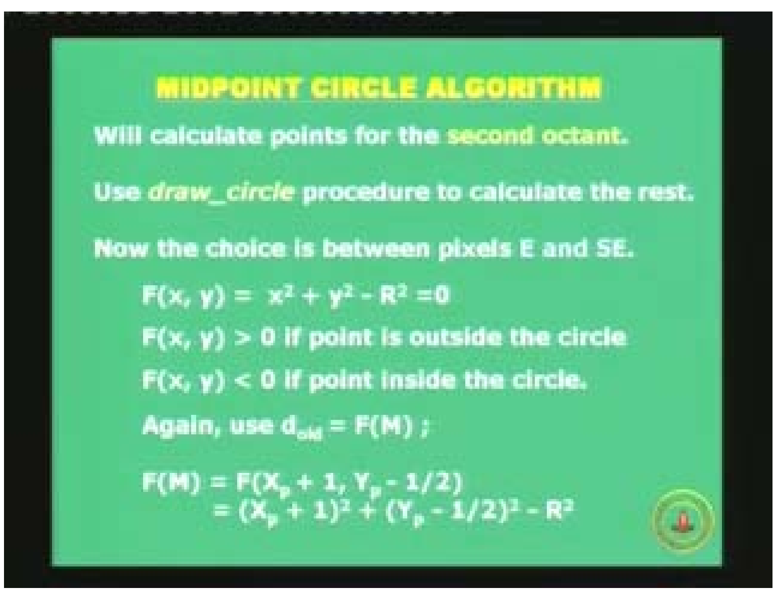
Computer Graphics
Prof. Sukhendu Das
Dept. of Computer Science and Engineering
Indian Institute of Technology, Madras
Lecture - 16
Scan Converting Lines, Circles and Ellipses

Hello and welcome you all once again to the lecture on computer graphics.
In the last few lectures we have been discussing about scan line algorithms, where we started with the method of drawing a straight line, from one point to another. Then, of course, we have moved to the case of the circle drawing algorithm and in both these cases, the criteria which was used has been the Bresenham's midpoint criteria. So, basically you evaluate the midpoint with respect to the line of the curve in the case of a circle and find out whether we have to select one of the two pixels. This method gives us a very fast integer based algorithms to select points in a discretised or digitized environment.

Towards the end of the last class, we have just finished the steps, the equations, and the mathematics related to the midpoints algorithm where we found out that the Increments were necessary based on the choice of the east and south east pixels. We move over to the corresponding pseudocode of the midpoint algorithm, but before that, to maintain continuity let us again look at those expressions of the equation of a circle and the corresponding Increment based on the choice of the midpoint criteria and the iterative loop. If we look back to the slide just I have rolled back the slide couple of slides back.

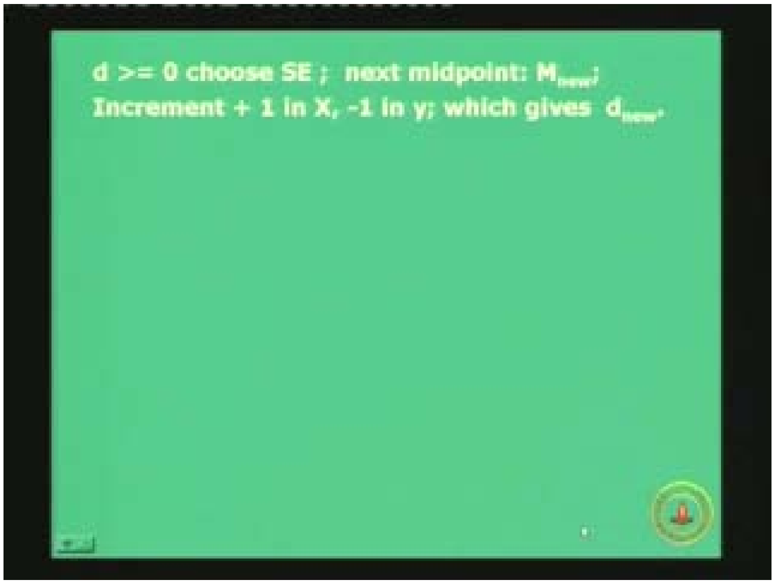
(Refer Slide Time: 00:02:39)



We are basically calculating points for the second octant. Draw a circle and proceed. Here, it takes care of all the other seven octants as we draw the circle and so when we are choosing points on the second octant, the choice is between east and the south east pixels. Between east and south east pixels, the equation of F(x, y) are given as x square plus y square minus R square is equal to 0. Thus, the expression, implicit form, and the midpoint criteria says that, if f is the function is positive then the point is outside the circle. If in the point, the value of the function is negative then the point is inside the circle. So, this is the midpoint criteria which we use to calculate and to find out whether the choice is east or south east.

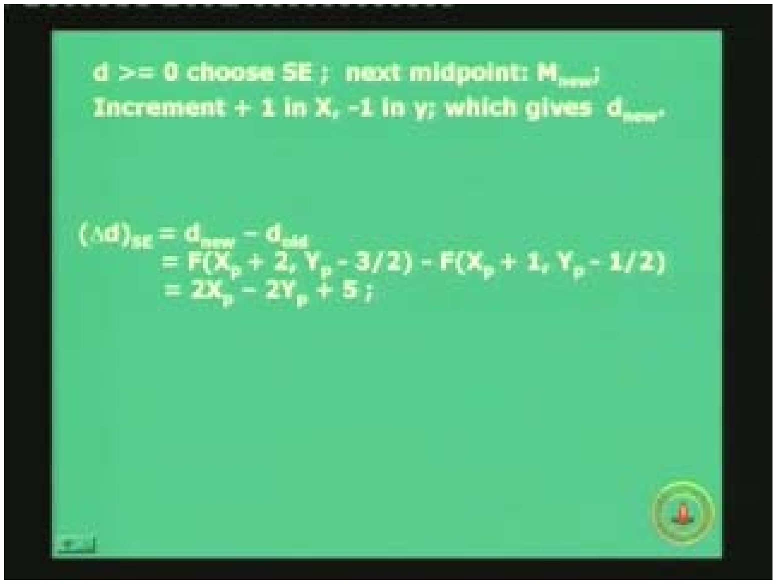
Remember, we also use the decision variable d old as a function of the midpoint at the next stage of iteration. Based on the current point xp comma yp, the functional value at the next midpoint will be f of xp plus 1, as given here at the bottom of your screen, you can see here, yp minus 1/2 of the y coordinates Incremental 1/2 down and 1 along the x direction to the right xp plus 1. So, if you break open the expression of f of m, you will get the expression given at the bottom of the screen, which we have also seen in the last class.

(Refer Slide Time: 00:04:08)



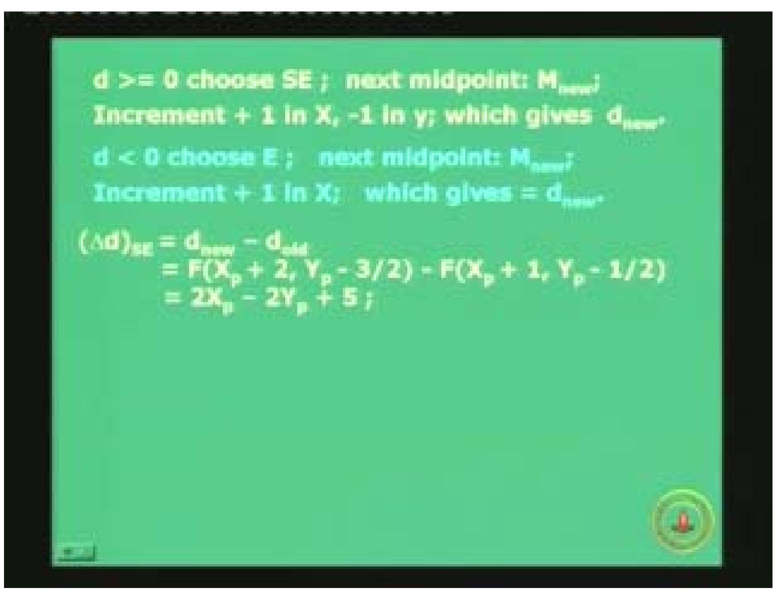
So, what was the choice for choosing the east or south east? We know that if the decision variable d is equal to 0 or positive, we chose south east, that is the next midpoint in new. It basically means, when we are choosing south east, the midpoint is above the line or the curve and that is why the curve passes through, closer to the south east point and so Increment positive 1 along x direction minus y along minus 1 along y direction which gives us d new. So, let us look at the expression of d new now.

(Refer Slide Time: 00:04:43)



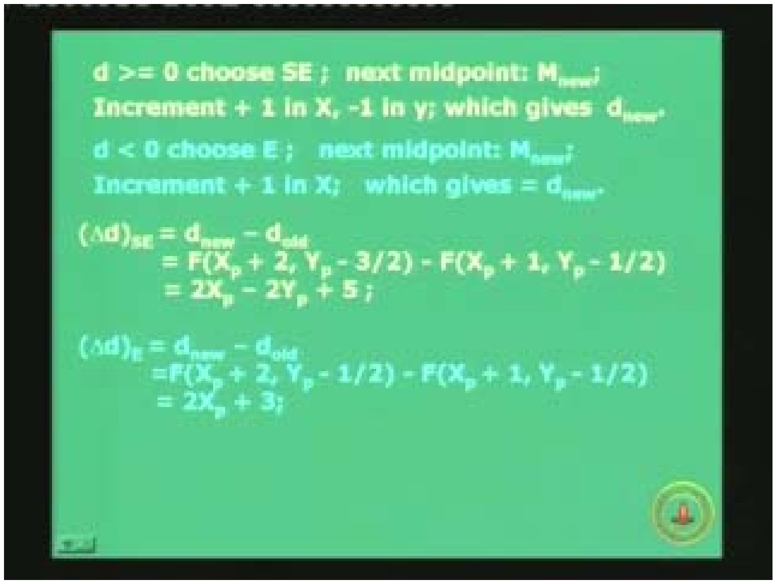
And also straight away, since we know how to substitute and get the value of d new, we look at the first order difference. When the choice is south east which is d new minus d old, the expression of d new is given here. Function of X_p plus 2 along for x coordinates and Y_p minus 1 and a ½, that is 3 by 2 in the y coordinate minus the d old which we have seen in the previous slide just now. So, d new minus d old will be given by these expressions, 2 into X_p minus Y_p or 2 X_p minus $2Y_p$ plus 5 is the Increment corresponding to a choice of the south east. Let us see if the decision variable is negative?

(Refer Slide Time: 00:05:22)



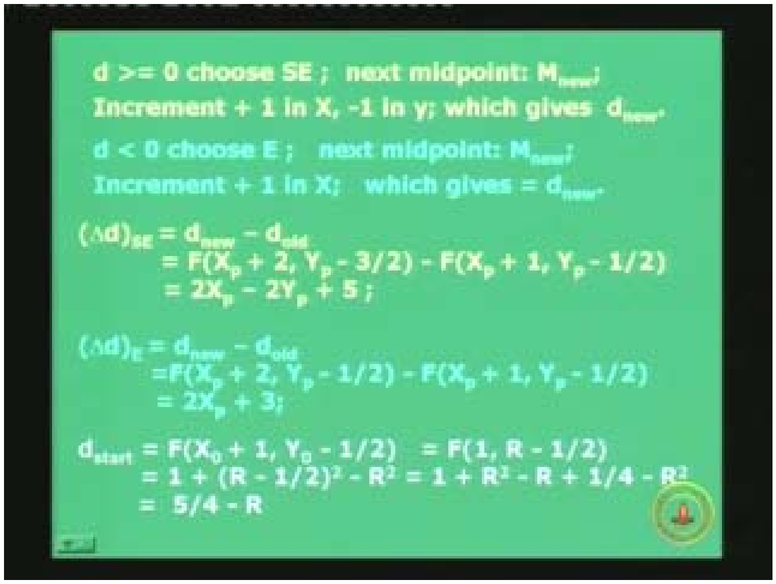
We have to choose east, the next point is m new which can be obtained by incrementing 1 along direction x which gives the d new.

(Refer Slide Time: 00:05:34)



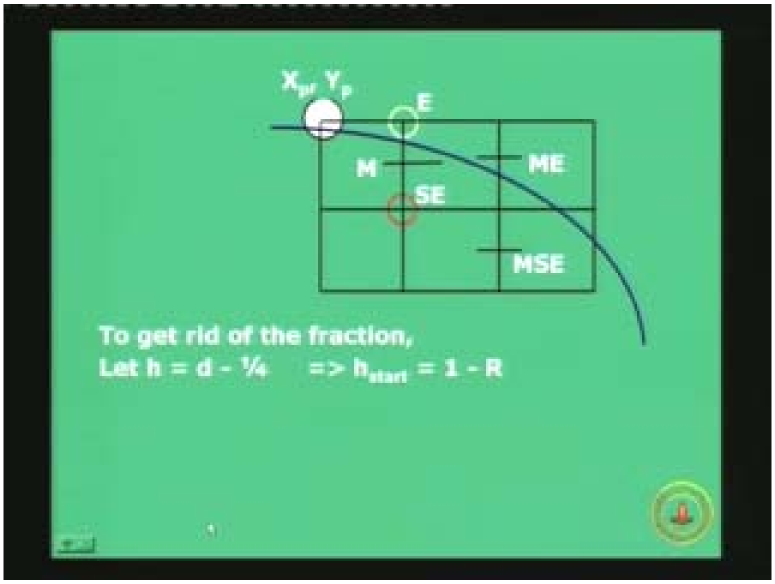
Let us find the delta d or the first order difference or the difference in d. If east is the choice, it is also the same like the previous case, d new minus d old where d new in this case will be function of X_p plus 2 and Y_p minus ½, instead of 3 by 2. For these, the choice is south east; since we are choosing east, it is Y_p minus 1/2 minus d old here, and that difference is given. You can calculate by yourself and it will be $2X_p$ plus 3 so that means if we are choosing east, you Increment the decision variable by value $2X_p$ plus 3, whereas if you choose south east you Increment by $2X_p$ minus $2Y_p$ plus 5. So, please note down these expressions because we are going to use these expressions with an example very soon, today.

(Refer Slide Time: 00:06:18)



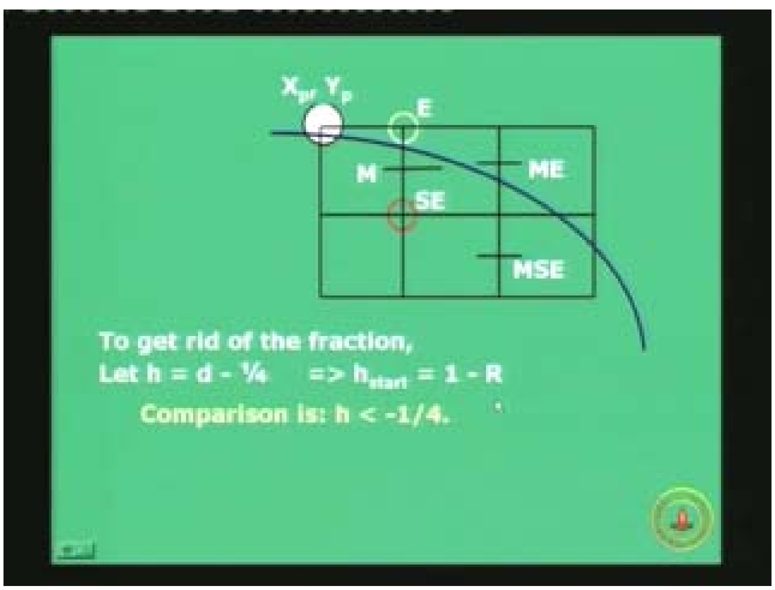
What should be the decision variable at the start of the iteration d? Start will be the first midpoint because the first point chosen is 0 comma R. So the next choice, the next midpoint will be plus 1 X_o Y_o is 1 is 0 and R and so it will be X_o plus 1, Y_o minus 1/2 and so it is F 1 comma R minus ½ because 0 comma R X_o is equal to 0, Y_0 is equal to R substitute. This is what you get of F, break open the expression for F, you get it as 5 by 4 minus R, 5 by 4 a fractional, so we got a fractional floating point to start with.

(Refer Slide Time: 00:06:56)



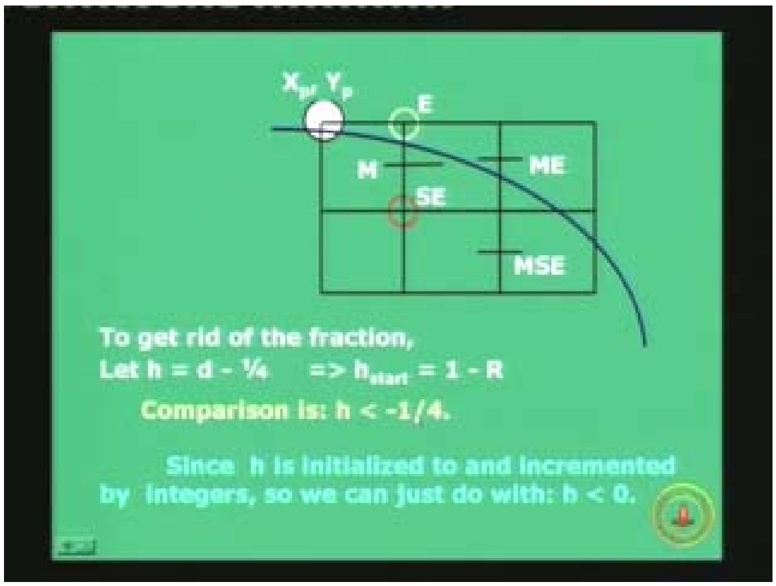
These diagrams typically show you the scenario at the start, when X_p Y_p has been chosen as the point of iteration at any stage and the curve is passing over the midpoint. So, the choice will be east if the curve passes below the midpoint the choice would have been south east and of course, if we have chosen east, we have to evaluate the next midpoint criteria at the m of east respectively. But, looking at the d start we eliminate the fraction by substituting a variable h is equal to d minus 1/4, so h of start will be 1 minus R which will be an integer value because we assumed at the beginning that R is an integer. Since R is an integer, h of start also will be an integer.

