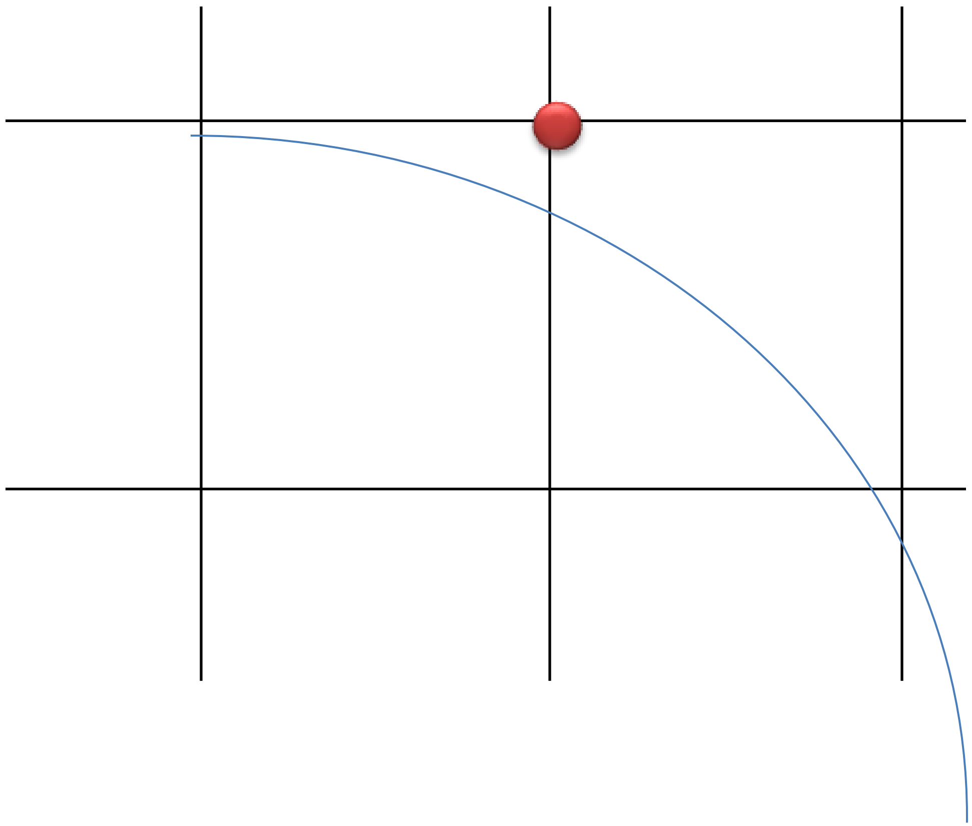
$$F(x, y) = x^2 + y^2 - R^2$$



If **F(X,Y) = 0,** the point (X,Y) on the circle

Equation of Circle and its function representation

$$F(x,y) = x^2 + y^2 - R^2$$

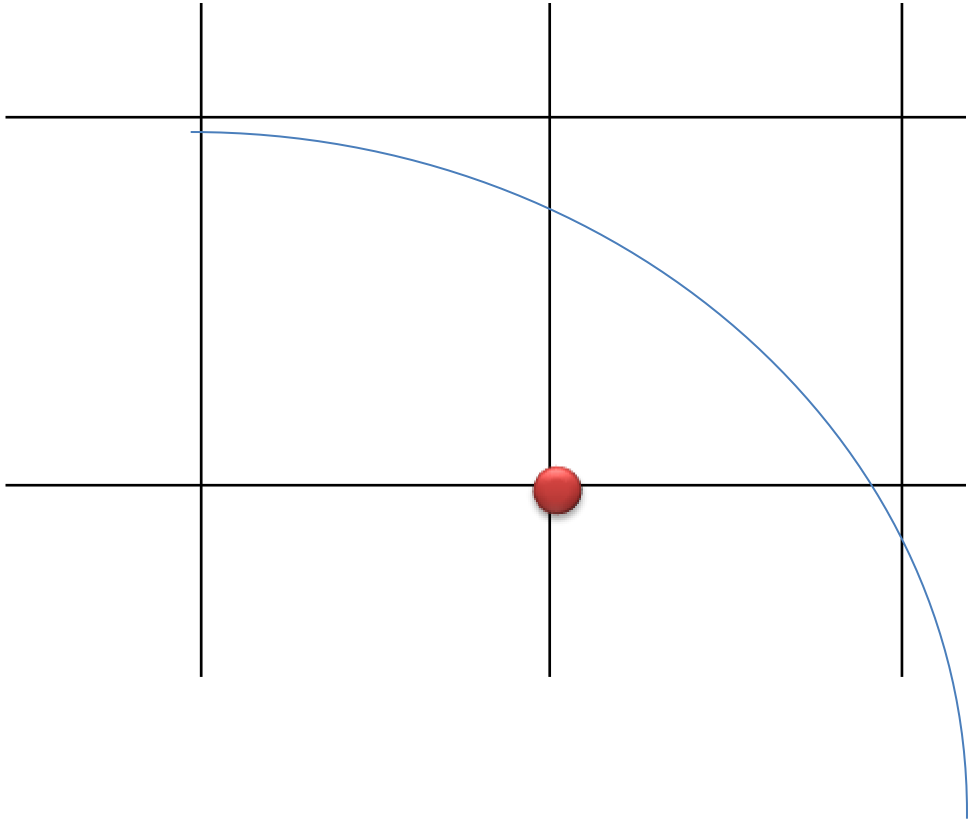


If **F(X,Y) = 0,** the point (X,Y) on the circle

If **F (X, Y) > 0,** the point (X,Y) is outside the circle

Equation of Circle and its function representation

$$F(x, y) = x^2 + y^2 - R^2$$

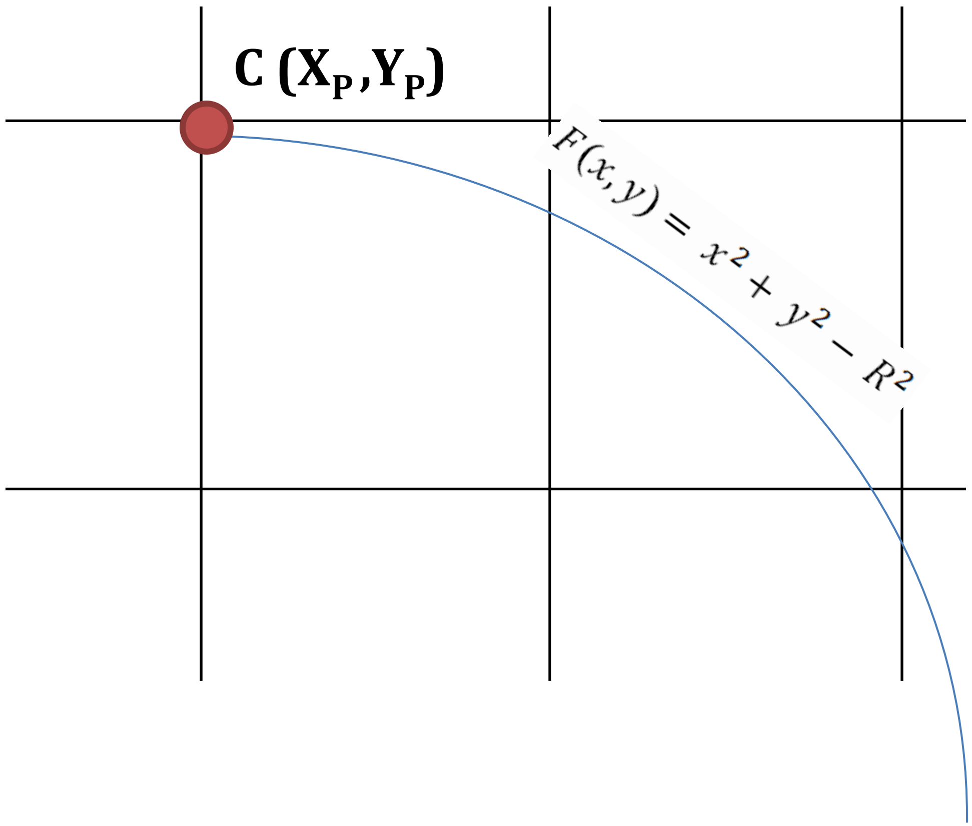


If **F(X,Y) = 0,** the point (X,Y) on the circle

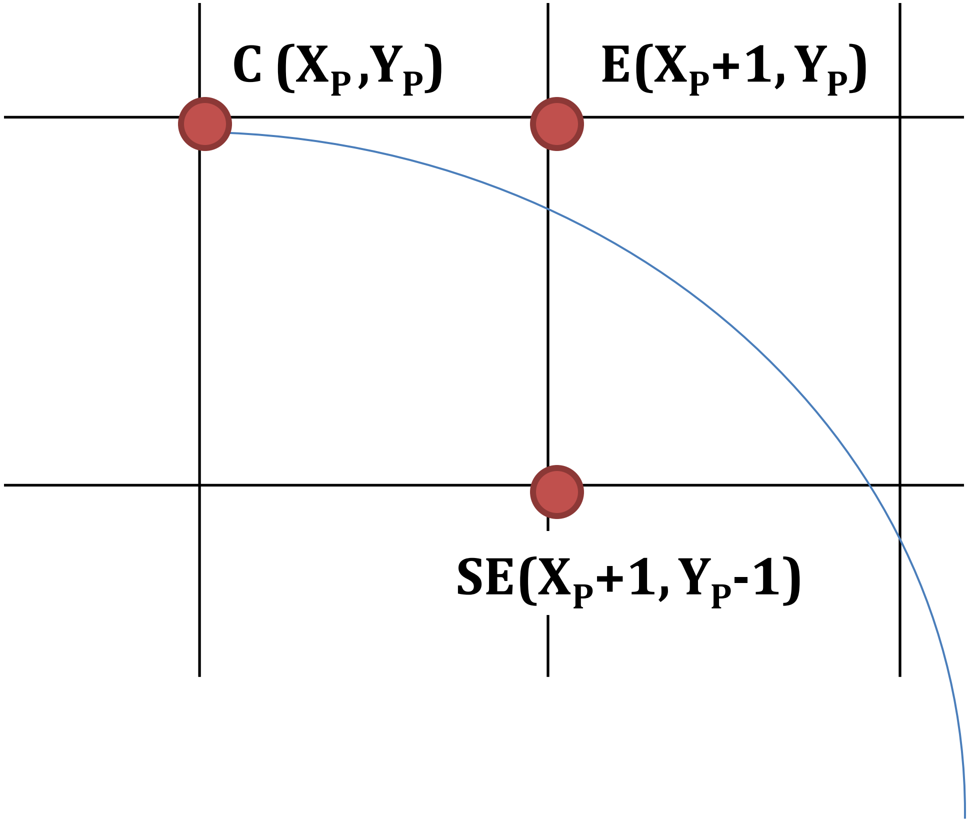
If **F (X, Y) > 0,** the point (X,Y) is outside the circle

If **F (X, Y) < 0,** the point (X,Y) is inside the circle

Selecting E or SE



Selecting E or SE

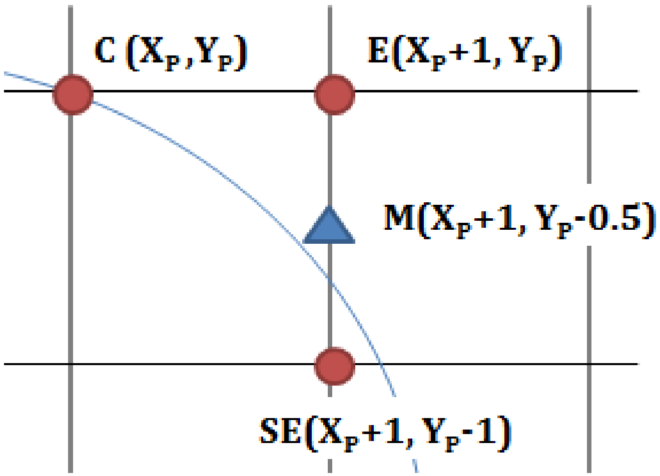


Selecting E or SE depends on closeness to the circumference.

