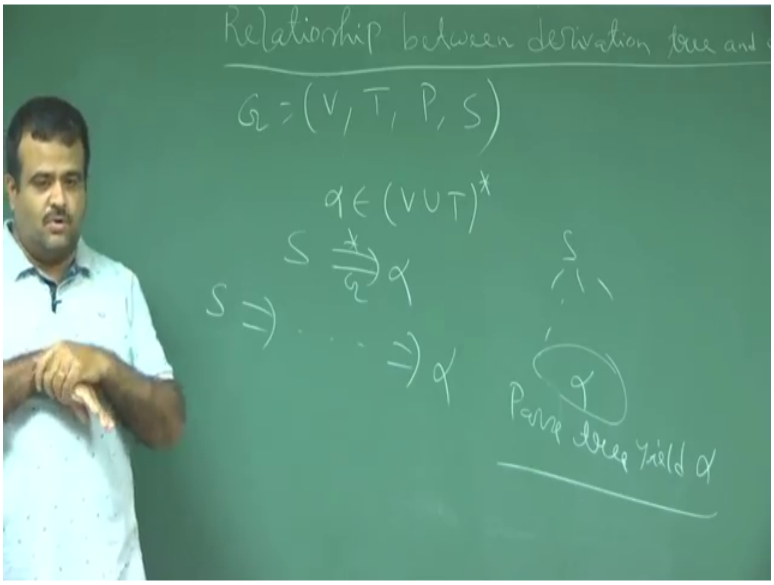
Introduction to Automata, Languages and Computation
Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology, Kharagpur

Lecture – 41
Leftmost and Rightmost Derivations

Ok. So, we are talking about the relationship between the parse tree or derivation tree and the derivations.

(Refer Slide Time: 00:27)

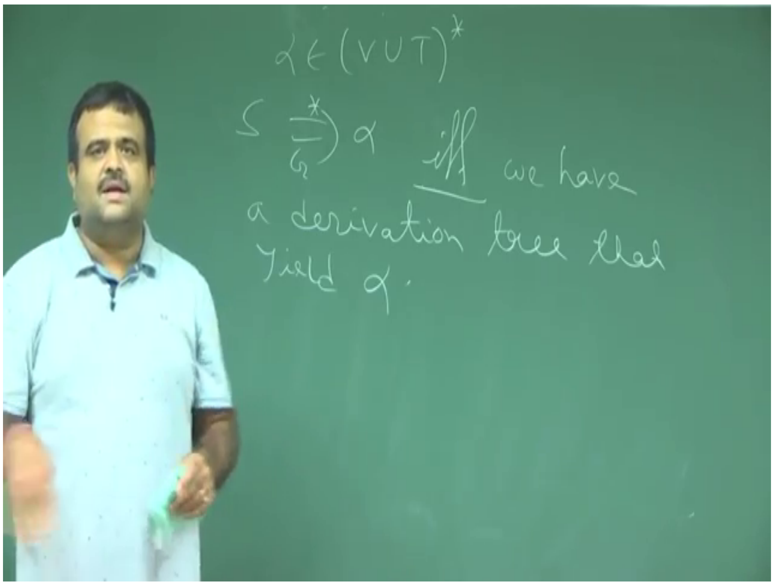


So, we have seen this we have a grammar context free grammar which is a 4 tuple V T and P S where, V is the set of all variables which is finite and T is the set of all terminals and P is the set of all productions or the rules. And, S is a variable special variable which is called start variable ok. And, we have we know that if a alpha which is a string of variable and terminal then alpha is derived from S by G. If we start from S and we follow the productions are in P and after some step, if we reach to the alpha then we say our S is alpha is derived from S; alpha need not be a terminal, but if it is language you are looking for a terminal string.

But, we can take some intermediate state also intermediate string of I mean the variable and the terminal the string ok. So, this is the derivation and we know in the last class we have seen if alpha derive from S then we have a parse tree which is from S if you have a parse tree which yields alpha. So, alpha the parse tree or the derivation tree parse tree that yield alpha ok. This is if and only if we have a parse tree which is yield alpha then alpha is can be derived from S. Because, basically in the parse tree we are following the productions in productions of P that is the way we form the tree parse tree and here also you are following the productions of tree.