(Refer Slide Time: 00:07:41)



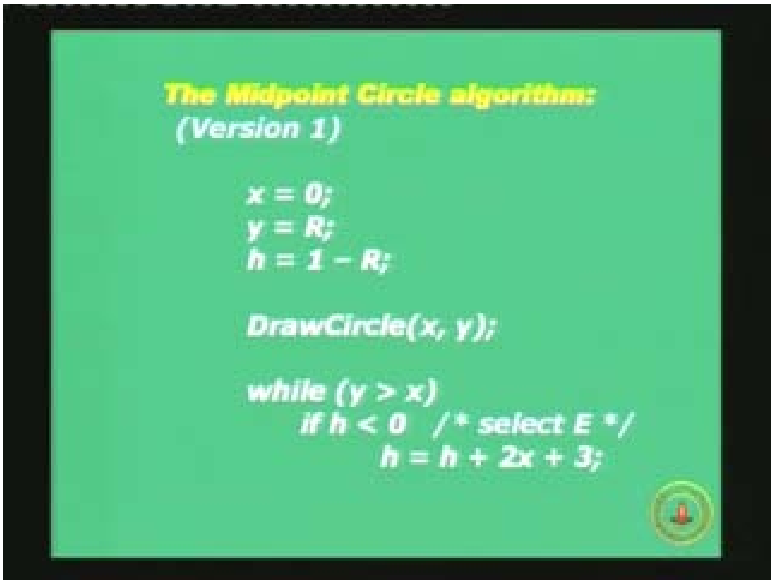
But the small problem is, the comparison because, if you are comparing a bipolar case d with positive or negative comparing with 0, if that is the case, you basically substitute d equal to 0. In this expression of h you will get h is equal to minus ¼, so you have to compare with a negative fractional number, but you do not worry about this because the last line that was given in the last lecture is, h is initialized to an integer and also incremented by an integer. So, we can always compare h with 0 instead of comparing with minus ¼.

(Refer Slide Time: 00:08:04)



So, where we stopped in the last class we had known. Now the expressions for d start or h start, we also know the expressions of the first order difference for the east, that is delta e or delta se for the south east case and all of these are now integers and in comparison also with respect to 0 positive or negative. You look at h instead of d and take a decision whether you move over to east or south east. So, this is the logic behind the midpoint criteria. Let us see the code behind it. Of course, we are not following any syntax in a particular language, but it could be a pseudocode. It should be a pseudocode. We can code that in Pascal or C depending upon the environment you have or C plus plus.

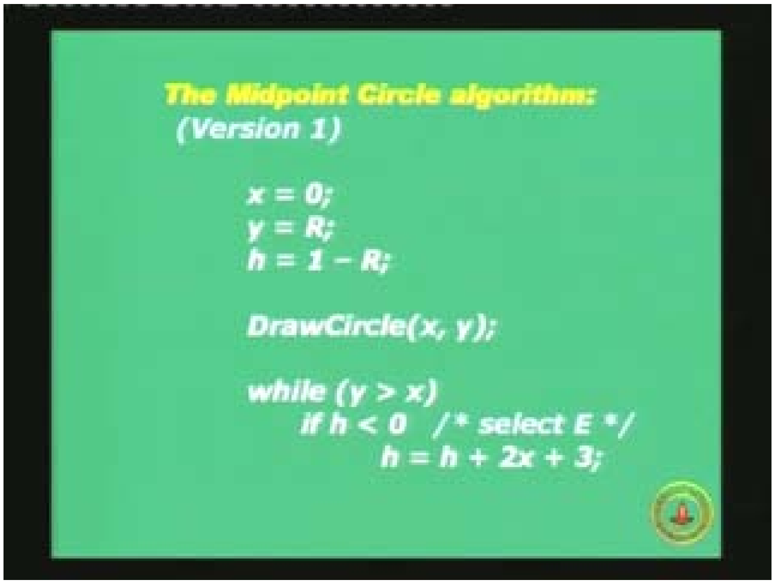
(Refer Slide Time: 00:09:03)



And let us look at the midpoint circle algorithm. Do not worry about the term version one for the time being. We will come with a version two which is [level] a sophisticated one very soon. So that is the initial condition. Put x equal to 0 y is equal to R. That is the initial condition. The other initialization is substitute h is equal to 1 minus R. We have seen that these are the three statements required for initializing. X equals 0 and Y equals R, that is the starting point and h start is 1 minus R and you first at the point itself at 0 comma R. That is, the drawcircle will not only draw the point at 0 comma R but it will also draw it in the other eight octants as well.

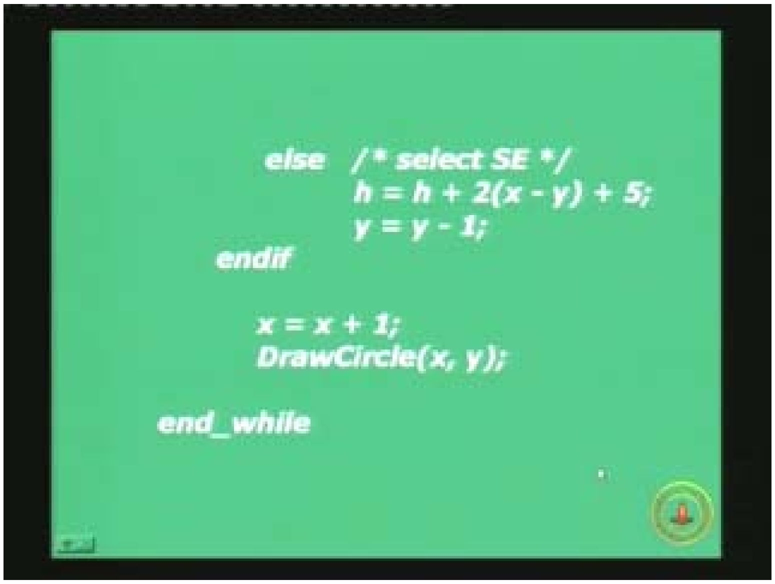
Drawcircle takes care of the other octants because we are only calculating the points in the second octant. So, all the seven other octants will be taken care of by the drawcircle command. It will not only plot the point at x comma y but also for the seven other points as well. Now, keep repeating this while loop as long as y is greater than x because when you reach a condition where y is equal to x or y is less than x, it means you have come out of the second octant and you have entered first octant. So, we start from the top 0 comma R and we keep moving towards east and south east typically. You will start selecting east initially and then south east in general. We will see with an example and you stop at the end of the second octant and you will suddenly have a condition when x is equal to y or x will be more than y, so keep doing this as long as x is less than y or y greater than x.

(Refer Slide Time: 00:10:35)

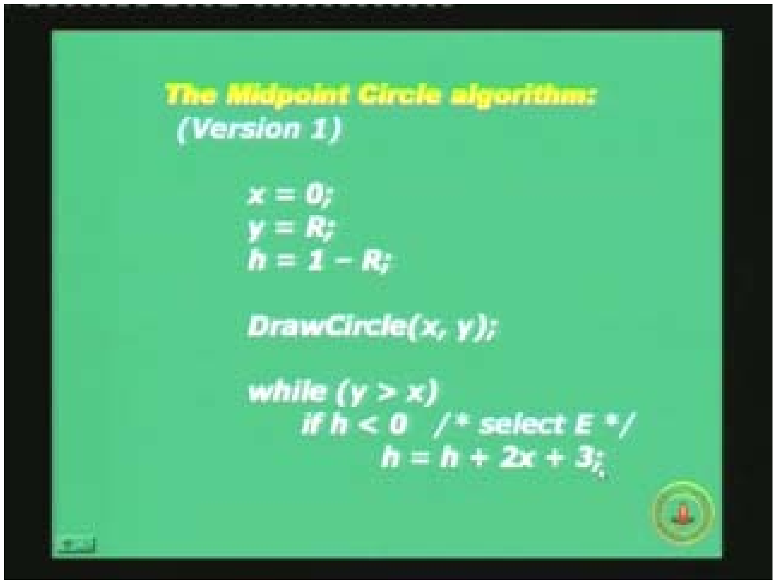


So come back to the code while y is greater than x. You look at their value of the decision variable in this case h instead of d compare it with 0. If it is negative select east and if you have selected east remember that the delta e or the increment for that in the case of east was 2x plus 3. So, it was 2x plus 3, so increment h by 2x plus 3. So, you see h is starting from integer which is 1 minus R and it is also going to be incremented by the integer 2x plus 3, we know that. Else the condition that means h as in the previous case h was negative.

(Refer Slide Time: 11:06)

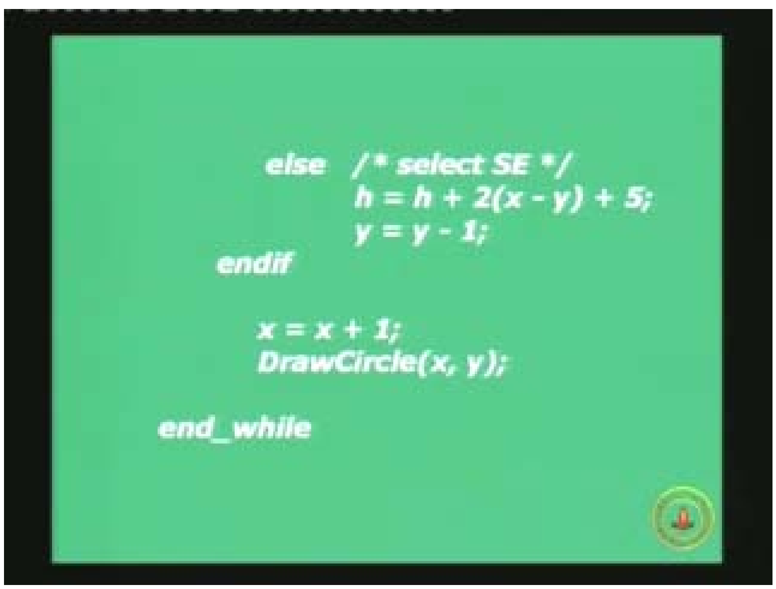


(Refer Slide Time: 00:11.10)



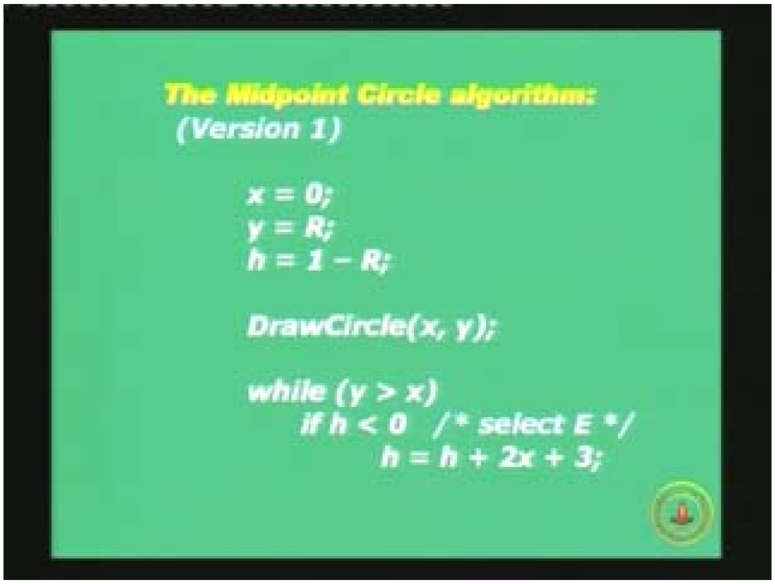
So, it is either equal to 0 or positive now and in this case select south east and if.

(Refer Slide Time: 00:11:13)



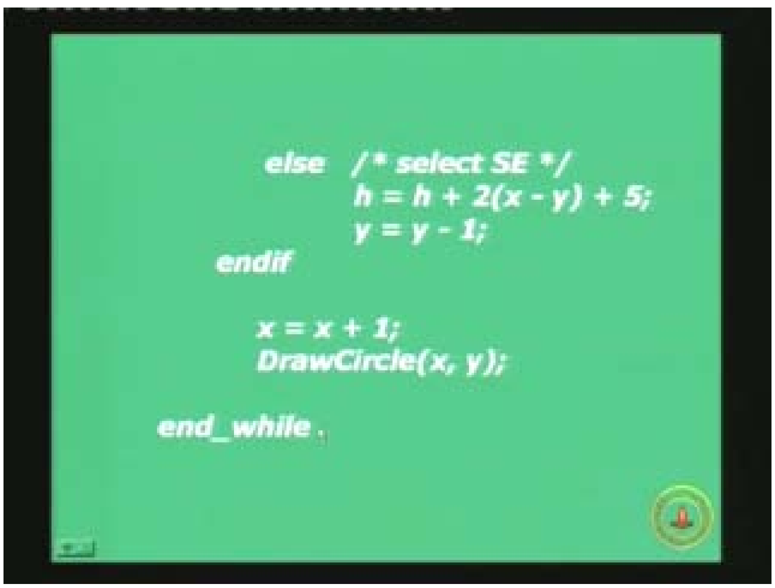
We all know that when we are selecting south east, the increment in the value of h will be 2 into x minus y or 2x minus 2y plus a 5 and in this case, since you are selecting south east, you decrement by y y 1 and the if condition ends here. Then based on the choice, based on the value of h we have selected east or south east and then you increment x by 1. You increment x by 1 and draw the circle and you keep doing this. And in while loop remember, let us go back to the previous slide.

(Refer Slide Time: 00:11:49)



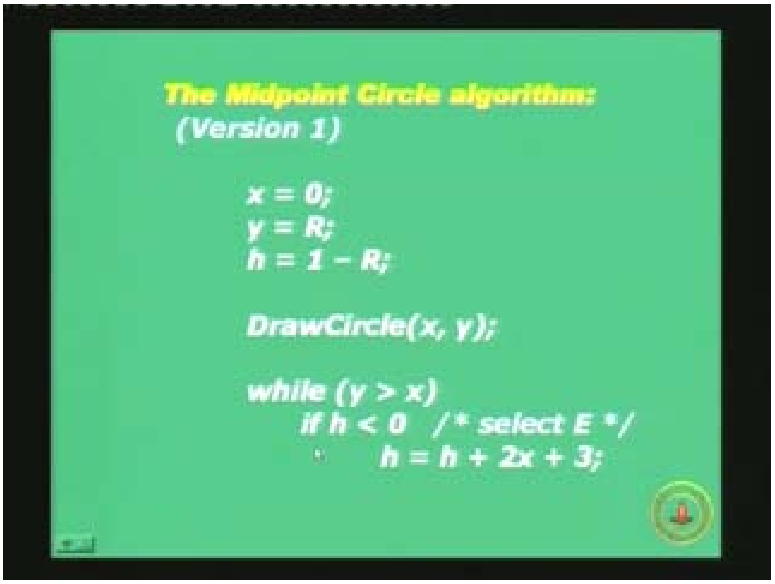
You started with the while loop when y was more than x, y was greater than x and the end and then of course, you had an if statement here.

(Refer Slide Time: 00:11:59)



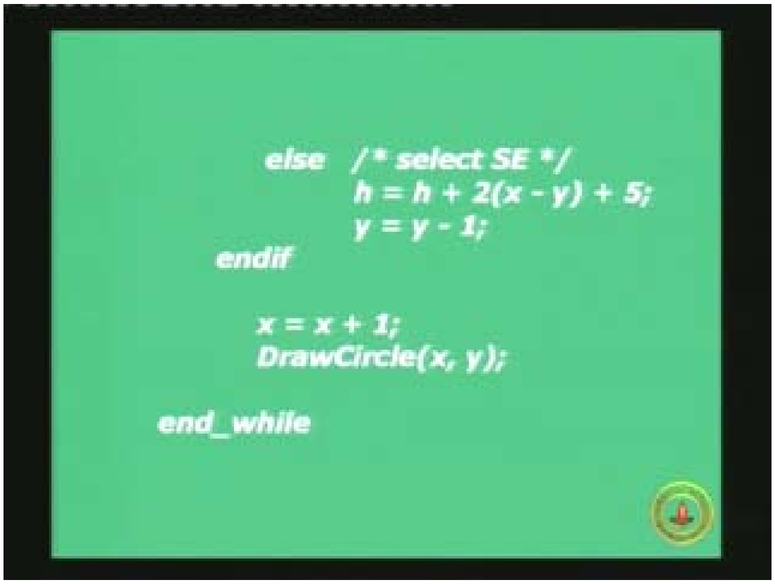
So, this endif and endwhile shows the condition as to when you will finish the function of if and when you will finish the function of y. Only remember, if the drawcircle commands, you draw all the eight points together, not only the point in the second octant but all the other seven octants as well.

(Refer Slide Time: 00:12:10)

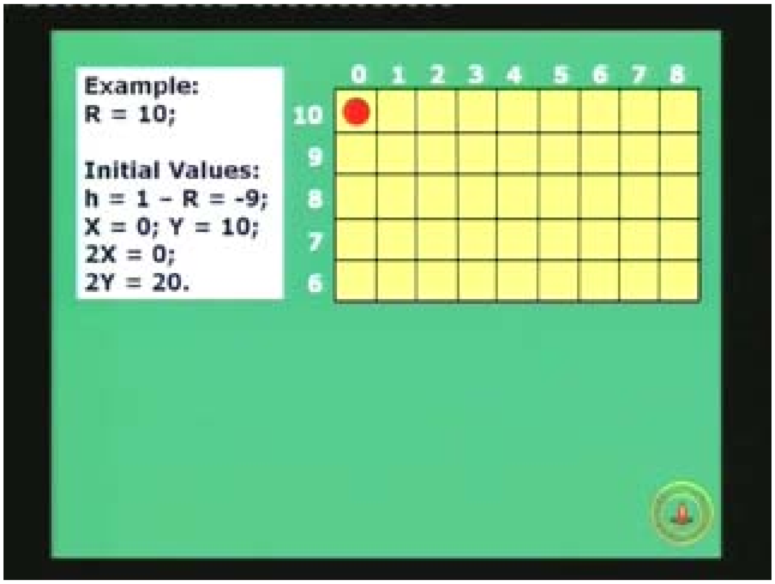


So, all the eight points in all the eight octants are drawn. The drawcircle command and you start with an initial value of h, which is an integer based on the decision variable h. You increment h by delta e or delta se which are also integers. So which will be an integer algorithm? It will be able to draw the circle for you. So let us take an example now.

