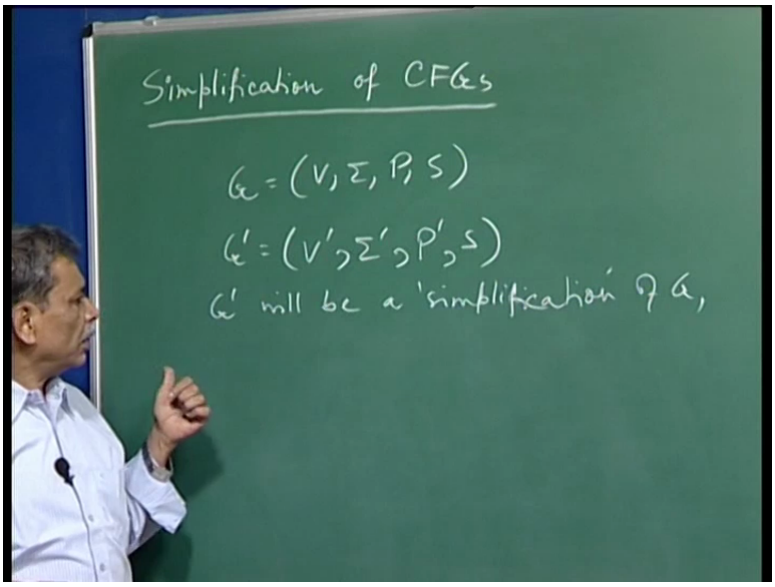
Theory of Computation
Professor Somenath Biswas
Department of Computer Science and Engineering
Indian Institute of Technology Kanpur
Lecture 23
Towards Chomsky Normal Forms-Elimination of Useless Symbols, Analysis of Reachable Symbols, Generating Nonterminals, Order of Sub-steps Matter

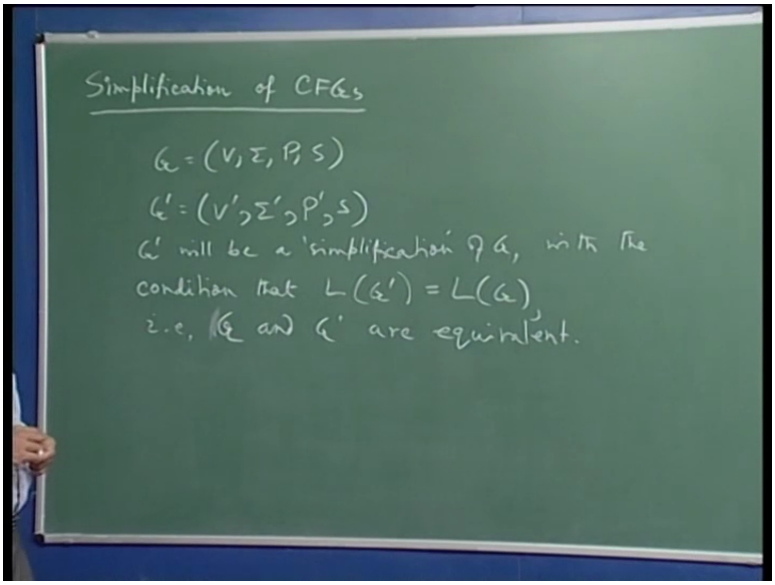
We will see in this lecture how we can simplify context free grammar? So let us say we have a context free grammar G, set of nonterminals, set of terminals, set of productions and the start symbol. Andwe would like to getanother grammar G dash which will have V dash, sigma dash, P dash and S such that this grammar G dash will be a simplified form of G.G dash will be a simplification of G. We will explain later on what we mean by the kinds of simplification that we would like to affect.

(Refer Slide Time: 01:27)



However both the grammar G dash and G they should produce the same language. So we will say with the condition that the language generated by this new simplified grammar is same as the language generated by the old grammar G. If two grammars are in this relationship that is they have the same languages generating then we call them equivalent.G and G dash are equivalent. The reason we would like to affect this simplifications will be that as we will see that this will help us cast every context free grammar in a certain standard form.

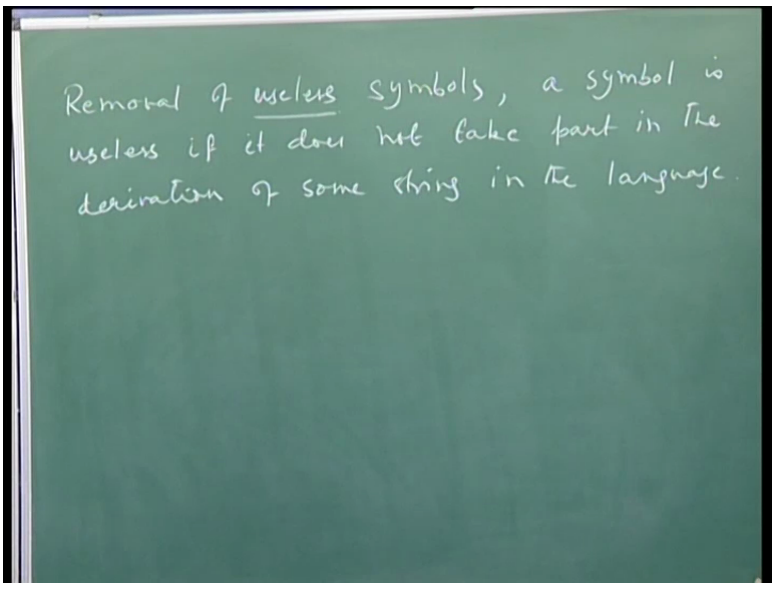
(Refer Slide Time: 02:53)



They call it normal form and using that normal form it will be easy for us to prove certain properties of context free grammars and languages. Now what are the kinds of simplifications that we would like to affect? The first kind of simplification that we would like to affect is removal of useless symbol. Remember the symbols in a grammar are the elements of the nonterminal set and the terminal set.

And we call a symbol useless if it does not take part in derivation of aterminal string which is in the language. So let us write it down. A symbol is useless if it does not take part in the derivation of some string in the language. So let us get this idea clear.

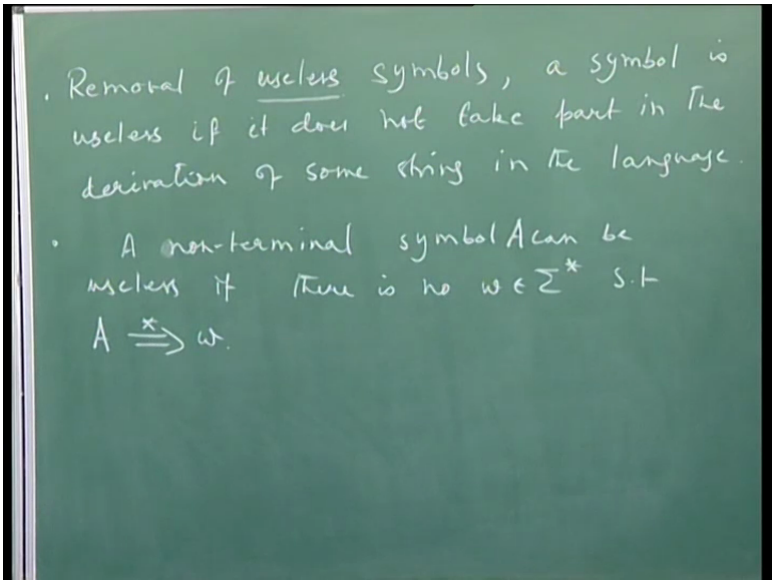
(Refer Slide Time: 04:45)



So we have a grammar G that is generating the language L G so that language is of coursesome set of strings over sigma. And now if it so happens that some nonterminal or even some terminal is such thatthese symbols are not taking part in any derivation whatsoever ofa string of the L G, the language generated by G, then such a symbol we will call useless. And if we think about it, a symbol may be useless in two ways. Let us saynonterminal symbol can be useless.

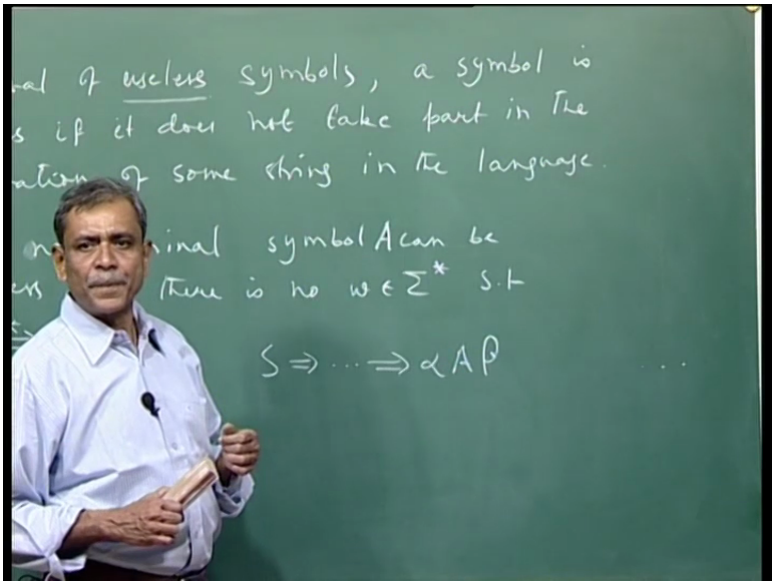
So nonterminal symbol let us say A. This symbol is A can be useless if it does not generate any terminals strings at all. What we mean by this or we can simply say one of the ways a nonterminal symbol A can be useless if there is no w in sigma star such that A derives this terminal string.

(Refer Slide Time: 06:40)



Sowhy such a nonterminal is useless? Because you see that suppose you start your generation as usual from the start symbol and somewhere down the line you get this string A, alpha A and beta. Alpha and beta are strings over V union sigma. Now eventually to get a string of the language this A needs to be rewritten into some string over the terminal strings.