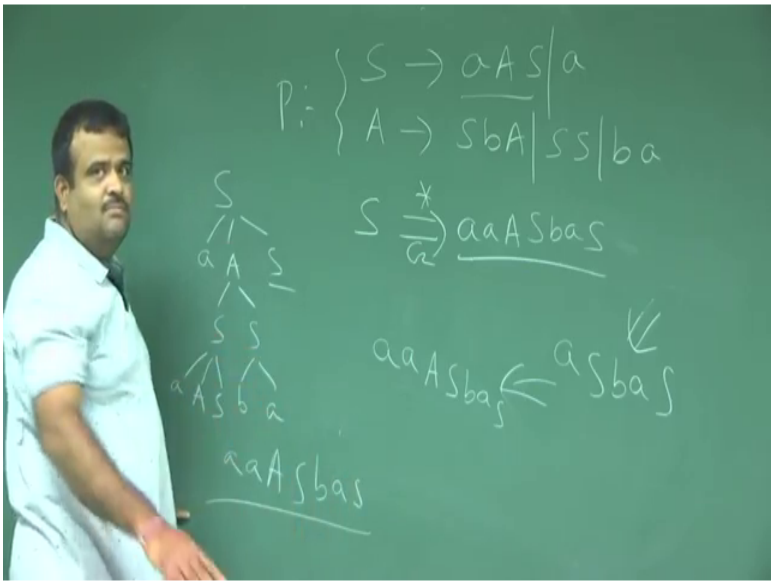
So, that are they are basically the same. So, we are just having the tree representation of the this process of the production process; so that is the relationship.

(Refer Slide Time: 02:57)



So, so this theorem we have seen in the last class that if we have a if alpha is belongs to alpha is a string and S is derive alpha is derived from S if and only if we have a we have a parse tree or derivation tree that yields alpha that yields alpha. So, this is the if only leaf if we have this then we must have a tree which is yielding alpha; yielding means if we count the left to right the leaf node that is the that three is yielding by the parse tree and the vice versa. If we have a parse tree basically it is coming from the productions; so, they are basically same. So, let us take an example of a grammar.

(Refer Slide Time: 04:08)



Suppose we take a grammar like this S is going to or a and A is going to S b A or SS or ba. So, these are the productions P, and what is the V? V consists of V is the set of variables you have only two variables S and A and what the terminals the alphabets. So, a b a b are the terminal and S is the starting state and this is the rule P is the production rule. So, this is a given grammar ok. So now, if we draw a parse tree like this say if S is we take this rule S is going to A S. Then this A we can take this rule S S a is going to say S or we can take this one S A is going to S S.

So, S S S then we can take one of the S by this say b a so, a S this S say we are taking b a S. Now, suppose this S we are again replacing by this a so, S is replace by S b A b a S. So, suppose this is our alpha so, alpha is say a S b A b a. So, this belongs to b this is a string consists of variables and the terminals, but when we talk about language we talk about context free language then we talk about only the string of terminals. There is no variable in the language string, but anyway this is the intermediate state we can say.

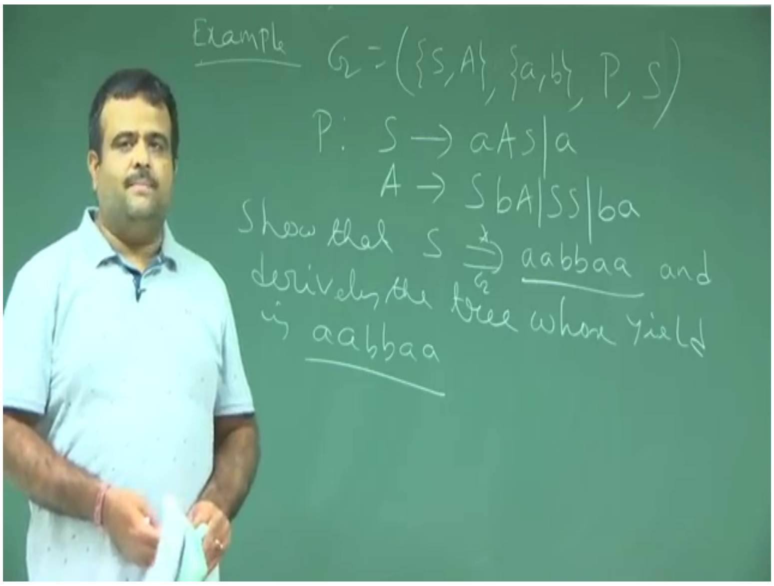
So, it consists of variable and the string by variable and the terminal string. So, this is this alpha is derived from this. So, you can write S is derived from the alpha is derived from a. So, a S b A b a; now we want to see whether we can get a parse tree to have this. So, the way we use the production and we will draw the tree by the similar way like this. So, we have S over here so, first we use the rule S is going to this. So, we can draw the parse tree a S A S then we follow this A A is going to S S.

