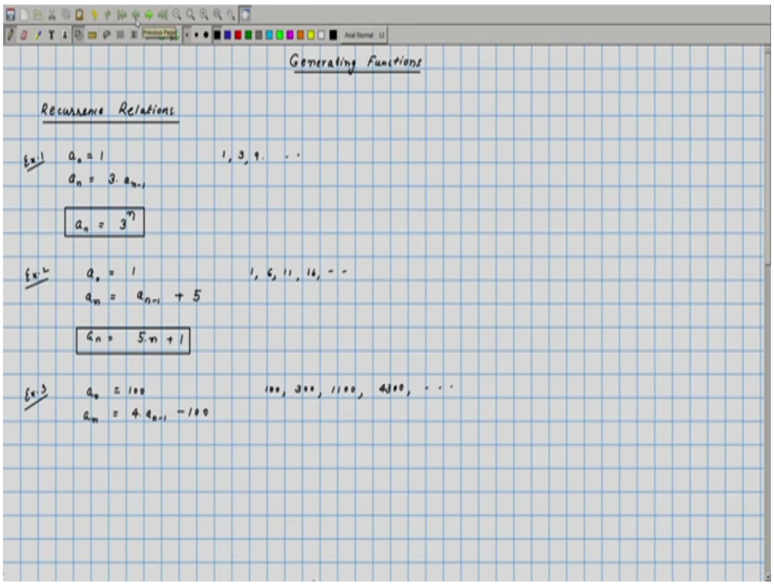
Discrete Mathematics
Professor Sajith Gopalan
Professor Benny George
Department of Computer Science and Engineering
Indian Institute of Technology Guwahati
Lecture 29
Generating Functions

In this lecture, we will learn about generating functions. Generating functions is a tool used to solve recurrence relation.

(Refer Slide Time: 00:36)



Let us first see some examples of recurrence relations. The first recurrence relation that we will see is a very simple recurrence relation. So, the first term a0 is denote is equal to 1 and a n is defined in terms of the previous value and it is 3 times a n minus 1. This is the familiar geometric series where the ratio of successive terms happens to be 3.

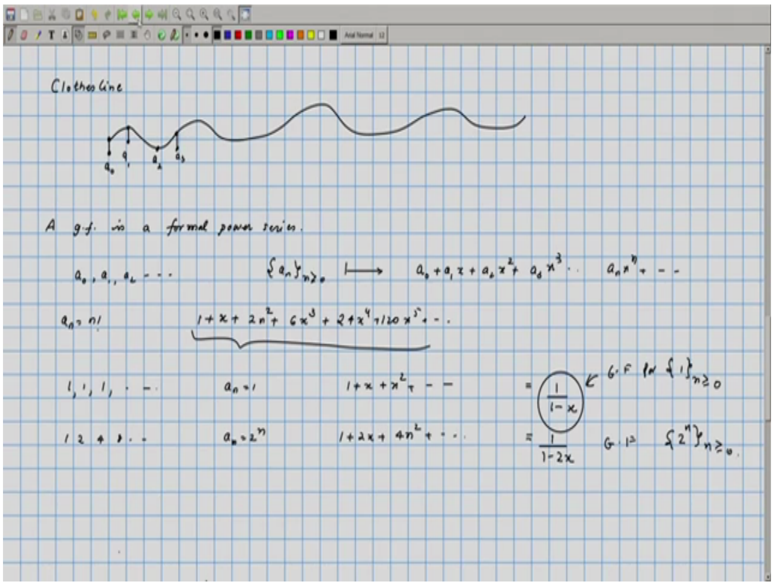
So, this recurrence relation basically gives a sequence of numbers. So, if you write down the sequence, the first number is 1, the next number is 3 times 1 that is going to be 3, 9 and so on. So, in this recurrence relation, it is very easy to give what is the nth term, the nth term by means of definition is going to be 3 times the previous term or the n minus first term, then we can therefore write a n in a closed form as equal to 3 raise to n, okay, so this can be viewed as a solution for this recurrence relation.

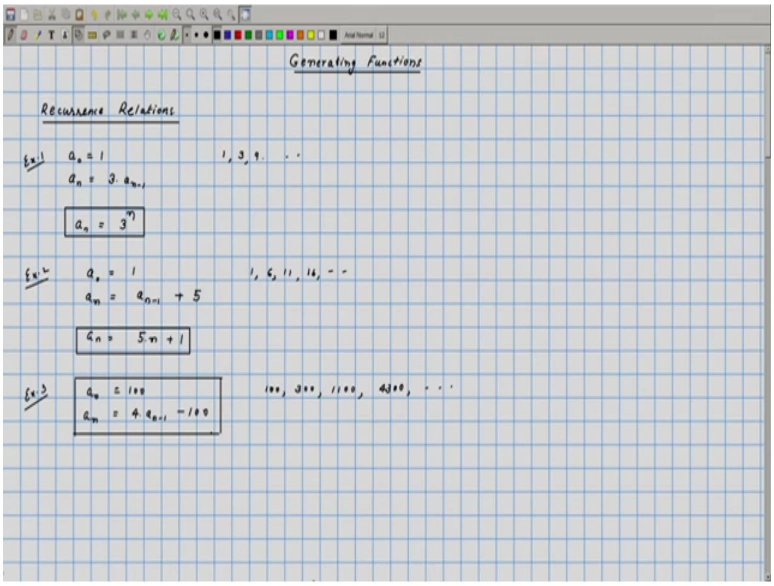
We could take another recurrence relation where say we take a0 is equal to 1 and a n is equal to let us say a n minus 1 plus 5. So, now what is the nth term, this is the arithmetic progression, this is also easy to determine then the nth term, if you write down the sequence you get the sequence 1, 6, 11, 16 and so on. Therefore, a n is just going to be 5 into n plus 1, okay. So, you can think of this is a closed form solution to the recurrence relation that we are given.