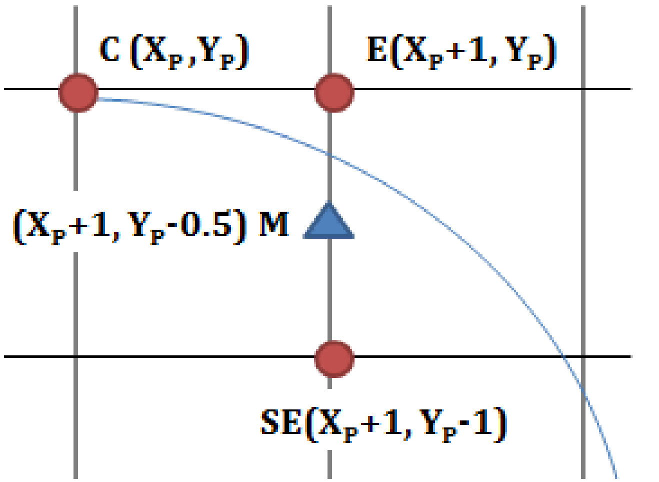
If E is closer to circumference, then E is selected

If SE is closer, then SE is selected

Selecting E or SE using Mid Point Criteria



If midpoint M is outside the circle, SE is closer to the circumference,
So, **SE** is selected



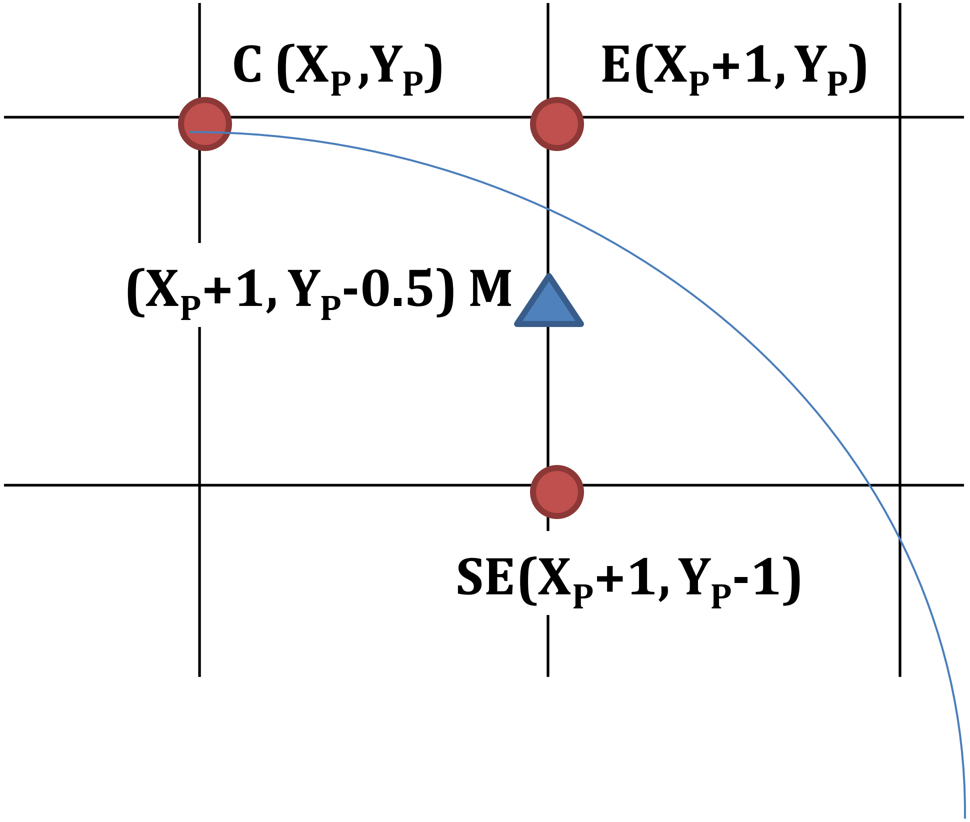
If midpoint M is inside the circle, E is closer to the circumference,
So, **E** is selected

Selecting E or SE using Mid Point Criteria

We know, $F(x,y) = x^2 + y^2 - R^2$

Lets put the mid point **M**'s coordinate in function F(X,Y)

$F(M) = F(\mathbf{X_P+1, Y_P - 0.5}) = \mathbf{(X_P+1)^2 + (Y_P - 0.5)^2 - R^2}$

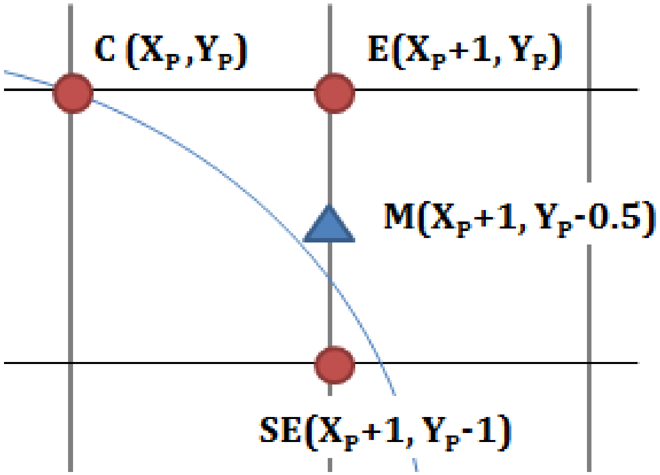


Lets store **F(M)** in a variable **d**

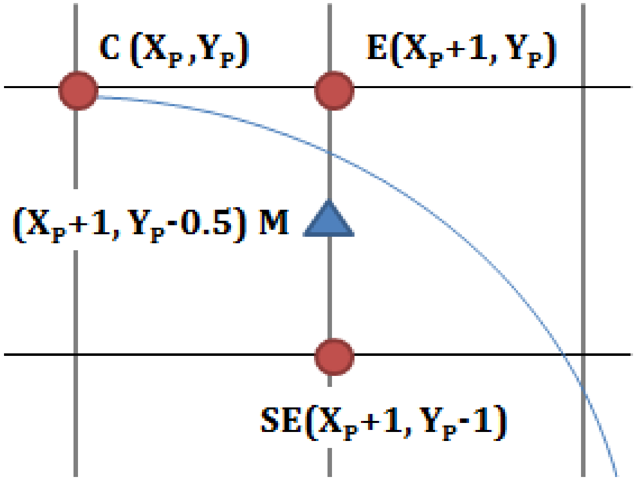
So, **d = F(M)**

d is called 'decision variable'

Selecting E or SE using Mid Point Criteria

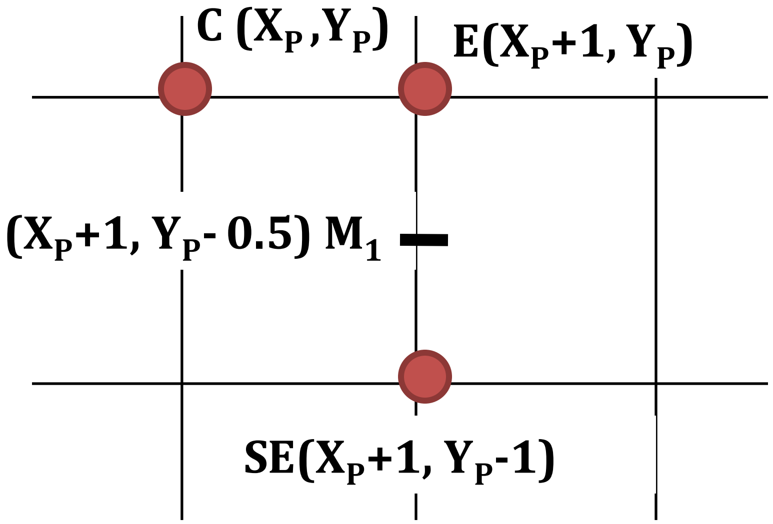


If **d >= 0**, then midpoint M is outside the circle, SE is closer to the circumference,
So, **SE** is selected



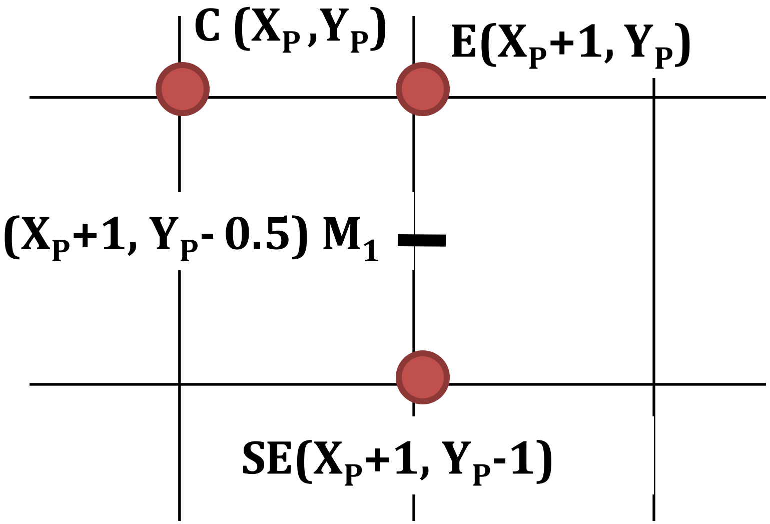
If **d < 0**, then midpoint M is inside the circle, E is closer to the circumference,
So, **E** is selected

