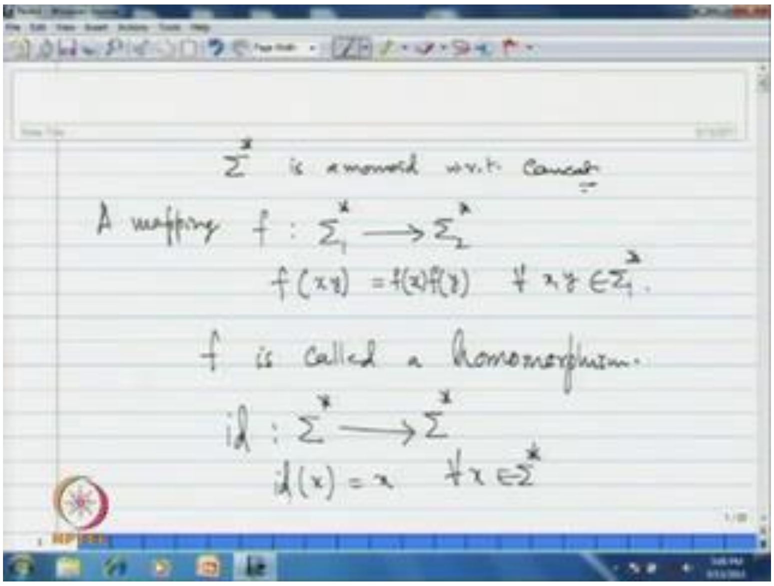
Formal Languages and Automata Theory
Prof. Dr. K. V. Krishna
Department of Mathematics
Indian Institute of Technology, Guwahati

Module - 7
Properties of Regular Languages
Lecture - 2
Homomorphism

In the previous lecture, we have discussed certain operations through which the class of regular languages is closed just to the elaborations. Thus, I reemphasize on the point that this closer properties are helpful to understand, some of the new languages to see that they are regular. In that context, we have observed that you know reversal of a regular languages is regular and right quotient. Similarly, one can define left quotient and understand that regular languages sort closed to the left quotient also with arbitrary languages, so that is where I have stopped my previous lecture.

Now, let me continue with of some of these closer properties of the regular languages. Now, I introduce a new concept in the context of regular languages, that we see using this also we see that the class of regular languages is closed with respect to this particular of portion. Now, before going to that new operation, so called homomorphism; first let me give you some worry about homomorphism that we normally discuss in case of algebras.

(Refer Slide Time: 01:53)

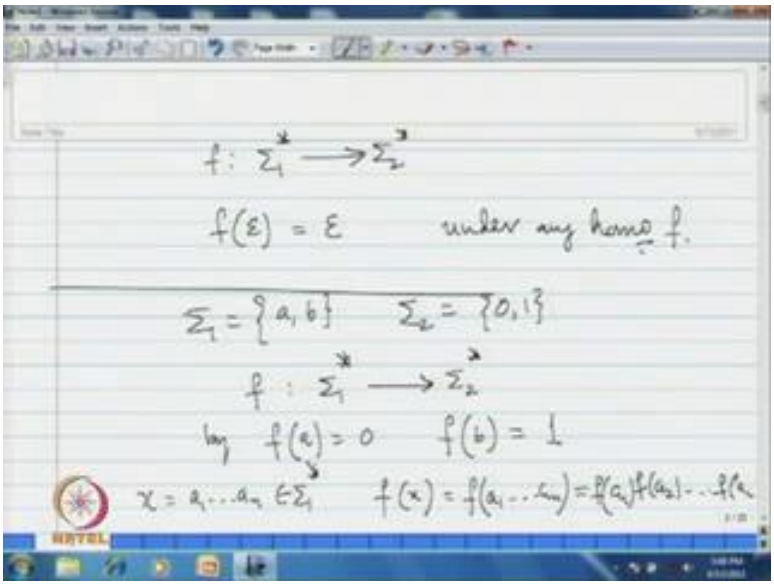


First, you understand or recollect, if you take an alphabet sigma, sigma star is a monoid with respect to the operation concatenation. Now, in case of algebras the way we understand the homomorphism, here if you consider a map f from 1 monoids say sigma 1 star to another monoids sigma 2 star to different alphabets, you may choose or two alphabets say may f a mapping is said to be homomorphism. If, it satisfies this property f of x y is equal to f x, f y for all x y in sigma 1 star, this is if a mapping, if it satisfies this property f is called a homomorphism.

Now, you look at the point that here sigma star it is not just the monoid, it is in fact, so called free monoid or the alphabet sigma, here this is a finite alphabet. And now what is a advantage in defining homomorphism in the context of you know this free monoids here, the set of all strings are an alphabet. It is sufficient to define the up mapping or the bases, you need not actually define the mapping for all the elements of sigma 1 star,, because sigma 1 is a countably infinite set.

So, you of course, one way of giving mapping you will define for every element here, but here the advantages you just define for the bases elements. Here, the elements of sigma and you can naturally extend using this property to for all the elements of this, so that it is a homomorphism. Now, to just to mentions about certain trivial homomorphism, if you consider the identity mapping from some alphabet sigma star to the same identity mapping is defined by f of x equal to x for all x in sigma star, you can quickly see that this mapping is identity mapping is homomorphism.

(Refer Slide Time: 04:58)

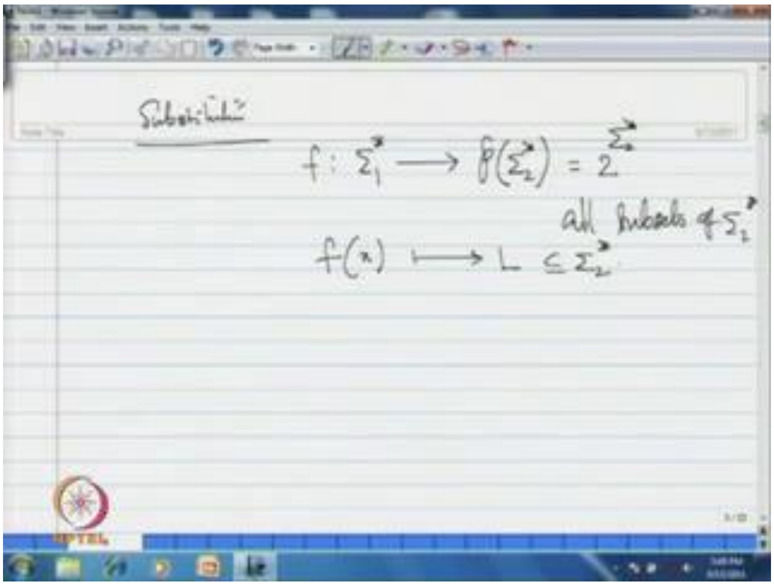


And, one important remark is this under any homomorphism f from say sigma 1 star to sigma 2 star, the empty string will be map to empty string only under any homomorphism under any homomorphism. So, here what you look at as I have remarked if you just defined the elements of for the elements of sigma 1, you can naturally extend using the property of that. Now, for example, if you take sigma 1 to b say a b, sigma 2 for example, say 01, if I define f from sigma 1 star to sigma 2 star by f of a is equal to 0, f of b is equal to 1, I have defined only for the bases elements.