(Refer Slide Time: 00:12:35)



(Refer Slide Time: 00:12:40)

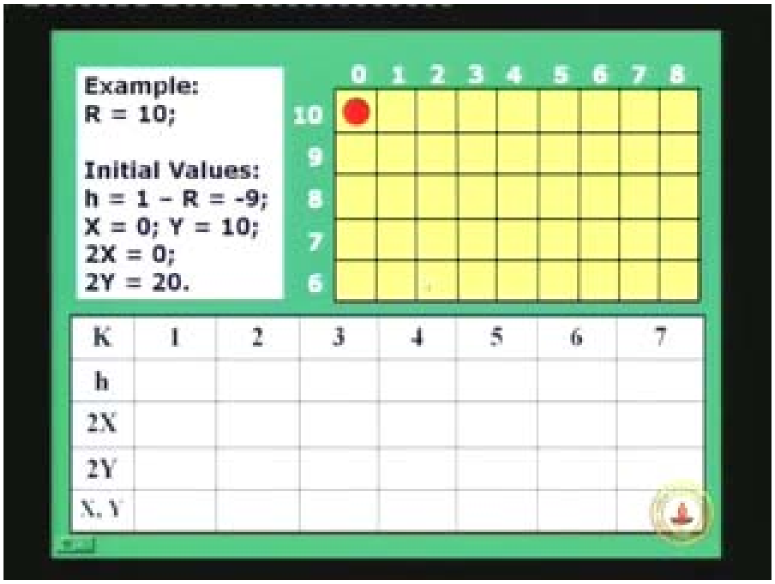


With this case we will take the first example today for drawing a circle and let us try to draw a circle with the center at 0 and the radius of the circle. Let us say it is R and I am only showing the points in the second octant. So, I have shown the plot of the graphics screen, the raster frame buffer or pixels on the screen. We have already plotted the first point at 0 comma 10.

The first point comes at 0 comma R. That is the first point in the second octant which is in the top. So, 0 comma 10 is already plotted as shown by the red circle. We will finish the initialization first. Direct initialization involves h which is 1 by R, 1 minus R since R is equal to 10. You will have the initial value starting of h, as minus 9 and the value of x is equal to 0. Initially, y is equal to R which is 10 and you remember, the increments of delta e and delta se typically requires a combination of 2 multiplied by the x coordinate or 2 multiplied by the y coordinate and that changes with time in each iteration loop.

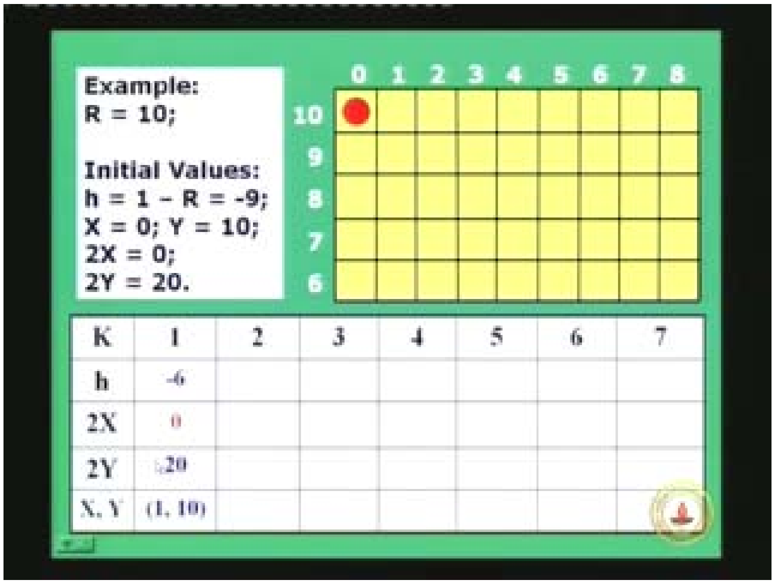
So, we start with 2X since x is equal to 0. Since x is equal to 0, 2X will also be equal to 0 and since y is equal to 10, 2Y also is hence equal to 20. You remember, the first point is plotted at 0 comma R which is 0 comma 10 and the drawcircle will also draw the other seven points in all the other seven octants. We only remember seeing this second octant part of the screen with the second octant. That is what we were viewing. Let us get into the first stage of the iteration.

(Refer Slide Time: 00:14:14)



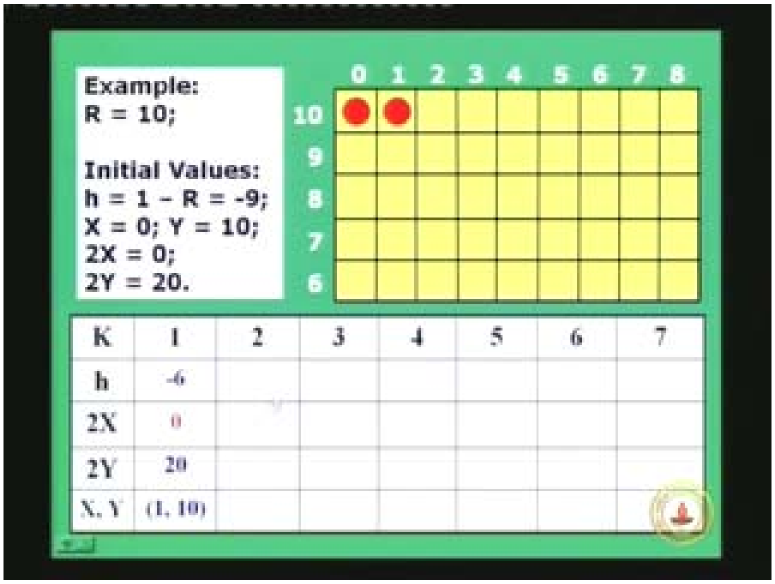
So this is the table which will show a stage by stage process of iteration. The K being the index of the iteration first, second and so on, h is the decision variable which will be incremented, 2X will be taken from the previous stage, x will be given 2X will be taken and 2Y. So, when you enter the first iteration, what happens is you first look at the value of h and decide what is going to be your next point. Since h is negative you remember the condition when h is negative what happens? You select the east. So automatically after 0, 10, the next point which will be selected is 1 comma 10 immediately to the right east pixel 1 comma 10. So you will see in the first iteration the value which will be written at the bottom of your screen. The last row will be 1 comma 10 and 2X, 2Y actually will be holding values from the previous iteration.

(Refer Slide Time: 00:15:08)



Let us see the values of the first iteration as we get in and select the point which is 1 comma 10 as selected here.

(Refer Slide Time: 00:15:13)

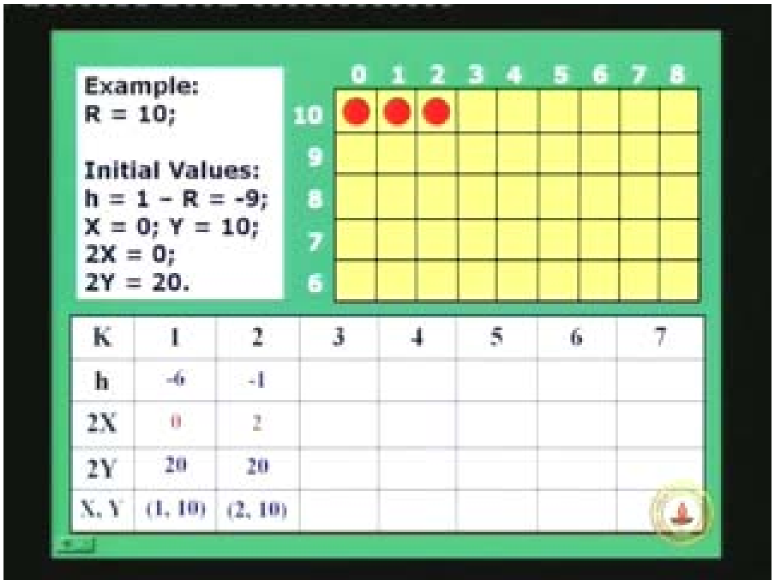


That is the point which will be selected and drawn after the combination. But the value of h will be incremented by minus n plus 2x plus 3. Remember, if that is the increment 2x plus 3, what is 2x 0? So, this will be incremented by the value 3. So for minus 9 plus 3 the h will hold the value 6. The h value of minus 6 will help you to select the next point not the 1 comma 10. The 1 comma 10 point which is the second point has been selected by the value h which is 9 minus 9 which helps you to select the first point that is 1 comma 10 as given in the bottom of your screen and also marked in red in the rasted screen within the grid here. Then h minus 6 actually will be used in the next iteration to choose the next point.

What is 2x and 2 y? Yes the 2x and 2y values have been selected from the current iteration based on the previous point 0 comma 10 at 0 and 20 as already given here. In fact 2x and 2y are computed within the loop. They are not part of the initial values. I remind you. Although it is given here, actually, it stops with selecting of h x and y only. So h is equal to minus 9x equal to 0, y equal to 10. The first three statements are initial values 2x and 2y, 0 and 20 are actually computed within the first stage of iteration. They then they pick up the older values 0 and 10 and 2x and 2y are used to compute the new value of h. So the new value of h becomes minus 6. I hope you would have followed the calculation. However, I repeat them again.

Here h was actually minus 9 and since you have selected east which is with coordinates 1 comma 10, the increment will be delta east. So h will be incremented by delta e which is 2x plus 3. Since it is 2x plus 3. The increment will be plus 3. So minus 9 plus 3 will give you minus 6. I hope you have written the expressions already in terms of what delta e and delta se and delta south east. So h minus 6 is also negative if h minus h is equal to minus 6 which is negative again. The next point selected will be the east. If it is so, you will basically select 2 comma 10.

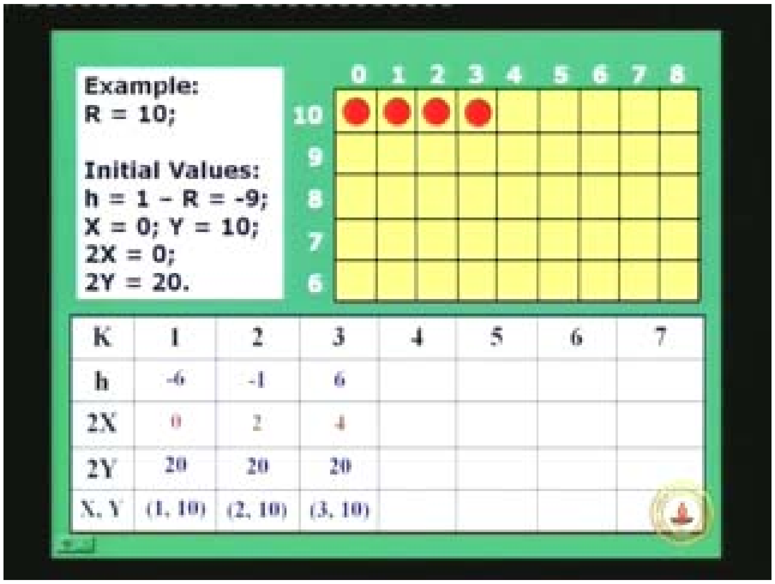
(Refer Slide Time: 00:17:21)



You will select 2 comma 10 as given by the second column of the table. The value at the last row is 2 comma 10. It is also plotted on the screen and then after that, let us see the values which will be updated. Since x is 2 and y is 10, 2x and 2y are easy to calculate. Let us see how you get h as minus 1.

What was h initially? Minus 6 and since we have selected east, the increment again will be 2x plus 3, where the value now of 2X is 2. This is the value which will be used to compute the new h which is 2 plus 3 equal to 5. So minus 6 plus 5 will give you minus 1. I repeat again, 2X plus 3, 2X is 2 plus 3 is 5. So, minus 6 plus 5 will give you minus 1. That is the new value of h which will take place at the end of the second iteration and the coordinates which will be selected is 2 comma 10. So, as you enter the third iteration, you use the second updated value at the second stage of iteration for h which is minus 1. It is still negative. So, you will be selecting the south east, I am sorry, the east pixel again because it is negative. So, the next pixel selected again will be 3 comma 10.

(Refer Slide Time: 00:18:34)

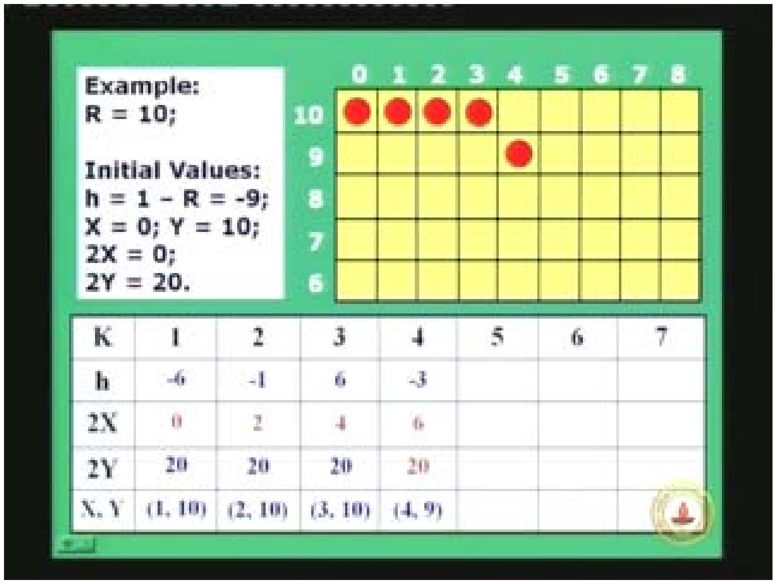


K	1	2	3	4	5	6	7
h	-6	-1	6				
2X	0	2	4				
2Y	20	20	20				
X, Y	(1, 10)	(2, 10)	(3, 10)				

As you can see here, this is because the h was minus 1 at the end of the second stage of iteration. So, let us enter the third stage of iteration 3 comma 10 that has been selected. It has also been marked here and based on the values of 2X and 2Y, what you typically have is 4 and 20, 2X and 2Y, because 1 comma 10 gave you 2 and 20, 2 comma 10. From the previous iteration before updating, you actually got 4 and 20.This is, delta east will be now computed or delta east south east will be computed. Based on these values you have to just follow the algorithm steps which I have given you earlier. Check the values with this table. Now, let us see how h has been updated.

Remember, at the third stage of iteration it is 3 comma 10. That means, you have actually selected east and since you have selected east you update the value of h which was minus 1 by delta east. What is delta east now? 2x plus 3 which is 4 plus 3 is 7, 7 minus 1 will give you 6. I hope you would have followed, as you have got 2 plus 3 which is 5. So, minus 6 plus 5 gives you minus 1 here, 4 plus 3 is 7, and when it is 7 minus 1 the remaining is 6. So, at the end of the third iteration, we have selected 3 comma 10. The new updated value of h is equal to 6, that is positive. That is the first change which occurs as h is equal to 6 and it is positive. When h is positive or 0, you basically select, what do you select? You select south east. So you select the south east pixel in the fourth iteration. Let us do that now. Let us enter the fourth iteration.

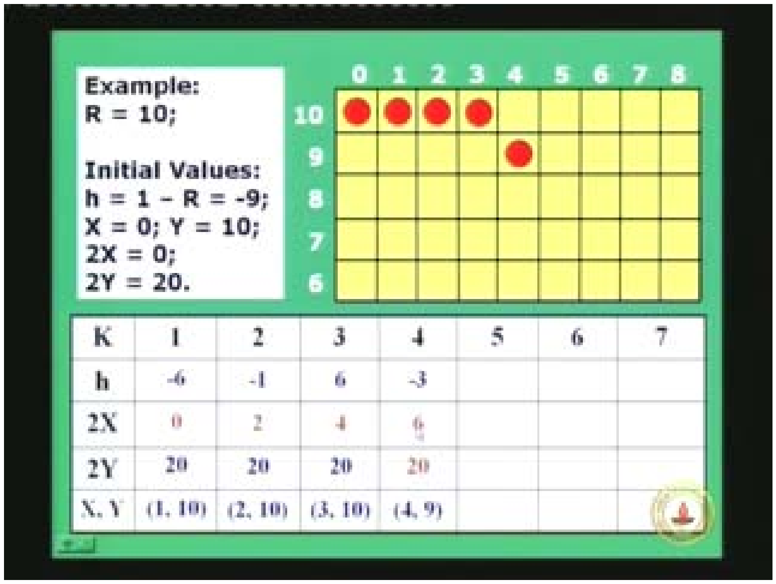
(Refer Slide Time: 00:20:16)



The X comma Y value after 3 comma 10 becomes 4 comma 9. You see, the new pixel here, the south east pixel selected coordinates given at the fourth column of the table, in the bottom row as 4 comma 9. So, what are 2X and 2Y? In fact, we actually update h and then select the pixel. So, 2X and 2Y from 3 and 10 in the previous iteration is basically 6 and 20. And, how do you get minus 3 as the new value of h? When h was 6, and since you have selected south east pixel from after the third iteration, at the fourth iteration your delta south east will be the increment for h.

What is delta south east? You remember? The delta south east has an expression 2 multiplied by x minus y plus 5. For delta east it was 2X plus 3 which we have been doing in the first iteration. Basically at the fourth iteration, since you have selected the south east pixel you have to increment the decision variable h by delta south east which is 2X minus 2Y plus 5 or 2 into X minus Y plus 5. So, let us look at the table.