So, we will just follow the way we derive the same way, if we draw the tree then that will give us the yield that same alpha that is the that is the example. So, this S then replace this S by no we replace this S by b a and then we replace this S by again this one a capital A S. Now, if we so what is that so, this is a tree parse tree which is not ending with the terminals, but still it is we are in the intermediate step. So now, what it is yielding? It is yielding you have to read from left to right.

So, a sorry a so, which S we are replacing a S is going to; so, this is our S just a minute. So, this S this S is going to SS b a sorry this we had a mistake over here. So, this S no this is not alpha; so, this S we apply the rule of A, but this is S. So, you have to apply a a A S then b a S this is the our alpha sorry a a A b a S this is our alpha and this is coming from a a A S b a S. So, this is derived from this parse tree and the vice versa; if we derive from this parse tree means we are applying some rules then can easily derive this by applying rule on one by one.

So, that is the example; so every parse tree will yield a string which can which is derived which can be derived from the starting variable.

(Refer Slide Time: 09:29)

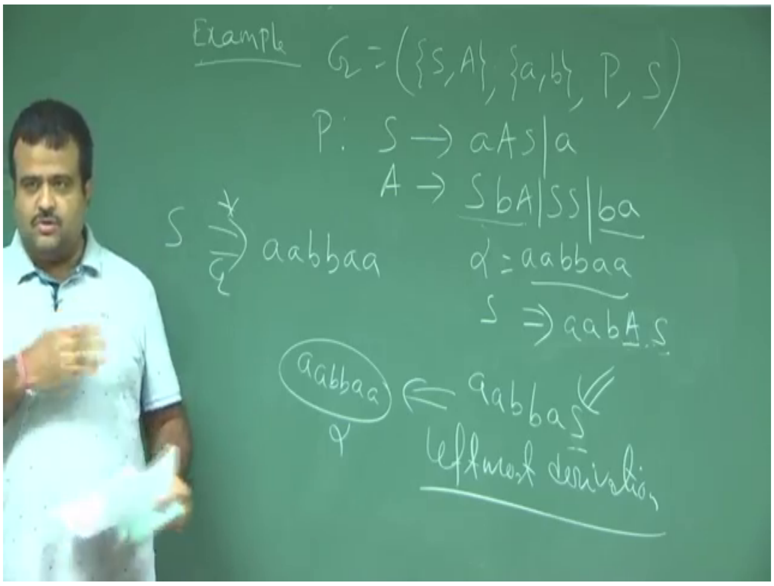


So, we will take some more examples with the same grammar. So, let us take this grammar G which is say S comma A and terminals are say a b and we have P this is 4 tuples. And, the P just now we have seen P consists of the rules or the productions a or A and A is going to S b A SS or b a. So, these are the 3 rules in where A is going and A production and this is the S production S is going to a this ok.

Now, you want to take a so, that we can derive a string a a bb a a this is our alpha and here this string called only consists of the terminal. So, this is a string of terminal there is no alphabet, but it could be mixture of terminal and the variables I mean. So, there is no variables over here it could be terminals or in the variables ok.

And so, we have to derive that and we have to find the parse tree and the derive the tree and the derive the tree; derive the tree parse tree whose yield is yield is yield is this alpha a a b b a a ok. So, this is our alpha so, this alpha you want to generate by the derivation applying the rules as well as we want to have the corresponding parse tree. So, let us try that.

(Refer Slide Time: 11:57)



So, let us first this is our alpha here alpha is a string of only the terminals ok. So, how we can derive alpha? So, we can start with S then you have to apply the rules. So, you can apply any one of this; we will apply the first rule ok. Now, this is actually we will use the what is called leftmost derivation. So, leftmost derivation means every time we will apply the production for the leftmost variable. So, here we have two variable S and A so, we will apply the production for A.

So, that is the left most variable. So, that way the derivation is called leftmost derivation. So, always we will apply the production of the variable which is the leftmost one. So, for that we have to apply the production some production on for A. So, there are main three productions on A so, we will choose one of this suppose we are choosing this one. So, this a is going to now capital A is going to a S b A and S.