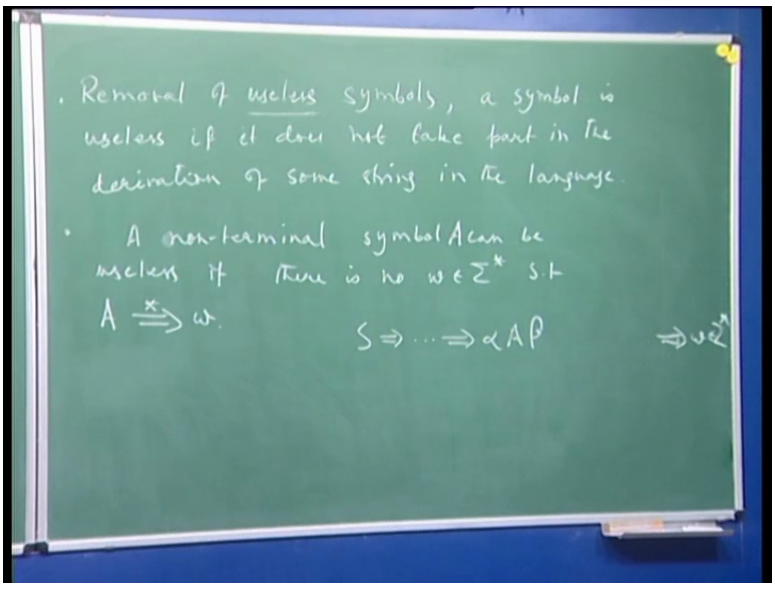
(Refer Slide Time: 07:22)



Now that string could even be empty. The point is this A needs to be rewritten ultimately eventually so that somewhere here you get some w in sigma star, right? So which should mean that there has to be a way of generating some terminal string from this nonterminal A, right? So this is one of the ways anonterminal symbol can be useless.

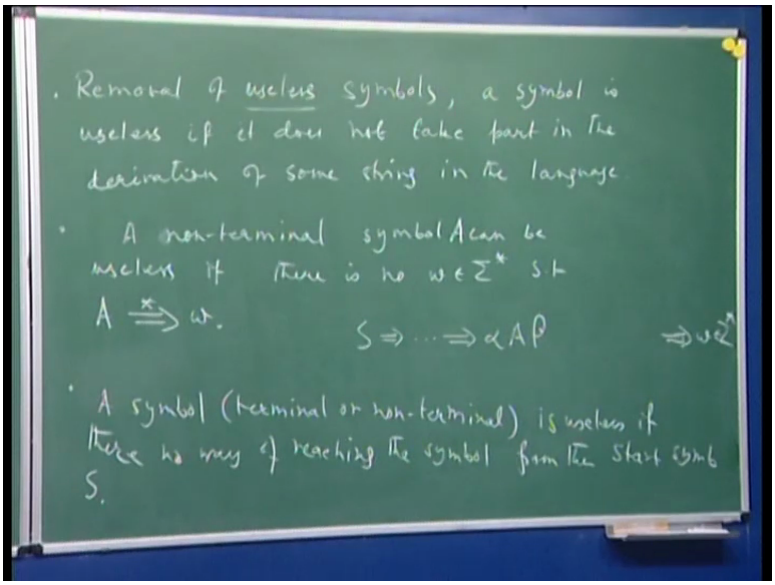
Can a terminal string be useless in this sense? Can some symbol A in sigma be useless in this sense? Of course not because A itself is a string over sigma. So that is why we said anonterminal symbol A can be useless if there is no string w in sigma star such that this is the situation.

(Refer Slide Time: 08:23)



So another possible way anonterminal or a terminal can be useless that if you can never reach such a symbol from the start symbol S. What we mean by this that a symbol, we will say terminal or nonterminal, is useless if there is no way of reaching the symbol from the start symbol S. We elaborate this a little more.

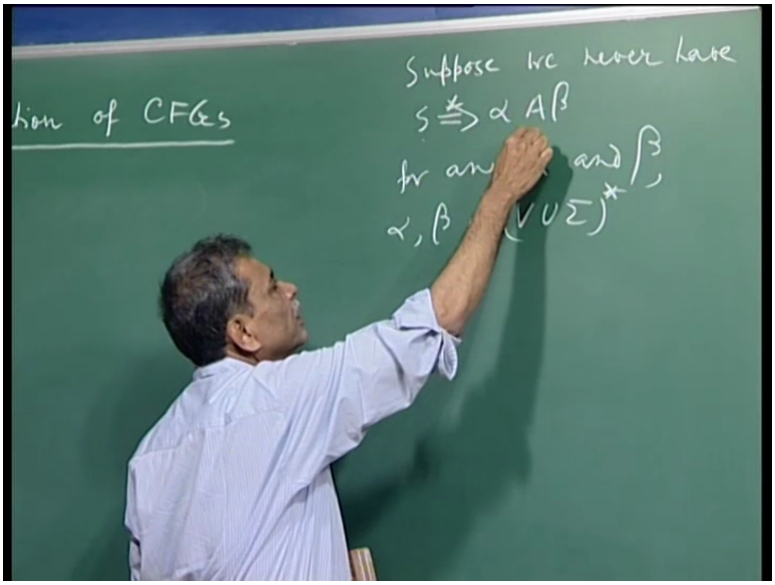
(Refer Slide Time: 09:49)



What we mean by the second point is that suppose there is no way you can generate some string over nonterminals and terminals such that starting from S. So let us write it this way that suppose we never have this situation for any alpha and beta, right? Alpha and beta are over V union sigma star. Thensee in that sense we say that we can never reach Astarting from S, alright? So if this situation happens then again such as symbol is useless.

Nowthe way we have written here A is a nonterminal but you see the same situation is proved. The same way even a terminal can beuseless thatif you would never have apartial derivation of this kind that starting from S you get to some string wherethat terminal is a part.

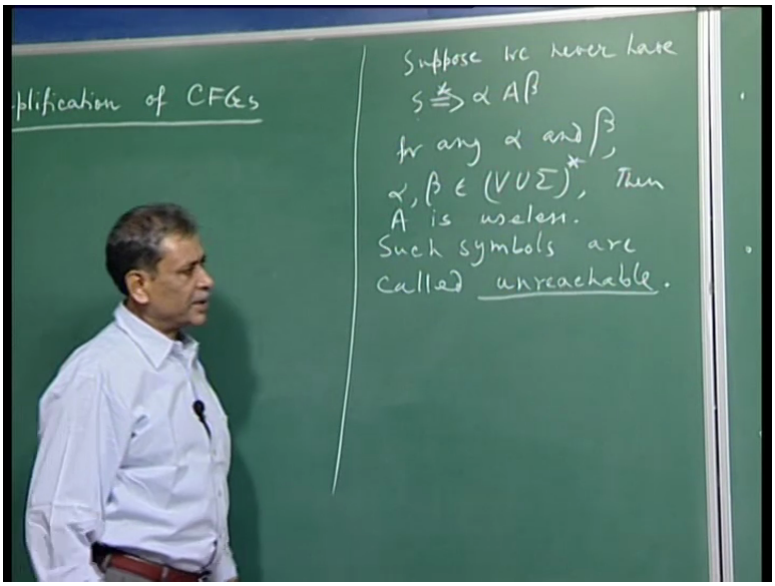
(Refer Slide Time: 11:41)



So this way the second point of being useless can happen with both nonterminal as well as terminal. Now there are you know people give names for this. Such anonterminalis called non-generating. And such symbolswhich are not reachable, they are simply called unreachable symbols.

Such symbols, let me complete this sentence.Suppose we never have S derives or can be rewritten as alpha A beta for any alpha and beta, right, then A is useless, okay. Suchuseless symbols are called unreachable symbols.

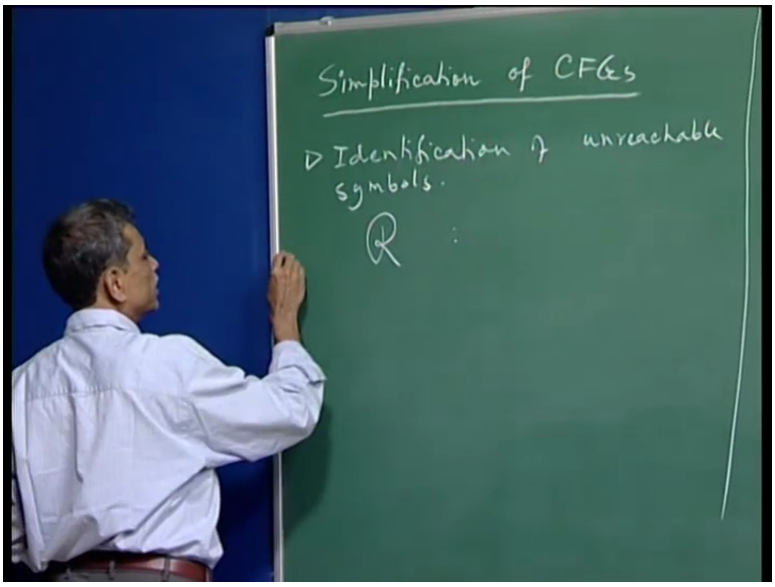
(Refer Slide Time: 13:30)



Notice as I said that this way of a symbol becoming useless which we call non-generating that makes sense only for nonterminals. Whereas unreachability, that of course can happen with both terminals and nonterminals. Let us now see how we can identify unreachable symbols and then non-generating symbols and then we will simply remove them from the grammar. We will see little more on that.

So or let us say here we should do identify of unreachable symbols. What we do is we inductively build a set. Let me call it script R. This will be the set of reachable symbols. The symbol which we can reach from the start symbol S.

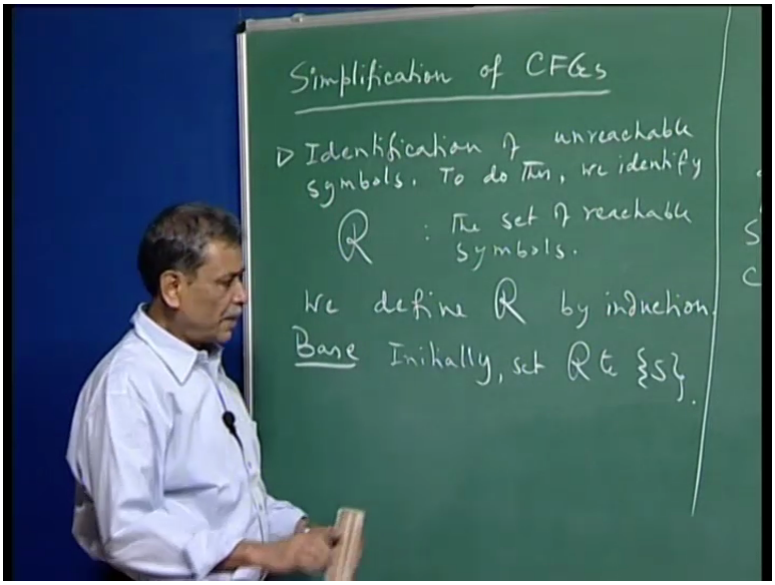
(Refer Slide Time: 14:51)



So let us sayto do this we identify R which is the set of reachable symbols. And then what will be the unreachable symbols? Basically take this set out from V union sigma,all the symbols which are left after taking out the reachable symbols from V union sigma,these symbols will be obviously unreachable. So as I said this set of reachable symbols we create or we define Rby induction.