(Refer Slide Time: 00:21:33)

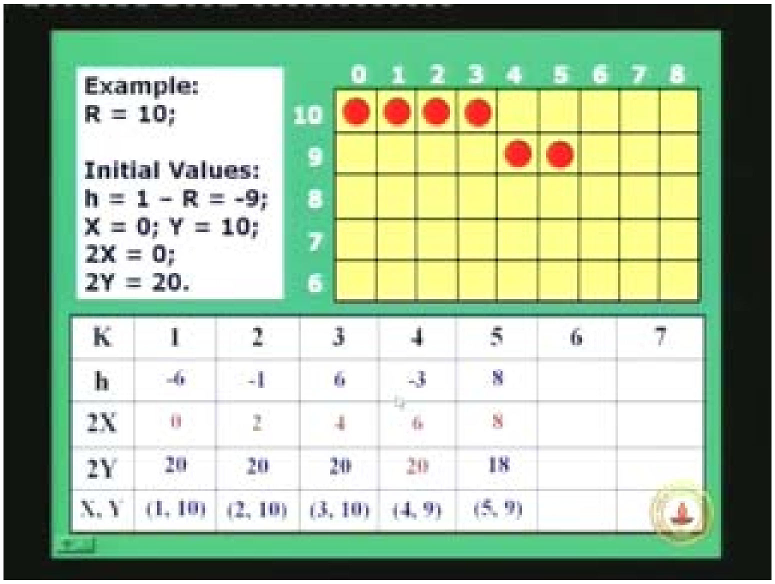


What are basically, what are the values of 2X and 2Y. If you carefully see, 6 minus 20 is basically 14, 6 minus 20 is minus 14, minus 14 plus 5 will give you minus 9, minus 9 plus 6 is minus 3. Again I repeat, how do you get minus 3? It is basically, blindly you can close your eyes and say, it is 6 plus 6 minus 20 plus 5. So 6 plus 6 minus 20 plus 5, so, 6 plus 6 is 12, plus 5 is 17, so, 17 minus 20 will give you minus 3.

I hope you are able to follow calculations. Otherwise, work it out yourself with formulas. Follow the algorithm step by step which I am also doing quite slowly. Now, you must follow, rather than looking at the table you should work it out yourself and check if these values of h. What you are getting is exactly what I am getting in this table. If not, please do and complete it and see.

I hope all are updated by yourself up to the four stages of iteration and at the fourth stage of iteration the values selected is 4 comma 9 for at the current pixel and then we look at h when as we enter the fifth stage of iteration. Remember, we will keep on doing that as long as the value of X is less than 5, otherwise there is no problem, we are in the second octant still. As we enter the fifth iteration now, the value of h is minus 3 which is again negative. So, you have to select the east pixel.

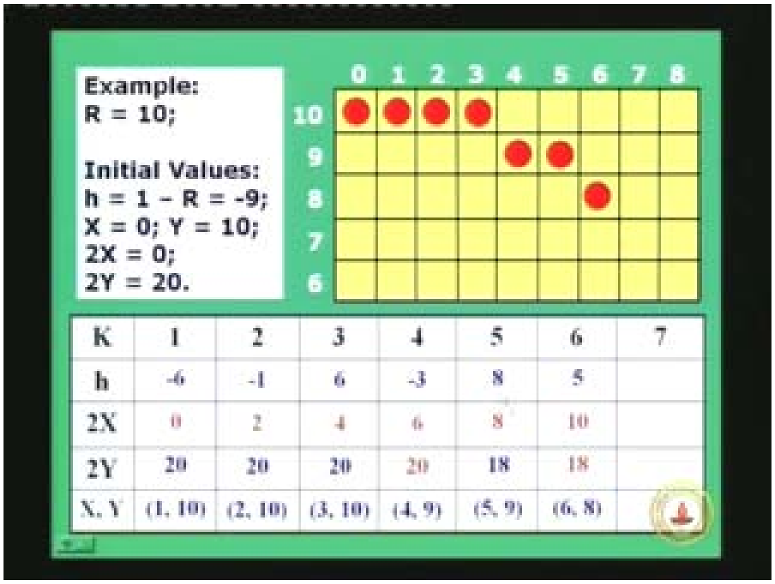
(Refer Slide Time: 00:23:11)



Let us enter fifth iteration and finish the choice of the east pixel. After the pixel 4 comma 9, the next pixel is 5 comma 9 as given in the bottom row of the fifth column and it is also marked in the screen here as 5 comma 9. As we can see very clearly here and that is what is drawn and the 2X and 2Y incremented from, since X is 4 and 9 in the previous iteration, it is 8 and 18 and let us see how you get the new value of as h of h.

How h is updated? Since you have selected east, each increment is again 2X plus 3. Since the increment is 2X plus 3, 2X is 8 plus 3 minus 3. I repeat again, delta e is 2X plus 3, 2X is 8 plus 3 minus 3. You can write those equations yourself, 2X plus 3 minus 3 will again give you 8. That is how you get the value of 8 and in the fifth iteration, at the end, we have chosen 5 comma 9. As we enter sixth iteration, still we are in the second octant because X is less than Y. What is the value of h now? It is 8, this is a positive number, so you have to select south east. So, select the south east pixel.

(Refer Slide Time: 00:24:25)



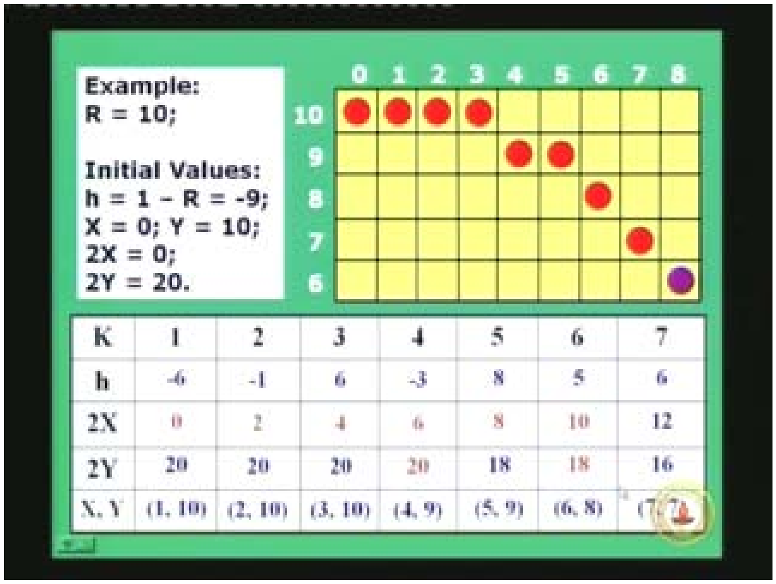
K	1	2	3	4	5	6	7
h	-6	-1	6	-3	8	5	
2X	0	2	4	6	8	10	
2Y	20	20	20	20	18	18	
X, Y	(1, 10)	(2, 10)	(3, 10)	(4, 9)	(5, 9)	(6, 8)	

The south east pixel is 6 comma 8. Now, I hope you are through with the calculations. You are getting used to, how these iteration stages are progressing. Now, basically in these highlighted colors, all colors are in blue in this table below. The colors which have been put in brownish yellow, brownish color or orange color basically, or in brownish color at the values of 2X and 2Y values which are used in the computation for getting the new value of h. That is why if they are not used they are kept.

Now, the fifth and the sixth iteration is based on the value of h equal to 8. We have chosen the south east which is 6 comma 8. Here 6 comma 8 is also marked on the screen. We have to only see what is the new value of h. h was 5. . h was 8 in the fifth iteration and since we are choosing south east, we have to increment the value of h by 2X minus 2Y plus 5. So, it will be 8 plus 10 minus 18 plus 5. Again I repeat, 8 plus 10 minus 18 plus 5.

We do the calculation again, you will find that, it will be equal to 5 because 8 plus 10 is 18 and it cancels with the minus 18. Here, you are left with 5. So, the new value of h is 5 at the end of the sixth iteration where we have selected 6 comma 8 as our point in the current iteration on south east point. So now, what is the value of h at the end of the fifth iteration? It is 5, it is a positive number, so again, you have to select south east. That means from 6 comma 8 you basically select 7 comma 7.

(Refer Slide Time: 00:26:06)

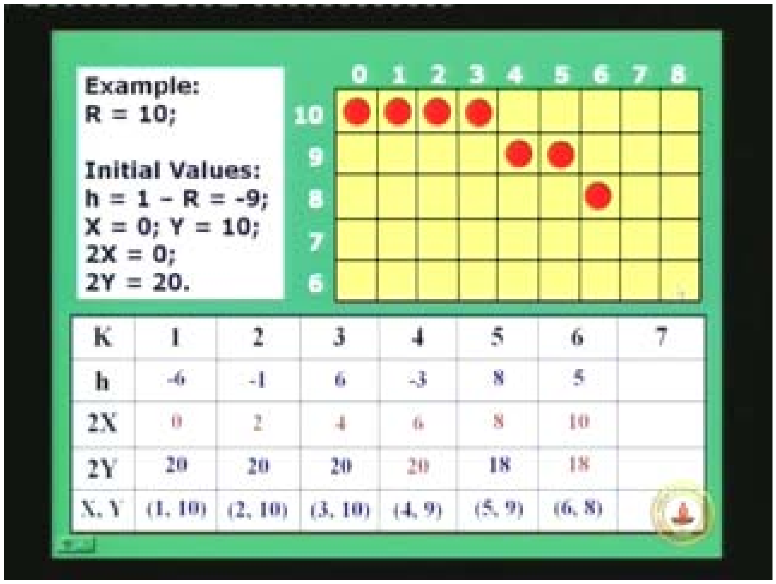


From 6 comma 8, 7 comma 7 is your south east which has been marked here. Do not worry about the non red plots of these points for the time being, assuming that you have drawn 7 comma 7 is your X comma X comma Y. You update a new value of h basically as 5 plus 12 minus 16 plus 5. I again repeat, 2X, 2Y because increment is delta south east. So h plus 2X which is 12 minus 2X which is 16 plus 5. So, 5 plus 12 is 17 minus 16 is 1 plus 5 will be equal to 6. So, this will be the current value of h at the end of the seventh iteration.

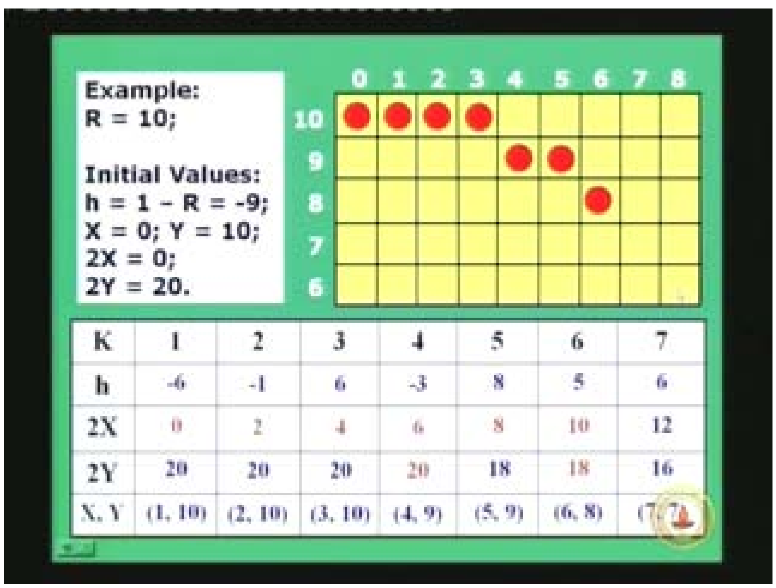
But at the seventh iteration, we now see that X is equal to Y and when you have selected a point when X is equal to Y, you have basically reached the end of the point in the second octant of your curve and you do not have any more points in the second octant. So, with respect to all these points which we have obtained, starting from 0 comma 10, 1 comma 10, 2 comma 10 here, then 3 comma 10, all these are for east, 4 comma 9, south east 5 comma 9, east 6 comma 8, south east 7 comma 7 south east. I am showing these points because the 7 comma 7 which is the last red point drawn in the sequence in your raster image is the last point in the second octant.

The drawcircle will draw all the other octants. This purple color point is actually drawn from 7 comma 7, because the drawcircle will generate it. All the other seven points and the other point which is at the right bottom of your screen, which is basically 6 comma 8 is the point generated in the first octant from 7 comma 7. So, just to show that, that point is in the first octant, it is automatically drawn from 7 comma 7. I have this point, so let us see that again.

(Refer Slide Time: 00:28:01)

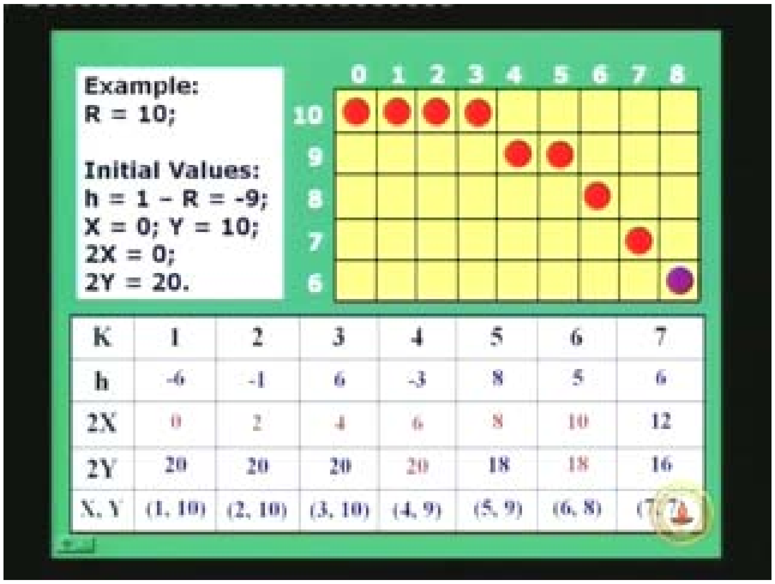


(Refer Slide Time: 00:28:03)



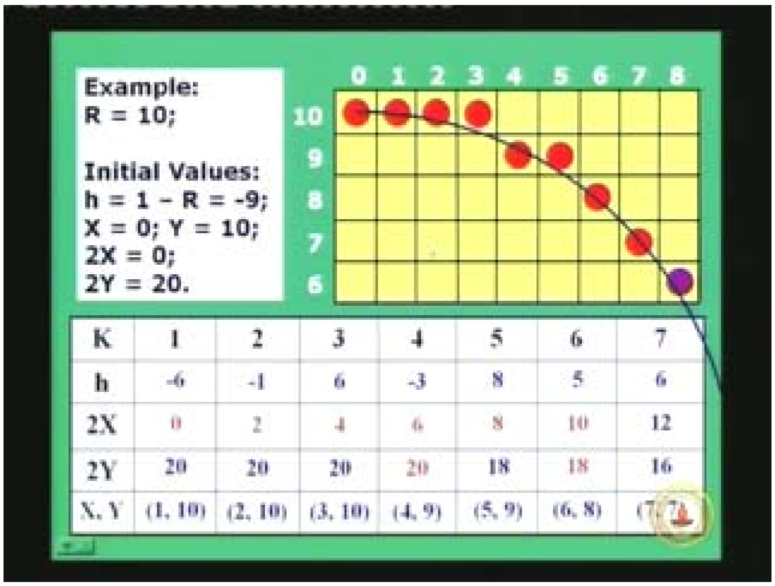
As you enter the seventh iteration the values are selected as 7 comma 7, h is updated and the points are drawn in the second octant as well as in the first.

(Refer Slide Time: 00:28:06)



Since we are only drawing the second octant and a part of the first octant, this is the one actually. Let us look at what the circle should look like? This is what the circle should look like.

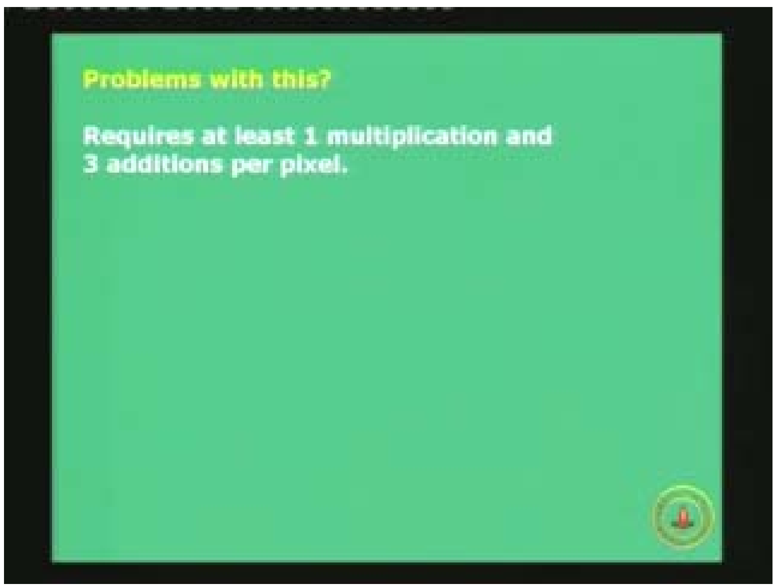
(Refer Slide Time: 00:28:18)



Do not worry that the circle is getting out of the screen, but these are the spread off points of the circle of radius 10 at this origin. And, you can see that the effect of aliasing or jaggies in the case of the circle. As we have to see in the case of a line, how it can be so different from the way we had used to seeing circles. So, this is the spread of the circles approximately. Of course drawn, I must admit here, but just to show the effect of jaggies, this is the circle which is passing through the second octant and getting into the first octant only. And that completes the example of the stage of iterations to get the points in the second octants from the radius of the circle.

So, given the radius, you basically assume the origin and the center and that you let the points in the second octant, all the seven other octants are drawn automatically. So, let us now look at, if this was the case, is this a good midpoint line algorithm because I did say, this is a version one algorithm. It works fine, I mean you get the points which are at the closest to the circle as we have seen from the diagram just now. The points at the closest to the circle the pixels are closest to the circumference of the circle but is this ok? Well, I see I will say no, why this is not ideal? Let us look at this line.

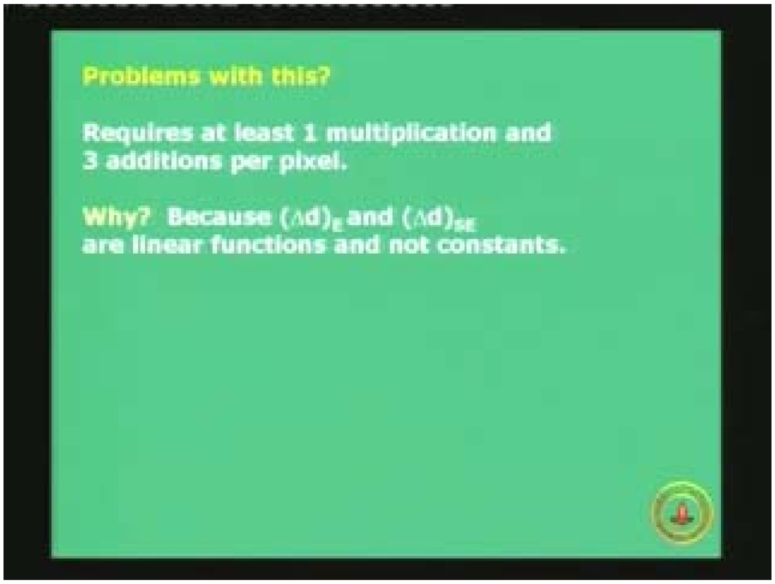
(Refer Slide Time: 00:29:16)



What are the problems with this? Can you identify the problems? Well the first problem compared to the line, drawing algorithm in the case of the circle, it is because of the equation being non-linear, each of this iteration is requiring at least one multiplication and three additions per pixel for each of these iterations. Compare the increment of east and north east pixel in the case of a line and the increment in the case of a east and south east. In the case of a circle delta e, it is 2X plus 3 and delta south east is 2into X minus Y plus 3. So, at least one multiplication and three additions or subtraction operations is what you need to do per pixel that will be very costlier.