Now, if you take any x in sigma 1 star what we do this f of x will be you know f of a 1 and so on, a n, since it has to satisfy the property what do we get f of a 1, f of a 2 and so on, f of a n. Because we want that property homomorphism property need to be satisfied and we are given for a and b the images, so f of a 1, if it is a you put 0, if f of a 2, if it is b you put 1 and so on.

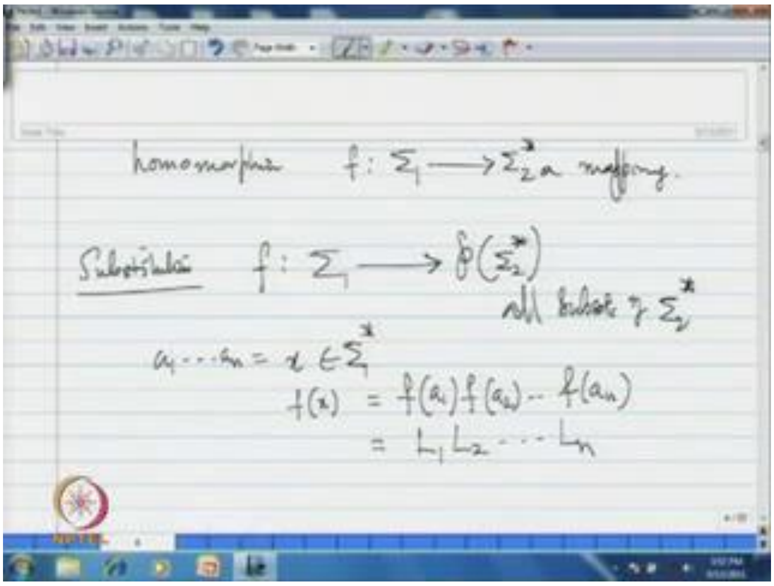
So, precisely the mapping is here defined by if you take any string over the alphabet a b, the images whenever there is a you will have 0, whenever there is b in the string in the of sigma 1 star, under consideration you put 1 that is clearly a homomorphism.

(Refer Slide Time: 07:31)



Now, a generalized to the concept of homomorphism is called substitution in which case what do we do in instead of giving string as an image. We give a language as an image for example, a mapping f from say sigma 1 star to now we consider language means I consider power set of sigma 2 star. That means, set of all subsets of sometimes people may write this as this sigma 2 star means all subsets of sigma 2 star; that means, all languages over sigma 2. Now, what do we do here instead of assigning a string in case of homomorphism here, we assign a language for each string you assign a language l over sigma 2. Now, as I have mentioned that it is sufficient to give images for the elements of the bases that is a alphabet here.

(Refer Slide Time: 08:48)

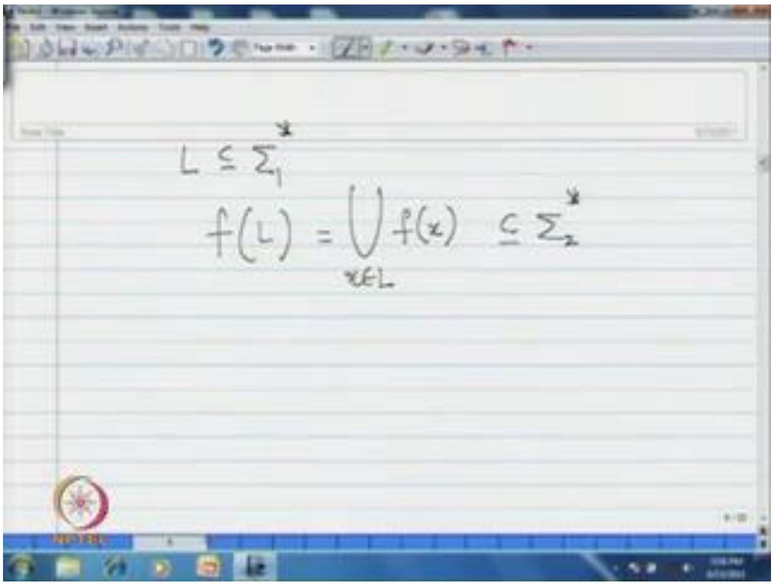


Now, homomorphism concept we may simply say it is mapping f from sigma 1 to say sigma 2 star a mapping. So, this is how we can now define and now to see for a string, we can naturally extend using the property of homomorphism condition of homomorphism. Now, a general concept to this homomorphism is available called substitution, where instead of assigning a string as an image what do we do, we assign a language or the alphabet sigma 2.

This is all subsets of sigma 2 star, so that means here, if you take any string x in sigma 1 star. Now, the image essentially for each symbol you will be assigning 1 language, now f of x will be as for the property f of a 1, f of a 2 and so on, f of a n; that means, here for a 1, a 2, a n, x if consider. This is the condition it has to satisfy like in homomorphism and what are the language that you are assigning for a 1, say let me call it as L 1, L 2 and so on, L n a concatenation of those languages, we consider.