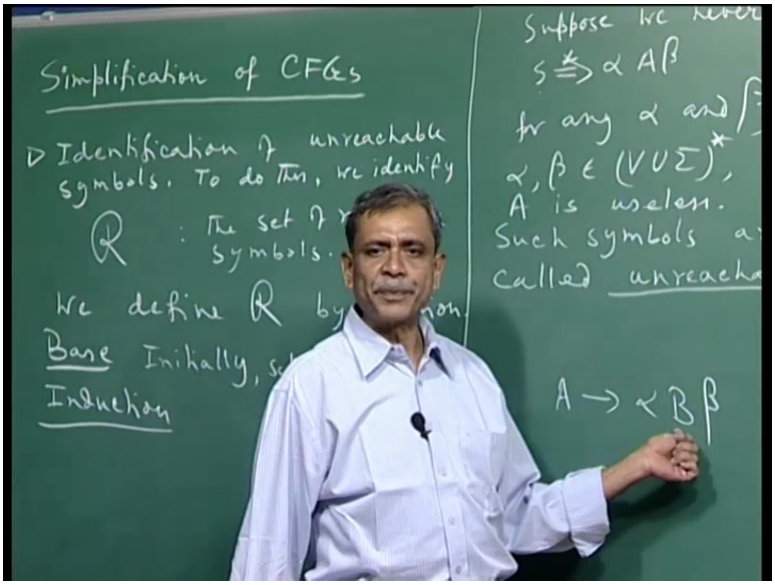
So we start you know wherever we define a set or any object through induction we first have a base case and then we say that suppose we had already builtyou know up to some point this setR thenhow to extend, right? So what is the base? That is pretty easy, is not it? Of V union sigma which are the symbolsor which is one symbol that you can think which is definitely reachable from S itself or S, right? That symbol is the symbol S. Because in zero step one can reach S from S. So we can say that initially set R to just this, right?

(Refer Slide Time: 17:06)



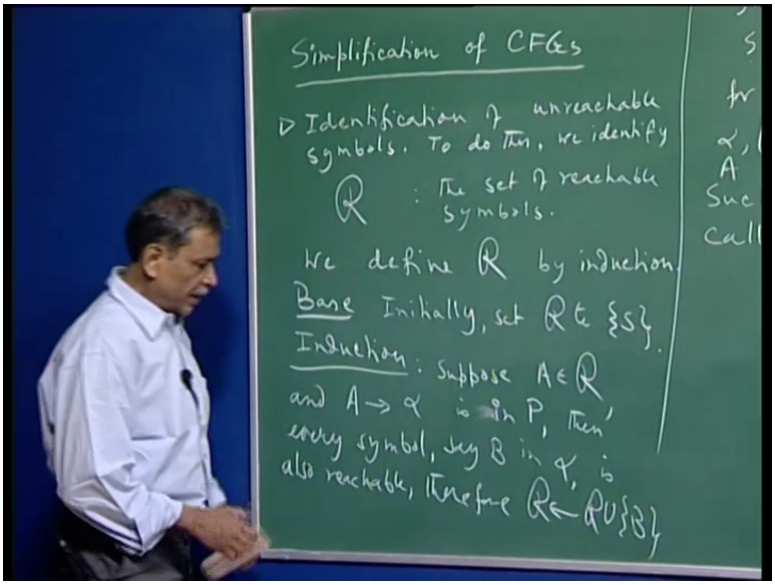
Absolutely no doubt that thestart symbol S isreachable from S, right? Ishould not say even definition, right? It is trivially so, correct. And now the induction process is this. Imagine I have a production let us say A goes to alpha B beta and I have already found A to be reachable. Then surely all these symbols here in particular B also will be reachable, right?

(Refer Slide Time: 17:57)



So induction step is that suppose A is an element of R, the set builtso far, okay. Then and A goes to let me say alpha, right,is in P then every symbol in alpha is also reachable. Then every symbol sayB inthis right hand side of this production in alpha is also reachable. Therefore what we can do? I can update my R with such a symbol if it is already not a member of R. So we can say that R is set to R union this symbol B.

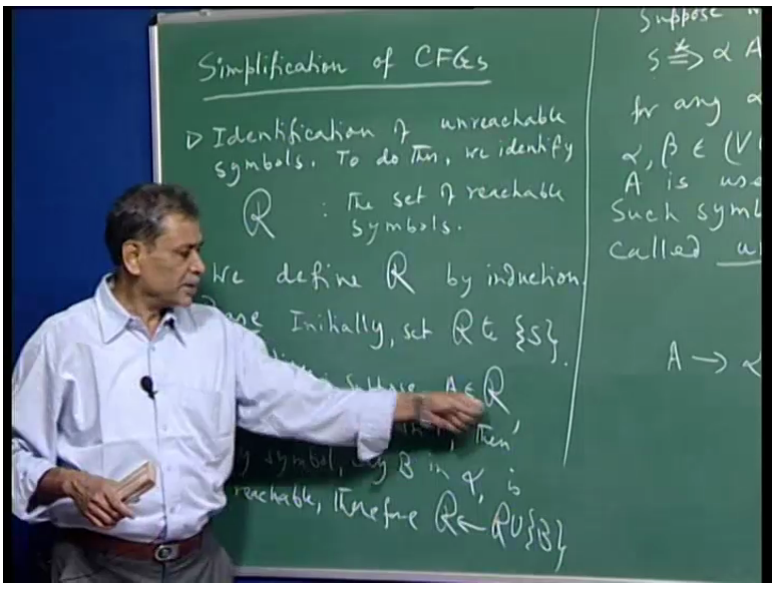
(Refer Slide Time: 19:39)



Of course again I am saying that this B the way I have written it is anonterminal. But the same is true even if it is a terminal. And in doing this what you really should do that if you look at the right hand side of the production and for every symbol check if it is already there in your set that you have built so far. If it is not there,add to it. And this way you go over all the productions. Then you have a set R, right? But then again everytimeyou change the set R.

See the set R changing in the sense is thescript R is actually growing. Every time it grows you need to again look at all the productions to see which all newmembers come into R because of an augmentation in the script R, this set, right?

(Refer Slide Time: 20:44)



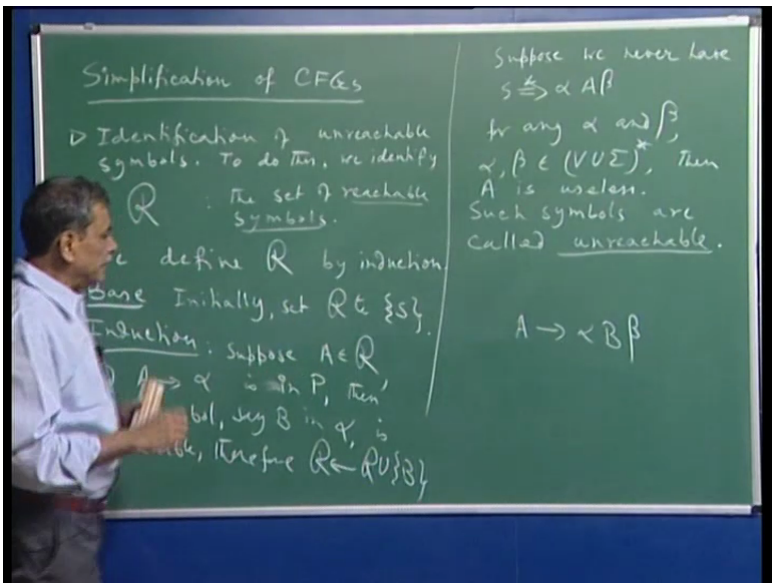
IfI find one more element which is reachable so I need to look at all the productionswhose right hand side is that particular symbol. Sorry the left hand side is that particular symbol. So that I can consider all the right hand sides if it is a nonterminal. If it is a terminal of course it goes. That itself does not add because the terminal symbol will never occur in the left hand side of a production.

So do you see what is happening? I start with the base case and thenas I keep growing the set R by putting newer and newer members into R, I check if this R keeps growing or not. Now anytime I find that I have added you know somesymbol and then even after consideration of all the productions in the grammar, the set R is not augmented, that means what? That means we have reached the finalvalue of R. The R cannot grow any larger because only way R can grow of course when some new member comes in.

Then its productions whose left hand side is that new memberwill give rise topossibly some more new reachable symbols. So you see this way it is not too difficult to see that this set R keeps growing monotonically because as you were adding more and more symbol and then finally it has to stop.

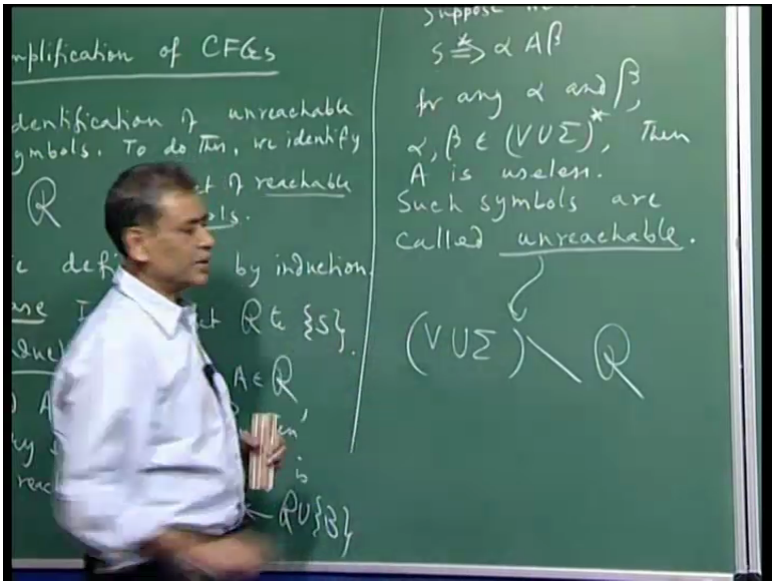
This cannot grow on indefinitely because after all at most all the symbols of V union sigma, all of them are reachable. So you know at some point perhaps before that when R contains everything of V and sigmathis process of growing R stops. So that is the set of reachable symbols.