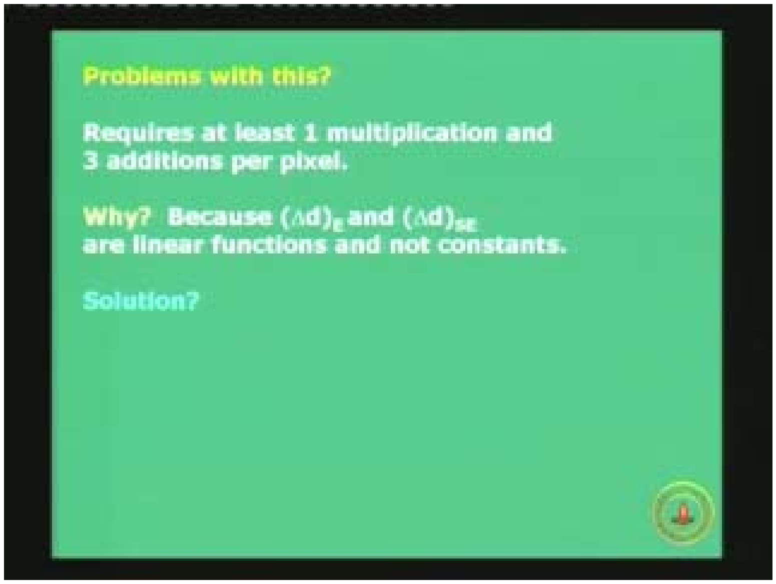
Remember, in the midpoint line algorithm we were just incrementing with integers and that is what was making it fast within a loop forgetting the operations like floating point around. We would like to avoid operations like multiplication and more than one addition if possible. So, the cost of the complexity rise is due to the overhead, associated with each iteration loops which requires at least one or sometimes two multiplications and two or three additions per pixel. And, that has been the main problem for the version one of the mid point circle algorithm. It is correct, it is giving out the correct points, but the complexity is a little bit higher because of the overhead, in terms of computation required at each loop.

(Refer Slide Time: 00:31:10)



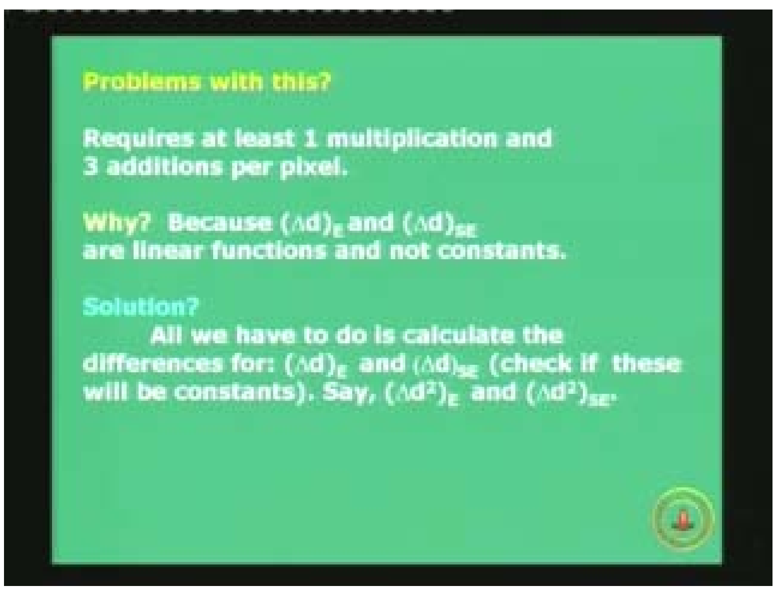
So that is in deep deep-rooted within the main problem, it requires at least one multiplication, three additions per pixel and I just told you the reason why? This is so because the delta d east and the delta d south east are linear functions of X and Y coordinates and not constants as in the case of a line. In the case of a line, the delta d east and delta d north east in the first octant were basically constants. But in this case, the delta d east and delta d south east are linear functions of X and Y coordinates and that is what is giving you the multiplication and more than 1 additions per pixel. So, how to get rid of this?

(Refer Slide Time: 00:31:45)



What is the solution now? Well there is and that is what we are heading towards. What we will call as the version two of the midpoint circle drawing algorithm?

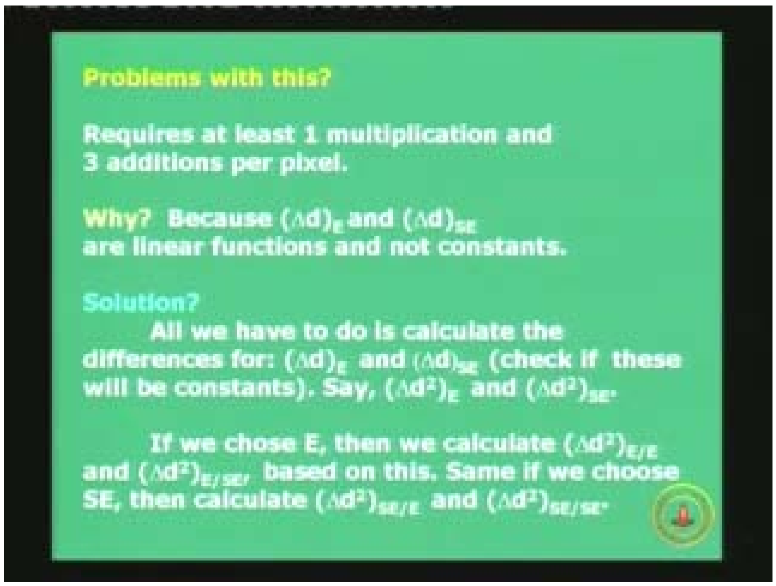
(Refer Slide Time: 00:31:58)



And all we have to do now, is to calculate the differences which occur in delta d east and south east from the one point to the other, from one midpoint to the next midpoint. If you check the differences of the increments delta d and delta d south east, which I will, from now, I will introduce the term, the first order differences in east and south east. The first order difference delta d, and delta d east, and south east are the first order differences which are functions of XY coordinates and not constants.

Let us see what is the difference in the first order differences and we will say these differences are delta d square for east and south east, which we will call as second order differences. So, instead of looking at the first order differences which we have done till now, we know the expressions of the first order differences. We have solved an example problem today to find out these points. What we will look for now is the second order differences and let us see by chance, if the second order differences are constants and not linear functions of XY coordinates. We should be easily able to derive an integer algorithm right now.

(Refer Slide Time: 00:33:07)



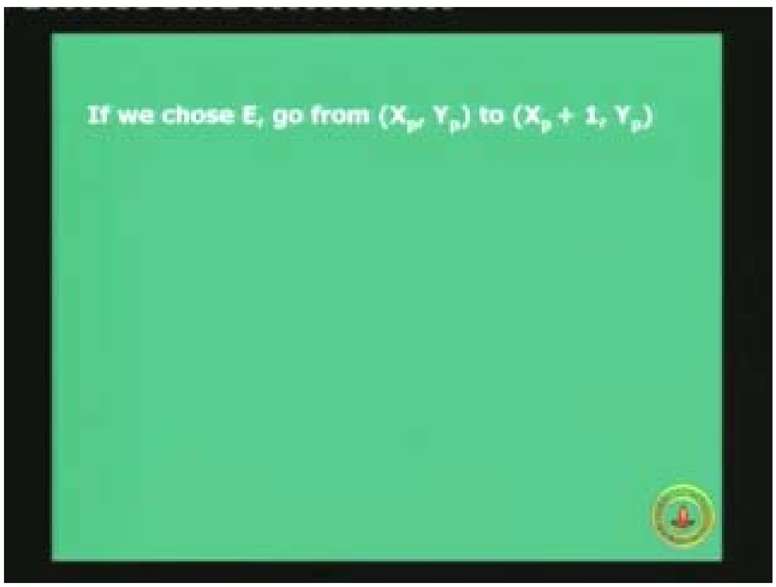
So, we will say, that under the choice, if you choose east and introduce terms, here if you choose east then we calculate the second order difference under the condition that we may choose one east after east or south east after east. That may possibly be a second order difference based on this. And the same, if you choose south east, we can then calculate the second order difference that, after choosing south east we may again choose east or south east.

So in fact, there are four possibilities of the second order differences. The first order difference has only two cases, you either choose east or south east, that is over. In the second octant, as we are, the curve proceeds down, in the second order differences there are four possibilities, under the condition you have chosen east. You can again choose an east in the second iteration.

First, if you choose east it can be followed by an east then east could be followed by a south east. So, the first two, whatever be the other two, well, you can choose south east first and then choose east. Successively, we are looking at two successive operations, east and east east, and south east south east, and east south east and south east. So, the two successive midpoints is what we are looking in the differences in the first order differences of two successive midpoints, because the value of the first order difference at two successive midpoints will not be the same now because they are the functions of XY coordinates. So, look into the second order differences which is the difference of the first order differences. That is, the second derivative function by mathematics as well as the difference in the first order difference is the second order difference. That is what we are looking in the discretised world or digitized space and we will try to find out if in the second order difference, if the easiest could be calculated? Whether these are integer constants?

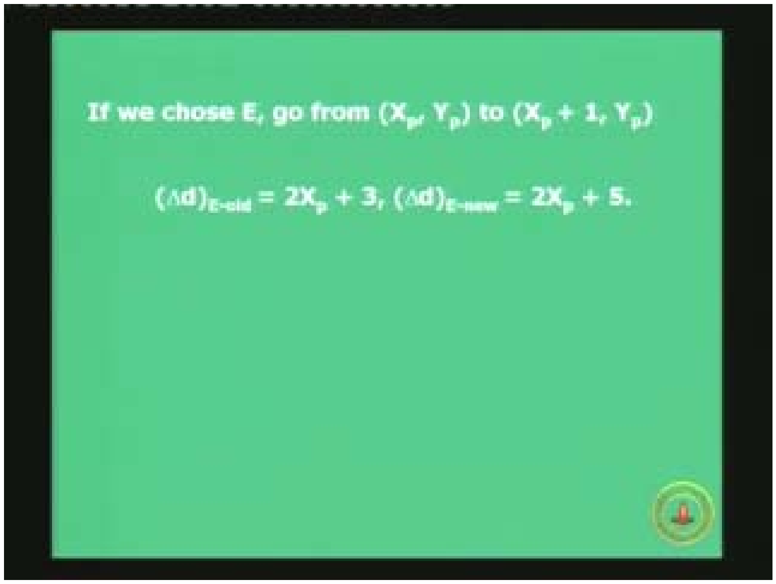
So, there are four possibilities of the second order differences which is east east, east south, east south east east and south east south east. We will calculate these in the next line.

(Refer Slide Time: 00:35:01)



So I hope you will be able to follow this. So, what is the first choice? If we choose east, we are basically going from a point at current iteration X_p Y_p to X_p plus 1 and Y_p that we all know.

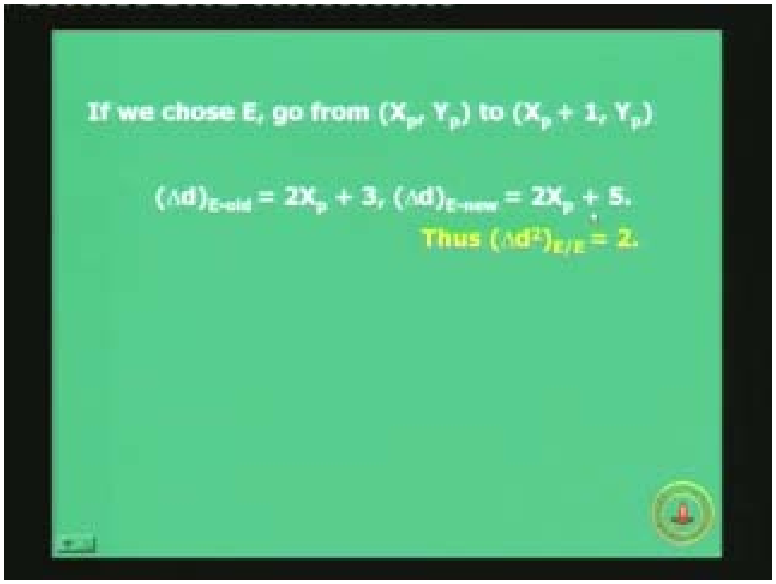
(Refer Slide Time: 00:35:15)



So we will say, the first order difference which is old, is 2X plus p plus 3. This function, we also know that the decision variable is which we are talking. So I replaced h by d, it does not matter, you can put the delta h here also. It is absolutely no problem. But let us take the old variable d itself, $2X_p$ plus 3 is the increment, for, if you choose east, that I will put it as old.

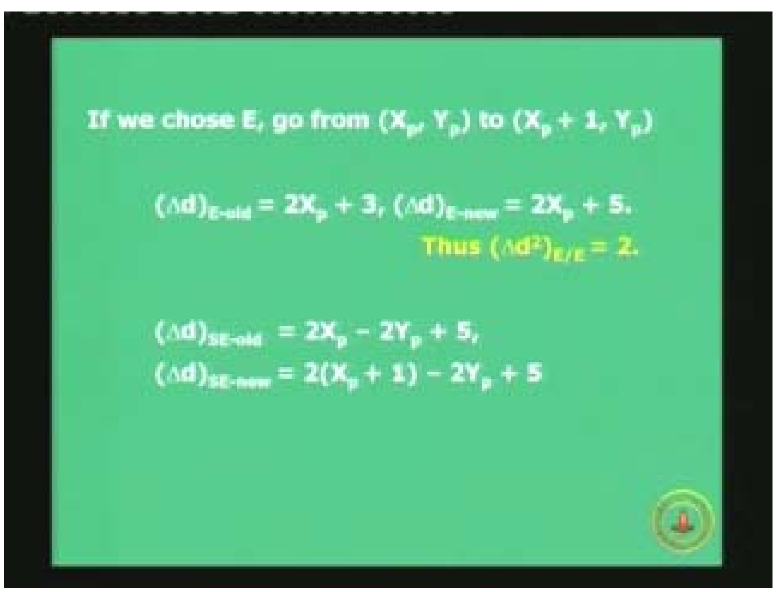
What is the new this new delta east first order difference under east is after choosing X_p plus 1. If you also choose the next ea east point, the point will be We are going from X_p plus 1 comma Y_p to X_p plus 2 comma Y_p and hence the delta d east first order difference new value will be 2 into X_p plus 2, and that gives you 2 X_p plus 5. If you get 2 X_p plus 3 from X_p comma Y_p then from X_p plus 1, you should be able to derive that this is the new value. This X_p plus 1 Y_p gives you the first order difference for the second iteration on new value and the old value is dx plus. Now, you can see that there is already a difference between the first order differences between the first mid point and immediately the next successive midpoint.

(Refer Slide Time: 00:36:40)



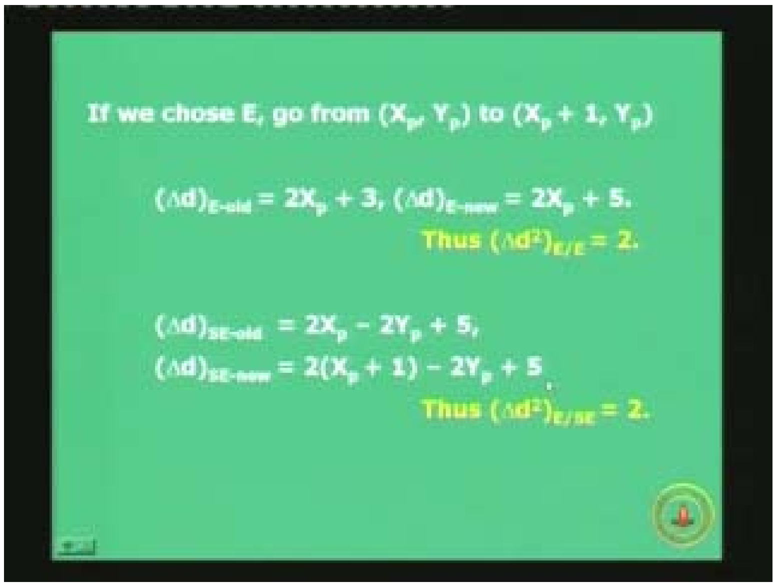
And this gives us the second order difference under the condition that, we have chosen east and east again is basically $2X_p$ plus 5 minus $2X_p$ plus 3 or delta d east new minus delta d e old is equal to 2because $2X_p$ cancels out on both sides and 5 minus 3 will be 2. So we have calculated the first second order difference and if you have followed these you should be able to follow the next order, three stages of this and other three different types of the second order differences. I hope you have followed what is the second order difference. In the first case when we have chosen two successive east points, one east and the second one east as well. Let us look into the second case.