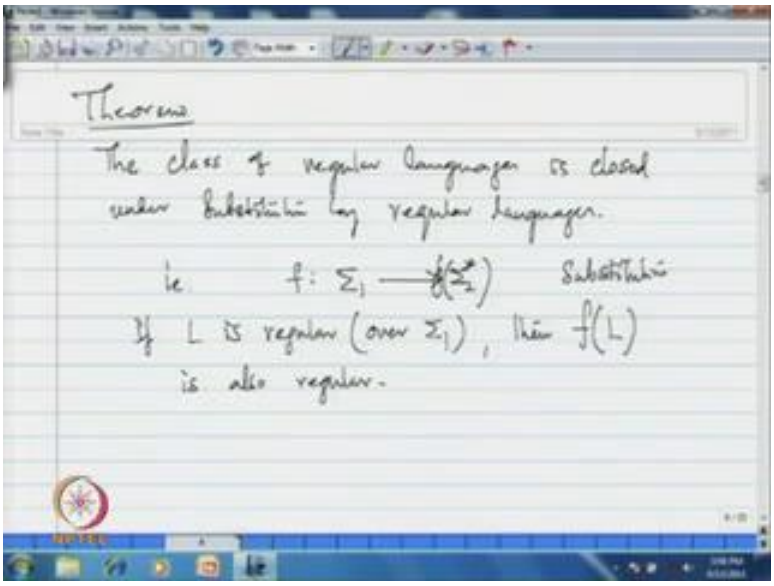
And naturally for each string, we have a language over sigma 2, under this substitution, so substitution essentially it is general concept to homomorphism. So, where for each string you will be assigning a language instead of defining, again since we are expecting this homomorphism property to be satisfied. In case of substitution as well what do we do we simply give or we simply assign a language for each symbol over sigma 1 and we naturally extend for each string and it is not only for string.

(Refer Slide Time: 11:17)



Now, as a general concept to as generalization of this for each language naturally for example, if you consider language L or sigma 1 star, under the substitution we can now talk about f of L image of L, that is for each string you consider the image. This is a language, you take the union here x in L, this is of course, a subset of sigma 2 star, because f x is a language over sigma 2 and union of languages here, that is a set over that is language over sigma 2, so under substitution we defined f L this way.

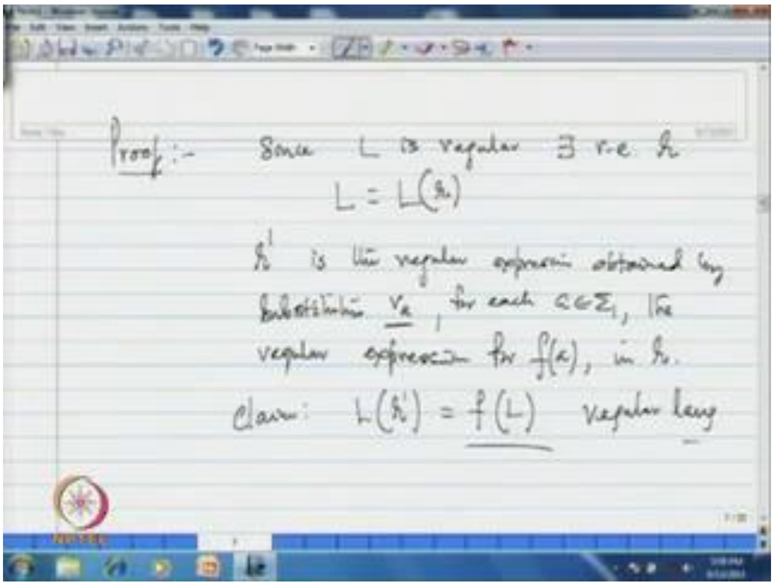
(Refer Slide Time: 12:21)



Now, let us look at some of the properties related to the homomorphism and substitution. The class of regular languages is closed under substitution by regular languages, this result essentially what it is saying that is let us consider f a substitution from say sigma 1 to sigma 2 star. Now, if L is regular of course, this is L is a language over sigma 1, then f of L as defined earlier is also regular, that is what is the statement here the meaning of this statement.

The class of regular language is closed under substitution by regular languages. Here, you consider the substitution f, this is substitution what we are targeting to understand that, if L is regular f of L is also regular. Now, what do we do we take the aid of regular expressions and understand this particular result.

(Refer Slide Time: 14:45)



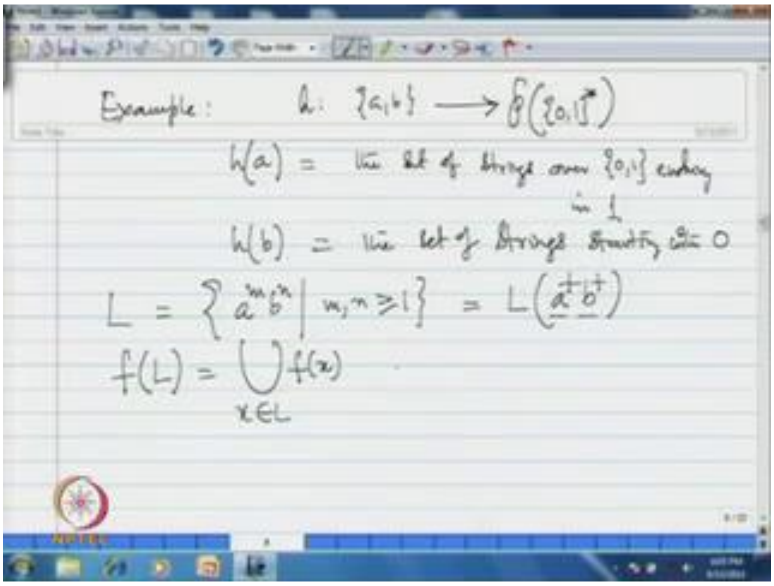
Now, the proof of this, what do we what do I suggest here, as L is regular you will have a regular expression for that. Since, L is regular there exists the regular expression, let me write r regular expression r, such that L is equal to L of that regular expression r. Now, whatever the substitution that we have assume that f is a substitution and now the substituting languages since they are regular languages for each such regular language you have a regular expression.

We substitute the regular expression at each symbol and the resultant 1, that we prove that this is this is what is representing that f of L. So, the claim essentially, so if I write r dash is a regular expression obtain by substituting. Let me call r a for each a in sigma 1, the regular expression for f of a, first f of a under the substitution f, this is a regular language that is a information given to us, since it is a regular language you will have a regular expression.

Let me, write it as r a, what do I suggest in r you substitute r a, wherever a is occurring the corresponding regular expression r a and obtaining the regular expression. Let me call it as r a dash, so the substitution of r a, this is the regular expression for f of a in r. Now, the claim is the language represented by this r dash is the desert language f of L.