We will look at a slightly more complicated recurrence relation. Let us say a0 is equal to 100 and a n is defined in terms of a n minus 1 as let us say 4 into a n minus 1 minus let us say 100. Now it is not so straightforward to determine just by mere inspection it is difficult to say what is the close form solution. We could say that a0 is 100, the next term is going to be 400 minus 100 that is 300 and the next term is going to be 1100, next term is going to be 4400 minus 100 that is 4300 and so on.

So, the general term is going to be slightly difficult to write down by mere inspection, okay. So, how do we find the solution to the nth term? For this we will use a very powerful method known as generating function that can be used to solve these kind of recurrence as well as much more complicated recurrence.

(Refer Slide Time: 04:04)





So, let us see the basics of generating function, so what basically is a generating function. You can think of the following analogy you can think of it as a clothes line. So, let us say the starting point of the clothes line is known, okay, this is the infinite clothes line and the first number of the sequence is put at the first position, the second then a1 is attached somewhere, a2 is attached somewhere when immediately after a1 and then a3 is attached and so on.

The reason why we are thinking about the clothes line analogy is we could wrap around the clothes line, we could tangle it and then you could pass it on somebody else and all they need to do is to stretch the clothes line and look at its starting point and from the clothes line they can just read of the values of the sequence.

So, we have an infinite sequence and we can think of the generating function as a clothes line on which various values are held at specific position, more formally the mathematical way to think of this is as a formal power series. So, a generating function is a formal power series.

So, if you have a sequence, let us say a0 followed by a0, a1, a2 so on. So, in general let say a n, so given a sequence, the generating function corresponding to this is the sequence is the formal power series given by a0 plus a1x plus a2x square plus a3x cube and so on.

So, a and x raise to n plus 1. So, given any sequence we can associate a formal power series with it. In case of some sequences the formal power series can be compressed and written in nice formats whereas for some other things the formal power series, it may not have a closed form expression.

For example, if your sequence was let us say the factorial sequence, so a n is equal to n factorial, if you look at this sequence that is going to be 1 plus x plus 2x plus 6 sorry 2x square plus 6x cube plus 24x raise to 4 is 120 x raise to 5 and so on. So, it is difficult to imagine some place where this sequence or this particular power series can be thought of as converging to some particular value, but we will not bother about the convergence aspects. This expression is basically what we will call as the formal power series or the generating function corresponding to the sequence a0, a1, a2 and so on.

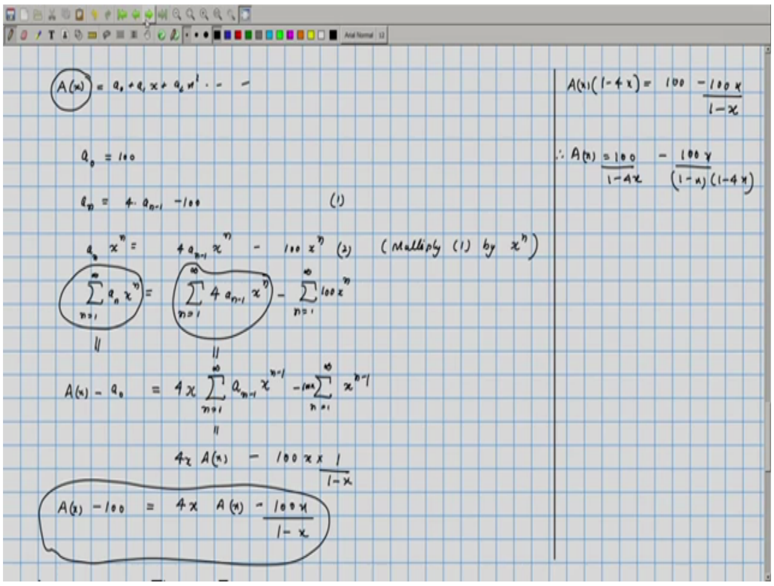
So, let us take some more examples, if you look at let say the sequence 1, 1, 1 so on. So, in other words a n is equal to 1 corresponding to it the formal power series would be 1 plus x plus x square and so on. So, this if we assume x to be less than or equal to 1 over the absolute value of x to be less than or equal to 1, we can just simply write it as 1 by 1 minus x.

So, this you can think of it as it is rolling together the generating function into a simpler expression, okay. So, instead of telling the whole power series, we can just say that we are looking at the generating function 1 by 1 minus x when we were thinking about the sequence 1, 1, 1, 1 and so on.

And if you had let us say the sequence 1, 2, 4, 8 so on, so the nth term you can think of as 2 raise to n, that will correspond to the sequence 1 plus 2 x plus 4 x square plus so on and this we can think of it is the I mean if you assume suitable values for x, you can say that it will be 1 by 1 minus 2x, okay.

So, this is the generating function corresponding to 1 and this is the generating function for 2 raise to n, n greater than 0, okay. So, now that we have the definition of generating functions in place, we can think of how to solve this particular recurrence relation using generating functions, okay.

(Refer Slide Time: 09:01)



So, little bit about the notation, whenever we think of a sequence a0, a1, a n its generating function we will denote by a of x. So, a of x is equal to a0 plus a1 x plus a2 x square and so on.

