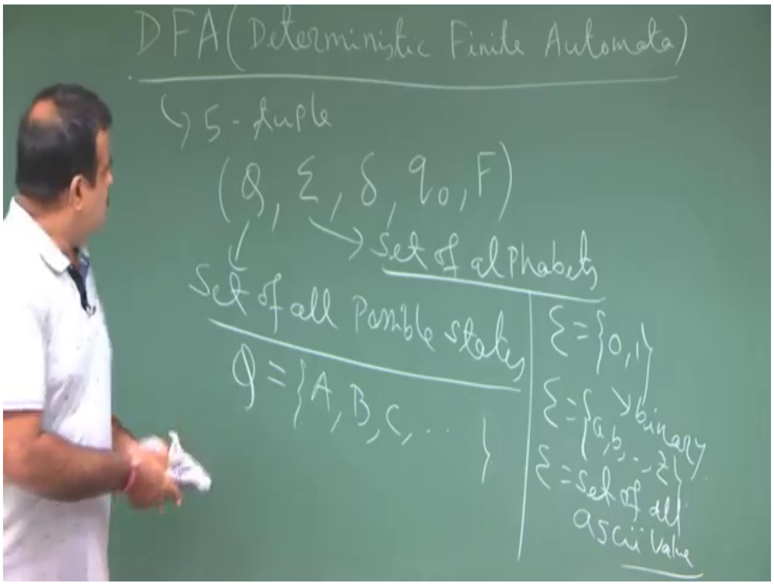
Introduction to Automata, Languages and Computation

Prof. Sourav Mukhopadhyay
Department of Mathematics
Indian Institute of Technology, Kharagpur

Lecture – 01
Deterministic Finite Automata (DFA)

(Refer Slide Time: 00:21)

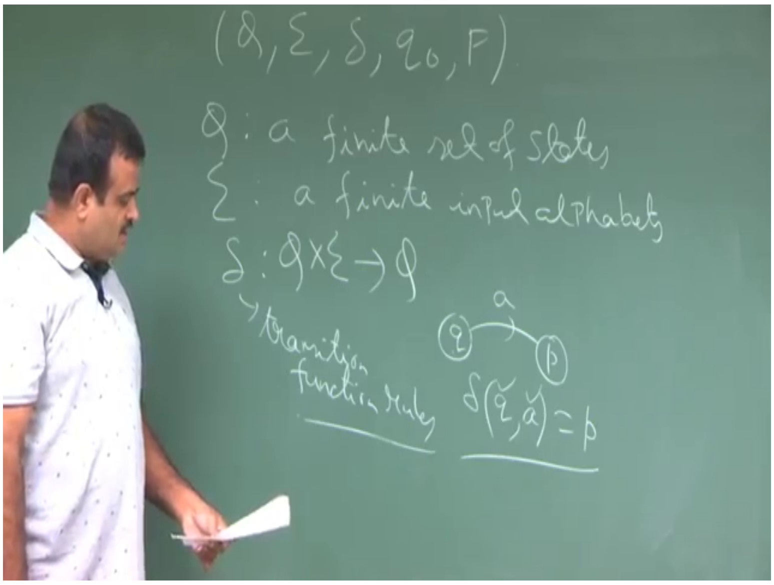


So, we will talk about deterministic finite automata or DFA. So, it is basically a five-tuple consists of Q, which is a set of all possible states, then sigma this is the set of all possible alphabets. So, these are all finite. Delta, this is called transition rule which takes one state and one alphabet and it will go to the new state. q 0 which is a starting state we have to start from a some state, so this is the starting state which is there is only one starting state for finite automata. So, there are some different type of automata which can be having many starting state, but for our case here finite automata DFA in a epsilon, it will all have the one state to start that is the q 0 or which is called starting state. And then we have a F, F consists of set of all final state. So, this could be many.

So, this Q is nothing but set of all possible states. This is the Q that means, all possible state our system can be. So, this usually we denoted by capital letter. So, Q is some state say Q A, B, C, D like this. So, use some capital letter to indicate this states.

And this is basically the set of alphabets; this is the another input. So, this we this could be 0, 1, this is the binary input. Alphabet could be binary, then it is 0, 1 or this would be some small letter like English alphabet from a to z, these are the small letter. Because capital letter we are going to use for the state, and small letter we can use for the alphabet. Or it could be a ascii value this could be set of all ascii value of ascii value. So, these are the alphabets that is the another part of the input of this finite state machine ok.