If, I can prove this claim, we understand that the f of L here, the image under the substitution it is a regular language. Because a language represented by regular expression is regular, thus you can conclude that this is a regular language, before going to this let me probably to explain this, let me consider an example first.

(Refer Slide Time: 18:11)



And then we can understand what essentially the mechanism here, we are considering in this result. Consider this example, consider this substitution h from a b to power set of 01 star; that means, for symbol a and b, I substitute the languages for 01. Let me define h of a all the strings the set of strings power 01 say ending in 1 for example, for h b order what I consider the set of strings starting with 0, let me consider these languages.

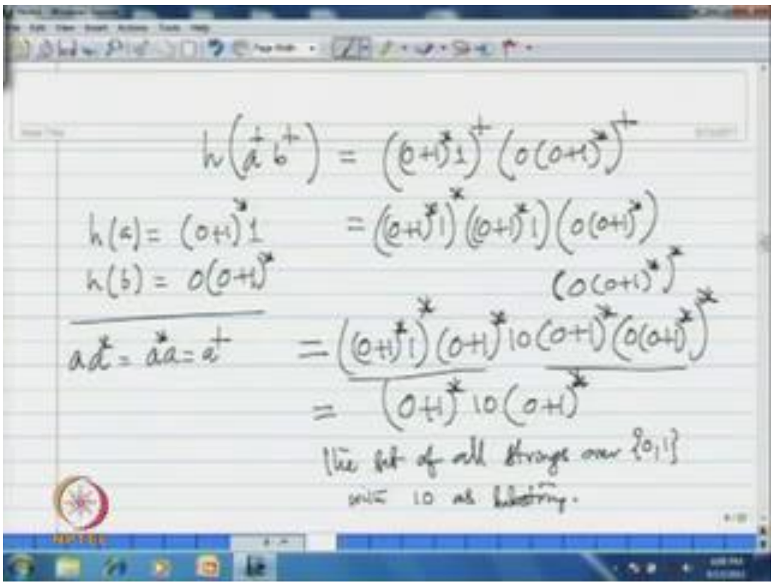
Now, if I consider the language L say a power m, b power n, such that m n greater than or equal to 1, you know this is a regular language. Consider this language, what is f of L, f of L is a essentially, wherever you take each string here from L by definition this is union of x in L, f of x. So, considering each string this is of the form a power m and b power n and each symbol a.

The corresponding language, whatever h of a is given you put it there, substitute there. And h of b, you consider whatever is given you substitute there and for each string you get a language. That language, you know concatenation of all those languages that is what is f of x by definition, consider all those languages and take union and this is what is f of L.

Now, what is the method that we are adopting to understand this f of L very quickly is considering, the regular expression corresponding to L. So, the regular expression corresponding to L is of course, this is a regular language and the regular expression corresponding to this is a plus, b plus, this is the regular expression. Now, in this regular expression for each occurrence of a, you substitute and you understand that h of a is regular, h of b is also regular.

So, these regular languages they correspond to regular expression is substitute in each occurrence of a, in a regular expression of L and each occurrence of b, the regular expression of h b is substitute. And look at the regular expression that you are getting and that is what is the process here we pursue to understand that the resulting language is regular.

(Refer Slide Time: 21:33)



So, here a plus, b plus is a given language and h of a is all those strings ending in ending in 1. So, the regular expression corresponding to that is 0 plus 1 whole star 1, this is the regular expression and h of b is given to you as those strings starting with 0; that means, this is a regular expression corresponding to this. Now, the suggestion is if I write this as in a loose sense like this. Of course, the language represented by a plus b plus, the image of that we substitute this way, that is 0 plus 1 star, 1 plus and 0 plus 1 star.

Now, you know this property a star for a regular expression a, concatenation with a, is a is equal to a plus. Now, we elaborate this using this property that is 0 plus 1 whole star 1, whole star and here 0 plus 1 star 1. And similarly this side I use 0, concatenation with 0 plus 1, whole star and now 0 concatenation with 0 plus 1 whole star, let me write it down here concatenation with 0 concatenation with 0 plus 1 whole star.

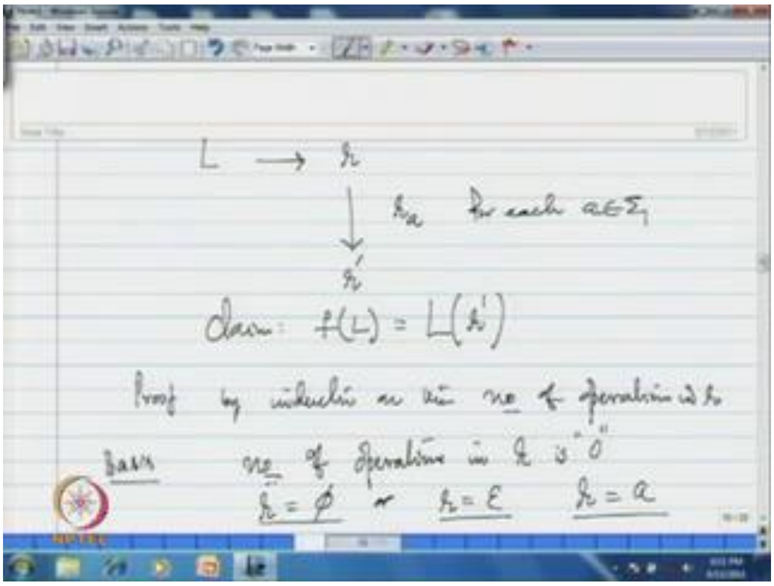
Now, you can quickly see that this is 0 plus 1 whole star 1, whole star, let it be this and here 0 plus 1 whole star 1 0. Because, the concatenation is associative and thus I can simply write you know this 1 0 and the remaining here. This is 0 plus 1 whole star and the remaining here this side is 0 concatenation with 0 plus 1 whole star, this is what is the resultant in string.

Now, let us look at this portion and this portion, here in this a string is of the form, if I write x y, here if I choose x to be Epsilon, I mean Epsilon, because Epsilon is in this. There is a regular expression for which there is a star, if I consider Epsilon, this is simply y and here y is an element of 0 plus 1 whole star; that means, any arbitrary string of 0 plus 1 whole star; that means, over 0 1, you can get in the left side.