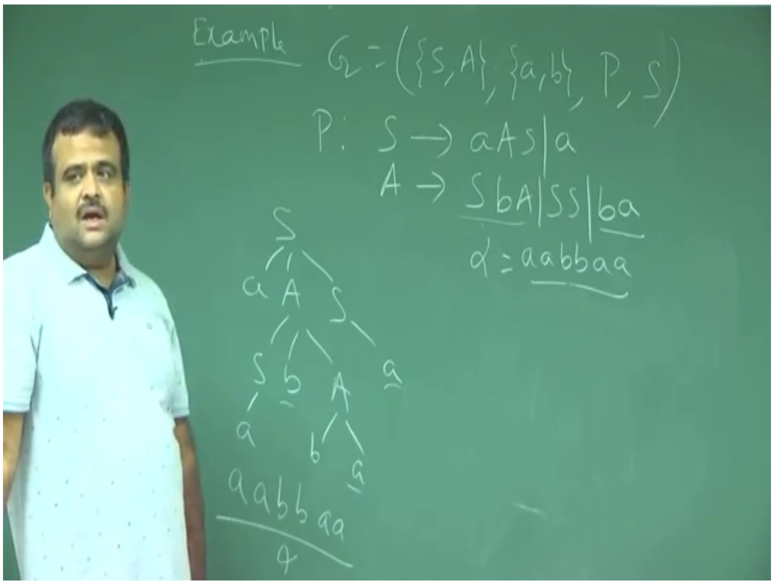
Now, again since we are in leftmost derivation so, again we take the leftmost variable this is again this S. So now, we can take any one of this rule; suppose we are taking the this S is going to a. So, this is a a b A S. Now, again this is a leftmost derivation for which is the left most variable these are all terminals only variable is a and this capital A and this S capital S.

So, then capital A is in the leftmost position so, we take the production in capital A. So, we can take any one of this production so, suppose we are taking this production a is going to b a. So, if you apply this a a b then A is going to b a then S. Now, there is only one variable which is S capital S which is in the leftmost there is only one variable; now we can say a is when S is going to a so, a a bb aa; so, this is our alpha. So, alpha is this a a bb a a is derived from this S by using the production in P in a leftmost way. So, this is called left most derivation leftmost derivation.

So, we are always applying the production in the leftmost variable of the string every time. So, this is the so, this alpha can be derived from S. So, we can write this a a b b a a can derive from S and in particular it is a leftmost derivation we will see whether we can have a rightmost derivation on this. So, every string which is can be derived from S is have both the leftmost and the rightmost derivation we will see that. Now, first of all you want to see the parse tree which is generating this; so let us draw the parse tree of this.

So, parse tree when we draw will follow the same way the sequence which we have a production sequence which have applied on this to get alpha. So, if you do that so, let us get the parse tree.

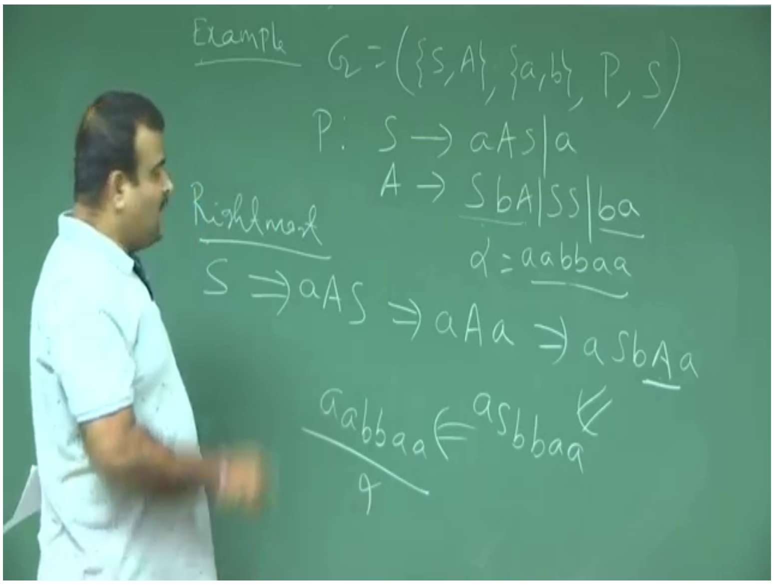
(Refer Slide Time: 15:47)



So, we start with S and we use this and then we use the leftmost derivation and A is going to this S b A and this S again leftmost derivation we use. So, this S is going to a and this A is b A is going to b a again leftmost derivation this is a this is the parse tree. So, what it is yielding? a a so, we just read the all the leaf node these are the leaf node which has no child; we just read the all the leaf nodes from left to right.