(Refer Slide Time: 23:13)



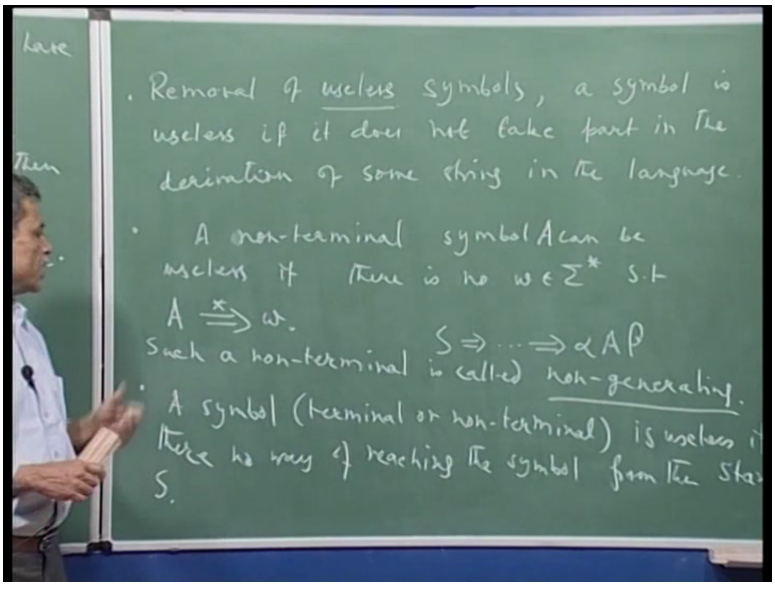
Then the unreachable symbols are simply here it will be V union sigma. Yousubtract from this set, this set of reachable symbols. So this is the set of unreachable symbols, okay.

(Refer Slide Time: 23:33)



So we know how to identify the set of unreachable symbols. And now we will also need to find out how we can identify non-generatingnonterminals because that A, this is the definition of non-generatingnonterminal that is anonterminal whichdoes not derive the terminal string.

(Refer Slide Time: 24:04)

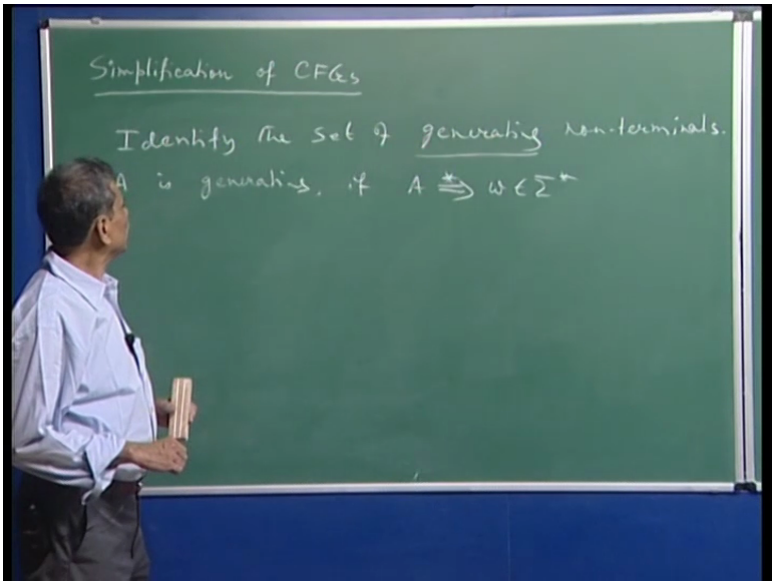


We will see now how to identify non-generatingnonterminals? What we are going to do to identify non-generatingnonterminals is likewisepreviously what we did. We will first identify the set of generating nonterminals. And clearly the definition of generating nonterminal is

that the nonterminal is generating if you can derive a terminal string for itstarting from that nonterminal.

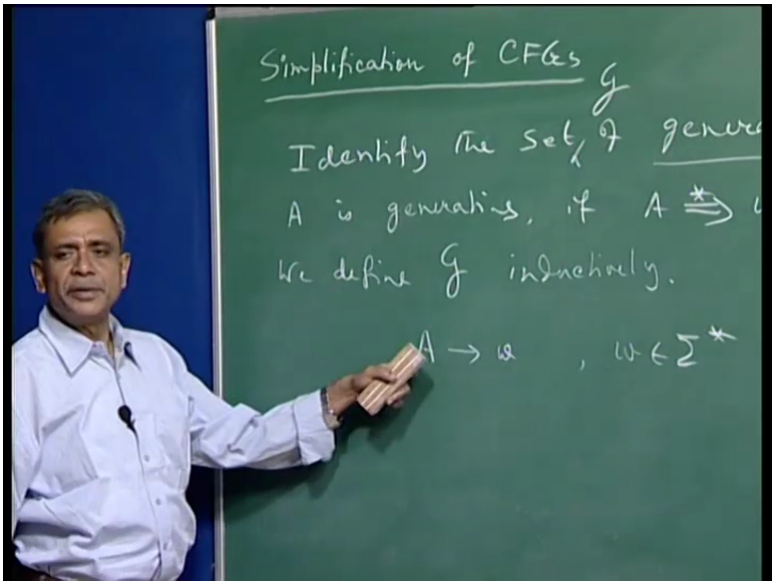
So you say A is generating if A derives some w in sigma star. Starting from A you can reach a string of only terminal. That is a generating nonterminal.

(Refer Slide Time: 25:31)



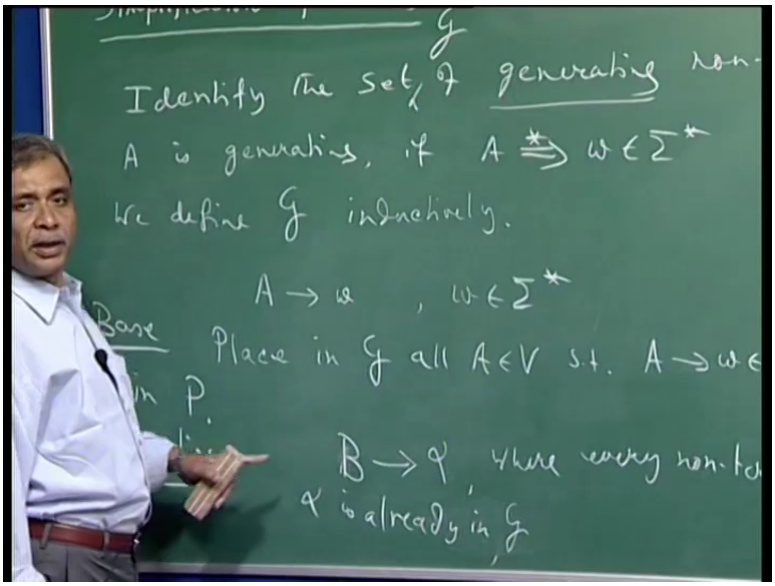
So this set let me call it script G. The setG of this generating nonterminal I am calling it script G and again we define G inductively, right? What is the base case? Here clearly suppose I have a production which is of the form A goes to w where w is in sigma star, right?Then clearlythis nonterminal is generating nonterminal A.

(Refer Slide Time: 26:23)



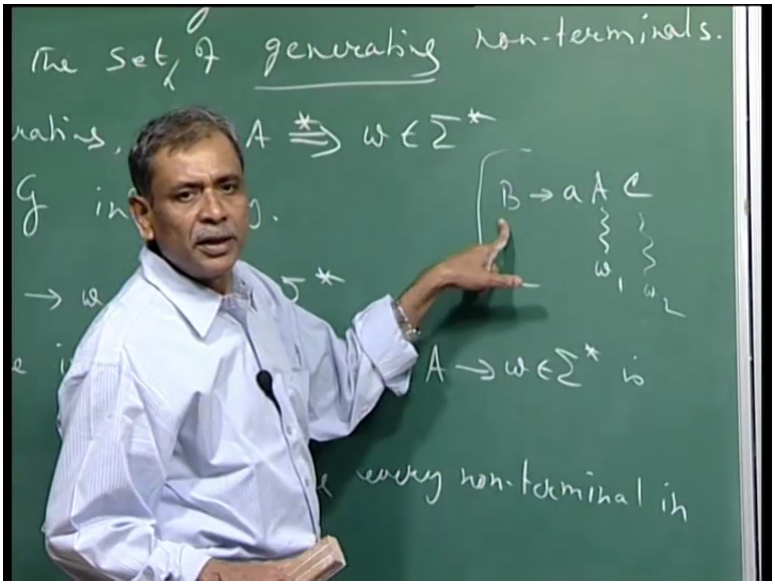
So the base casein the inductive definition of script G is thatplace in G all A such that A goes to w in sigma star is an element of the set of production, okay. So you will start with some subset of V. And how do you grow G? Now again we willable to grow G by looking at productions. So for example suppose I have a production of the kind that B goes to alpha, right, where every nonterminal in alpha is already in G.

(Refer Slide Time: 28:15)



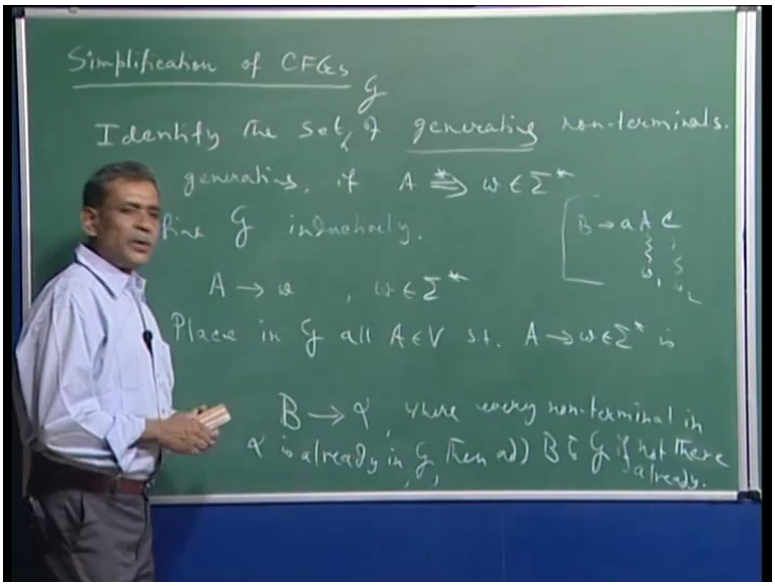
And then if B is not in G then I should add B. Why? Because you see it is like this. Suppose thissituation is B goes to let us say a A C and I can derive some terminal string w 1 from here, I can derive some terminal string w 2 from here. This is the terminal. So what happen that means I can derive from B, a, w 1, w 2. So therefore B will be also generating.