Similarly, on the right side any string over 0 plus over 0 1, you will get and hence you understand this. In fact, equal to; that means, and the left portion 0 plus 1 whole star is a subset of this and since it is a whole set, I can quickly conclude that, this is 0 plus 1 whole star 1 0, 0 plus 1 whole star. Now, this is a regular expression you get and in fact, from this we can understand, the image of consider the given language, under this substitution h is all those strings in which 1 0 as substring.

So, this is the set of all strings power 0 1 with 1 0 as substring, so this is a image of the given language under substitution. Now, whatever the mechanism that we have pursued here, exactly the same thing we wanted to adopt in the proof and see that it is a regular language and in fact, it is representing h of L.

(Refer Slide Time: 26:19)



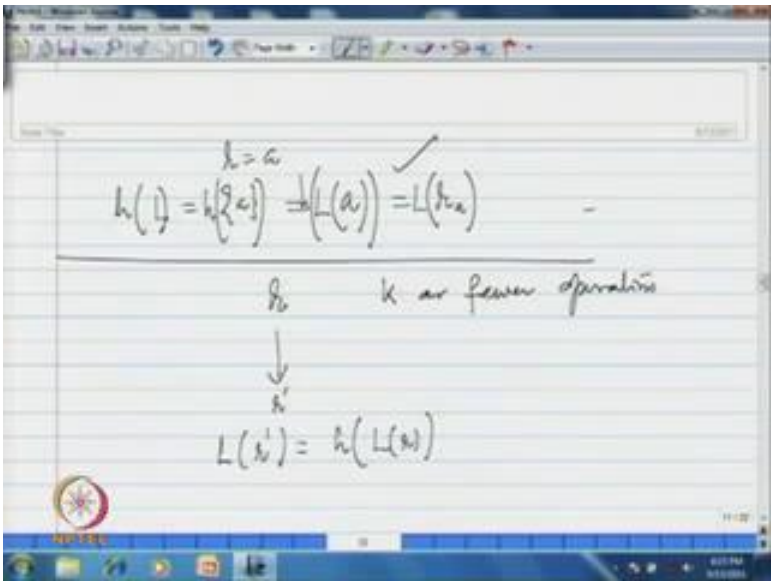
So, here what do we do given L, we have considered the corresponding regular expression r and we have obtained r dash, that is the regular expression by substituting r a for each a in that sigma 1. And, as I have mentioned the claim here is f of L is the language represented by this, we prove this result by induction on the number of operations so far, this proof of the claim by induction on the number of operations in r.

Since, r is a regular expression there are three possible operations, concatenation, addition and cleanly star, so we use induction on the number of operations on r. Now, for the bases you assume the number of operations in r is 0, there are only 0 number operations; that means, the possibility for r is maybe empty r. This r can be the regular expression Epsilon r, r can be simply a, if there is no operation involved the corresponding regular expression is should be 1 of this it can be empty Epsilon r a.

Now, in this first case since there is no symbol of sigma 1, so the substitution is prevail here and hence the corresponding r dash is again phi. The regular expression phi and similarly if the regular expression is this, in this case there is no symbol of sigma is involved.

And hence here also trivially r dash is Epsilon and we can see that this substitution this f of L, that is also empty language and here that is single to the Epsilon. And these two cases you can quickly see that trivially f of L is equal to the image of the language L is the same. In case of r is equal to a, whatever the symbol that it is a of sigma 1, the corresponding regular expression r a will be consider.

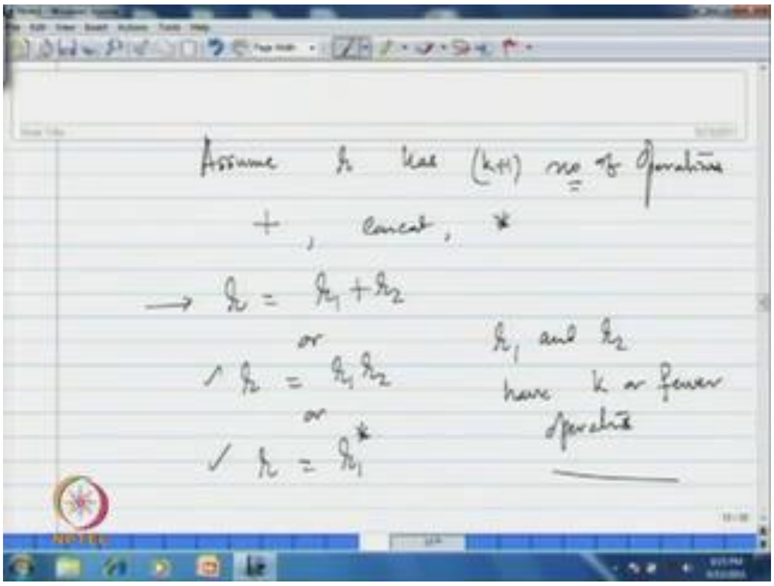
(Refer Slide Time: 29:22)



So, in case r is equal to a here L is single tan a and now what we are doing we are considering the regular expression a and we are substituting r a in place of that and we are considering that. And, it is given that r a is a regular expression, we have considered that r a is the regular expression, corresponding to this f of a and this is corresponding to f of a and this is what is f of L.

So, by substituting here, so h of L, now can understand that h of L by substituting r a in place a, we get this and thus we can understand that. So, if there is for the bases, we can see that the result is true, now we assume this result is true, if a regular expression r has k or fewer operations assume that there is this is happening, if r has k or k or fewer number of operations. Now, whatever that r dash we are obtaining what is meaning here by substituting each our assertion is true that is L of r dash is h of L of r, we are assuming this is happening.