So, now what we have is we have this following recurrence relation a0 is equal to 100 and a n is equal to 4 times a n minus 1 minus 100. Let us call this as equation 1, if we multiply both sides by x raise to n we will get a raise to n, x raise to n is equal to 4 a n minus 1 times x raise to n minus 100 x raise to n, okay, and then we will sum this equation 2 for values of n starting from 1 to infinity, okay.

So, both sides if we sum up the LHS and the RHS summation n is equal to 1 to infinity a n x raise to n is going to be equal to summation 4 a n minus 1 x raise to n; n is again varying from 1 to infinity. The reason why we took n is equal to 1 to infinity instead of 0 to infinity is the term a n minus 1 would not have made sense when n is equal to 0, okay and furthermore this I mean in other words the equation 1, which we started off holds only for values of n greater than or equal to 1, okay.

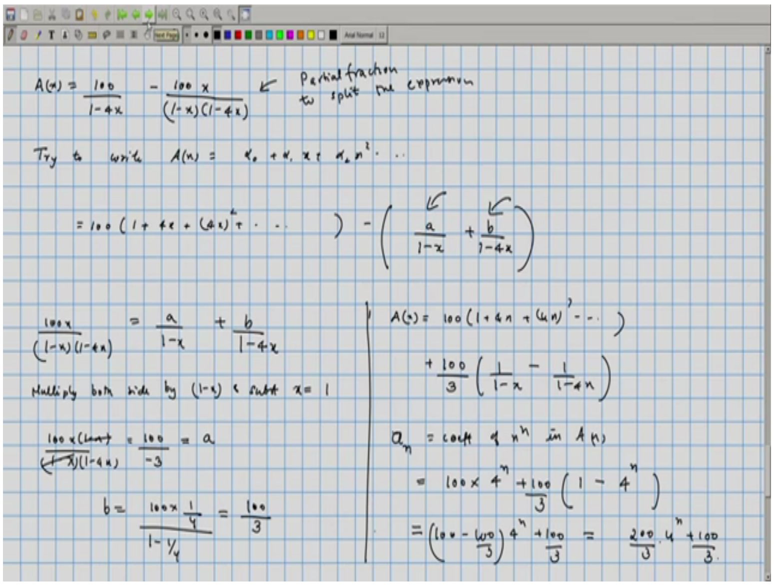
So, n minus summation n is equal to 1 to infinity, 100 x raise to n. So, if we look at look at this equation closely, we can see that some parts of these equations is starting to resemble ax, for example, the LHS term is almost the term on the LHS is almost equal to ax just that the first term is missing.

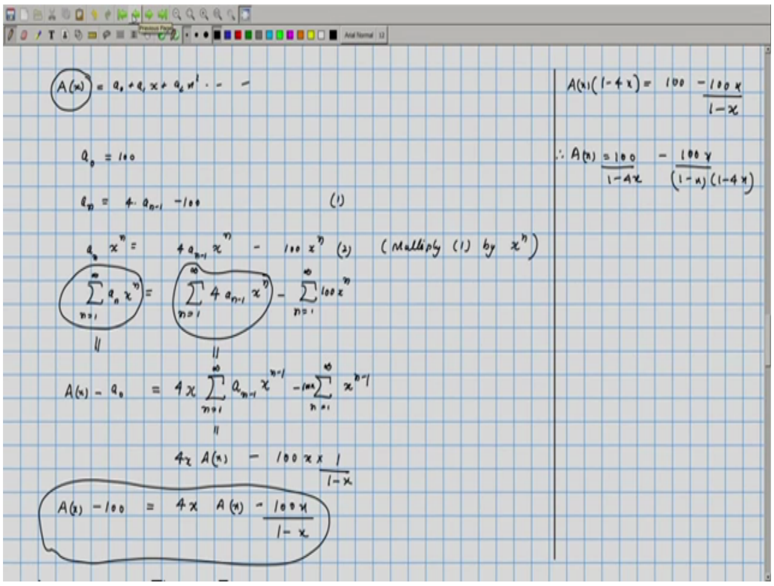
So, this we can rewrite as so this is equal to ax minus a0 whereas the second term here, this is equal to if we take x outside and 4 outside, what we have is summation n is equal to 1 to infinity a n minus 1 x raise to n minus 1. Now, since the n are starting from 1 so the first term of the summation is a0 x raise to 0, the next term is a 1 x raise to n and so on and so this will basically be equal to 4x ax, okay.

And the the last term on the LHS is summation n is equal to 1 to infinity, we can take 100 x outside and we will get x raise to n minus 1, okay, and that is going to be equal to minus 100x into the summation is 1 by 1 minus x, 2 as 100 we will get ax minus 100 is equal to 4x times ax minus 100x by 1 minus x.

So, whatever recurrence we had, we had used that to write a new equation involving the generating functions of the sequence. This we can rewrite as ax into 1 minus 4x is equal to 100 minus 100 x by 1 minus x and therefore ax is equal to 100 by 1 minus 4x minus 100x by 1 minus x into 1 minus 4x. So, now instead of thinking about the recurrence relation, we have generating function of x, which basically contains all the information corresponding to the sequence in a succinct form.

(Refer Slide Time: 13:50)





So, what we know is ax is equal to 100 by 1 minus 4x minus 100x divided by 1 minus x into 1 minus 4x. Now, from this, how do we get the value of the nth term? So, we can look at this generating function carefully and we can try to determine the x raise, the coefficient of x raise to n if you write this as a0 plus if you were to write this as let us say alpha 0 x raise to 0 plus alpha 1 x raise to 1 and so on.