(Refer Slide Time: 28:45)



Where every nonterminalin alpha is already (gen)in G. That means we have already found every nonterminal in alpha to be generating. Then add B to G if it is not already there. If not there already, right? This isthere. This makes sense.

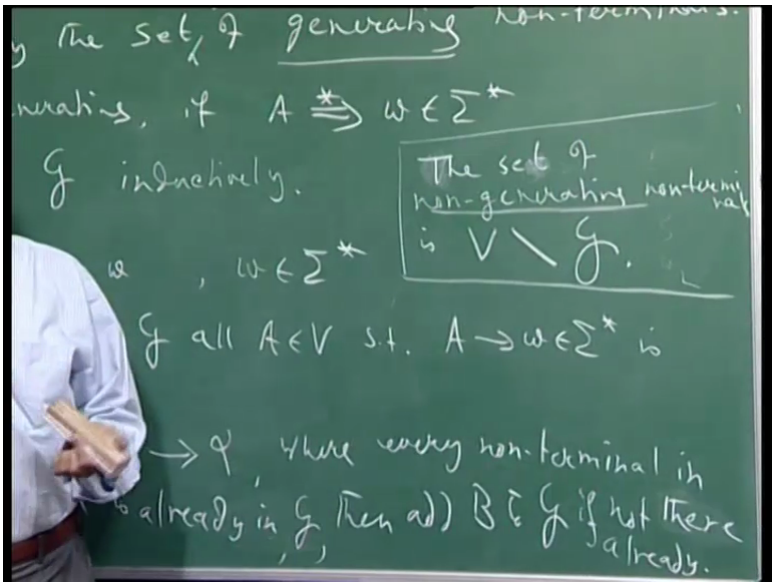
(Refer Slide Time: 29:40)



So again you know anytime I have this setG thenI look at productions and try to find a way of augmenting G using essentially thisstrategy. So again the G will keep growing. At some point you will find G is going no further even when you look at all the production. And that is the time that is the final set of generating nonterminal.

Then the set of non-generatingnonterminals will be simplythis. The set of non-generatingnonterminals isfrom this set V I subtract the setG. That is the site of non-generatingnonterminals.

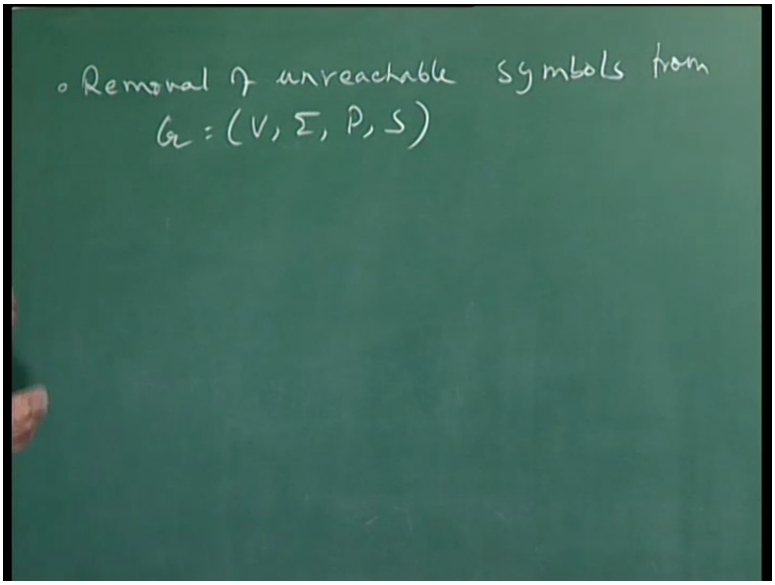
(Refer Slide Time: 30:58)



So I have found simple. Actuallyboth these algorithms are fairly simple. The way is to identify non-generatingnonterminals as well as non-reachable symbols which can be of course either a nonterminal or a terminal. After we have identified the set of unreachable and non-generating symbols we should simply remove them from the grammar. Now by that what we mean? Removing such a symbol from the grammar means that not only they go out of the corresponding nonterminal set or terminal set, right?

Also we must get rid of any production were such a symboltakes part, right? So there will be a problem which you shouldaddress now that separately we can do both of these without any problem.So let us say what I can do is that removal of unreachable symbols from G which is let us sayV, sigma, P, S. And recall that the set of unreachable symbols, this set may contain some nonterminal and some terminals.

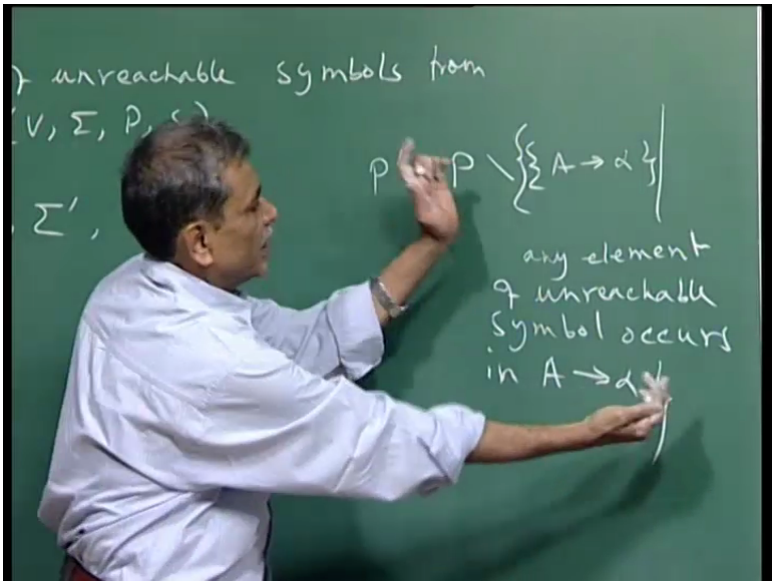
(Refer Slide Time: 32:52)



So we will get a new grammar G dash by removing from V all the unreachable nonterminals, from sigma I remove all the unreachable terminals, I get these. So therefore Iam writing V dash and sigma dash. From P, I remove all productions where any of the unreachable symbol occurred, right? Sobasically what we are saying is, so let me write it down clearly that P dash is P minus the set of all productions of the kind A goes to alpha, right, such that any element of unreachable symbol occurs in A goes to Alpha.

It can maybe A itself is unreachable. So in that case of course we must remove that production from P. Also it could be that you knowone of the right hand side symbol of this production, one of the elements of alpha is unreachable. Then again there is no pointkeeping thisproduction. So this is theset of all productions where this set starting from here the set that I represented this, out of curly brackets.

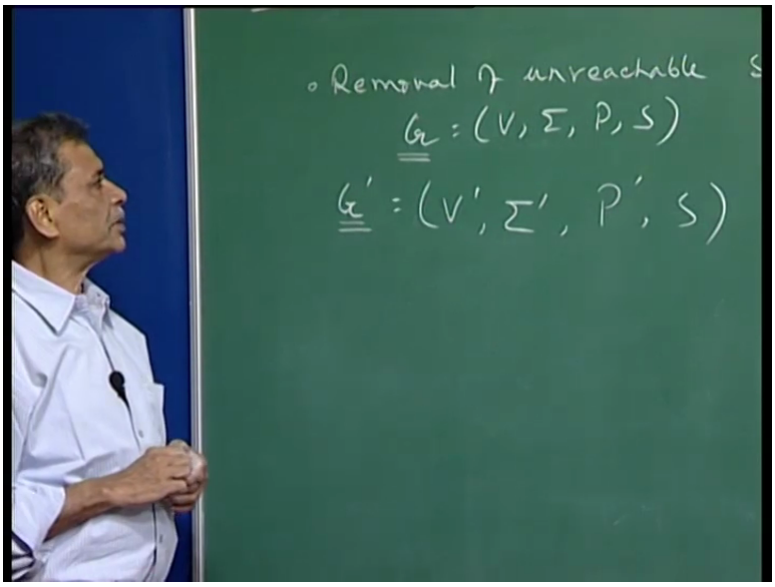
(Refer Slide Time: 34:57)



This is the set of all the production wheresome unreachable symbol occurs. All thoseproductions I removed from P to get P dash, right? And S can never go out because S is of course is itself always reachable from itself, right? So this is the grammar G dashwhich will not contain any unreachable symbol.

And because of removal of these unreachable symbols we have managed to get rid of some productionswhich would never be used in deriving a terminal string, right? So this grammar G dash is therefore a simplification of the grammar G after removing unreachable symbols.

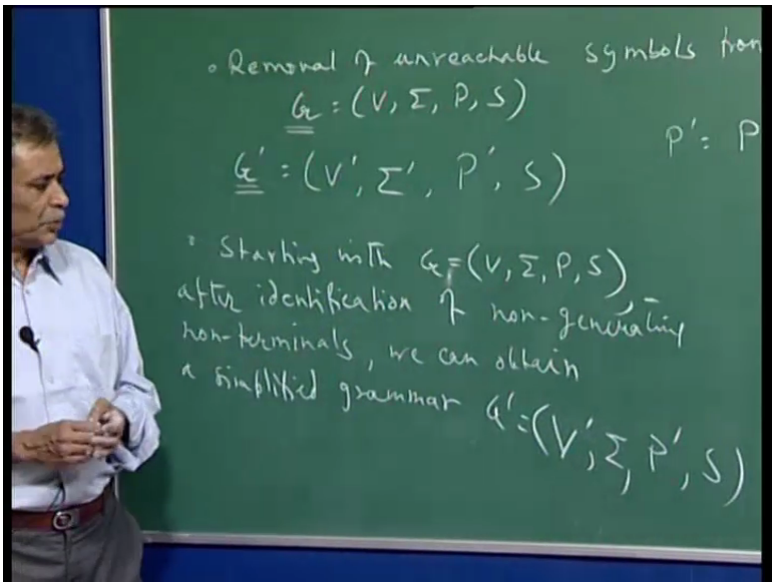
(Refer Slide Time: 36:00)



In the same manner we can obtain separately starting with Gwhich is again let us say some V, sigma,P, S. Afteridentification of non-generatingnonterminalswecan obtain a simplified grammar G dash where of course some nonterminals which we have non-generating, they have been removed from V.