(Refer Slide Time: 00:37:24)



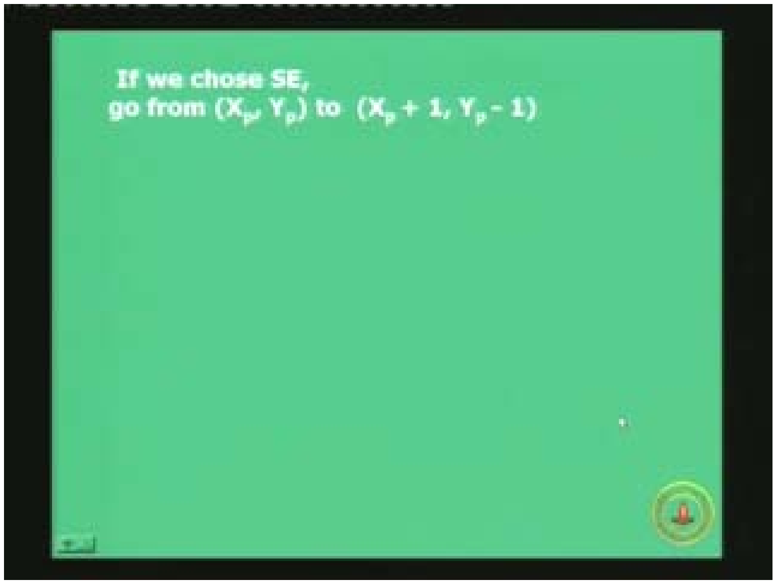
Where delta d south east is old which is $2X_p$ minus $2Y_p$ plus 5. This expression is also known to us that means, we have chosen south east in the first iteration and again if you choose south east, the new value of the first order differences will be based on these values X_p plus $1Y_p$. That is the next coordinate. South east will be X_p plus 1 and also will be Y_p. X_p plus 1, sorry, that is the one and the south east new will be $2X_p$ plus 1 minus $2Y_p$ plus 5. That is going to be south east new.

(Refer Slide Time: 00:38:03)



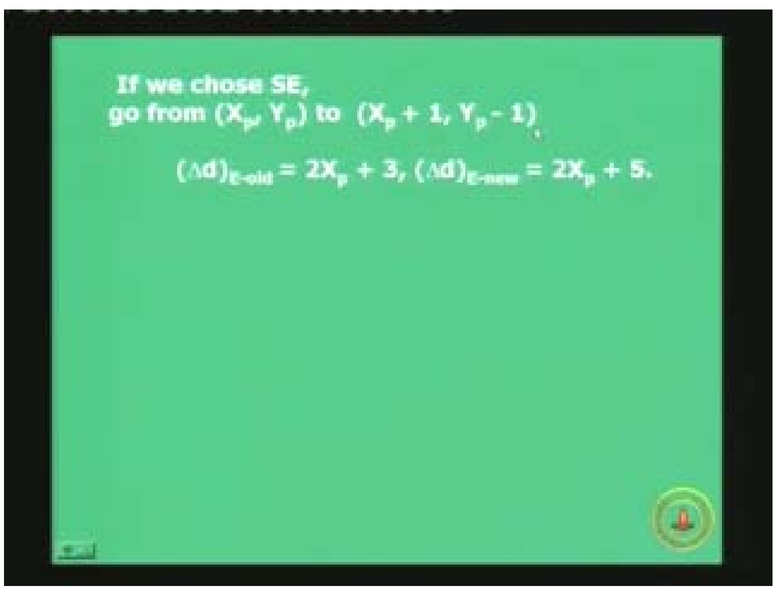
So, this is under the condition that you are choosing east and then south east will be. You have to subtract the second order difference for south east under the condition that we have chosen east earlier. This is very interesting because you must follow what is being done here. I am saying that, in the previous case of second order difference between east and east, in the first choice, again east, the second order difference will be the difference of the first order differences for e new and e old which is equal to 2 but in this case, again you have chosen east, but the second one, you could be choosing south east, then this difference between these two expressions and it will again be equal to 2. That is very simple and straight forward.

(Refer Slide Time: 00:38:44)



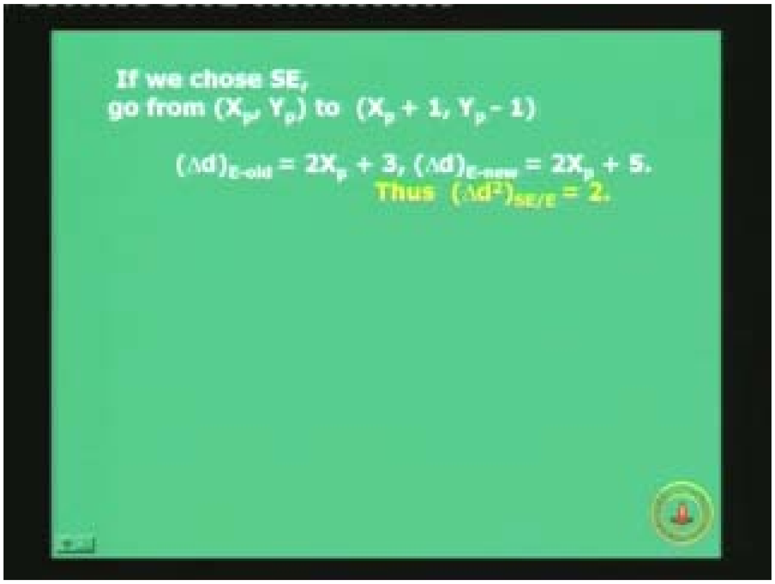
Let us look at the choice. If we go south east, I hope you have understood the previous case where the choice was purely based on east. Now, if you have chosen south east, the point goes from X_p Y_p to X_p plus 1 Y_p minus 1 south east point.

(Refer Slide Time: 00:39:01)



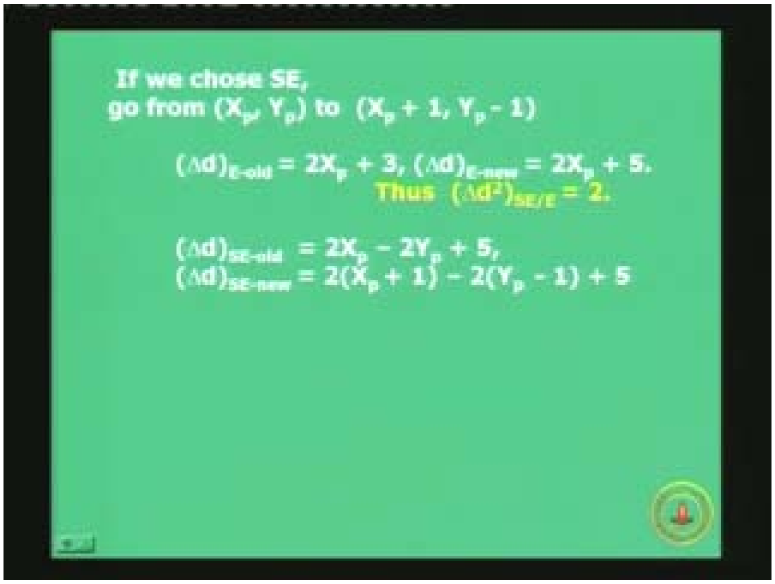
And the first order difference between old and new will be the expressions as given earlier is the same thing now.

(Refer Slide Time: 00:39:10)



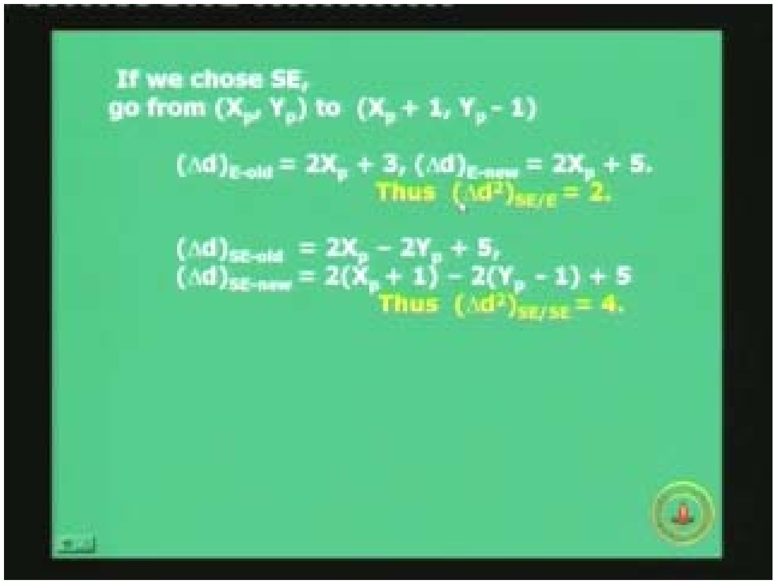
And hence, the second order difference delta d square under the choice of south east but again looking at east after south east is equal to 2 and similarly you can look at south east old and south east new. The values will be different now, Y_p minus $1X_p$ plus 1.

(Refer Slide Time: 00:39:19)



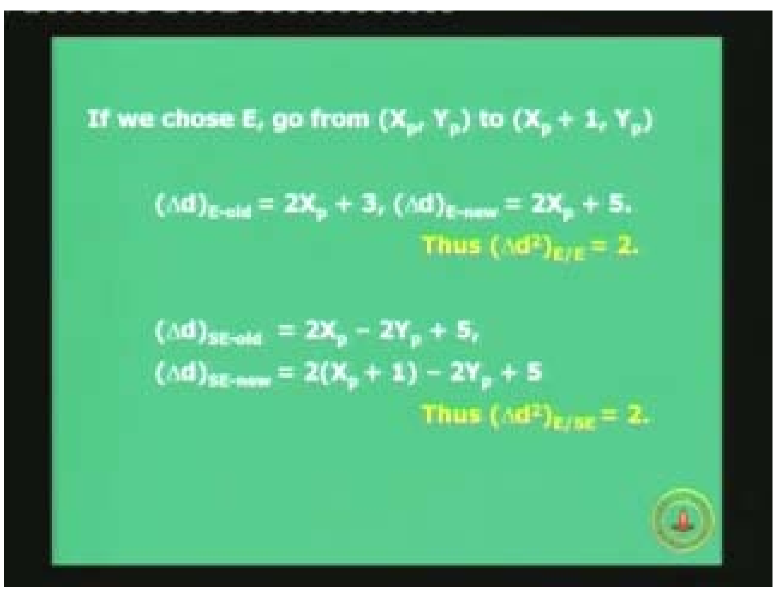
You just substitute these second coordinate here, X_p plus $1Y_p$ minus 1 in the expression of the first order differences of old and new old, based on X_p Y_p new, based on X_p plus $1Y_p$ minus 1 which is the south east pixel. Those are the expressions for south east old and new for the first order differences.

(Refer Slide Time: 00:39:45)



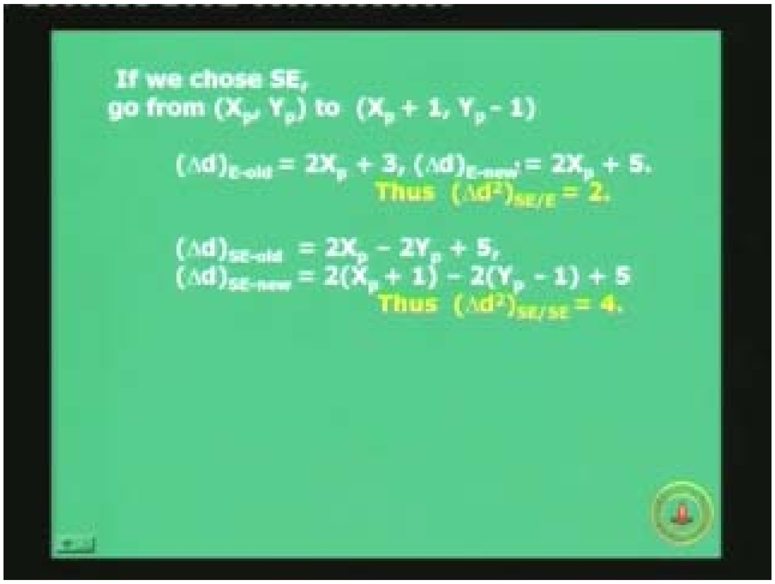
Take the difference is what you will get as your second order differences which is equal to 4. If you subtract south east old from south east new, you have a 2 from this term coming out and another 2 from this term coming out. So, it is 2 plus 2 minus 1 minus 1 will give you a positive number 2 plus 2 will give you 4. So I hope you will be able to calculate these and have understood this. I will probably revise it once again.

(Refer Slide Time: 00:40:11)



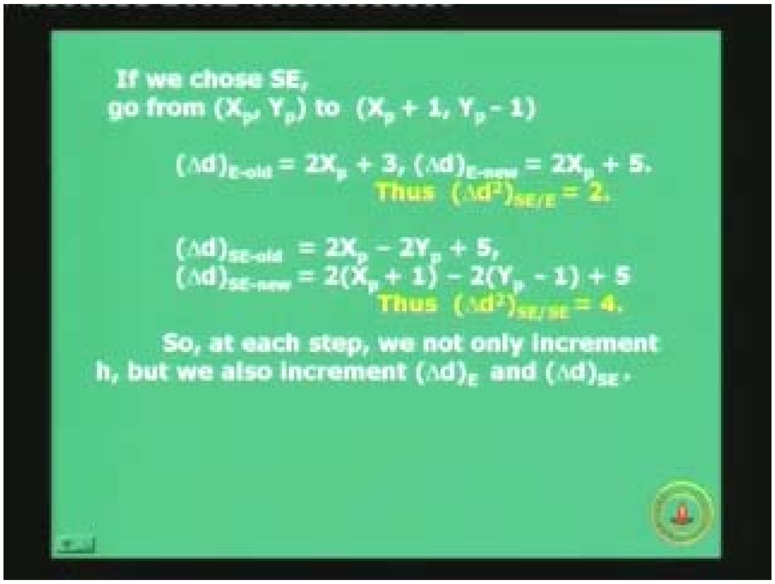
Let us go back to the previous case of east when you have chosen east, these were the cases, because the point is moving from X_p Y_p to X_p equals 1 and Y_p. So, use this expression to compute the delta d old and delta d new. Take the differences, it will give you the first order difference under east and east. The substitution indicates that you have actually chosen east and looking at east again in this. Say, in this case, the second time I am choosing east but I am looking at south east, in fact I have chosen south east after east. Then I have to increment south east variable by 2 because the expressions are these, as given by the two coordinates.

(Refer Slide Time: 00:40:53)



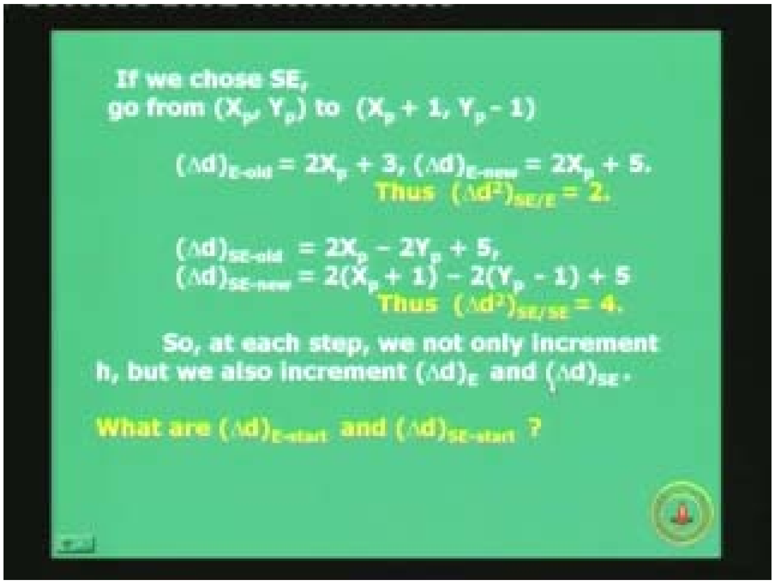
So, these were the first two cases and in the second case when we are choosing south east, the difference has come out in this form as 2 and 4. So, the interesting point in this is, if you have followed the calculations, please try to do it yourself which will be interesting for you. Following that, the second order differences are all integer coordinates, integer values, they are all integer values and those integer values were like the case when we were doing scan line algorithm for a straight linear case. You also had integer values. Second order differences are integers. It helps to come out with a new version of the algorithm based on the second order differences. Let us look at that algorithm now. Before that we will just rewind the methods.

(Refer Slide Time: 00:41:29)



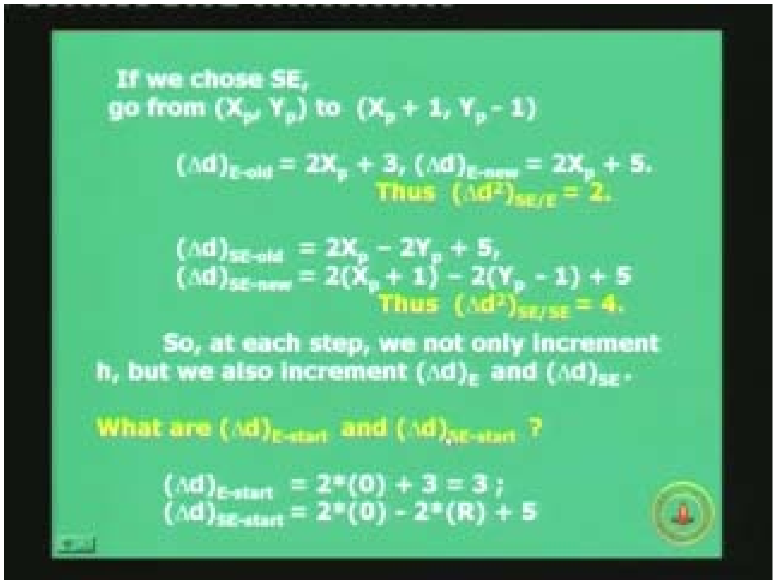
So at each step, we not only increment the value of h, but we also increment now, the first order differences east and south east. In the version one, we were only incrementing h, using the first order differences. Now, in each loop we not only increment h, but we also increment the first order differences of east and south east and how do you increment these because h is incremented by the first order differences delta d e and delta d south east. These are again incremented by the second order differences.