If you could write it in such a format and alpha n must be equal to x raise to n, so try to write ax as alpha 0 plus alpha 1 x plus alpha 2 x square and so on, okay, so there are two terms 100 by 1 minus x we can simply write it as 100 by 1 plus 4x plus 4x whole square plus so on because 100 by 1 minus 4x is a sum of a geometric progression, if you think of 1 plus 4x plus 4x square and so on that is a geometric progression that will sum up to 1 by 1 minus 4x.

And the next term is little more tricky, but we will use what is known as the partial fractions method. So, we will write this as, this is a by 1 minus x plus b by 1 minus 4x, okay. If you write it in this format, this expression easily can be thought of as a geometric series and their expansion is easy to write, okay, so we want 100 x by 1 minus x into 1 minus 4x to be equal to a by 1 minus x plus b by 1 minus 4x, okay.

Now, what are the values of a and b, which will make this equation true. We can multiply both sides by 1 minus x and substitute x is equal to 1, that will give the value of a, so when you multiply that the left hand side will be 100x into 1 minus x by 1 minus x into 1 minus 4x, these gets cancelled.

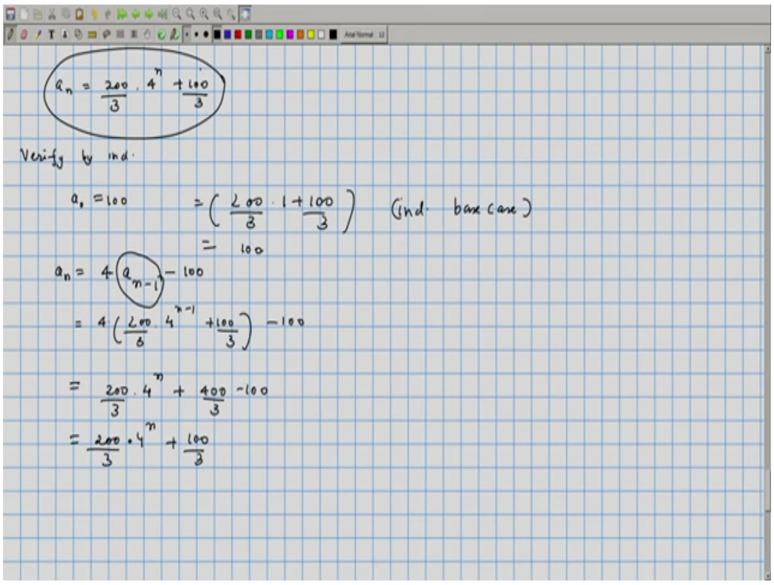
Now, when you substitute x equals 1 you will get 100 by 1 minus 4, that is minus 3, okay, and on the right hand side 1 minus x term cancels off with so 1 minus x terms will cancel off whereas b by 1 minus 4x into 1 minus x when you put x equals 1, the 1 minus x term becomes 0 and hence the right hand side of this equation is going to be equal to a.

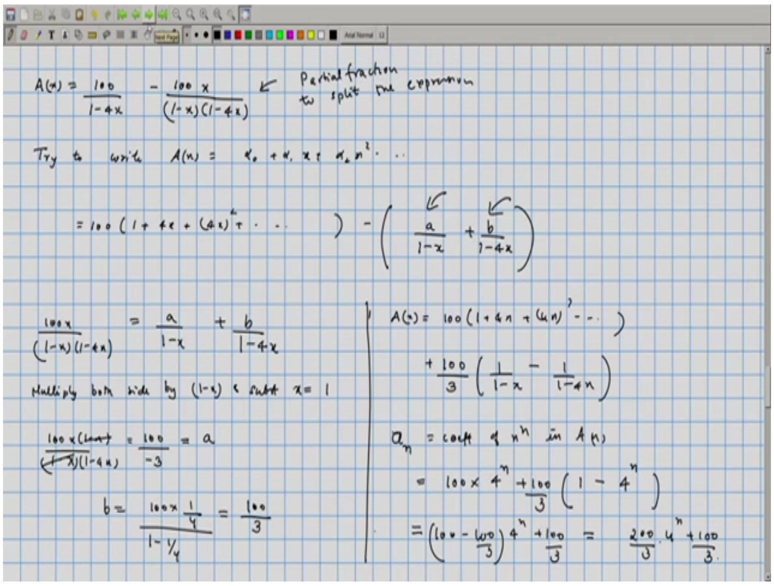
So, this 100 by minus 3 is going to be a and similarly if you multiply both sides by 1 minus 4x and put x is equal to 1 by 4 what you will get is the value of b. Therefore, we can write b is equal to 100 into 1 by 4 divided 1 minus 1 by 4, that is going to be 100 into 1 by 4 by 3 by 4 that is 100 by 3 and therefore we can simply write this generating function ax as 100 into 1 plus 4x plus 4x square so on minus 100 by 3 say plus 100 by 3 into 1 by 1 minus x minus 1 by 1 minus 4 x.