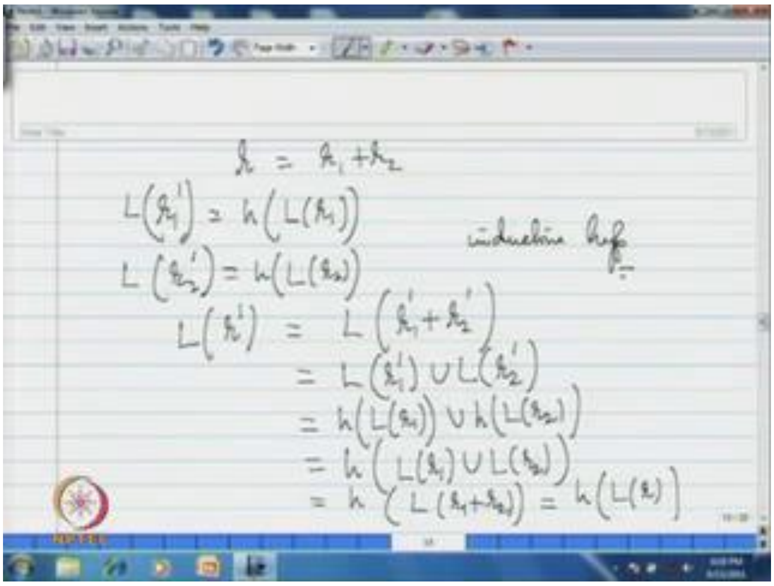
(Refer Slide Time: 31:43)



And now consider r has k plus 1 operations for induction, then if there are k plus 1 number of operations at least 1 operation, we can see there, the possibility is either it can be plus or the concatenation r star at least 1 of them is involved. Thus we can write r to be say r 1 plus r 2 in this form or if can be it is of the form r 1, r 2 or we can say that r is of the form say some say r 1 star, and understand that in each case any of this case r 1, r 2 should have k or fewer operations.

So, what do we do we apply induction on r 1 and r 2 and we take that assertion and proceed to understand that the result is true for k. If r has k plus 1 number of operations, I consider this case and observe that our assertion is true, when r is equal to r 1 plus r 2 and 1 can similarly observe these other 2 cases and understand that the induction from induction the result follows.

(Refer Slide Time: 33:41)



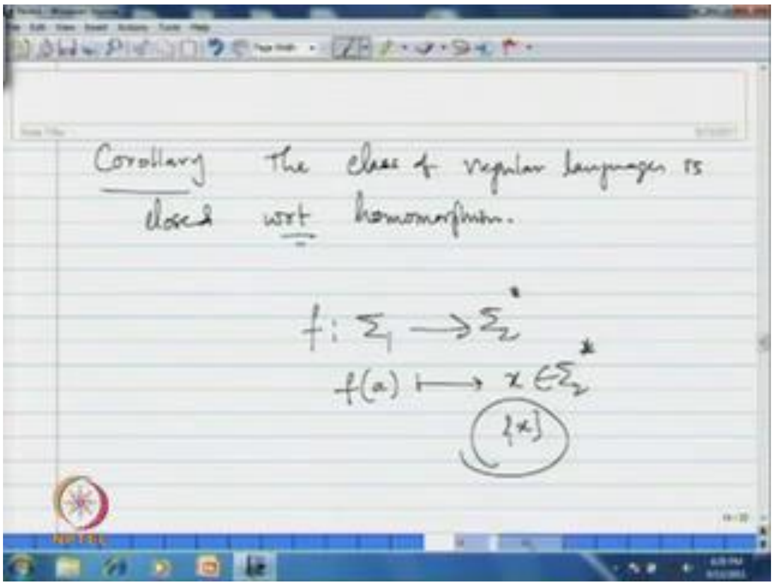
Let me consider the case r is r 1 plus r 2, let me write r 1 dash, this is obtained from r 1 by substituting r a for each occurrence of a and by induction what we have L of r 1 dash is h of L of r 1. And similarly let me write r 2 dash is the regular expression obtained from r 2 by substituting r a for each occurrence of a of sigma 1 and from by inductive hypothesis. Because r 1 and r 2 have k or fewer operations, this is inductive hypothesis what do we do what we have observed that L of r dash is h of L of r that is what we have to observe.

So, let me start with L of r dash the language of r dash, r 1 dash, r 2 dash why the reason because r is the form r 1 plus r 2, r 1 dash is the regular expression obtained from r 1 by substituting each occurrence of a b y r a similarly in r 2. So, r dash by definition it is nothing else, but r 1 dash plus r 2 dash, now you proceed to understand this is the manipulation with regular the regular expressions and the corresponding languages.

So, this is union r 2 dash and from inductive hypothesis what have this is h of L of r 1 union, h of L of r 2, this is from inductive hypothesis and since h is substitution you can write it this way. We have this property and now again the regular expression h of L of r 1 plus r 2 and this is what is r, so when plus is involved has operation here we observed that l of r dash is h of l of r.

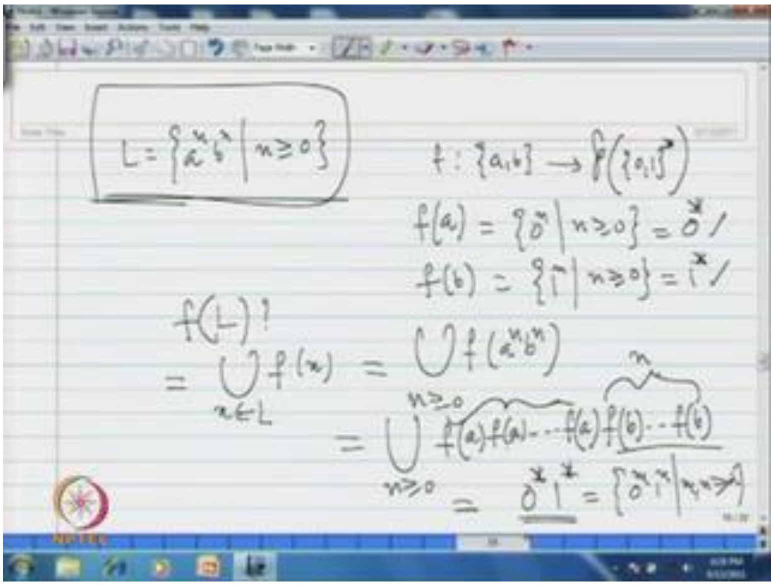
Similarly, if the concatenation is involved, you can proceed this way and understand this absorption is true in when it is having k plus 1 operations r. In case of cleanly star and thus by induction, we have observed that substitution of regular languages is you know in regular languages, it is a regular operation and thus the class of regular languages is closed with respect to substitution by regular languages.

(Refer Slide Time: 37:13)



Now, an immediate consequence here is this the class of regular languages is closed with respect to homomorphism. Now, the question is when we have talked about substitution, I have mentioned that substitution by regular languages and you see homomorphism is a special case of substitution. And in this special case what we do from sigma 1 to sigma 2 star for each symbol, you are just assigning a string so; that means, in this special case the substitution is by single tan x of string. Essentially of sigma 2 and we know this is a regular language and thus from the above result as a consequence. We can quickly say that the class of regular language is closed with respect to homomorphism.