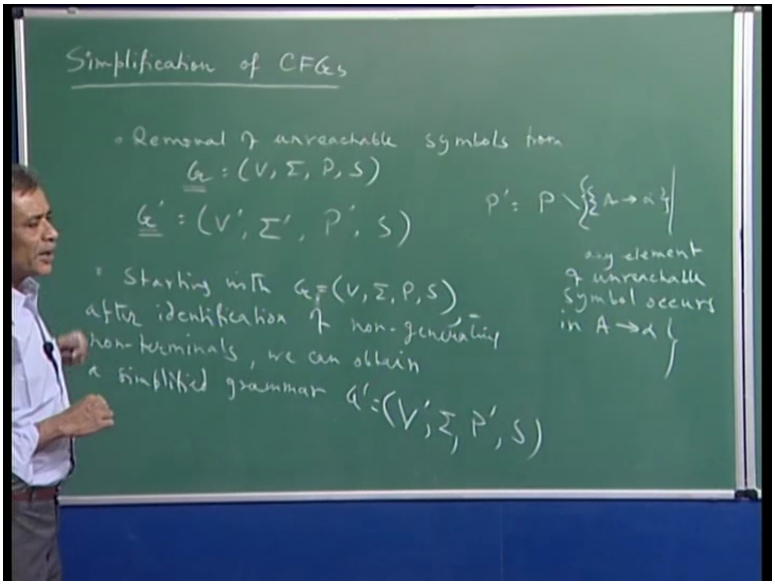
Sigma does not change because we are just talking of non-generatingnonterminals. P possibly obviously would change if we have removed some symbols from V. That is if we have identified some nonterminal to benon-generating.Andwe will assume that S is always generating.

(Refer Slide Time: 37:45)



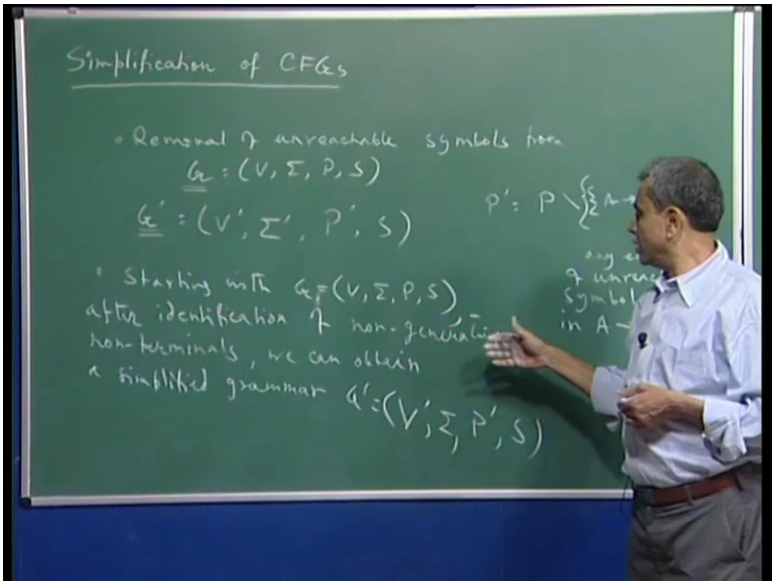
S, in other words the grammar G originally was such agrammar that L G was non-empty, right? The language is non-empty. So therefore S must be deriving some terminal strings and therefore S would remain in the simplified grammar also. So these two simplifications we can do separately and it is easy to see thatwhatever we have said is correct in the sense that this G dash is indeed generate the same language as G.

(Refer Slide Time: 38:26)



And similarly here also that this G dash after removal of non-generatingnonterminals will also generate the same language as G. At the same time boththese grammars are simplified. Now our goal was to remove all useless symbols. And we said in the beginning that a symbol can be useless because either it is non-generating or it is unreachable. Soas I said here we know how to do these things separately. Removal of unreachable symbols and the removal of non-generating symbols.

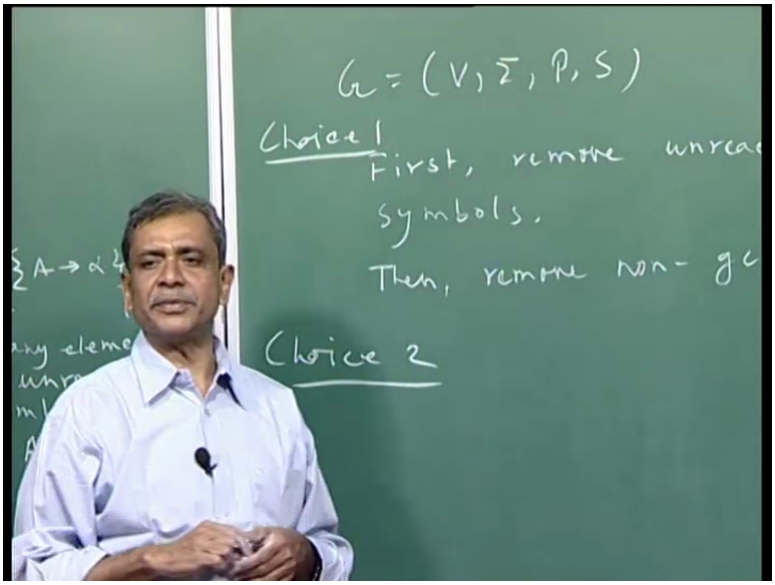
(Refer Slide Time: 39:13)



Nowin which order we should do this? Point I ammaking is that you have given me a grammar G, V, sigma, P, S. The same state I have a choice that first remove unreachable symbol. Then remove non-generating symbols. So this is my choice 1. And choice 2 is the

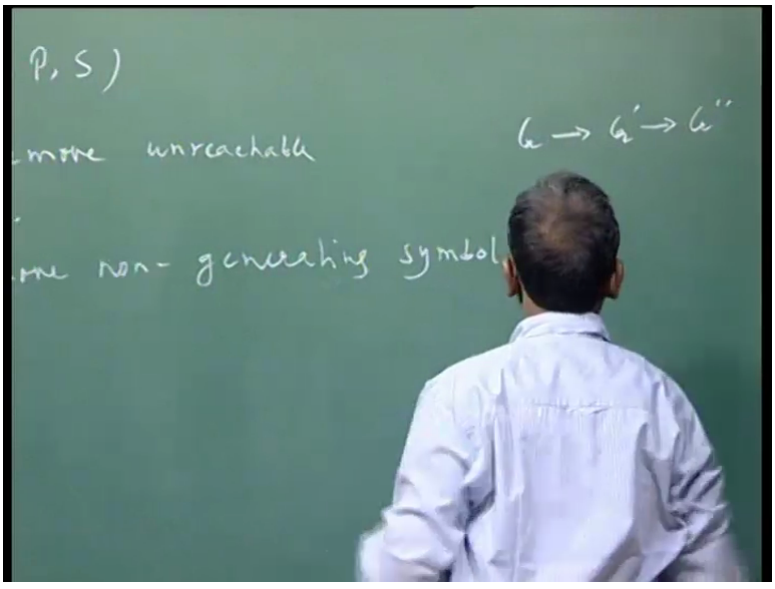
other way. We first remove the non-generating symbols and then remove the unreachable symbols.

(Refer Slide Time: 40:28)



So is it clear what we are saying? That from G, after removing unreachable symbols I get a grammar G dash and then I removed in this case, in choice 1, all the non-generating symbols and maybe I get the new grammar G double dash.

(Refer Slide Time: 40:46)

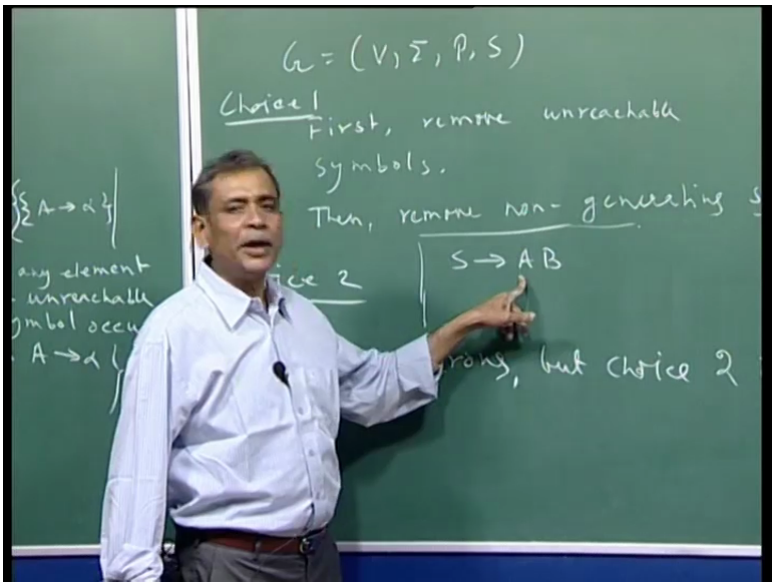


And in choice 2 I first remove all the non-generating symbols and then remove all the unreachable symbols, right?What I would like to point out is that choice 1 is wrong but choice 2 is correct. See why would choice 1 be wrong? So let me show it here very simple.

You seeso let us say that I have this S goes to A B and so therefore bothA and B are reachable, right?

And then later on you found that B is non-generating, okay. So then you would remove this production. Now it might be in the process because of the removal of this production the link from S to A also goes.