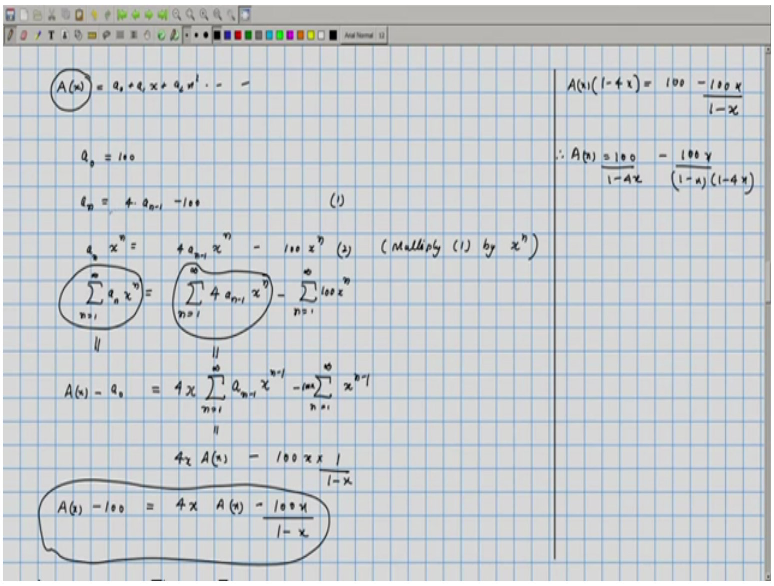
So, this can be written as so if you look at the nth term, so from this you can read off the nth term, so a n is equal to coefficient of x raise to n, that is going to be equal to 100 into 4 raise to n plus 100 by 3 into 1 by 1 minus x, the term was going to be 1 minus 1 by 1 minus 4x that is again will be minus 4 raise to n. So, this is going to be equal to 100 minus 100 by 3 into 4 raise to n plus 100 by 3, so the nth term is going to be 200 by 3 into 4 raise to n plus 100 by 3.

So, just for this particular case, we will verify that our answer is actually agreeing with generating function, I mean with the recurrence relation.

(Refer Slide Time: 19:06)







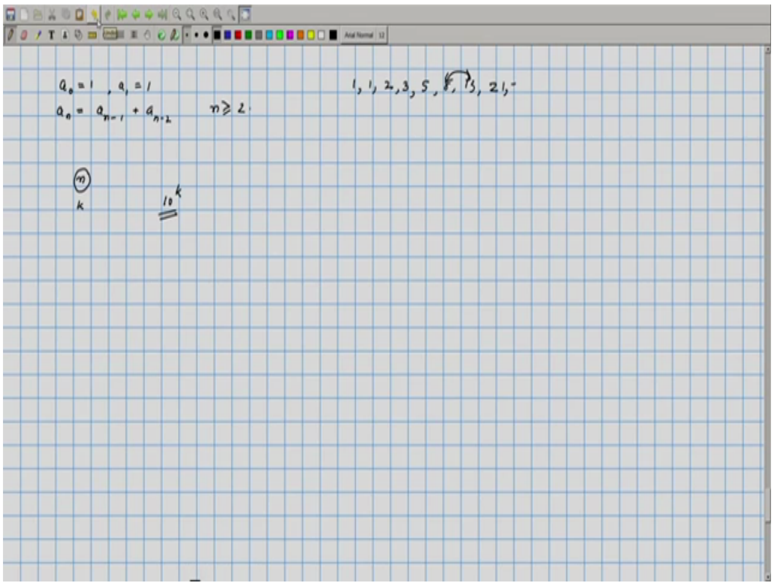
So, we got an is equal to 200 by 3 into 4 raise to n plus 100 by 3, we will verify by using induction, so if you look at a0, which is given to be 100 by the recurrence relation, if we substitute in formula, we will get 200 by 3 into 4 raise to 0, that is going to be 1 plus 100 by 3 that is 300 by 3 that is going to be equal to 100, okay.

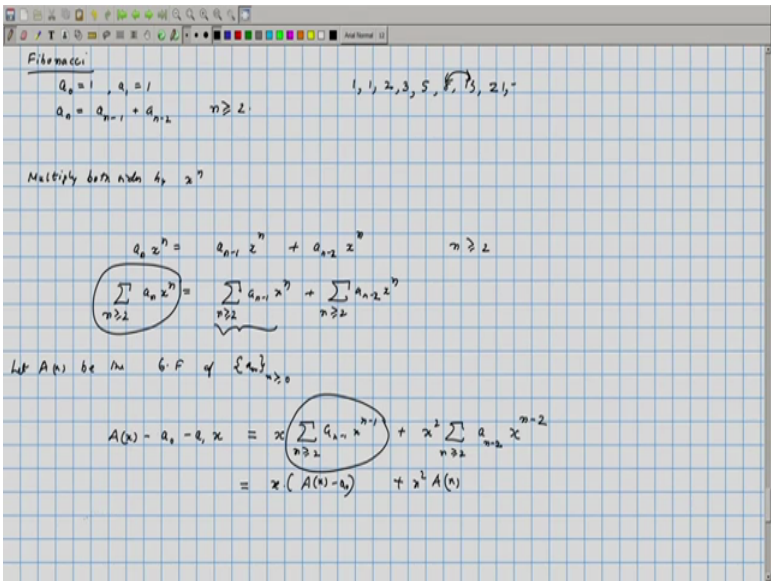
So, that is induction base case and we if we assume that the formula is true up to n minus 1 we need to check that a n is equal to 4 times a n minus 1 minus 100, so up to n minus 1 we will believe that expression was correct and therefore this is going to be equal to 4 into 200 by 3 into 4 raise to n minus 1 plus 100 by 3 minus 100.

The recurrence relation gives this is the answer and that is going to be equal to 200 by 3 into 4 raise to n plus 400 by 3 minus 100. So, that is going to be equal to 200 by 3 into 4 raise to n plus 100 by 3 that agrees with the formula that we had computed via the generating function method.