(Refer Slide Time: 38:38)



Now, let me discuss 1 example here, let me consider the language say a power n, b power n, that is that n greater than or equal to 0 or n greater than or equal to 1, whatever. Let me, consider this language and consider the substitution say f from a b to of course, power set of 0 1 star. Let, me consider this and we give the substitution f of a, I consider 0 to the n greater than or equal to 0 and f of b, I give 1 to n greater than or equal to 0.

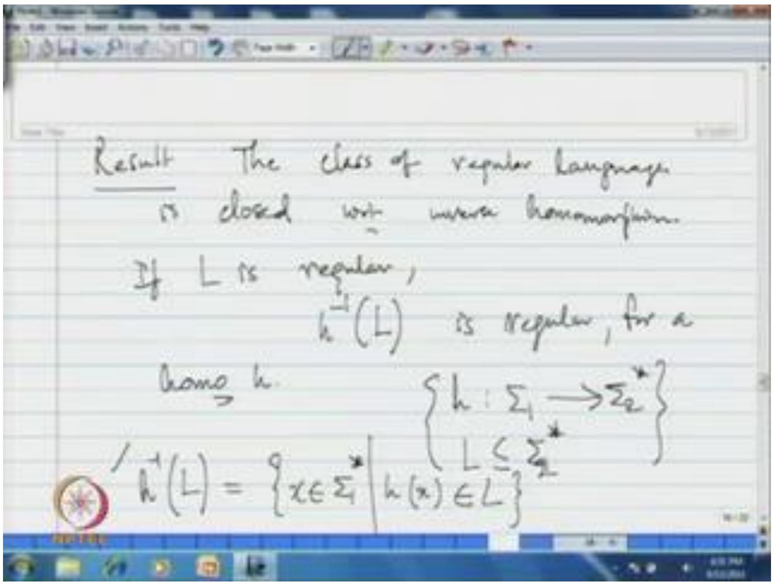
Now, what is f of L, if you look at this example you quickly see here of course, we so far, we have not obtained any regular expression for this language. But the reason is, because as an application of mainly road theorem have observed that this is the index of this language is infinite and from mainly road theorem, we have observed that this is not regular and hence we cannot expect any regular expression for this.

So, the previous approach, because taking the regular example substitute to the corresponding regular expressions, that we do not we cannot follow here. Now, let us analyze from the definition this is equal to union h of x is L, now I can write it as union n greater than or equal to 0, h of a power n b power n. So, this is a typical swing in L, union n greater than or equal to 0, here since h is I am writing h and f interchangeably here, whatever is considered let us follow the same consistently.

So, this is f of a, and so on, this is for n times, because this f is satisfying this property and similarly here f of b for n times, this is for n times and this is also for n times. But, whatever it is f of a is simply 0 star f of b is 1 star, so f of a when we have considered this is 0 star concatenation 0 star and so on. Because, 0 star for n number of times a concatenation of 0 star with 0 star for n times is nothing else, but 0 star.

Similarly, this f of b case, so whatever is the number of n times that you concatenate here you understand that this is equal to this 0 star, whatever is the number of n times you concatenate you get just 0 star. So, this is 0 star 1 star, so that is essentially 0 power m, 1 power n such that m n independently greater than or equal to 0, that is the language we are getting. Now, look at here we have considered a non regular language and we have used this substitution. We are substituting regular languages and of course, as a result this is here f of L, we have got a regular language.

(Refer Slide Time: 42:55)

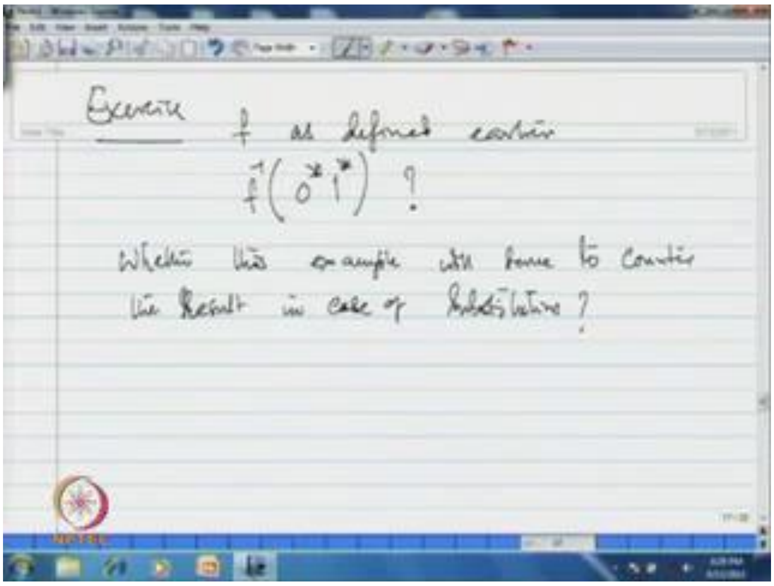


Now, let me state a result that is class of regular languages is closed with respect to inverse homomorphism. So, let me elaborate this result, that means if L is regular, because we understood that if L is regular f of L that is homomorphism image of regular language is regular. Now, inverse homomorphism image; that means, if L is regular and if you consider a homomorphism f h say for example, h inverse L is also regular.

If L is regular, h inverse L is regular, for a homomorphism h that is the statement. Of course, here h is a mapping, h is a homomorphism, say for example, sigma 1 to sigma 2 star and here L is a language over sigma 1. So, these things I am not mentioning, but what is h inverse L, if l is not clear, the inverse image of a set is essentially collect all those strings of sigma 1 star, such that the image of that should be in L, this is what h inverse L.

So, here you look at this result the class of regular languages is closed with respect to inverse homomorphism. Look at the previous example here, I have considered the non regular language, we know this is a language, that is not regular and I have substituted here regular languages and I have obtained a regular language, now the question is under this substitution, what is inverse image of this.