(Refer Slide Time: 00:42:00)



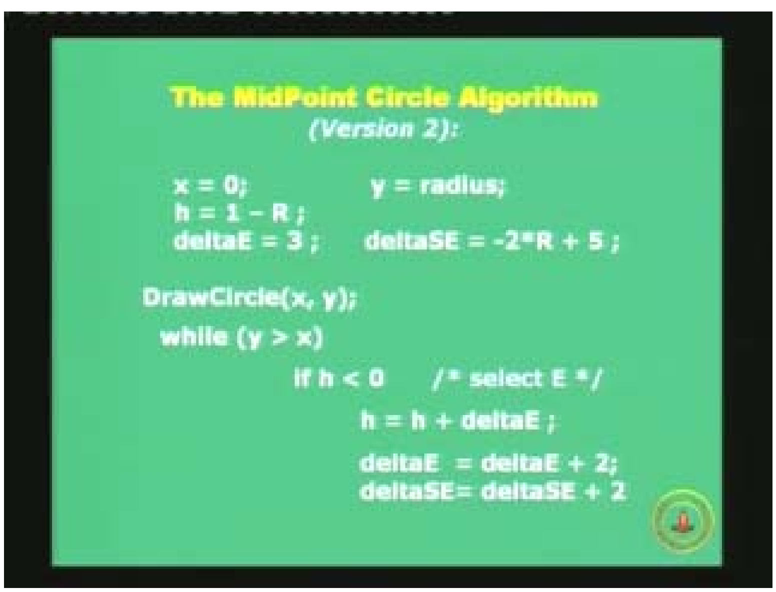
Since you had an h start, we need to have a delta d e start and delta d e south east start as well. That means, the first order increments and first order differences at the start for east and south east at the starting.

(Refer Slide Time: 00:42:16)



Just substitute the value of 0 and R in each of the two cases because you know, the expression you get the first value as 3 and another is 5 minus 2 R. So, these are the starting values, 5 minus 2 R and 3 is what you get as the starting value for delta d. We know the starting value of h. We know the starting value of the first order differences.

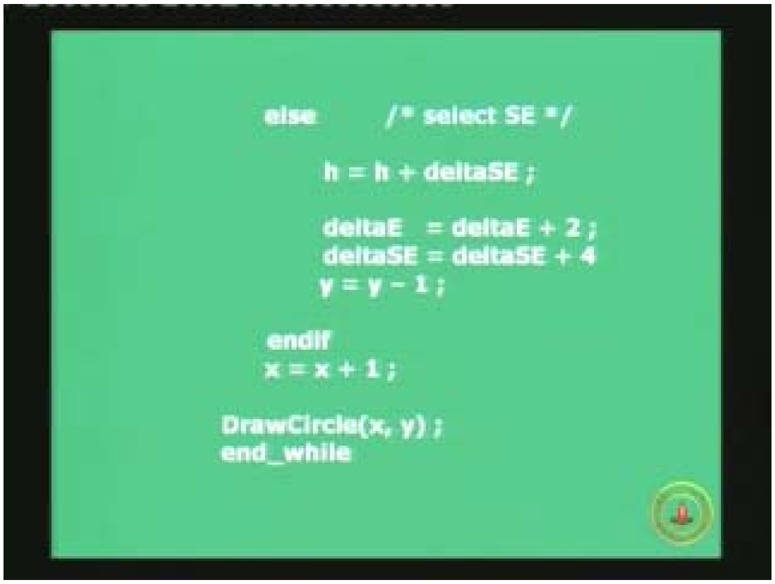
(Refer Slide Time: 00:42:36)



And now, we can look at a midpoint circle algorithm version two. What does it say? The initialal lisation initial liaison part, unlike the three steps for the version one has five steps, where we initialize x were 0 and r. I use the word radius and instead of r, but you can use capital R there. It is absolutely no problem. We have interchanged r and radius. You can assume this to be capital R, x is equal to 0, y is equal to r, h is equal to 1 minus R and the first order difference is 3.

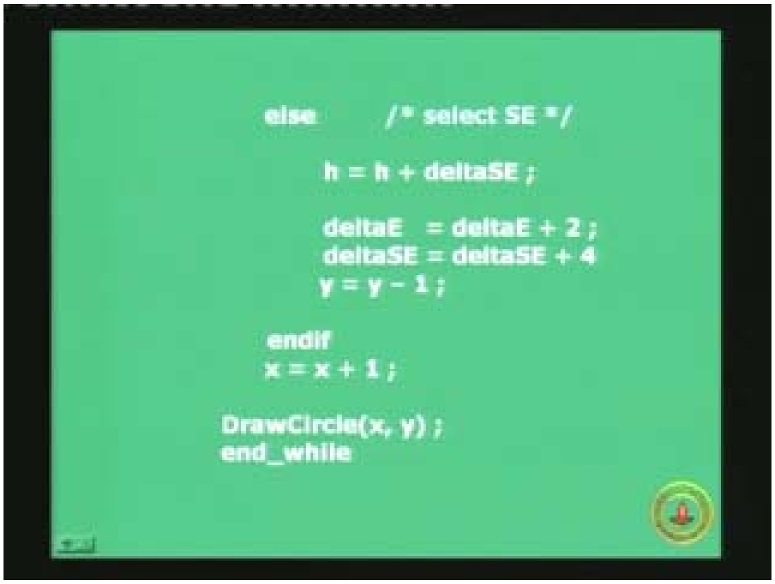
The first order differences for south east is minus 2 R plus 5 or 5 minus 2 R. So the initialization consists of five statements here. The drawcircle as, like the previous case, now let us look at the while loop and the first if condition when you have selected east, you first increment h based on del d, the first order difference delta e in east direction. You do that, that is the one which you are doing in version one but in version two what you also do is based on this you increment the first order differences delta e and delta south east by the second order differences two and two as have been evaluated earlier. This is under the condition that we had selected east. I hope you have noted down the expressions earlier and you are able to follow the algorithm.

(Refer Slide Time: 00:43:51)



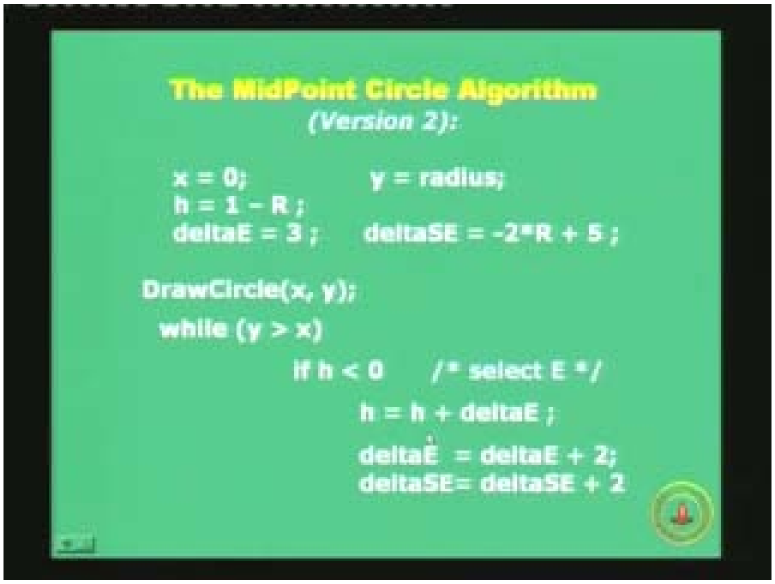
Let us look at the other if condition when we are selecting the south east. Yes, we do increment h by the first order differences of the south east direction, that is delta dSE and the first order differences of east and south east are incremented by their second order differences respectively 2 and 4. Remember, we got three integers and four values of the second order differences. The first two cases were 2 and 2, the second were 2 and 4. I hope you understood the derivation of that and you noted down those values what has been done.

(Refer Slide Time: 00:44:20)



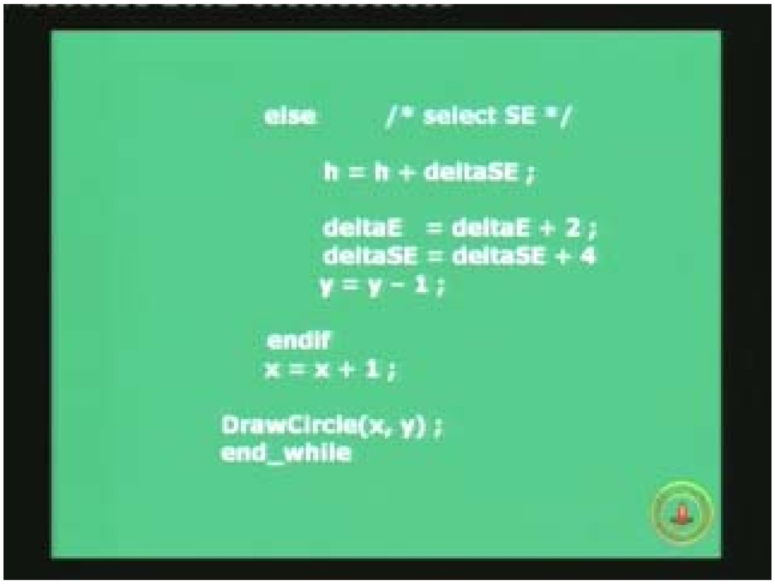
In the previous case, when we selected east, the increment is by 2 each in delta d and delta south east, and in the second case, it is basically 2 and 4. If you look in to the slide, you will happen to select south east, here you increment by the second order differences 2 and 4. You increment or decrement y by 1 always, when you select south east and come out of the if loop, not the loop, the if condition and basically increment x by 1 and then you do the drawcircle again to draw all the points in all the seven other octants using x and y and then wind up the while loop. So, this is the version two algorithm.

(Refer Slide Time: 00:45:03)



These were the initial steps. I repeat again, then there was a while loop, there was a if loop and if h is negative, select east, increment h and increment first order differences.

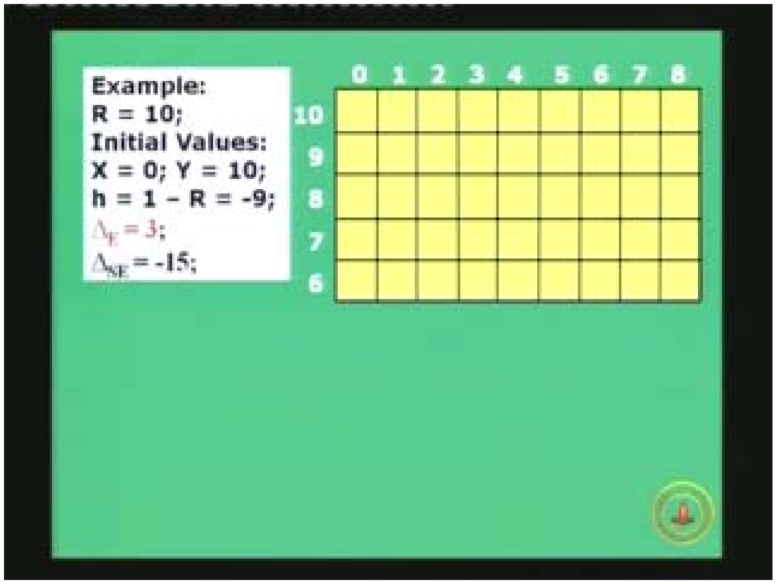
(Refer Slide Time: 00:45:13)



Else condition, select south east increment h by the south east first order differences increment, the first order differences decrement y, come out of the if condition, increment x and draw all the points and keep on doing this as long as you are in the second octant. So, this is the algorithm which we have for the case of drawing the circle in the version.

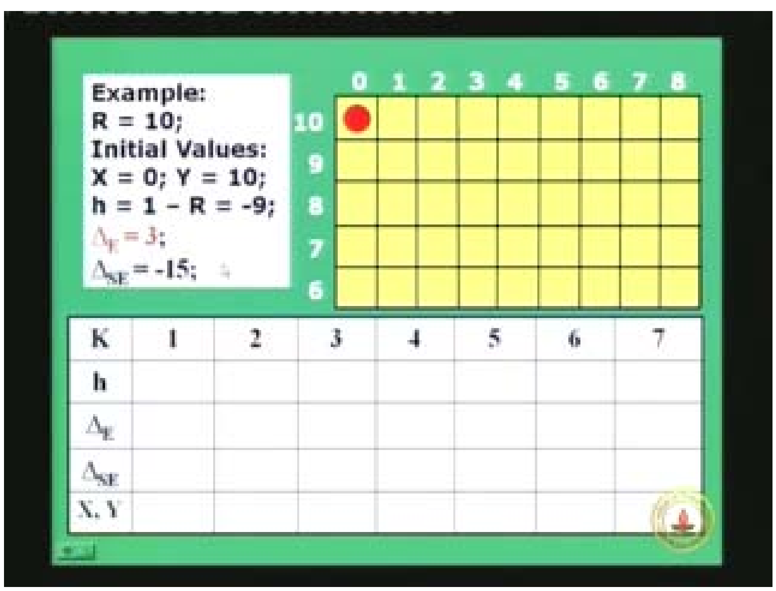
We will quickly go through the example, the same example which we used for the version one to draw a circle with center at 0 and radius. What was the radius? 10. So we did that with the version one where we have to calculate at each iteration, the first order difference delta e and delta south east using some the x y coordinates. Remember the calculations, some multiplication of x into 2 plus 3, 2X minus 2Y, now, that is not necessary. You now start with an initial condition, not only of h but of the first order differences and within each loop, you increment that with second order differences. That will be interesting. So let us look at this table once again.

(Refer Slide Time: 00:46:18)



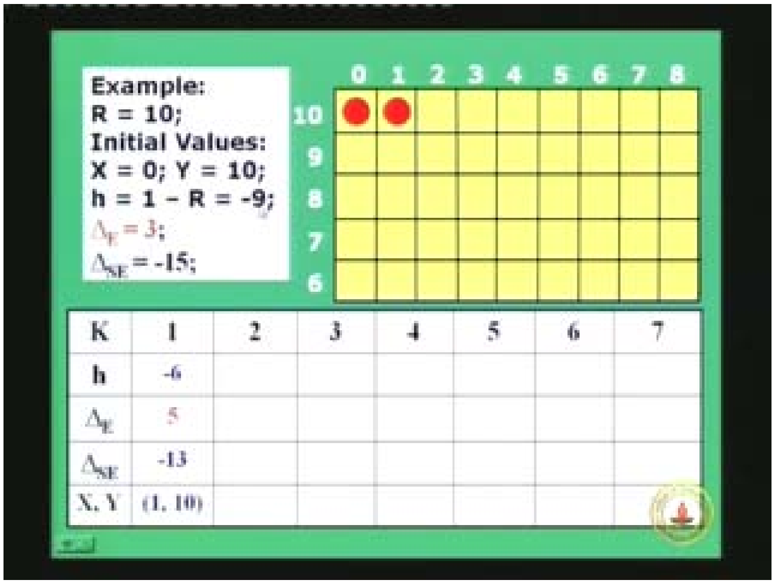
And start with the example where the radius is equal to 10 and the initialization consists of five steps. First point is of course 0 comma R. So, it is 0 comma 10, no doubt about it. h value, of course will be the same which is 1 minus R that is minus 10. But in the initialization steps, we have two values of delta e and delta south east, two values of the first order difference is a part of the initialization step itself, where delta e was, what was delta e? It was 2x plus 3. So since x is equal to 0, you have 3 and what was delta south east? 2 into x minus y plus 5. So minus 22 into 10 is minus 20 plus 5 minus 15. So I hope the calculations are easy to follow. Those are the starting values for the first order differences and we now open the table and start to look at the steps of the operation.

(Refer Slide Time: 00:47:08)



This time I can go a bit fast because you followed what I was doing in the previous case of version one incrementing all the decision variables. And in this case, I have to increment also the decision variables of the first order differences. So, you entered the first iteration with h value minus 9 which forces you to select as usual the east, so you select 1 comma 10 and plot that point 1 comma 10.

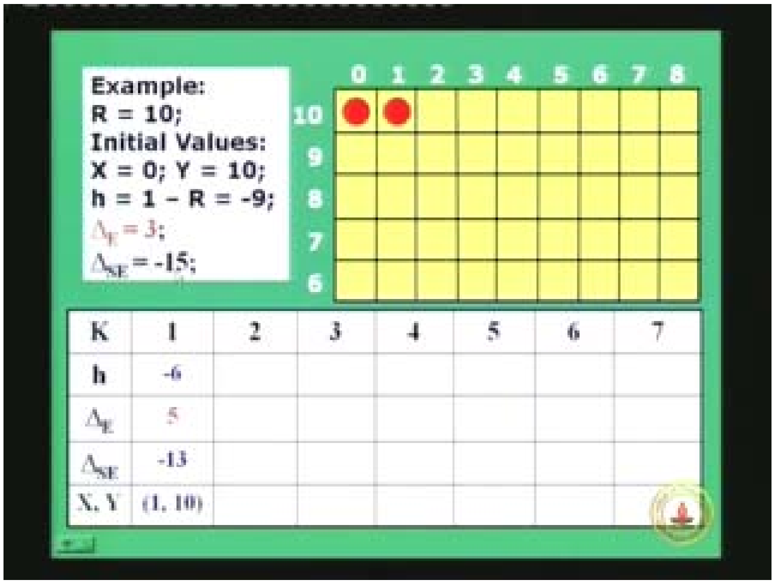
(Refer Slide Time: 00:47:32)



And what you do, you have to keep incrementing h delta E delta south east.
What you do for incrementing h? It is minus 9, h is minus 9 plus 3 because you have chosen east, so delta E will be 3 which will be used to increment h minus 9 plus 3 marked in, on that color, will give you minus 6 and what you do to increment delta east and south east?
Since you have chosen east you increment both by 2, in fact it will be interesting to note. The delta E will be always incremented by 2. In this third row, delta E will be always incremented by 2, delta south east may be incremented by 2 or 4 depending upon whether you choose east or south east. So blindly follow the logic.

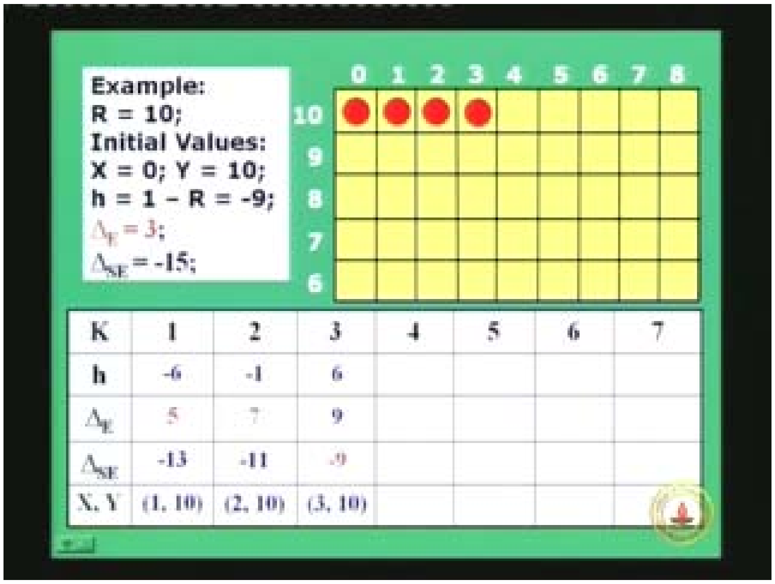
The first order increments of the first order differences are the following increment delta e always by two invariant of what they were choosing east or south east, the increment, first order increment of first order or the increment of the first order difference is for south east, can be incremented by either 2 or 4. If you are choosing east, you choose 2. If you are choosing south east, you choose 4. That is what we derived at.