So, we have seen the generating function method used to solve a particular recurrence relation. We will see one more recurrence relation just for practice slightly different from the one that we had seen. Here, we had just one term 4 times n minus 1 and there is a constant term.

(Refer Slide Time: 21:07)





So, we will look at a more famous recurrence relation namely the Fibonacci recurrence, so if we think about the Fibonacci sequence, the sequence is given as a0 is equal to 1 and a1 is equal to 1 and a n is given by a n minus 1 plus a n minus 2 where n is greater than or equal to 2, okay.

So, the sequence if you write the first few terms, it is going to look like 1, 1 next term is the sum of these two 2, next term is going to be 3, 5, 8, 13 and so on, okay. So, add up the two terms their sum is the next term 21 and so on. So, we can find the nth term by just repeating this process, if you were to write a computer program to do this.

If n is input and if we want to write the nth term of the sequence, it is going to take a long time, it is going to take time, it is an exponential time algorithm because the input is given as n which is given in decimal, so this will have some k digits and the time taken is going to be proportional to n which is something like 10 raise to k.

So, if you just apply the recurrence relation we going to get an exponential time algorithm to compute the value of a n or f n here, the nth Fibonacci number. So, let us see how we can do this via generating functions. Method is identical to the one that we had looked at earlier, we look at the main recurrence relation, which says a n is equal to a n minus 1 plus a n minus 2.

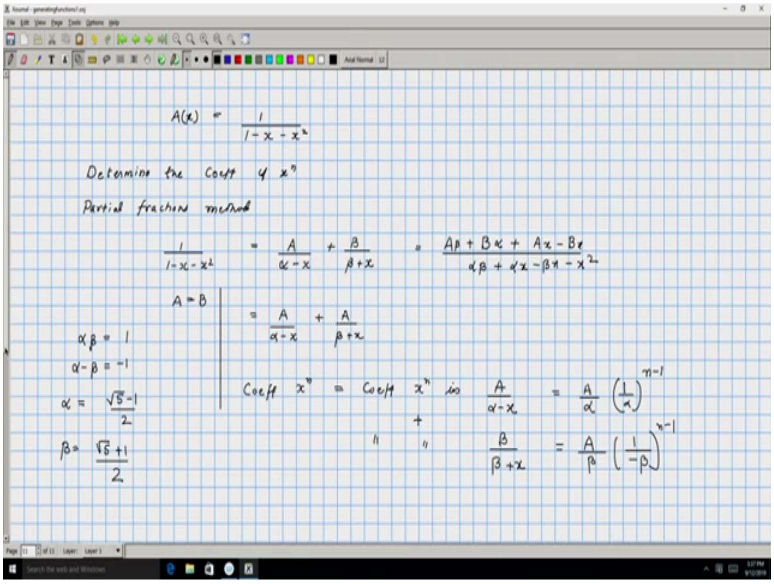
So, multiply both sides by x raise to n, and we will get a n x raise to n is equal to a n minus 1 x raise to n plus a n minus 2 x raise to n, so this equation is valid only for n greater than or equal to 2. So, we sum this up for all values of n greater than or equal to 2, so sum over n greater than 2 a n x raise to n, that is going to be equal to the summation a n x raise to n, n minus 1 x raise to n the n greater than or equal to 2 plus summation n greater than or equal to 2 n minus 1 x raise to n, okay.

Now, the first term is going to be if you assume that ax, so let ax be the generating function, then the first term in this expression is going to be equal to ax, there are two terms missing namely a0 and a1 x, so this can be written as ax minus a0 minus a1 x.

This is going to be equal to the first summation in the right hand side. It resembles ax if you take an x outside, you will get this as summation n greater than or equal to 2, a n minus 1 x raise to n minus 1, and the second term if we take x square outside this is going to be summation n greater than or equal to 2 a n minus, sorry, this is a n minus 2, a n minus 2 x raise to n minus 2 and this is going to be equal to x times and the summation every term in generating function is present except, the when n equals 2 what you get is A 1 x raise to 1, the next term is a2 x square, only term missing is A 0.

So, this is going to be ax minus a0 plus x square into here n starts from 2 and a n minus 2 is going to be 0 times x raise to 0 is a1 x raise to n, a1 x raise to 1 and so on, so this is going to be equal to ax. We can bring all the ax's together and substitute the values for the constants a0 and a1. So, look at the equations involving the generating functions and by rearranging terms and plugging in the value of a0 and a1 we will get ax the generating function as equal to 1 by 1 minus x minus x square.

(Refer Slide Time: 26:00)



Now, what remains is to determine the coefficient of x raise to n, it is if we expand out this particular function 1 by 1 minus x minus x square what will be the coefficient of x raise to n? So, we will use the method of partial fractions. So, we will write 1 by 1 minus x minus x square as A divided by alpha minus x plus B divided by beta plus x, okay.