(Refer Slide Time: 45:54)

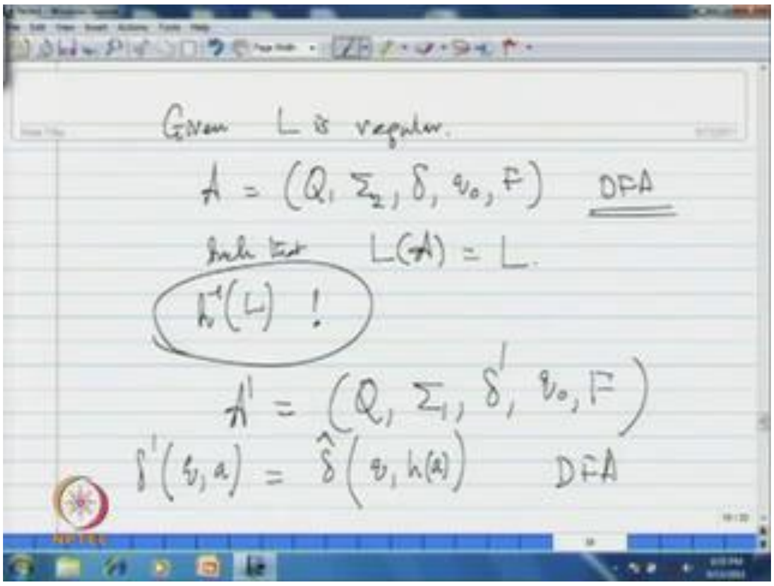


So, here I have f, let me use the same f as defined earlier in that example shown here consider this and what is f inverse of 0 star, 1 star, question number 1 and whether this example will serve to counter the result in case of substitutions. Take this an exercise and so what is the result in here I have stated the class of regular language is closed with respect to inverse homomorphism.

Now, whether the class of regular language is closed with respect to inverse substitutions, suppose if 1 question here I have considered a non regular language and I have observed that the image of that is this. Now, understand what is the pre image of this particular language, here is a regular language and here is a substitution and now understand what is the pre image, essentially from which what we can conclude whether this result is not true, in case of substitutions or not.

Let us, prove this result, the result is if L is regular, h inverse L is a regular for a homomorphism h.

(Refer Slide Time: 47:49)



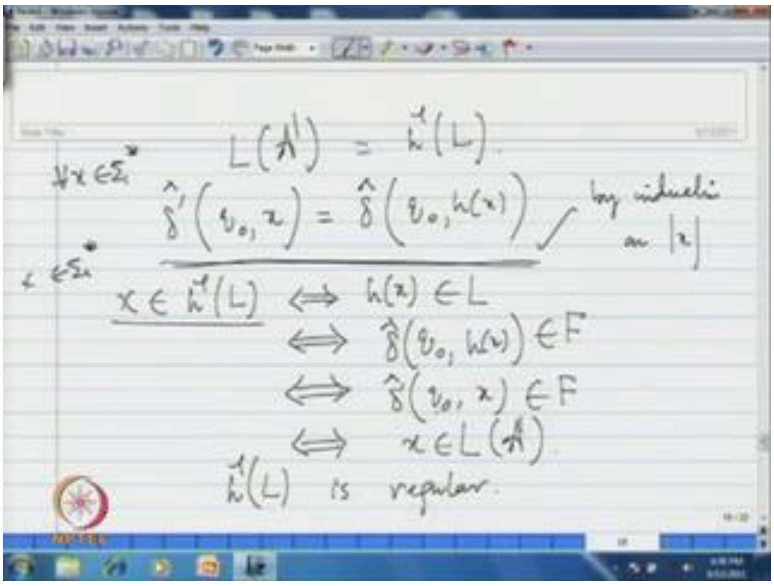
So, what is the proof idea here is given l is regular, so consider the corresponding you know some DFA, let me say a is a DFA accepting L. Let me write it as Q sigma 2 say delta q naught f of course, here this is over sigma 2, because we are talking about inverse image. So, L is a subset of sigma 2 star and its pre image h inverse L is a language over sigma 1, so this is sigma 2, so sigma 2 is alphabet here and let me consider this is be a DFA accepting L.

Now, we will con construct a DFA for h inverse L to show this is regular, what is a proof idea here, because h is a homomorphism carrying the languages are you know the strings from sigma 1 to sigma 2. We defined new transition, because we consider the same a state set of course, the alphabet will be sigma 1 here if I call it as a dash. So, same state set I consider and a alphabet here is sigma 1, new transitions delta dash I call and same initial state, but here the final states will be consider to be same.

Now, how this transition map is define, this is defined as essentially composition of the original map with h, that is how we define that means, this delta dash at any state for a symbol a in sigma 1, we define it as delta cap of q h a. Because in the given automaton h a is an element of sigma 2, that is a string. In the state q, If you apply the string h a wherever it is going since it is a DFA, this is a DFA you have a unique state here and thus delta dash as defined here in the state q.

If you apply a the h a will be applied in the state q of the original automaton, you will get a unique state. And thus you see here for each state in the symbol you will get exactly 1 state and hence this is a dash is a DFA, so this is a DFA.

(Refer Slide Time: 51:09)



We define and we prove that the language accepted by this a dash is h inverse L, to prove this first we understand this property for each string x in sigma 1 star, what do you have the image of this. If you substitute in q naught, in fact in any state this is nothing else, but delta cap q naught h x, you look at the definition, this is the same is defined for each symbol of sigma 1.

Now, we are saying this is true for all x and sigma 1 star, so this can be prove quickly by induction on length of x, because when you take m t string you can observe and for each a in sigma 1 star; that means, length 1. This is straightforward from the definition of delta dash in fact you can extend this and understand that for each string of sigma 1 star, you get this property.

Once you understand this, let me just show you that if you take any arbitrary string x in h inverse L, by definition this image of this is in L, that is a definition a string is in h inverse L. Now, h of L is in l; that means, in the initial state of the original machine, if you apply h x you are going to your final state that is an element of f.

Now, we know that this is nothing else, but this because that is a property 1 can observe, so this is an element of f. So, in the initial state if you apply x, if you are going to the final state of a dash; that means, this string x is in language accepted by a dash and thus you see any string, any arbitrary string x in sigma 1 star. If you take any string x in sigma 1 star, what is happening a x is in h inverse L, if and only if x is in L of a dash, and since a dash is a DFA, h inverse L is regular. So of course, you can take this as an exercise and prove that this property holds that is what we have used here to prove that h inverse L is regular and hence we have this same.