Bresenham's Mid Point Criteria : Successive Updating (for selecting E)



$$d_1 = F(M_1)$$
$$= F(X_P+1, Y_P- 0.5)$$
$$= (X_P+1)^2 + (Y_P- 0.5)^2 - R^2$$

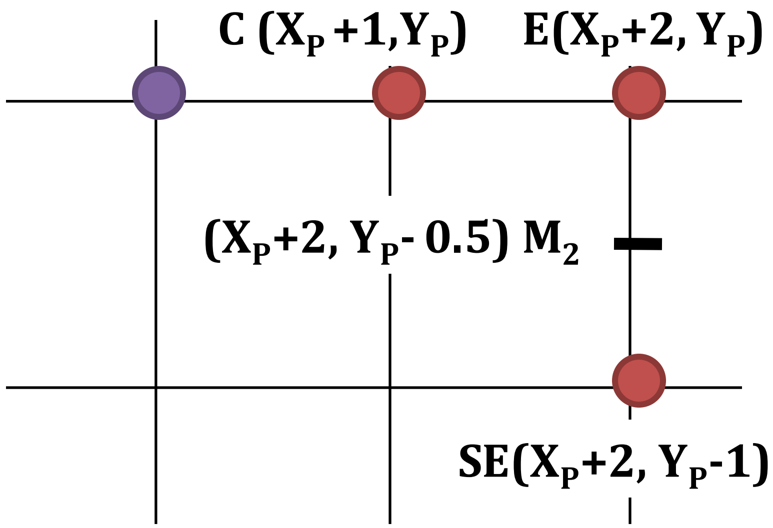
Bresenham's Mid Point Criteria : Successive Updating (for selecting E)



$$d_1 = F(M_1)$$
$$= F(X_P+1, Y_P- 0.5)$$
$$= (X_P+1)^2 + (Y_P- 0.5)^2 - R^2$$

If $d_1 < 0$, E $(X_P=X_P+1, Y_P)$

Bresenham's Mid Point Criteria : Successive Updating (for selecting E)

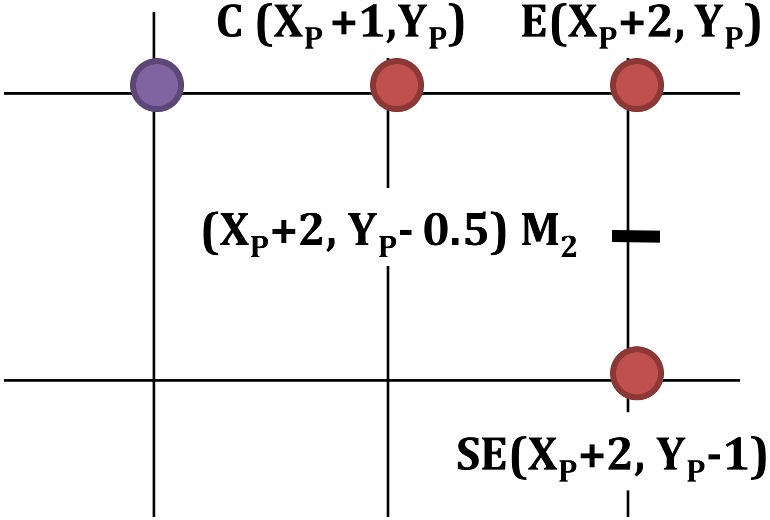


$$\mathbf{d_1 = F(M_1) = F(X_P+1, Y_P - 0.5) = (X_P+1)^2 + (Y_P - 0.5)^2 - R^2}$$

If $\mathbf{d_1 < 0}$, **E** $\mathbf{(X_P=X_P+1, Y_P)}$

$$\mathbf{d_2 = F(M_2)}$$
$$= F(X_P+2, Y_P-0.5)$$
$$= (X_P+2)^2 + (Y_P - 0.5)^2 - R^2$$
$$= X_P^2 + 4X_P + 4 + (Y_P - 0.5)^2 - R^2$$
$$= X_P^2 + 2X_P + 1 + (Y_P - 0.5)^2 - R^2 + 2X_P + 3$$
$$= \mathbf{d_1 + (2X_P + 3)}$$

Bresenham's Mid Point Criteria : Successive Updating (for selecting E)



$$\begin{aligned} d_1 &= F(M_1) \\ &= F(X_P+1, Y_P - 0.5) \\ &= (X_P+1)^2 + (Y_P - 0.5)^2 - R^2 \end{aligned}$$

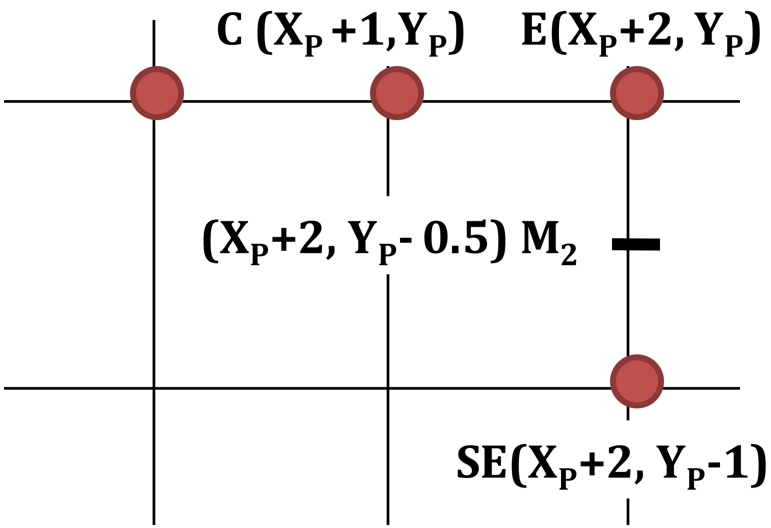
If $d_1 < 0$, E $(X_P = X_P+1, Y_P)$

$$\begin{aligned} d_2 &= F(M_2) \\ &= F(X_P+2, Y_P - 0.5) \\ &= (X_P+2)^2 + (Y_P - 0.5)^2 - R^2 \\ &= X_P^2 + 4X_P + 4 + (Y_P - 0.5)^2 - R^2 \\ &= X_P^2 + 2X_P + 1 + (Y_P - 0.5)^2 - R^2 + 2X_P + 3 \\ &= d_1 + (2X_P + 3) \end{aligned}$$

Similarly, If $d_2 < 0$, E $(X_P = X_P+1, Y_P)$

Then $d_3 = d_2 + (2X_P + 3)$

Bresenham's Mid Point Criteria : Successive Updating (for selecting E)

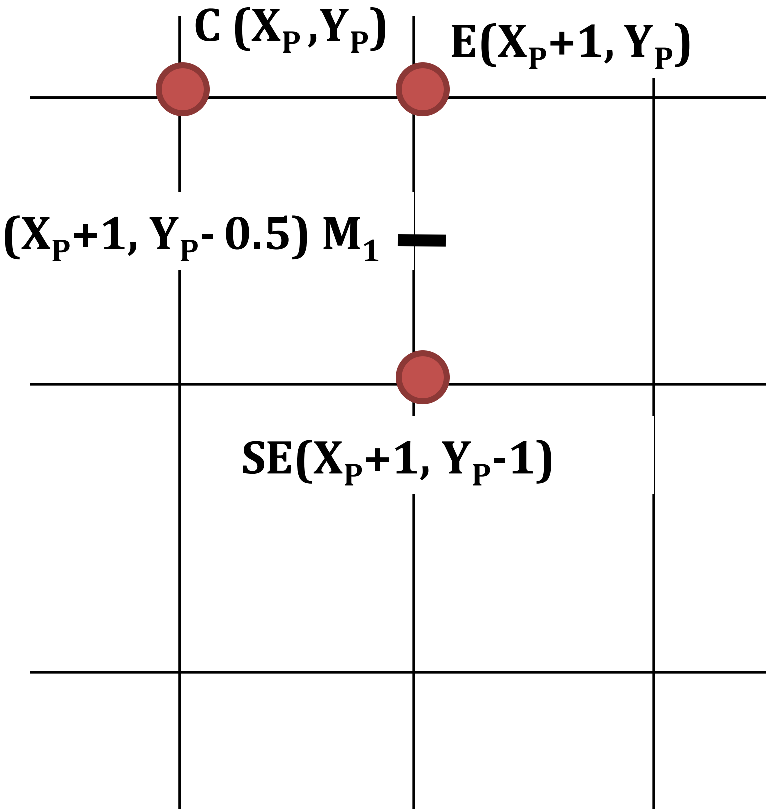


Every iteration after **selecting E**, we can successively update our decision variable with-

$$d_{NEW} = d_{OLD} + (2X_P + 3)$$

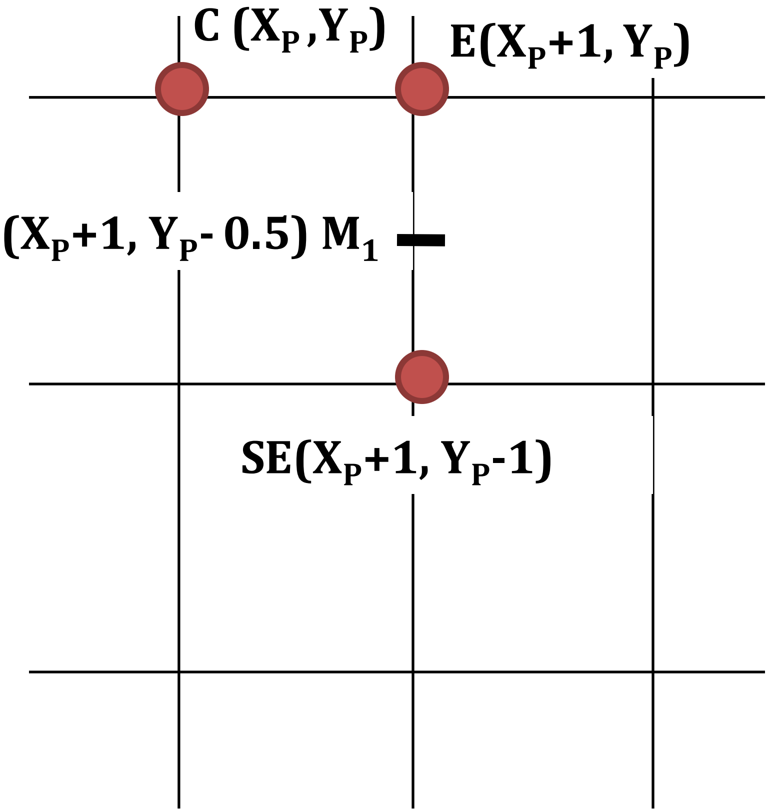
Bresenham's Mid Point Criteria : Successive Updating (for selecting SE)

$$d_1 = F(M_1)$$
$$= F(X_P+1, Y_P - 0.5)$$
$$= (X_P+1)^2 + (Y_P - 0.5)^2 - R^2$$



Bresenham's Mid Point Criteria : Successive Updating (for selecting SE)