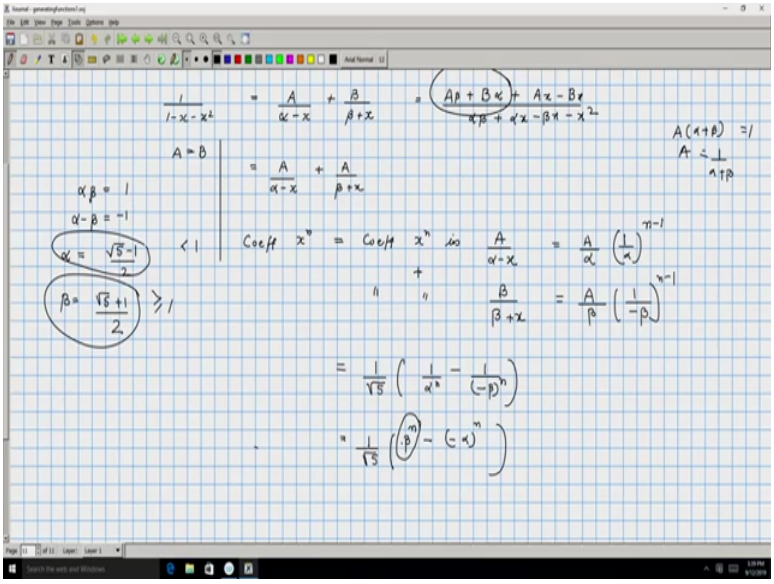
Now, A, B, alpha and beta they have satisfy some conditions in particular Ax plus I mean if you multiply them out, what you get is A beta plus B alpha plus A minus Bx divided by alpha beta plus alpha x minus beta x minus x square. So, comparing the terms we will get A is equal to B because Ax minus Bx should be 0 because there is no coefficient for x in the numerator.

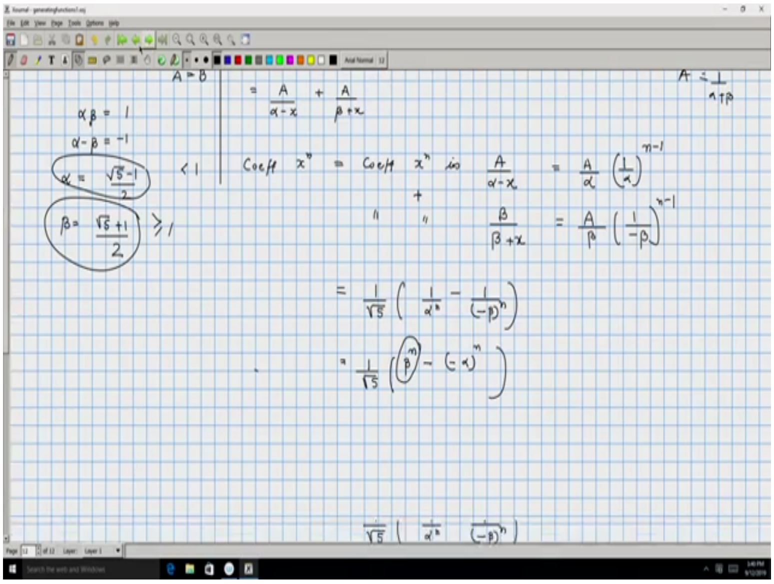
So, A is equal to B, and therefore we can simply write this is this is equal to A by alpha minus x plus A by beta plus x where alpha and beta has to satisfy additional conditions by comparing the coefficients we will get alpha beta should be equal to 1 and alpha minus beta should be equal to minus 1.

So, those will be the roots of this particular denominator polynomial, therefore alpha will be equal to root 5 minus 1 by 2 and beta will be equal to root 5 plus 1 by 2, okay. Now, if we plug in these values we can compute the coefficient of x raise to n, okay.

So, the coefficient of x raise to n will be the coefficient of x raise to n in A by alpha minus x plus coefficient of x raise to n in B by beta plus x and this is equal to A by alpha into 1 by alpha raise to n minus 1 and for the other term the coefficient will be B by beta that is going to be B is same as A so that is A by beta into 1 by minus beta because here the term is beta plus x minus beta raise to n minus 1.

(Refer Slide Time: 29:48)





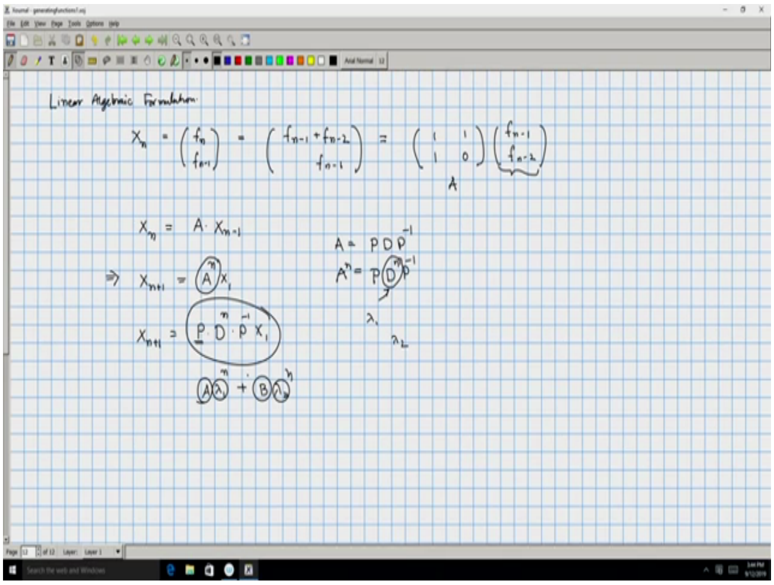
And we can further infer the value of A by observing that A times alpha plus beta that is this particular term that should be equal to 1, so A is equal to 1 by alpha plus beta and alpha plus beta is equal to root 5, so 1 by root 5 that is A times minus 1 by minus beta raise to n.