So, a a b b a a so this is nothing, but our alpha. So, alpha is yielding so, this is the tree which is those whose yield is this alpha. So, this is one derivation; now we can have another derivation we will see whether we can get it by right most derivation or not this alpha.

(Refer Slide Time: 16:59)

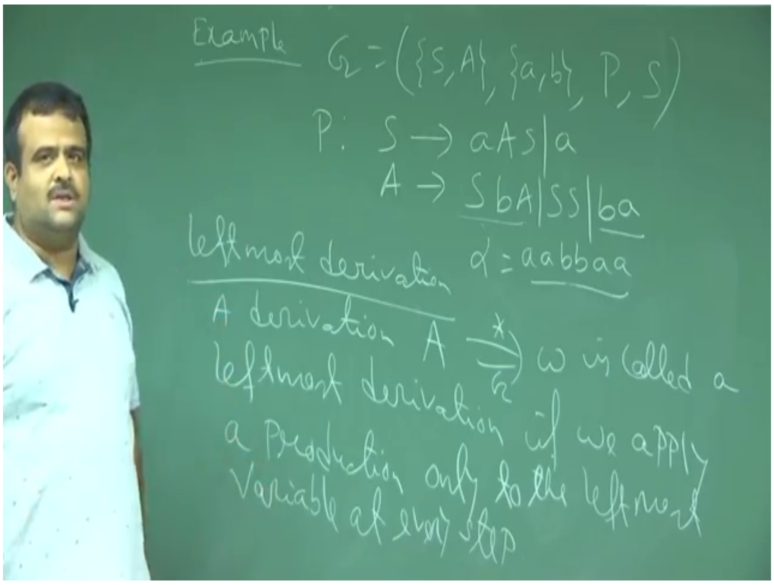


So, let us try that this is rightmost derivation. So, if we apply rightmost derivation we start with S we will take one of this production in S ok. So now, who is the right most variable? This S is the rightmost variable so, if S is the rightmost variable you have to apply that derivation for S; I mean you have to take the production of this S. So now, we have two option we can take a this way ok. So now, here from here we can go to again we have only one variable S a.

So, we can take the derivation in a suppose we are taking this one a S b A. So now, again this is the rightmost derivations so, we have to take the derivation in A. So, if you take that we take the derivation in a. So, we can yeah so, a is going to b a. So, just a minute so, a we have a over here sorry this a. So, a is now this is the rightmost derivation rightmost variable; so, we have to take a derivation we have to take a production in that rightmost variable a.

So, if you take this one it is going to a S b b a a; now this is only one variable this is also rightmost variable. So, we can go to this can go to a so, a a b b a a this is our alpha. So, alpha can derive from the rightmost way also. So, that there are basically two ways we can get the alpha. So, one is leftmost derivation another one is rightmost derivation; so, let us formally have that.

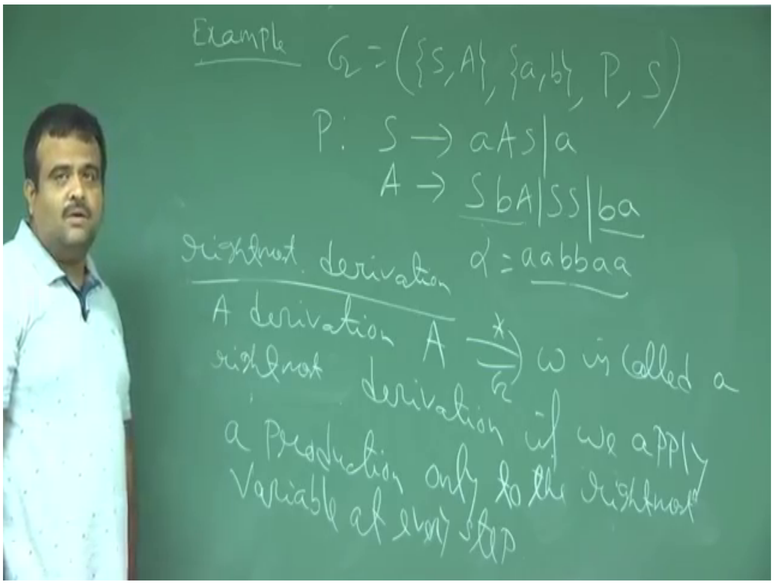
(Refer Slide Time: 19:22)



So, yeah so, let us just formally define left most derivation. So, leftmost derivation a derivation A is going to w; w is a string is called a leftmost derivation if we apply a production if we apply the production rule production only to the leftmost variable at every step only to the leftmost variable at every step.