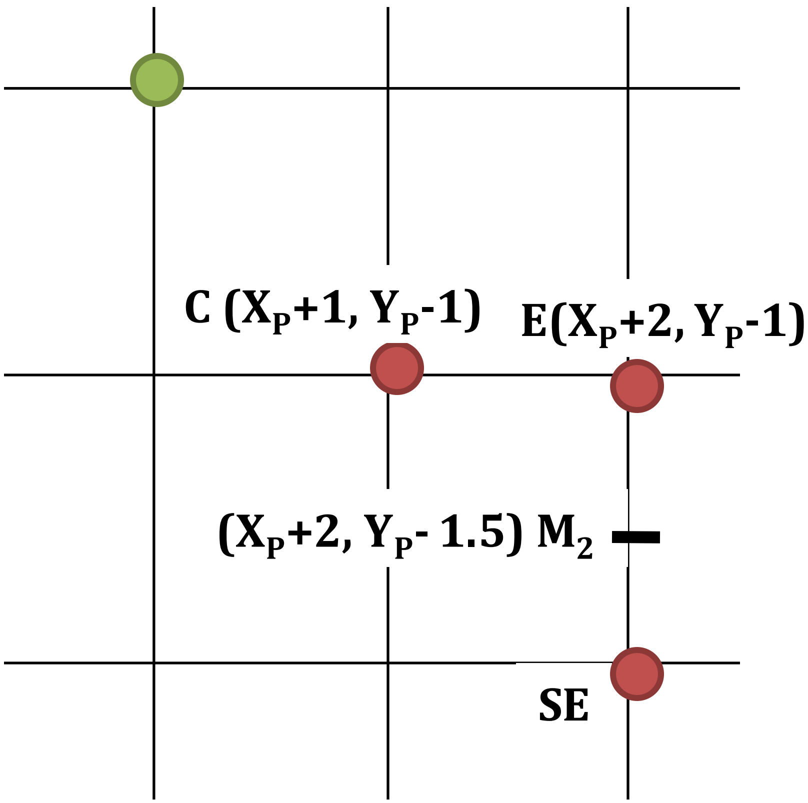
$$d_1 = F(M_1)$$
$$= F(X_P+1, Y_P - 0.5)$$
$$= (X_P+1)^2 + (Y_P - 0.5)^2 - R^2$$



If $d_1 >= 0$, SE $(X_P = X_P+1, Y_P-1)$

Bresenham's Mid Point Criteria : Successive Updating (for selecting SE)

$$\mathbf{d_1 = F(M_1)}$$
$$\mathbf{= F(X_P+1, Y_P- 0.5)}$$
$$\mathbf{= (X_P+1)^2 + (Y_P- 0.5)^2 - R^2}$$

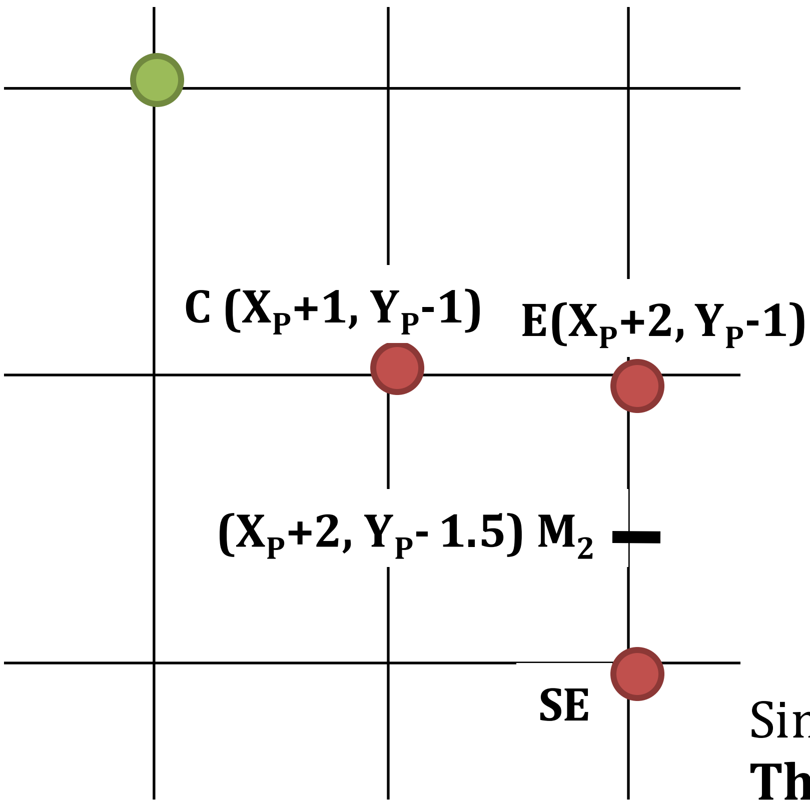


If $\mathbf{d_1 >= 0}$, **SE** $\mathbf{(X_P=X_P+1, Y_P-1)}$

$$\mathbf{d_2 = F(M_2)}$$
$$= F(X_P+2, Y_P-1.5)$$
$$= (X_P+2)^2 + (Y_P- 1.5)^2 - R^2$$
$$= X_P^2+ 4X_P + 4 + Y_P^2 - 3Y_P + 2.25 - R^2$$
$$= X_P^2+ 2X_P + 1 + Y_P^2 - 1Y_P + 0.25 - R^2 + 2X_P - 2Y_P + 5$$
$$= (X_P^2+ 2X_P + 1) + (Y_P^2 - 1Y_P + 0.5^2) - R^2 + 2X_P - 2Y_P + 5$$
$$= \mathbf{d_1 + (2X_P - 2Y_P + 5)}$$

Bresenham's Mid Point Criteria : Successive Updating (for selecting SE)

$$d_1 = F(M_1)$$
$$= F(X_P+1, Y_P - 0.5)$$
$$= (X_P+1)^2 + (Y_P - 0.5)^2 - R^2$$



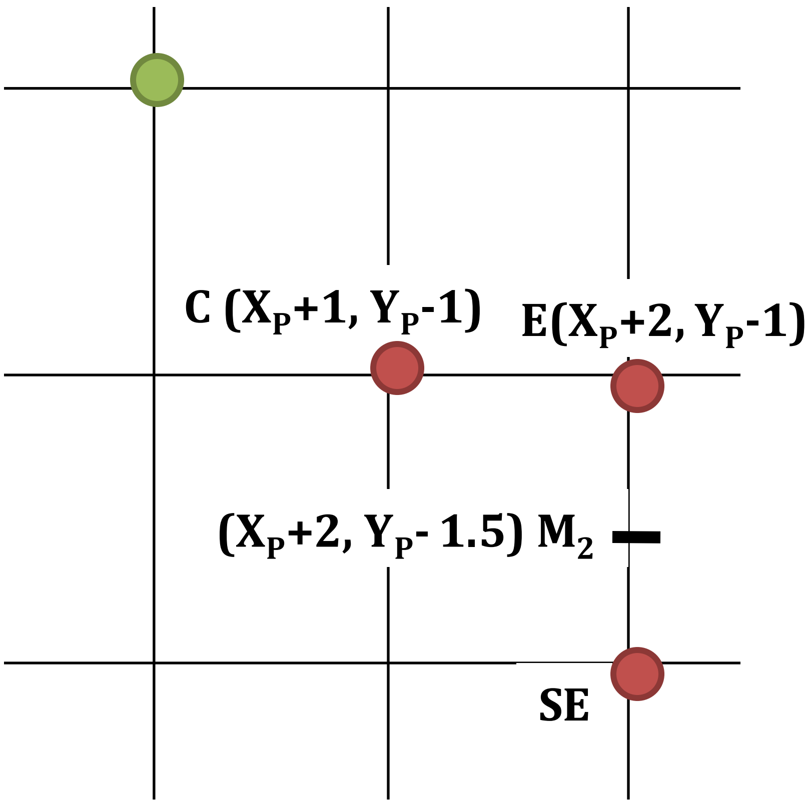
If $d_1 >= 0$, SE $(X_P=X_P+1, Y_P-1)$

$$d_2 = F(M_2)$$
$$= F(X_P+2, Y_P-1.5)$$
$$= (X_P+2)^2 + (Y_P - 1.5)^2 - R^2$$
$$= X_P^2 + 4X_P + 4 + Y_P^2 - 3Y_P + 2.25 - R^2$$
$$= X_P^2 + 2X_P + 1 + Y_P^2 - 1Y_P + 0.25 - R^2 + 2X_P - 2Y_P + 5$$
$$= (X_P^2 + 2X_P + 1) + (Y_P^2 - 1Y_P + 0.5^2) - R^2 + 2X_P - 2Y_P + 5$$
$$= d_1 + (2X_P - 2Y_P + 5)$$

Similarly, If $d_2 >= 0$, SE $(X_P=X_P+1, Y_P-1)$

Then $d_3 = d_2 + (2X_P - 2Y_P + 5)$

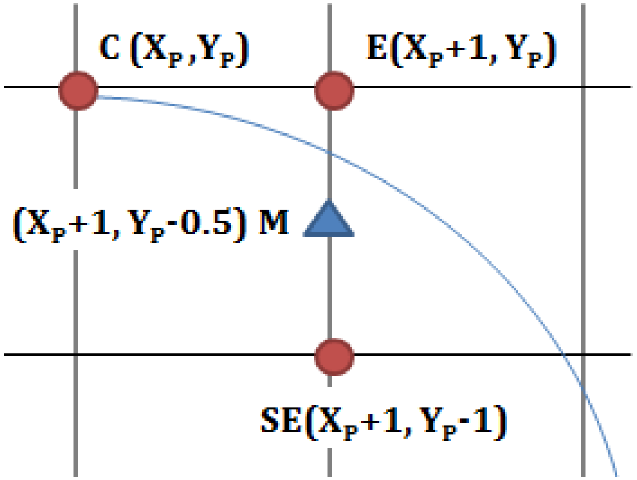
Bresenham's Mid Point Criteria : Successive Updating (for selecting SE)



Every iteration after **selecting SE**, we can successively update our decision variable with-

$$d_{NEW} = d_{OLD} + (2X_P - 2Y_P + 5)$$

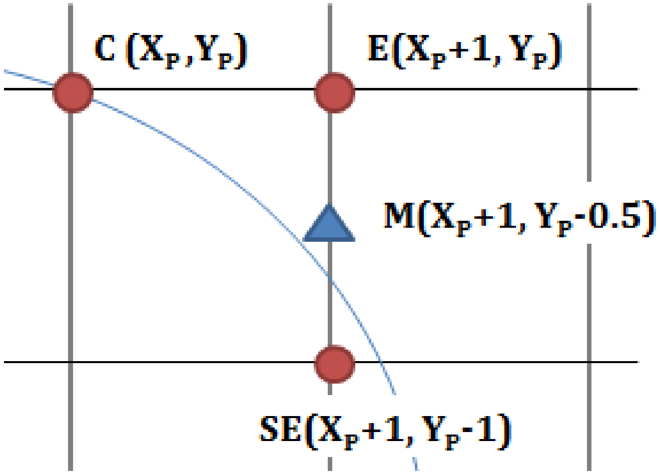
Bresenham's Mid Point Criteria : Successive Updating (summary)