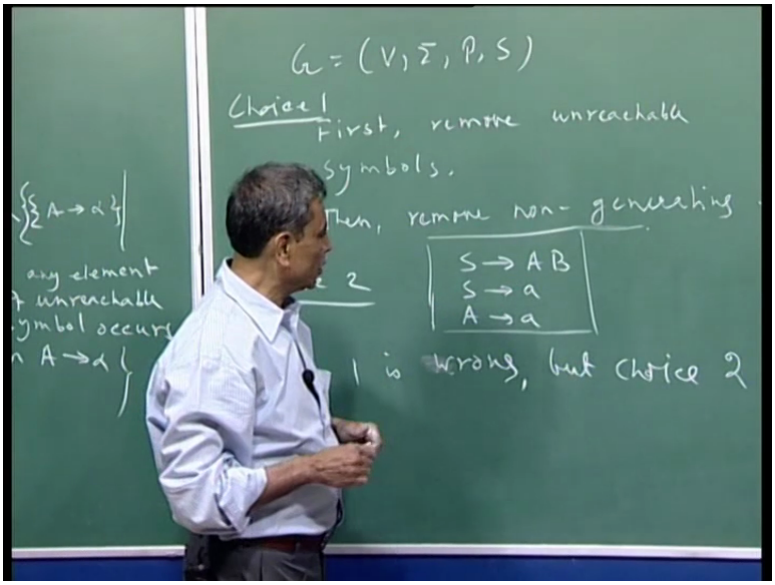
(Refer Slide Time: 42:18)



Although A itself is generating. In fact there is a simple example given in some textbooks. So let us sayS goes to A aswell as I have A goes to small a, right? So this is the grammar and first when youdo choice 1, what you are going to find? That all these symbols A, B, S of course as well as small a, they are all reachable from S, is not it? From S or set thatscript R.

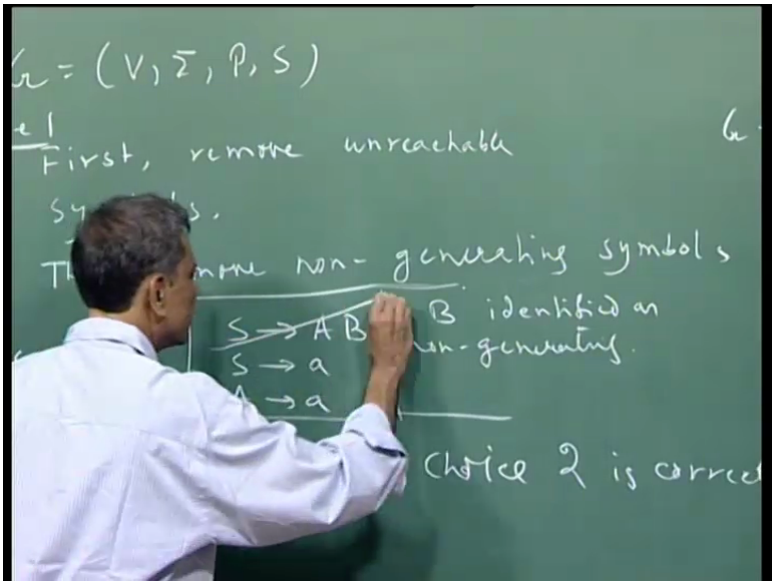
First it will have S then immediately when I look at this I will see S, A and B they will also go. And then I see any one of these productions you see this symbol small a is also reachable, right?

(Refer Slide Time: 43:14)



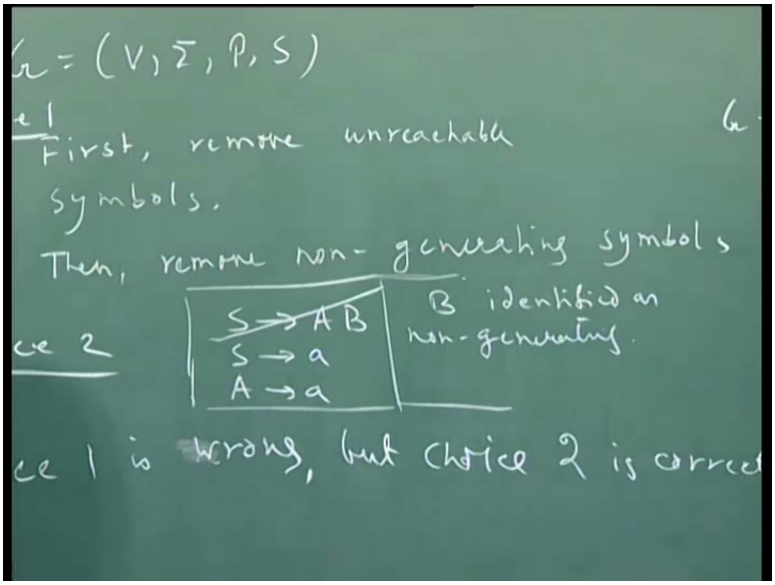
So all the symbols here are reachable. So the grammar really will not simplify if I consider the unreachable symbols removal. But now I see what? Now I look at non-generating symbols. I clearly identified that B to be non-generating, right? Then B identified as non-generating, therefore this production will go out.

(Refer Slide Time: 43:54)



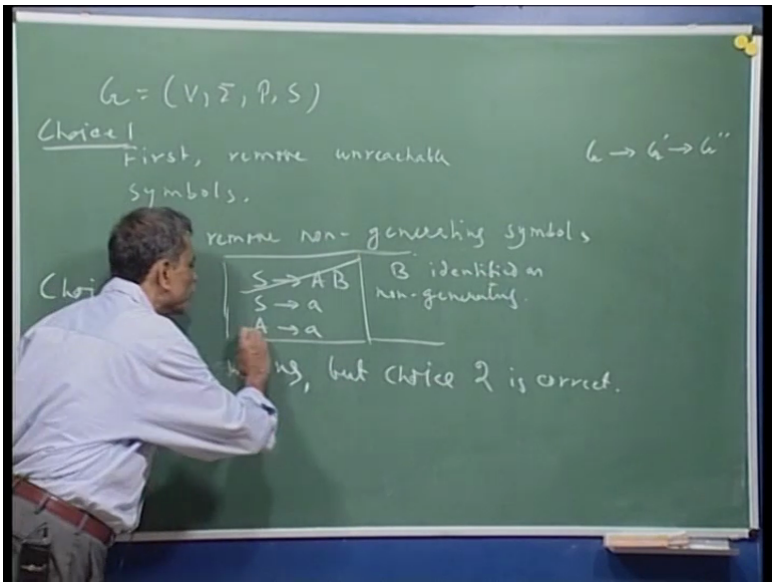
And now that is all that you can do, right? You will remove all productions and of course you will remove the nonterminalB also. And then you are left with this.

(Refer Slide Time: 44:14)



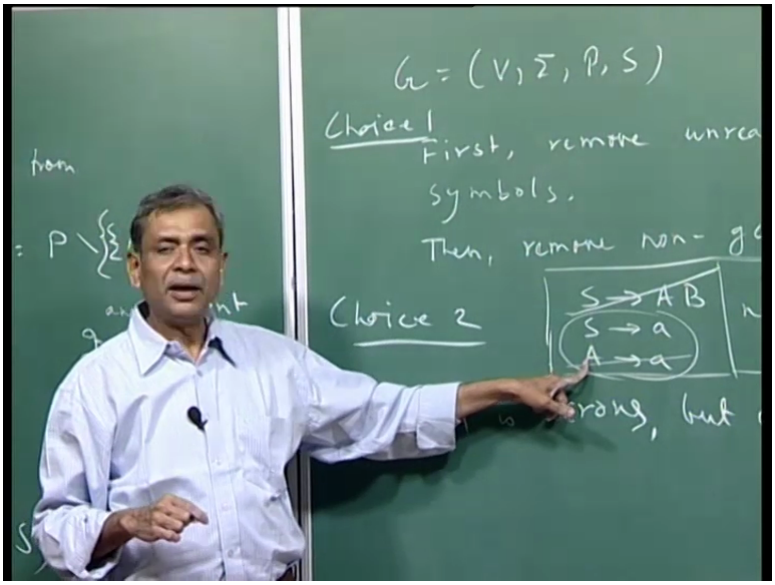
Now do you see what has happened? So this is the simplified grammar that you are getting. But is this grammar okay for us? Because nowbecause of the removal of B we have introduced a new non-reachable element which is A. So actually nowthis should also go.

(Refer Slide Time: 44:36)



In fact the simplified grammar after removing all the useless symbolswill have only one production which is S goes to a. So what has happened? So you see the point I am making is that if I first remove unreachable symbols and then remove non-generating symbols I may end up as in this case with these two productions. But that grammar which has these two productions is not totally simplified because I will retain a symbol A which is unreachable.

(Refer Slide Time: 45:19)

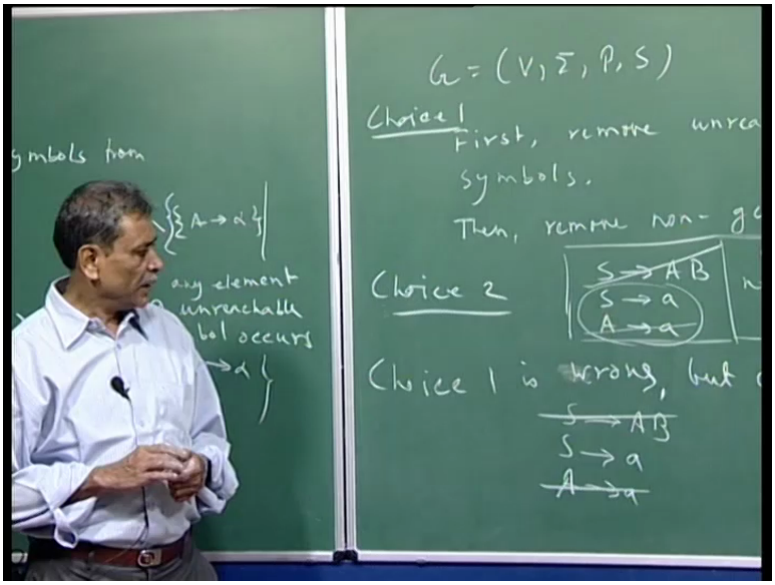


On the other hand for the same if I do choice 2, what is going to happen? S goes to A B. Again start with this. S goes to a, A goes to a I find. So here in choice 2 what I am going to do first identify all the unreachable symbols, remove them from the grammar. So here you will remove this particular production because B is non-generating.