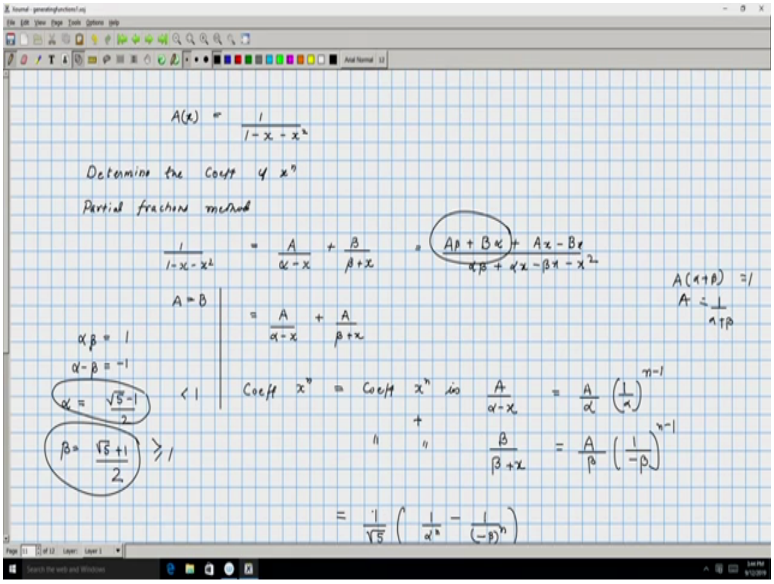
We know that alpha and beta are inverses of each other so this can be written as 1 by root 5 into beta raised to n minus minus alpha raise to n, okay, because 1 by beta is minus alpha, and observe that beta is a quantity, which is greater than 1 and alpha is a quantity whose absolute value is less than 1 because root 5 minus 1 by 2 you can show that this is going to be less than.

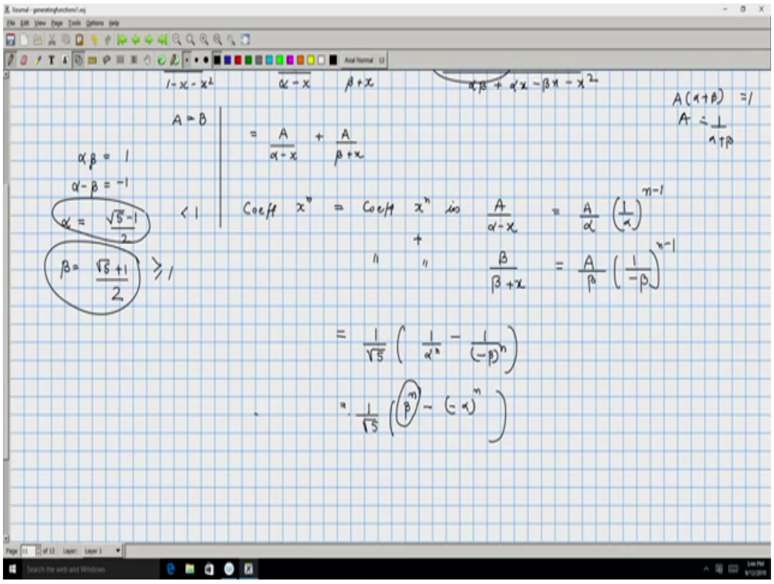
So, the dominant term is going to be beta raise to n, so although both these numbers alpha and beta are irrational numbers they will the irrational part will cancel out with each other and if you want an approximation, okay, you can even ignore the part alpha raise to n after a certain point because alpha being less than 1 it will quickly go towards 0 and if you just round off whatever is the integer part of beta raise to n, if you round off to the nearest integer, you will get the correct answer the exact answer without any approximations.

Now, we were counting some combinatorial object namely the we were estimating the Fibonacci the nth Fibonacci number we saw that the strings numbers alpha and beta are coming in.

(Refer Slide Time: 32:00)







We can also see that the same thing arises in another way as well. So, let us look at the linear algebraic formulation. So, now instead of thinking of one particular Fibonacci number, instead of thinking of the nth Fibonacci number, we will think of a vector xn which denotes the pair of Fibonacci number namely fn s and fn minus 1.

So, this pair is what we will denote by a vector of size 2 and note that this vector will be equal to fn minus 1 plus fn minus 2 and fn minus 1 will be just define minus 1 and therefore this can be written as 1, 1, 1, 0 into fn minus 1, fn minus 2 because you can say that this matrix A multiplied by this particular vector will give you fn, fn minus 1, okay.

So, we can simply write this as xn is equal to A into xn minus 1 and therefore you can repeatedly apply this and say that xn is equal to A raise to, let us think of xn plus 1, so xn plus 1 is equal to A raise to n times x1, okay. So, A raised to n is going to be a matrix is going to be 2 cross 2 matrix, if we can somehow estimate that matrix from that information, we can determine the nth Fibonacci number.

Now, in order to determine the nth Fibonacci, I mean the nth power of this matrix, we can diagonalize it, so suppose we can write A as PDP inverse then A raise to n is going to be equal to P D raised to n P inverse and then we can write xn plus 1 is equal to P into D raise to n times P inverse times x 1, okay.

And D is now going to be a diagonal matrix and this diagonal matrix it is diagonal values if it is, lambda 1 and lambda 2 this entire expression will basically be something of the form A lambda 1 raise to n plus B lambda 2 raise to n, of course A and B are not going to be the same A and B from there but it is going to be something which is dependent on what the P D etc, okay.

But you can see that the lambda 1 and lambda 2 which comes here will be equal to will be the same as what we saw earlier, in fact it will be our alpha and beta and the coefficients will be 1 by root 5, okay. We will stop here and learn more about generating functions in the next lecture.