If **d < 0**, then midpoint M is inside the circle, E is closer to the circumference,
So, **E** is selected and do-

d = d + ΔE

Where, $\Delta E = 2X_P + 3$

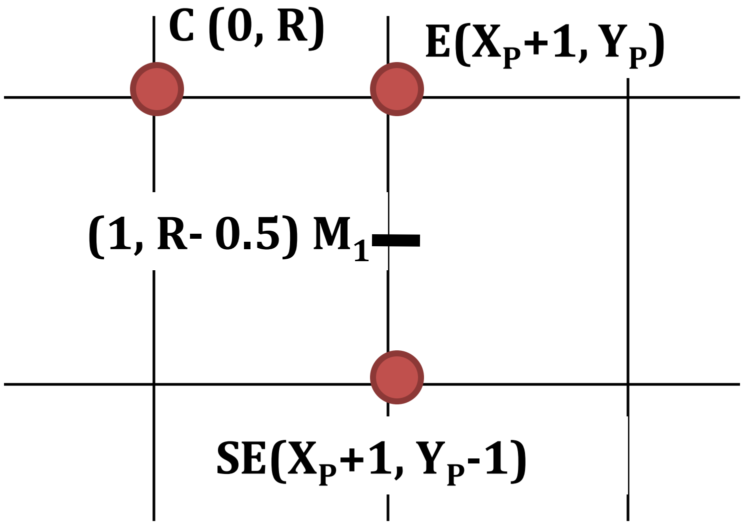


If **d >= 0**, then midpoint M is outside the circle, SE is closer to the circumference,
So, **SE** is selected and do-

d = d + ΔSE

Where, $\Delta SE = 2X_P - 2Y_P + 5$

Initialization



$$
\begin{aligned}
d_{INIT} &= F(M_1) \\
&= F(1, R-0.5) \\
&= (1)^2 + (R - 0.5)^2 - R^2 \\
&= 1 + R^2 - R + 0.25 - R^2 \\
&= 1.25 - R
\end{aligned}
$$

Initialization

We get, d = 1.25 – R

Lets say, h = d – 0.25
= 1.25 – R – 0.25
h = 1 – R

'h' is our new decision variable.
so -

d = 0	**h = -0.25**
d > 0	**h > -0.25**
d < 0	**h < -0.25**

Initialization

We get, d = 1.25 – R

Lets say, h = d – 0.25
= 1.25 – R – 0.25
h = 1 – R

'h' is our new decision variable.
so -

d = 0	**h = -0.25**
d > 0	**h > -0.25**
d < 0	**h < -0.25**

For, new decision variable 'h', it will be checked whether it is greater than or less than 0.25, rather than 0

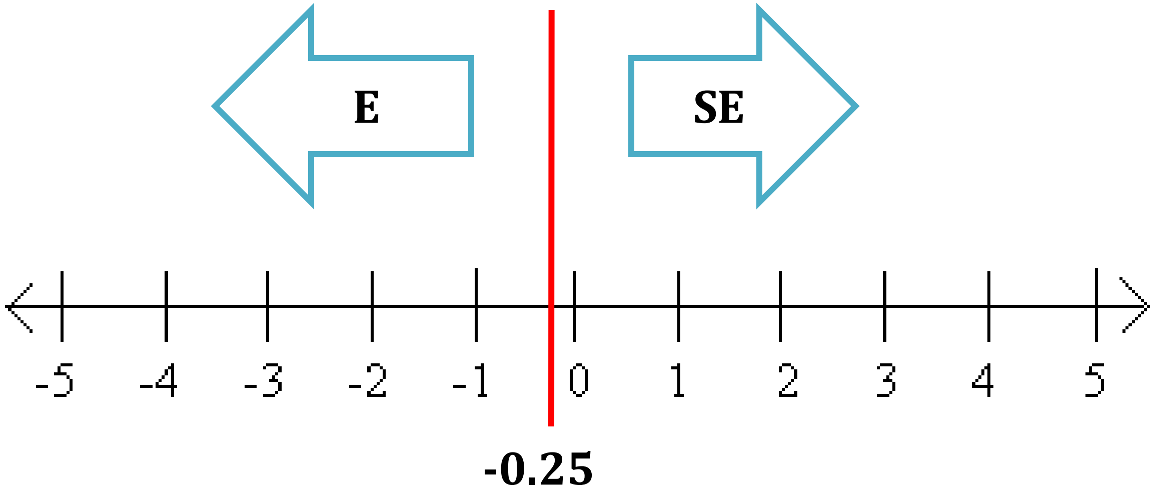
$h_{INIT} = 1 - R$

If **h < -0.25**, then **E** is selected, **h = h + ΔE**

If **h >= -0.25**, then **SE** is selected, **h = h + ΔSE**

Since **h** starts out with an **integer** value and is **incremented** by integer value (**ΔE or ΔSE**), we can change the comparison to just **h < 0**

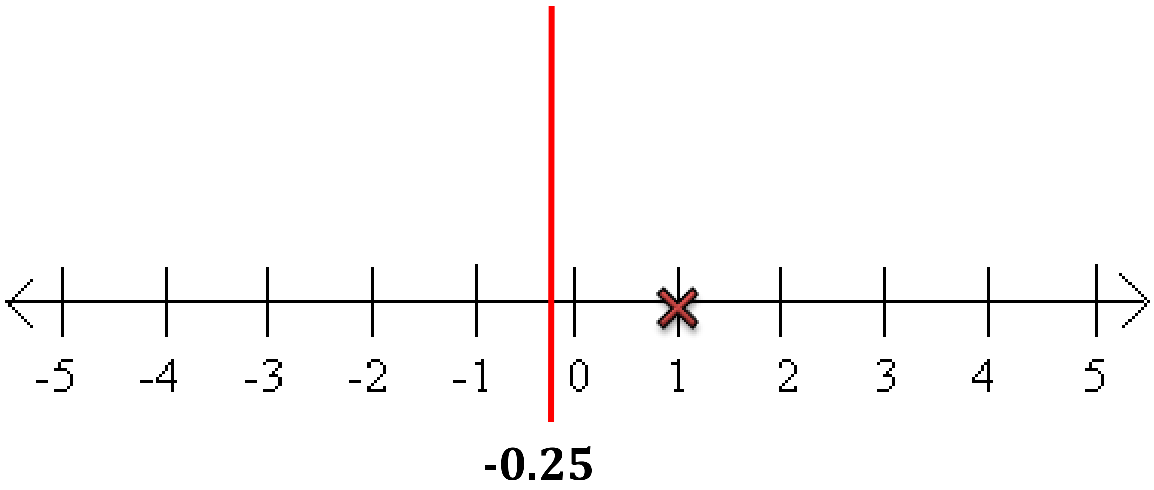
Comparing h with 0



– 0.25 is the threshold.

Comparing h with 0

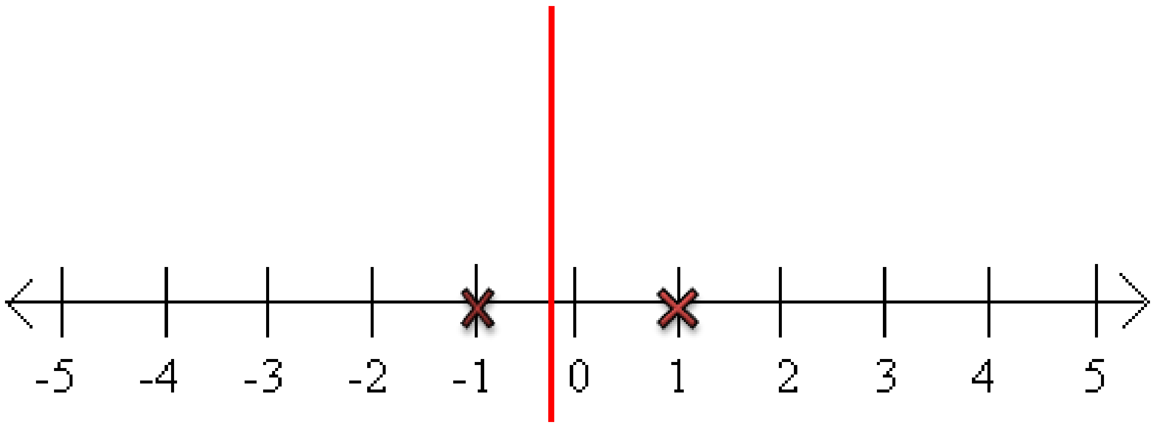
Let, h = – 2,
Δ = 3
h = – 2 + Δ
= – 2 + 3
= 1 > – 0.25
Select SE



– 0.25 is the threshold.

Comparing h with 0

Let, h = – 2,
Δ = 3
h = – 2 + Δ
= – 2 + 3
= 1 > – 0.25
Select SE



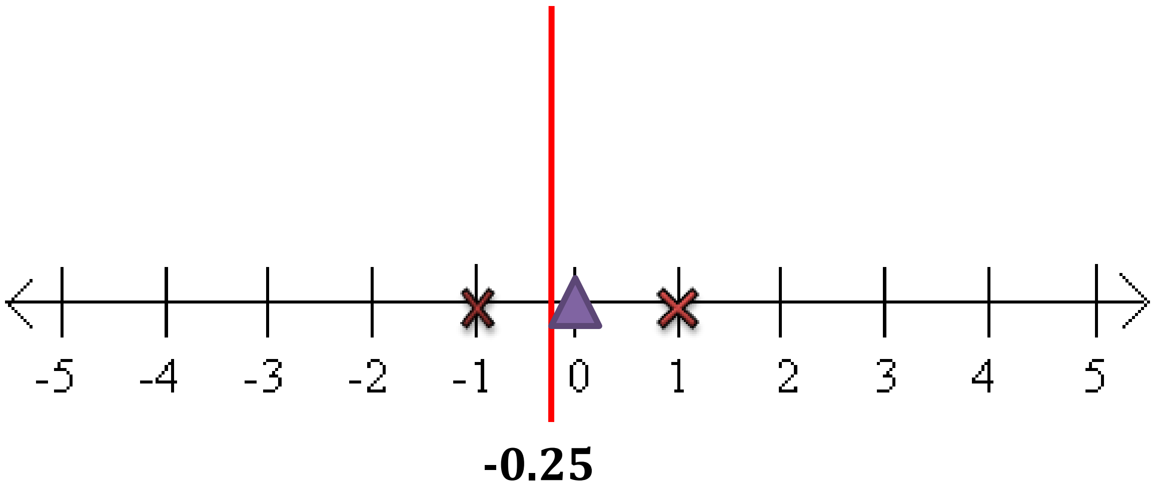
-0.25

Let, h = – 2,
Δ = 1
h = – 2 + Δ
= – 2 + 1
= – 1 < – 0.25
Select E

– 0.25 is the threshold.