In choice 2 first you figure out all the non-generating symbols, remove them from the grammar and that would mean that I will remove this production and of course the symbol itselfwill go away from the set of non-terminals. And then both of these are generating S and A.

And now I will try figuring out if there is any non-reachable symbol. And yes indeed I find that A is unreachable because starting from S, I just get this and that is it. S and small a will be the only two symbols which are reachable. So capital A is not reachable. So again this goes. So I have got the right kind of simplification.

(Refer Slide Time: 46:49)

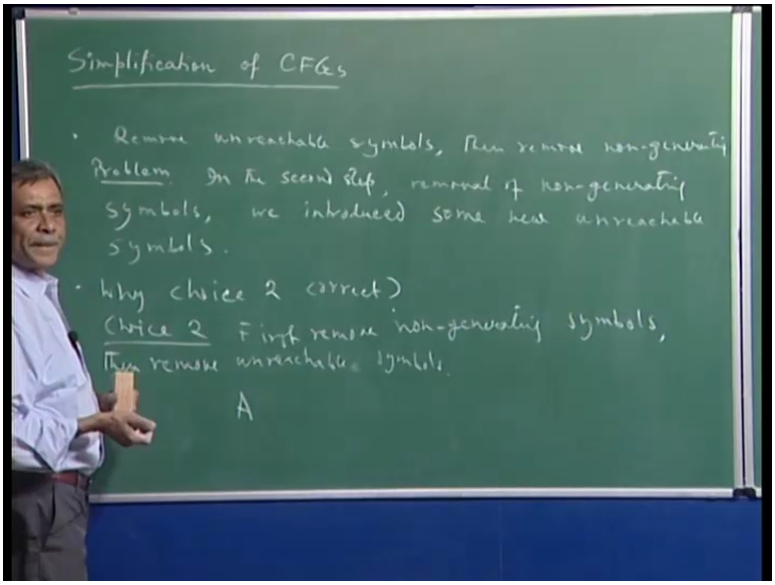


Can we prove this orif not formally can I atleast justify that choice 2 is right and choice 1 is wrong? Why choice 1 is wrong? Because of the simple thing that it is possible after the removal of unreachable symbols, right? So let usmake this point clear. Why choice 1 can go wrong? Choice 1 was first remove unreachable and then remove non-generating. So what can happen? And in fact there is an example that we have seen that after you have removed whatever symbols that you found originally to be unreachable.

When you started removing non-generating symbols, some symbols which were reachable previously became unreachable, right? So that was the problem which was one. So let me writethe problem. In the second step that isin the process of removal of non-generating symbols we introduced some new unreachable symbols. So this is the problem. Andthere was an example we have already seen. But interestingly why choice 2 is correct?

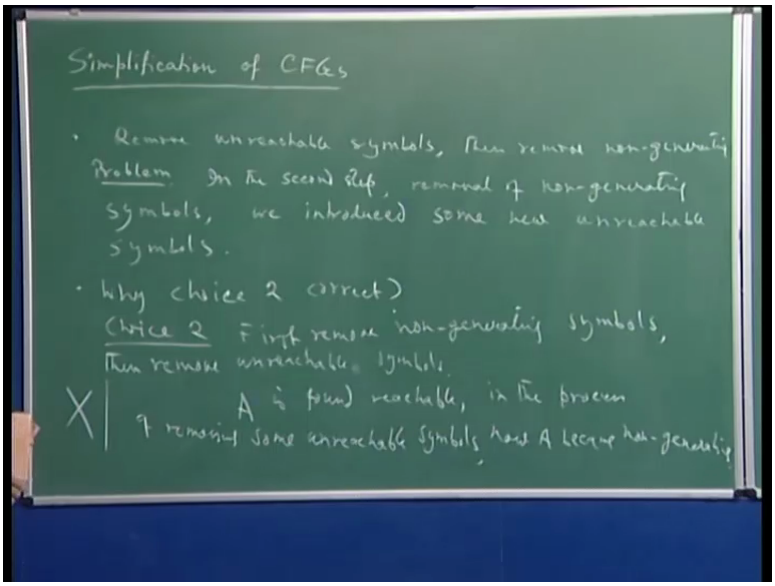
Let us understand that at least informally. What is choice 2?Choice 2 is first remove non-generating symbols andthen remove unreachable symbols. Sowhat happened in case of choice 1, happened with choice 2. That would happen if you have foundthat some A to be reachable, right? But in the process of removing some other symbols which are not reachable you made A to be non-generating, right?

(Refer Slide Time: 50:49)



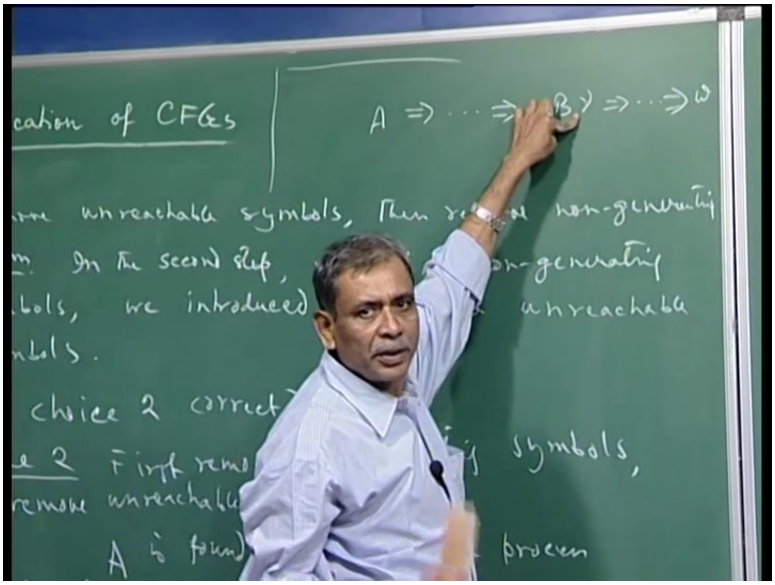
Only in that casesuch a choice 2also be unsafe. Is it clear? The situation is choice 2 will be badif A is found reachable. A was generating before that is why youcame to secondphase when you found A is reachable. Butin the process of removing some unreachable symbols now A becamenon-generating, right? This is the way this choice 2 can go also wrong, right? Now I claim this can never happen.

(Refer Slide Time: 52:20)



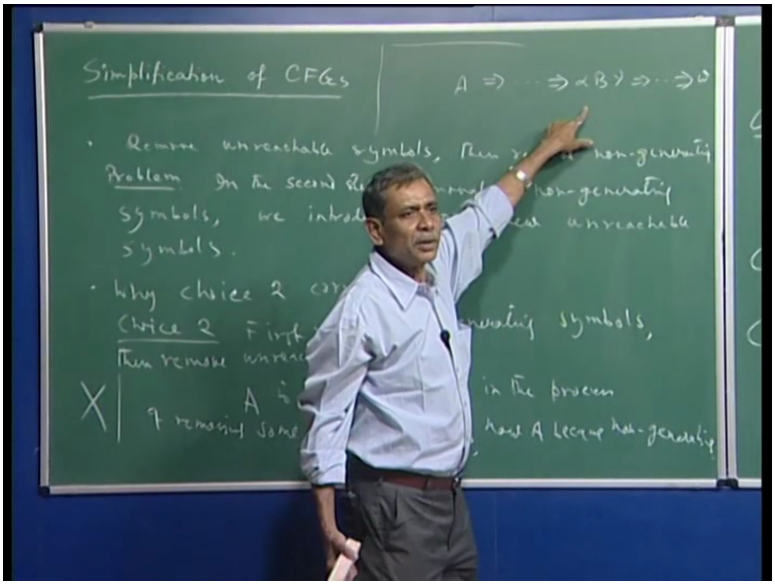
Why? Because thatidea is very simple. Seeyou could reach A fine, and A was generating. So imagine I have a sequence of steps through which I derive w. And now there is a possibility of A becoming non-generating if I remove B.

(Refer Slide Time: 53:03)



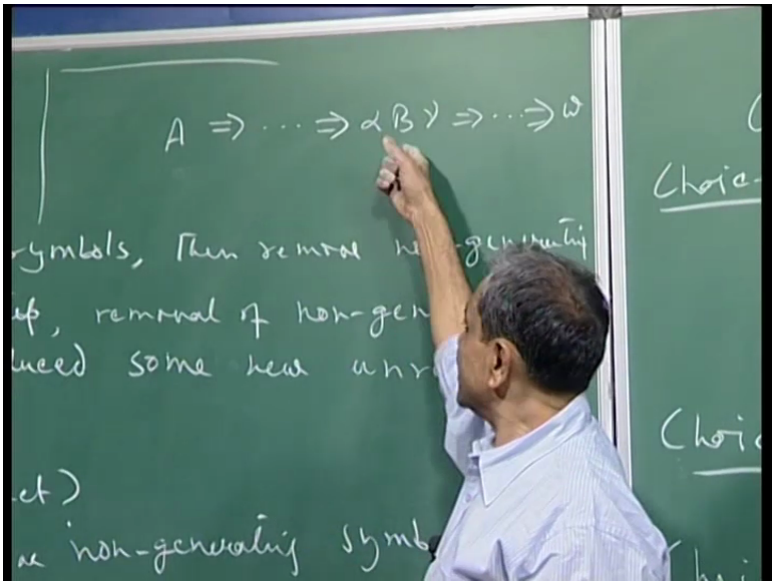
But now remember in which phase we are in? We are in the phase we are looking at whether some symbol is not reachable and then we are removing them. But if we have found A to be reachable and then we can B reach from A as this shows, then surely B will also be reachable, is not it?

(Refer Slide Time: 53:25)



Reachability is transitive, right? If you can reach from S to A and in this caseas we see we can reach B from A. Therefore we can reachfrom S to B also. So this B will not be thrown out because it is not reachable. Because it is clearly reachable. If A is reachable then B is also reachable. So we will never throw out such a B.

(Refer Slide Time: 53:54)



And therefore it is not possible because of throwing out of some unreachable symbols in the second part I will make something which was already generating to become non-generating. So this situation can never happen. So therefore choice 2, that is first identify the non-generating symbols, remove them from the grammar, get the simplified grammar and now identify the unreachable symbols and nowremove those unreachable symbols.

In the process you are not going to go wrong and therefore the grammar that we will getwill not have any useless symbols.