(Refer Slide Time: 00:48:43)



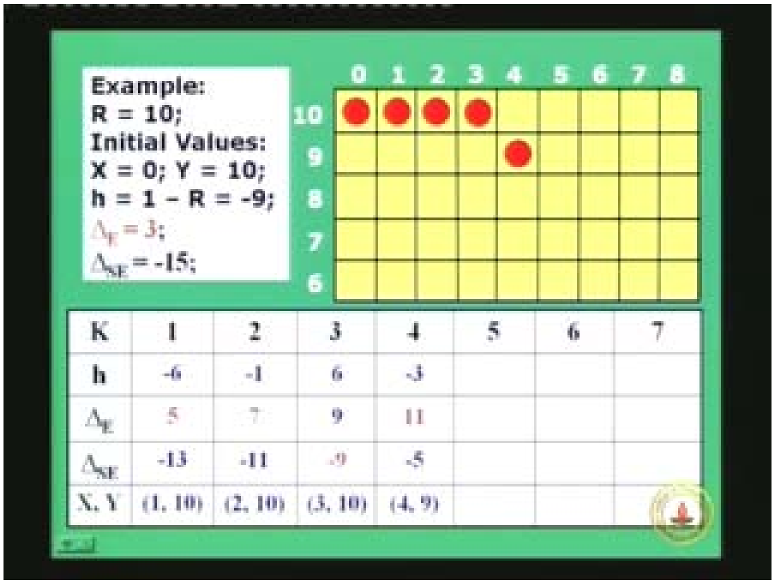
So, now if you see the values, it was 3 and minus 15. increment both by 2 because you have chosen east. So 3 plus 2 is 5 minus 15 plus 2 is minus 13. Look at minus 6. Obviously the next choice is 2 comma 10. That is what we have selected and since we are choosing east, 5 plus 2 is 7 minus 13 plus 2 is minus 11 which is very straight forward. The next two iterations are even very straight forward. Look at minus 1, you have to choose the next one. What is the next one? 2 comma 10 is chosen. You basically choose 3 comma 10.

(Refer Slide Time: 00:49:10)



You basically choose 3 comma 10. First the value was 0 comma 10. The first iteration gives you 1 comma 10 which is this. Second is 2 comma 10, third 3 comma 10 because, from minus 1 is less than 0 and minus 1 plus 7 will give you 6. Increment 7 by 2 and that gives you 9. Increment minus 11 and it gives you minus 9. Now, for the first time, the h value has become positive, so you have to choose south east and if you have to choose south east, the next value after 3 comma 10 will be 4 comma 9, so choose 4 comma 9 and mark it.

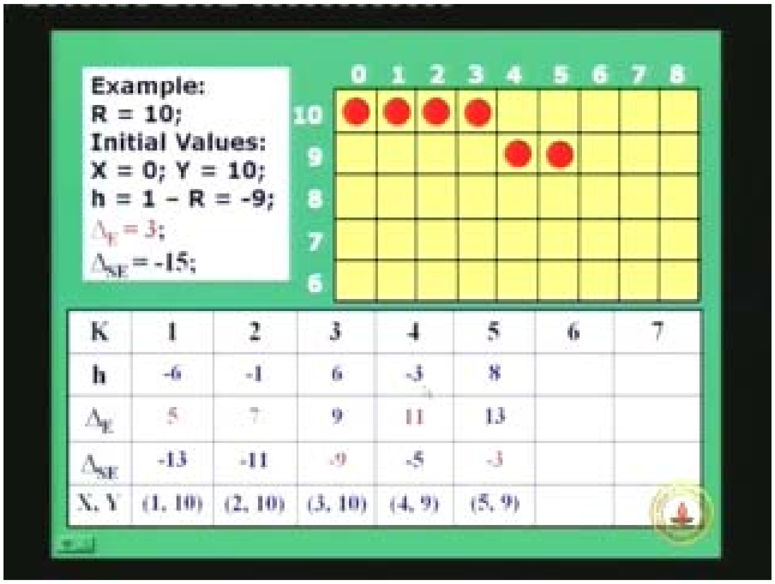
(Refer Slide Time: 00:49:43)



And then, what you have to do is, increment 9 by 2. The first order difference for east 9 becomes 11 plus 9 plus 2 is 11. And minus 9 has to be incremented by 4 because you have chosen south east for the first time. Anytime you choose south east, you have to choose increment by the second order difference as 4 is what you have to use. So minus 9 plus 4 will give you minus 5 and this is minus 9 which is delta south east. Since you have chosen this, this is now the first order difference which you use to increment h. So, h is 6 plus minus 9 will give you minus 3.

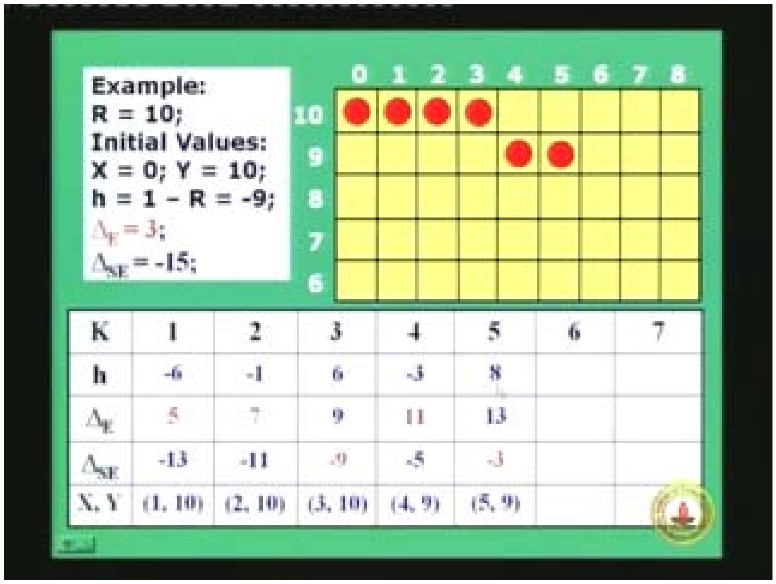
I hope you have followed the calculation. Minus 6 plus 5 was minus 7, minus 1 plus 7 was giving you 6 and 6 minus 9 gives you 3. In the first stage of the iteration 5, 7, 9, and 11are the increments of the first order differences for east. First order differences of south east is minus 13 plus 2 minus 11 minus 11 plus 2 minus 9 minus 9 plus 4 is minus 5. Why 4 here? It is because we have chosen in the fourth iteration, a south east pixel 4 comma 9 from 3 comma 10. Look at h, it is again negative, so choose south east which is the point 5 comma 9 as marked here.

(Refer Slide Time: 00:51:01)



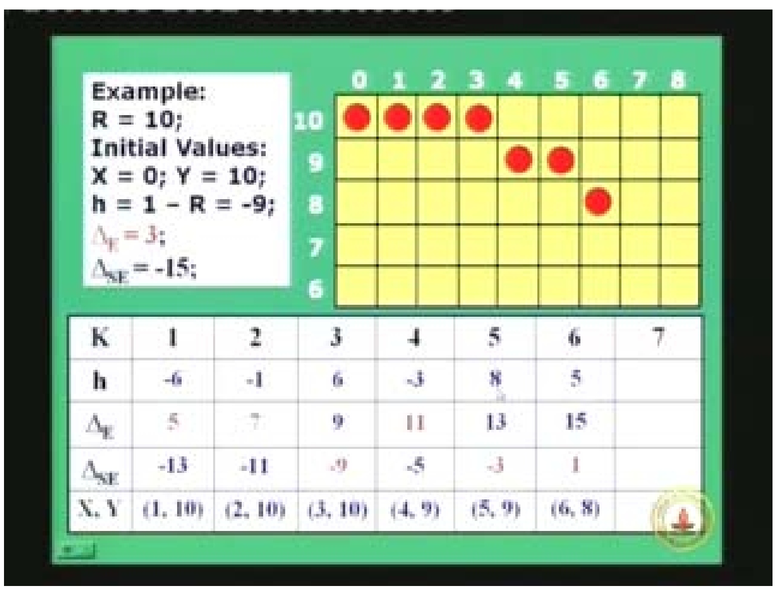
In the fifth iteration, as given below at the bottom of the screen, 5 comma 9 and it is marked here and then 11 is incremented to give 13. Since you have chosen, basically the east, so you have to increment h by delta east, so it is minus 3 plus 11 which is the non blue color, minus 3 plus 11 as the color is different which is used you see to increment h. So, minus 3 plus 11 will give you 8. So 8 is the current value of h at the fifth stage of iteration. That is what you have and then what you see is, you use h to decide the next choice east and south east. Since h is positive, you will choose south east. Since h, I hope you are following these stages, please try to work this out yourself to get the new values of h and the next coordinates as you get it out. Otherwise, if you miss one particular point, you will not be able to follow the steps yourself.

(Refer Slide Time: 00:52:04)



Try to predict yourself, so look at the fifth iteration now where we have chosen 5 comma 9 as your east pixel, but you now look at the value of h which is 8. h is equal to 8 and enter the sixth iteration and choose your pixel. If h is positive, we choose south east we know that. So, choose the south east pixel 6 comma 8 and mark it there.

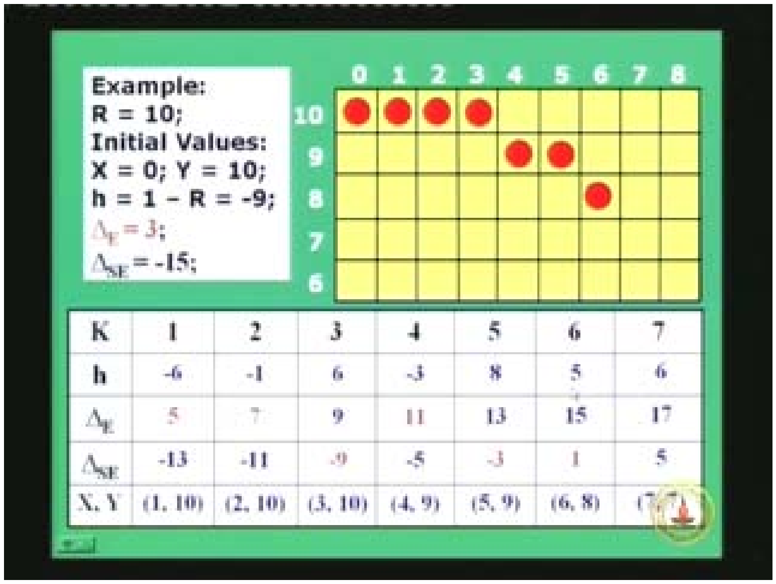
(Refer Slide Time: 00:52:20)



So it is marked, 6 comma 8 is marked in a red circle and then what you do is you increment 13 which is the delta e by 2. So, that gives you 15 delta south east. Since you have chosen south east, you have to increment that by 4. So, minus 3 plus 4 will give you 1 and you actually use delta south east to increment your h.

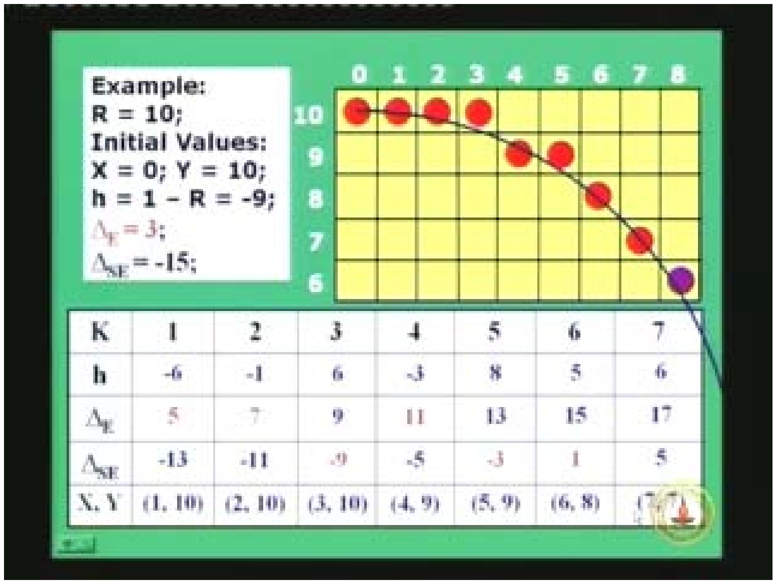
So, h which is, h plus delta south east, which is minus 3 h. Minus 3 will give you 5, so that is what you use to compute the new value of h. So in the sixth iteration you have chosen the value 6 comma 8.

(Refer Slide Time: 00:53:02)



Look at 5 remember X is still less than Y and you basically will choose the south east pixel because h is positive. So choose 7 comma 7 increment 15 to 17. That is the first order difference in east and since you have chosen south east, this delta south east will also be incremented from 1 plus 4 which is equal to 5. Use 5 plus 1, increment h by delta south east, so 5 plus 1 will give you 6 and you have chosen 7 that is X equal to Y, which is the last point in the second octant.

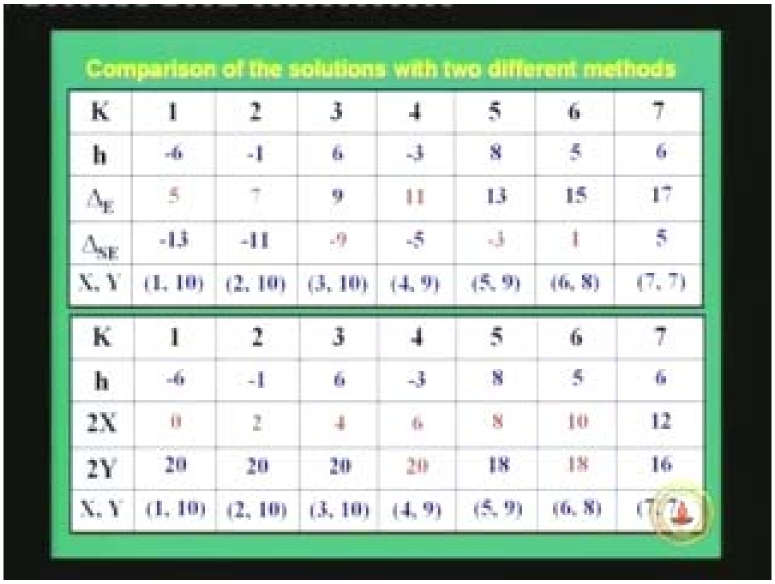
(Refer Slide Time: 00:53:29)



And that is where you have finished the stage of the iterations and the curve is here. Actually, you got the same set of points with the different calculations and these calculations never at each stage of iteration. It involves only integer, addition or subtraction. One integer addition, one or two integer additions or subtractions like the scan line algorithm for drawing the straight line. Unlike the previous version we had, we had to multiply and add two or three times under each loop. We did not have to do that.

The version two will definitely work faster because we are only doing integer manipulation. Delta east is incremented always by 2 and delta south east by 2 or 4 and h is incremented by delta east or delta south east which is the first order differences depending upon whether you have chosen east or south east. So, as you can see here that, this is a much better algorlithm in terms of implementation; it will be yielding much faster. We will wind up today’s class with this comparative table.

(Refer Slide Time: 00:54:24)



Comparison of the solutions with two different methods

K	1	2	3	4	5	6	7
h	-6	-1	6	-3	8	5	6
Δ_E	5	7	9	11	13	15	17
Δ_{SE}	-13	-11	-9	-5	-3	1	5
X, Y	(1, 10)	(2, 10)	(3, 10)	(4, 9)	(5, 9)	(6, 8)	(7, 7)

K	1	2	3	4	5	6	7
h	-6	-1	6	-3	8	5	6
2X	0	2	4	6	8	10	12
2Y	20	20	20	20	18	18	16
X, Y	(1, 10)	(2, 10)	(3, 10)	(4, 9)	(5, 9)	(6, 8)	(7, 7)

Comparison of the solutions with two different methods and the table on the top actually gives you the values at each iterations using the second order differences and the table below which we have completed in the middle of the class today. Earlier, the values at each iteration which have been computed was based on the first order differences. One alone as you can see, you just compared the last two rows of X Y, you got the same set of points 1 comma 10, 2 comma 10, 3 comma 10, 4 comma 9, and 5, 9,6,8 and 7. So the result is the same. Not only that, it is interesting to note that the values of h as you look in the second row of both these tables are all same.

So, whatever method you use, either the second order differences or the first order differences alone, you will actually be getting the same values of h and same values of X and Y. So it typically means that your calculations are correct you get the same result here. The algorithm is wrong if you do a mistake in calculations. When you are doing, using hand or the calculator, you will get two different results and something would be wrong. This is the best way to check. Pick up any other radius value instead of 10 you can put which is a larger value of 14 and 15, lesser value of 7 and 8 to compute all the values of X and Y using the first order differences alone which is 2X plus 3, 2X minus 2Y plus 5.

Use those formulas to get the new values of h. Get the new values of X Y, then use the version two algorithm, where do you use the second order differences to increment if? Within each stage of iteration, you are using the second order differences to increment the first order differences as well as the h. Do that and find out the values of h. All the values of h at each stage of iteration should match in the two stages of the algorithm independent of whatever is the value of R. If it is not, then you are making a mistake. So, we have solved an example today using two different versions of the midpoint scan line algorithm for drawing a circle, the first order differences and second order differences.

We got same results, we should be happy but I expect you to probably keep trying this again and again so that not only this particular example you should try but you should also try another example with a different value of R and if you get both the results as same you should be happy that you have not done a mistake. So, this ends the discussion on midpoint circle algorithm today. In the next class we will actually move to the other craft drawing mechanism of the other drawing, which is the midpoint ellipse, drawing an ellipse.

So we can continue that in the next class. Thank you very much.