Comparing h with 0

Let, h = – 2,
Δ = 3
h = – 2 + Δ
= – 2 + 3
= 1 > – 0.25
Select SE

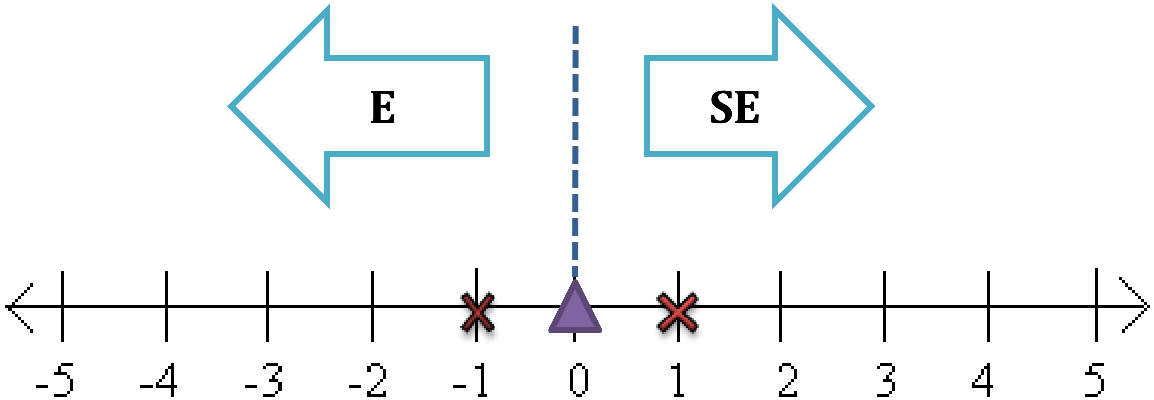


Let, h = – 2,
Δ = 1
h = – 2 + Δ
= – 2 + 1
= – 1 < – 0.25
Select E

In every case, the decision will remain same if we determine 0 as threshold, rather than – 0.25

Comparing h with 0

Let, h = – 2,
$\Delta = 3$
$h = -2 + \Delta$
$= -2 + 3$
$= 1 > 0$
Select SE



-0.25

Let, h = – 2,
$\Delta = 1$
$h = -2 + \Delta$
$= -2 + 1$
$= -1 < 0$
Select E

In every case, the decision will remain same if we determine 0 as threshold, rather than – 0.25

Comparing h with 0

So, finally.....

$\mathbf{h_{INIT} = 1 - R}$

If $\mathbf{h < 0}$, then **E** is selected**,** $\mathbf{h = h + \Delta E}$

If $\mathbf{h >= 0}$, then **SE** is selected, $\mathbf{h = h + \Delta SE}$

Where, $\Delta E = 2X_P + 3$
$\Delta SE = 2X_P - 2Y_P + 5$

Algorithm

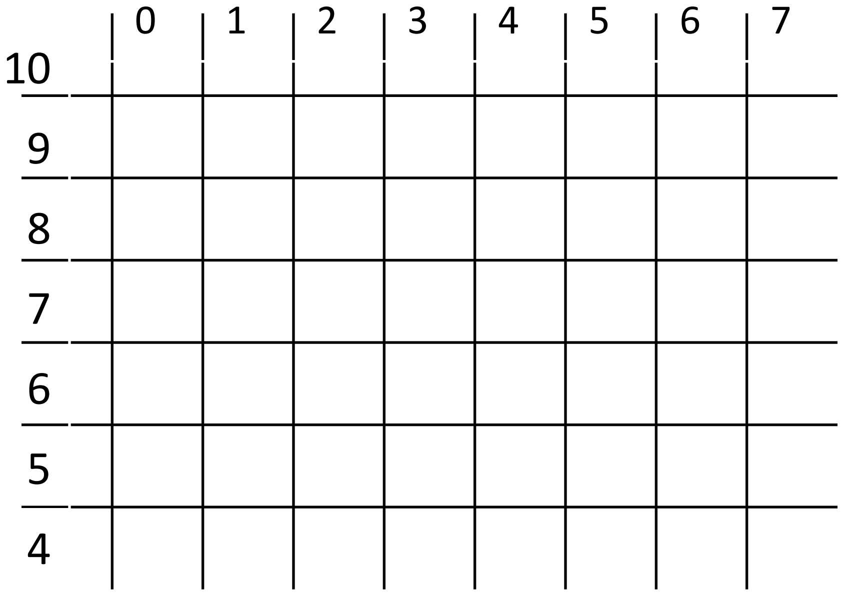
```
void MidpointCircle(int radius, int value)
{
        int x = 0;
        int y = radius ;
        int h = 1 – radius ;
        CirclePoints(x, y, value);
        while (y > x) {
        if (h < 0) {       /* Select E */
        h = h + 2 * x + 3; }
        else {       /* Select SE */
         h = h + 2 * ( x –  y ) + 5;
        y = y – 1; }
        x = x + 1;
        CirclePoints(x, y);
        }
}
```

Algorithm

```
void MidpointCircle(int radius, int value)
{
        int x = 0;
        int y = radius ;
        int h = 1 – radius ;
        CirclePoints(x, y, value);
        while (y > x) {
        if (h < 0) {      /* Select E */
        h = h + 2 * x + 3; }
        else {      /* Select SE */
         h = h + 2 * ( x –  y ) + 5;
        y = y – 1; }
        x = x + 1;
        CirclePoints(x, y);
        }
}
```

```
CirclePoints (x,y)
        Plotpoint(x,y) ;
        Plotpoint (x,-y) ;
        Plotpoint(-x,y) ;
        Plotpoint(-x, -y) ;
        Plotpoint(y,x) ;
        Plotpoint(y, -x) ;
        Plotpoint(-y, x) ;
        Plotpoint( -y, -x) ;
end
```

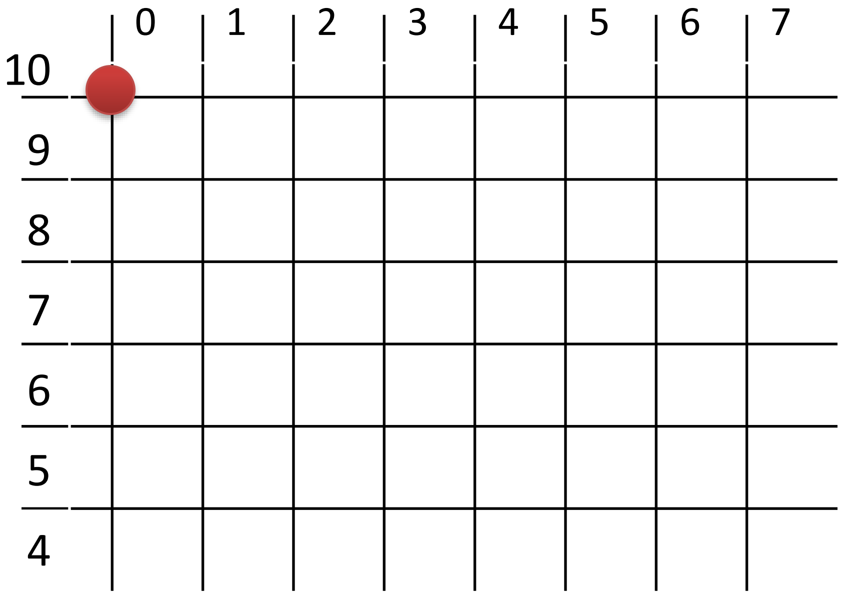
Example



Given:

Radius , R = 10

Example



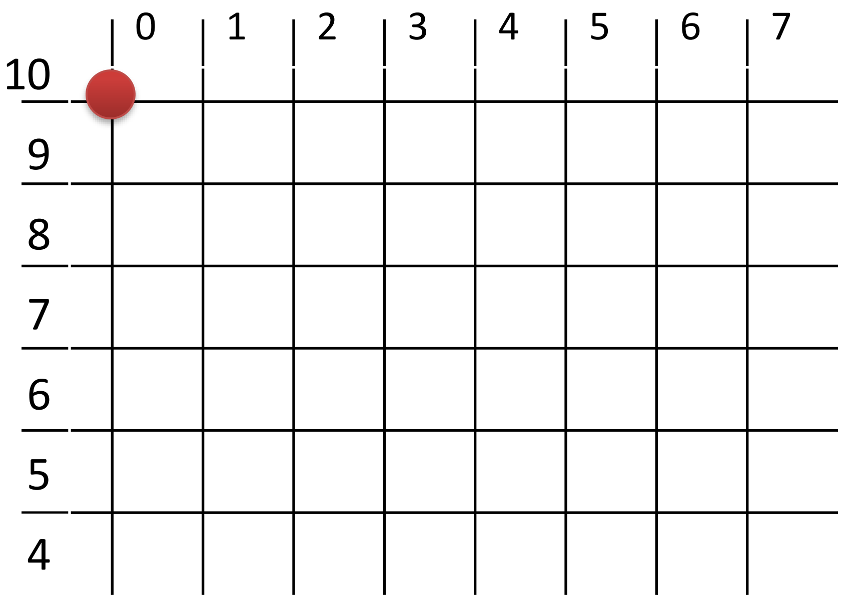
<u>Given:</u>

Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

Example



Given:

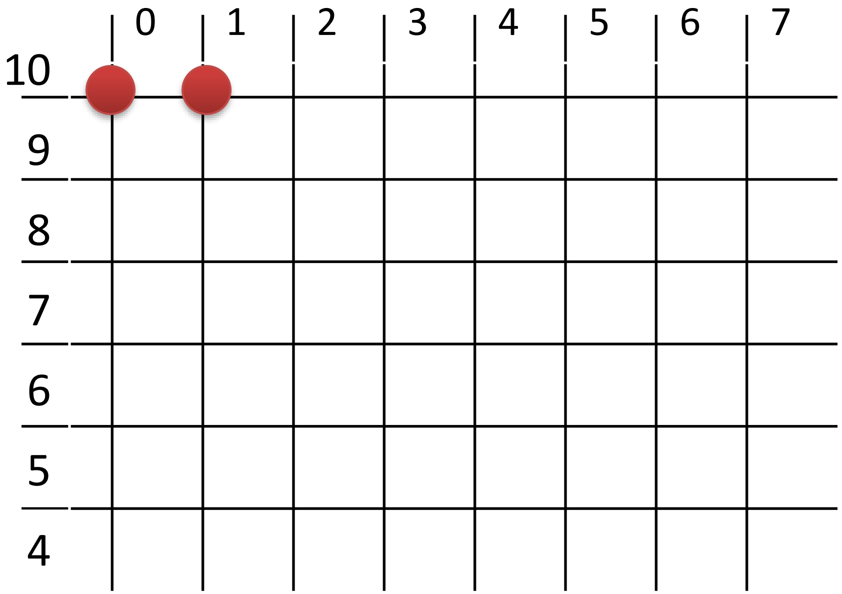
Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