So, we keep on applying the production. So, whenever we have a string of variable and the terminals then we look at the leftmost variable. And, then we apply a production in the on that variable every time we will do that in each step then that that derivation is called leftmost derivation.

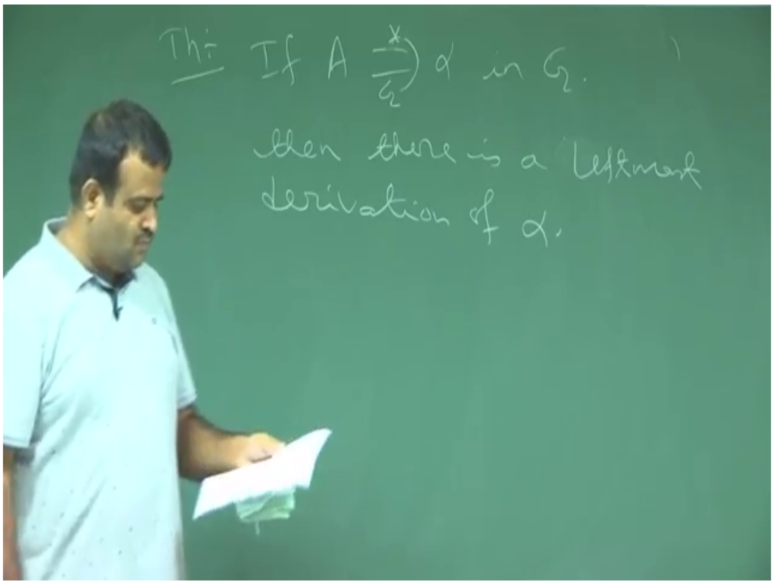
(Refer Slide Time: 21:04)



Similarly, we have the rightmost derivation we will just apply the; so what is the rightmost derivation? So, this is the rightmost derivation if we apply production only to the rightmost variable at each step rightmost variable at each step; then this is the that will be the called rightmost derivation. So, we have last we have this example we have seen alpha can be derived by the left leftmost derivation and also alpha can be derived from the rightmost derivation ok.

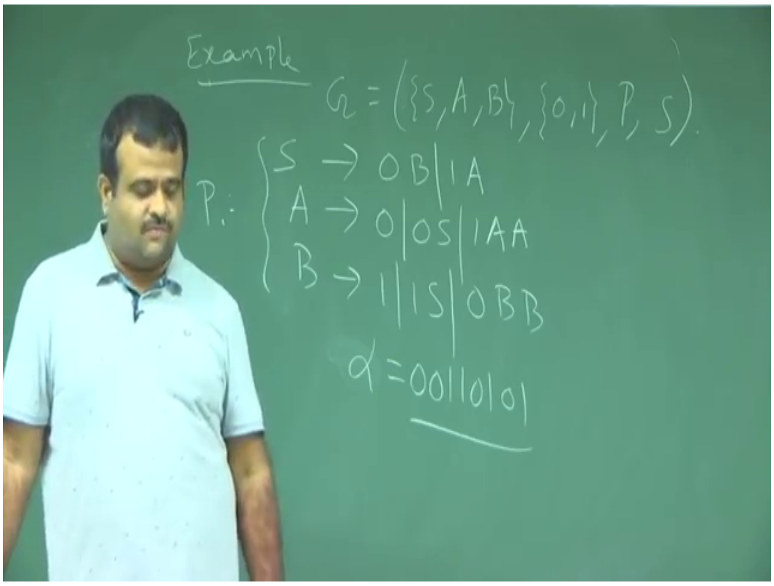
Now, if you only stick to the leftmost derivation we can show that this is for parse tree also. So, if alpha is derived from a variable a then we must have a leftmost derivation which can derive alpha; so that we can prove by theorem.