K	1						
2x	0						
2y	20						
h							
(x, y)							

Example



Given:

Radius , R = 10

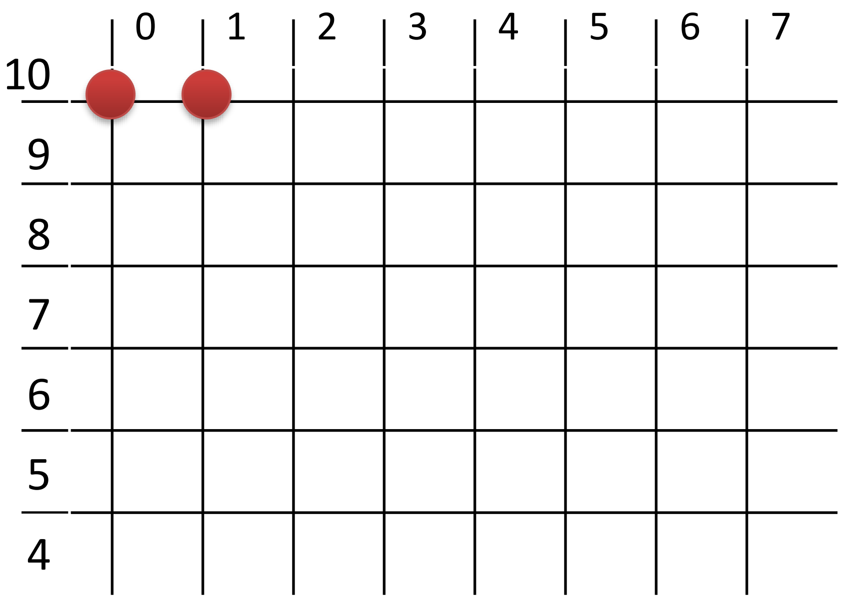
(x,y)=(0,10)

h = 1 – R = -9

K	1						
2x	0						
2y	20						
h							
(x, y)	E(1,10)						

h <= 0, E

Example



Given:

Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

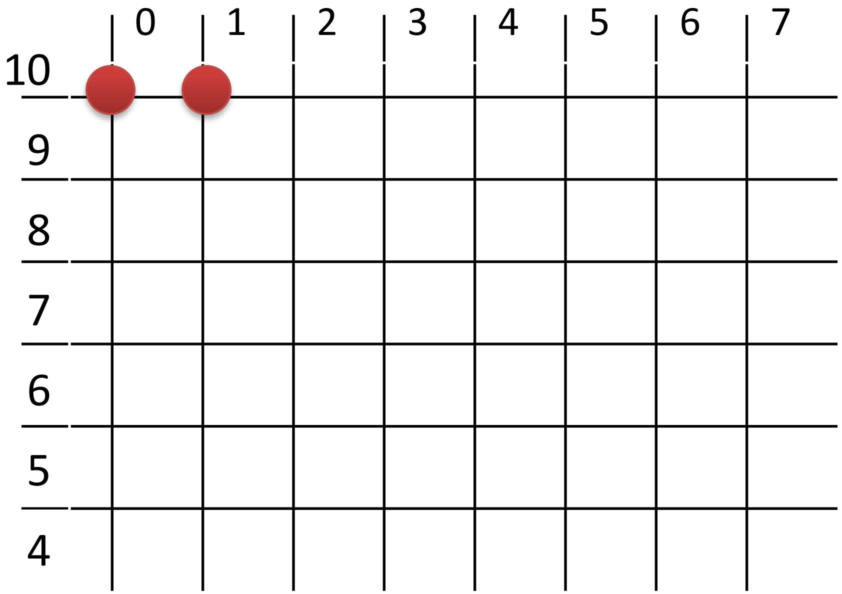
h = h+ΔE = h+2x+3

=-9+0+3

=-6

K	1						
2x	0						
2y	20						
h	-6						
(x, y)	E(1,10)						

Example



Given:

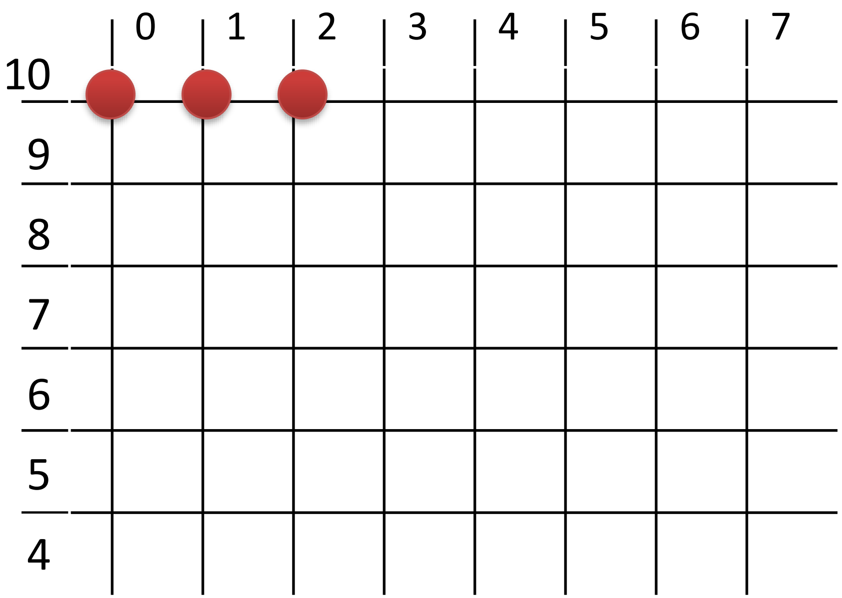
Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

K	1	2					
2x	0	2					
2y	20	20					
h	-6						
(x, y)	E(1,10)						

Example



Given:

Radius , R = 10

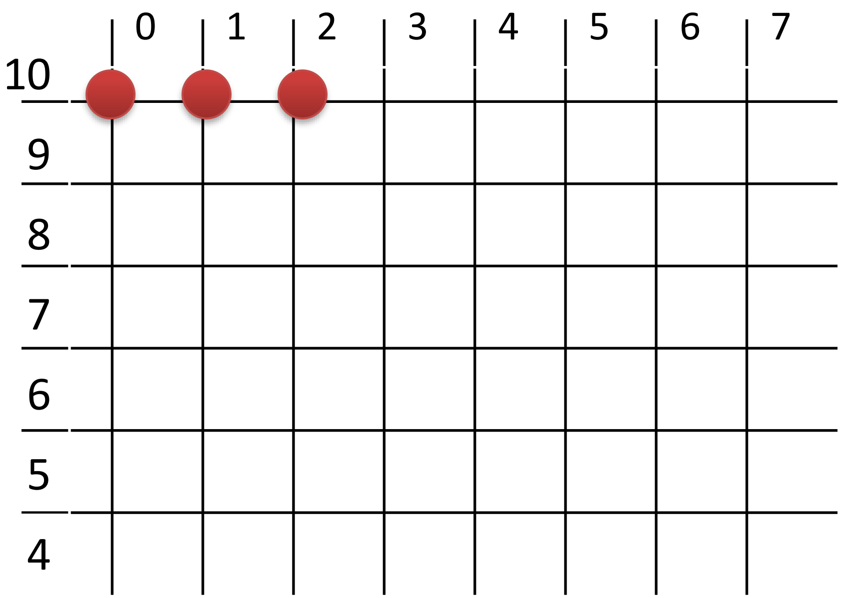
(x,y)=(0,10)

h = 1 – R = -9

K	1	2					
2x	0	2					
2y	20	20					
h	-6						
(x, y)	E(1,10)	E(2,10)					

h <= 0, E

Example



Given:

Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

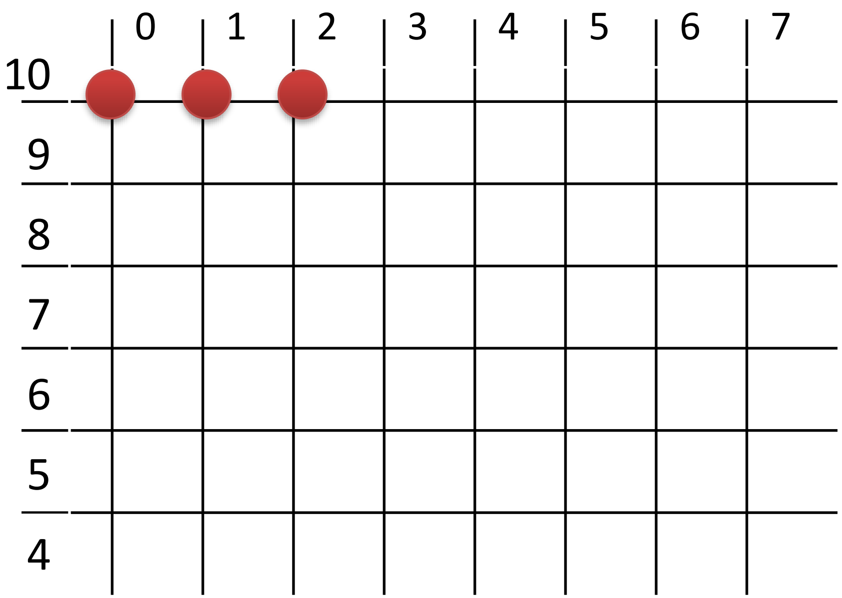
h = h+ΔE = h+2x+3

=-6+2+3

=-1

K	1	2					
2x	0	2					
2y	20	20					
h	-6	-1					
(x, y)	E(1,10)	E(2,10)					

Example



Given:

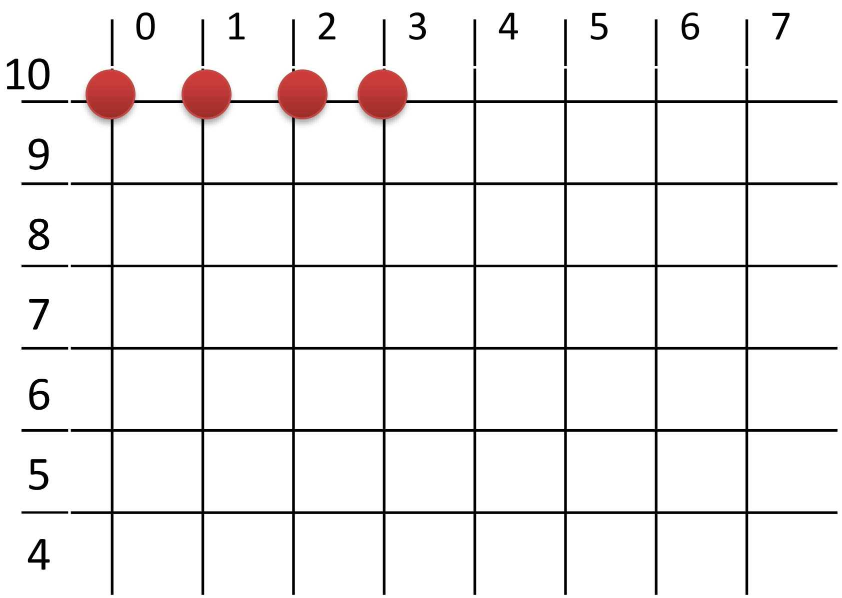
Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

K	1	2	3				
2x	0	2	4				
2y	20	20	20				
h	-6	-1					
(x, y)	E(1,10)	E(2,10)					

Example



Given:

Radius , R = 10

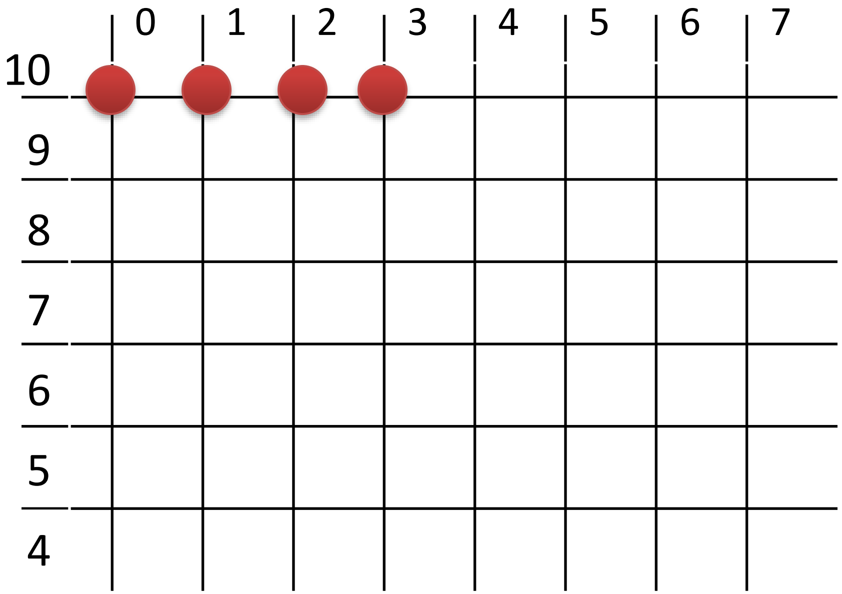
(x,y)=(0,10)

h = 1 – R = -9

K	**1**	**2**	**3**				
2x	0	2	4				
2y	20	20	20				
h	-6	-1					
(x, y)	E(1,10)	E(2,10)	E(3,10)				

h <= 0, E

Example



Given:

Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

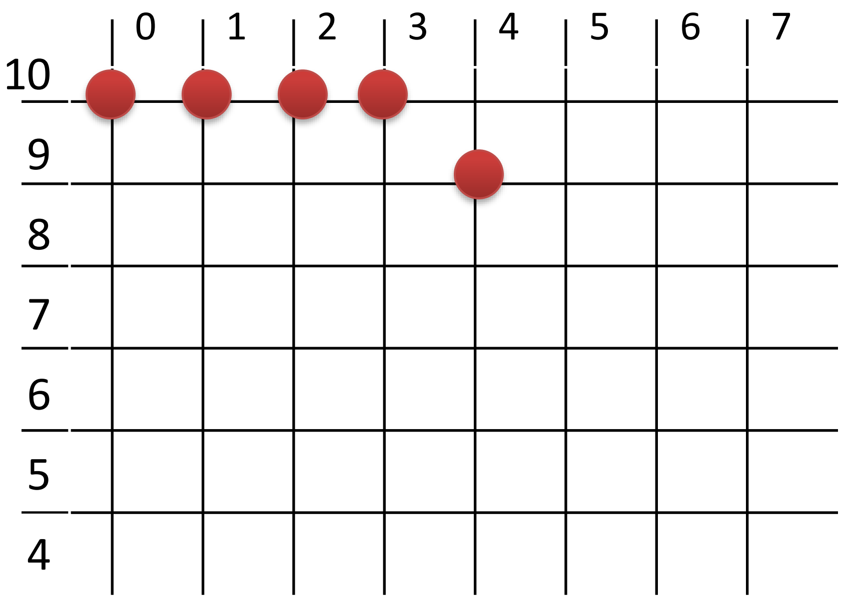
h = h+ΔE = h+2x+3

=-1+4+3

=6

K	**1**	**2**	**3**				
2x	0	2	4				
2y	20	20	20				
h	-6	-1	6				
(x, y)	E(1,10)	E(2,10)	E(3,10)				

Example



Given:

Radius , R = 10

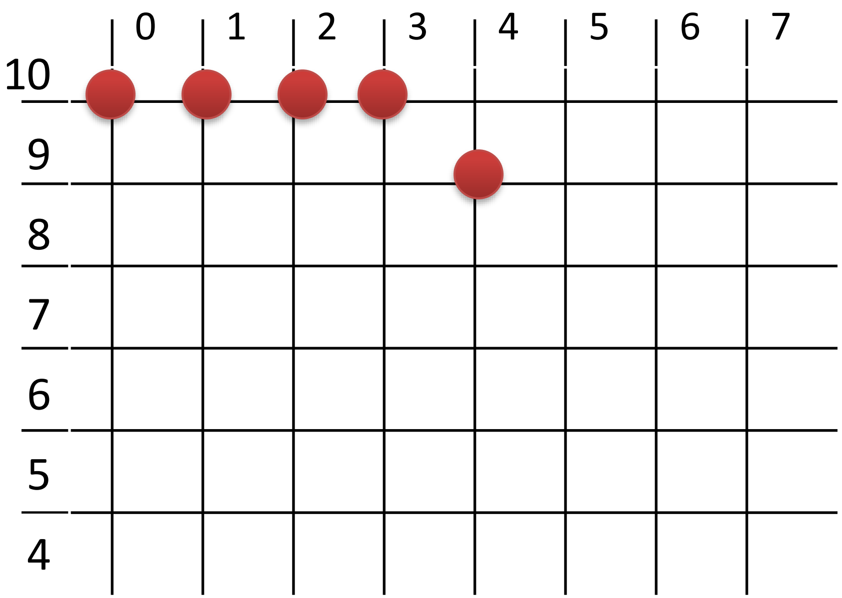
(x,y)=(0,10)

h = 1 – R = -9

K	**1**	**2**	**3**	**4**			
2x	0	2	4	6			
2y	20	20	20	20			
h	-6	-1	6				
(x, y)	E(1,10)	E(2,10)	E(3,10)	S(4,9)			

h > 0, SE

Example



<u>Given:</u>

Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

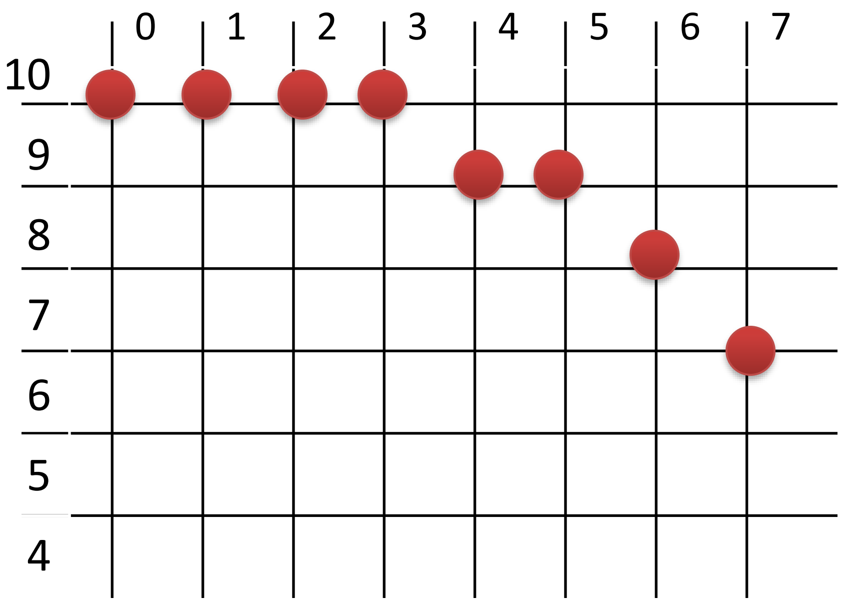
h = h+ΔSE = h+2x-2y+5

=6+6-20+5

=-3

K	1	2	3	4			
2x	0	2	4	6			
2y	20	20	20	20			
h	-6	-1	6	-3			
(x, y)	E(1,10)	E(2,10)	E(3,10)	S(4,9)			

Example



Given:

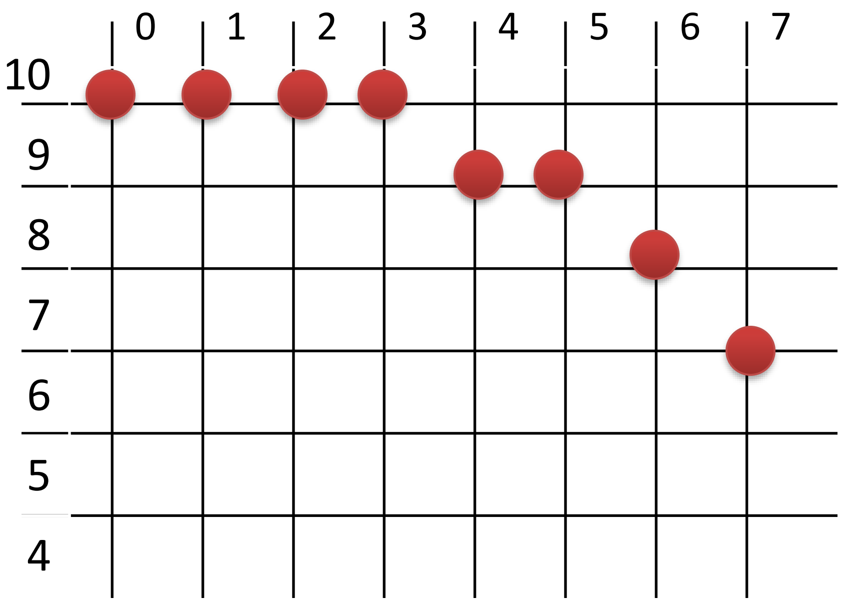
Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

K	1	2	3	4	5	6	7
2x	0	2	4	6	8	10	12
2y	20	20	20	20	18	18	16
h	-6	-1	6	-3	8	5	6
(x, y)	E(1,10)	E(2,10)	E(3,10)	S(4,9)	E(5,9)	S(6,8)	S(7,7)

Example



Given:

Radius , R = 10

(x,y)=(0,10)

h = 1 – R = -9

Untill y > x

K	1	2	3	4	5	6	7
2x	0	2	4	6	8	10	12
2y	20	20	20	20	18	18	16
h	-6	-1	6	-3	8	5	6
(x, y)	E(1,10)	E(2,10)	E(3,10)	S(4,9)	E(5,9)	S(6,8)	S(7,7)