(Refer Slide Time: 22:15)



So, if so, if alpha is in G if alpha is derived from A in G then there is a leftmost as well as rightmost anyway there is a leftmost derivation of alpha ok. So, any string if we can derive I mean from a variable A or a set of variable and the terminal then this alpha can be derive only through the leftmost derivation. So, this proof will be in the lecture note.

(Refer Slide Time: 23:28)

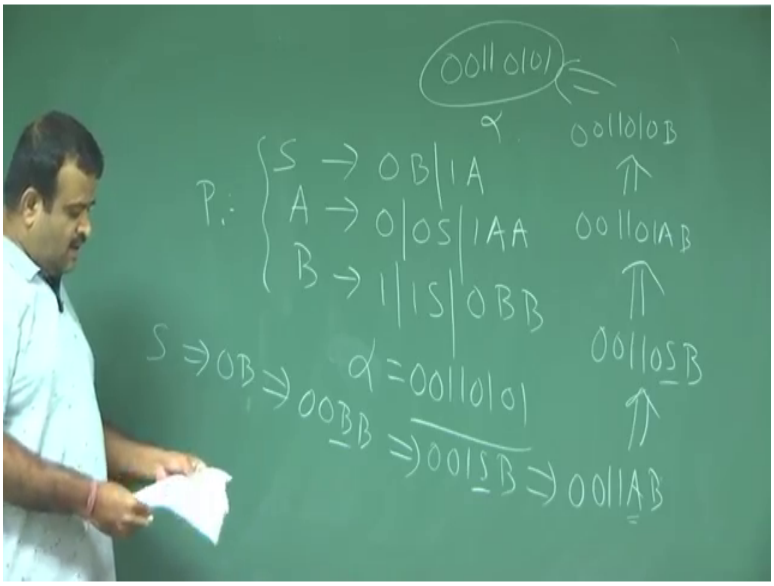


So, we are not proving this instead of that we will take some more examples and this is to for also rightmost derivation. We can have a rightmost derivation which will which will give us the alpha ok. So, let us take a grammar like this. So, we have these variables

S A B and we have terminals small a b is 0 1 P and S. So, what are the rules? Production is like this S is going to 0 B or 1 A and A is going to 0 or 0 S or 1 A A. And, B is going to 1 or 1 S or 0 B B ok. This is the grammar these are production P, this is our production P.

Now, suppose let us take it alpha string string had string we can take any string consists of variables and the terminal, but here we are taking some terminal strings. Say alpha is say alpha is say 0 0 1 1 0 1 0 1 this string and we have to derive this string from the S ok. Now, we have to use the I mean we want to use a leftmost derivation; we want to get leftmost derivation of this string from S. And as well as the rightmost derivation the string form S and then we want to find the derivative tree. So, let us try that let us get the leftmost derivation of this string S and the rightmost derivation of this string S.

(Refer Slide Time: 25:57)



So, so these are the so, let us try this. So, S is going to 0 B so, we are we want to apply the leftmost derivation first. So, there is only one string one variable so, this is the leftmost. So, B we have to take one of the rule of B. So, suppose we are taking this rule. So, 0 0 BB; now this is since this is leftmost we take this B and we apply one of this rule say we are taking this rule. So, B is going to 1 S so, 0 0 1 S B; now this S is leftmost. So, you have to apply the leftmost derivation of this say suppose S is going to 1 A. So, 0 0 1 1 A ultimately we have to reach to this 1 A then B ok. So, then B so now, this A is the leftmost one.

So, if we apply this A so, we have to take one of the rule of A. So, suppose we are taking this A is going to 0 S this one. So, this is 0 0 1 1 0 S B; now again this S is the leftmost. So, we have to take 1 over here; so, based on that we can decide which rule you should take. So, again 1 A we can take; so, this S is going to 1 A. So, 0 0 1 1 0 1 A B ok; so now, we can take this this A A is now leftmost now we can take A to be 0 because this is 0.

So, 0 0 1 1 0 1 0 and then B now, we can take B B is the leftmost now B is the only variable B to be 1; it will give us this alpha. So, 0 0 1 1 0 1 this is our alpha. So, alpha this is the leftmost derivation of alpha. So, similarly we can have the rightmost derivation of alpha. And, then we can have the tree for this corresponding to the leftmost derivation and corresponding to rightmost derivation. So, we will continue this in the next class.

Thank you.