(Refer Slide Time: 03:40)

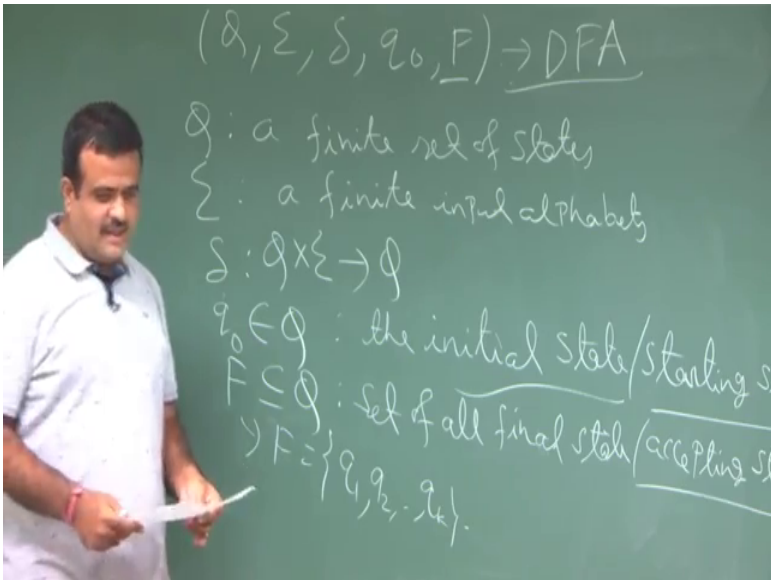


So, far this is. So, let me write it again. So it is basically five-tuple Q, sigma, then the rule, and q 0, F. And Q is a finite set of states. So, if that is our Q set of all possible state. This sigma is the a finite input alphabets.

And the delta is the transition rule. So, our system is at any state, suppose our system is here some state say q, and then it will take an input, input it is a a, and it will go to another state p. So, this is in other word we can write delta of q, a is equal to p this is the way. So, we are at state q, and we take an input alphabet a, and it will go to the new state p, so that will by that this rule this is called a state transition rule. So, this is the transition function or transition rule. This is called transition function or rules.

So, this is a function form Q, because this is one input state, another input is an alphabet. So, Q cross delta hey sorry Q cross sigma to the it will go to the another state Q cross sigma to Q. So, it is taking one input as a present state and another input is the input alphabet and it will move to the new state. So, this is the state transition function.

(Refer Slide Time: 06:00)



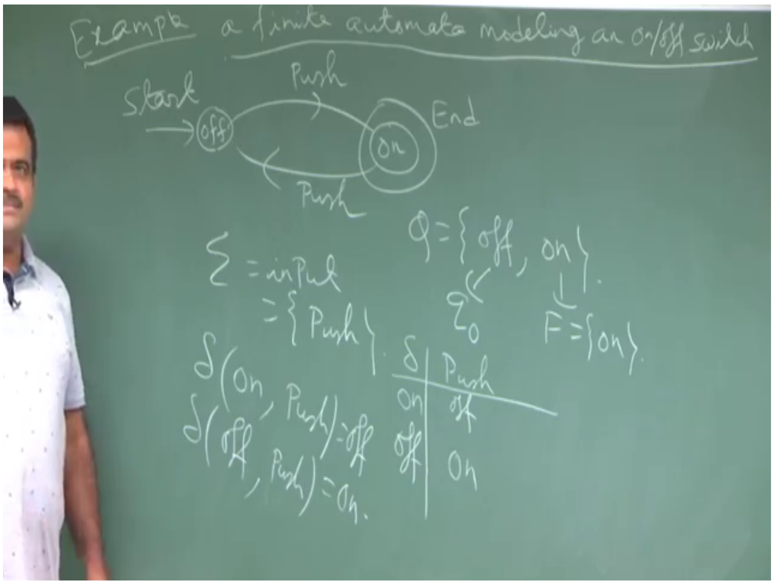
And then we have a starting state. So, you have to start with a, from a state. So, that is q 0. So, q 0 is belongs to Q is called the initial state or the starting state of the system initial state or the starting state. So, it needs to start from a state ok, so that is the q 0.

And we have the F, F is also consists of some states which is called final state. There could be one final state there could be many final state, so that is why f is a subset of Q is the set of all final states. So, this is the final state or it is also called accepting state, set of all final state or accepting state set of all final state or accepting state. So, this is the, this is a, this could be 1 also, this could be also. So, F consists of some state say F is nothing but some q 1, q 2, q k, it could be only q k or q 1. So, F consist of some states which is defined to be as accepted state.

So, this five-tuple is referred as DFA, so any DFA. So, this is the deterministic finite automata. So, deterministic finite automata is basically five-tuple. This is the set of all possible state, this is finite. This is the set of all possible alphabet input, this is also finite. This is the rule, the transition rule currently we are at some state, we take input, we will go to another state, so that rule. That rule will come from this function. So, this is a function from delta cross sigma to delta, I, this is a function for this Q cross sigma to Q. And this is the initial state our system is on initial state, and then this is this is referred as final state. So, any DFA consists of this five-tuple.

So, now, we will take an example, very simple example of modeling or switching, switch on off. So, if we have a switch, it has all these two state either it is on or off. And depending on the I mean we will just push where there is only one input, alphabet that is push.

(Refer Slide Time: 08:43)



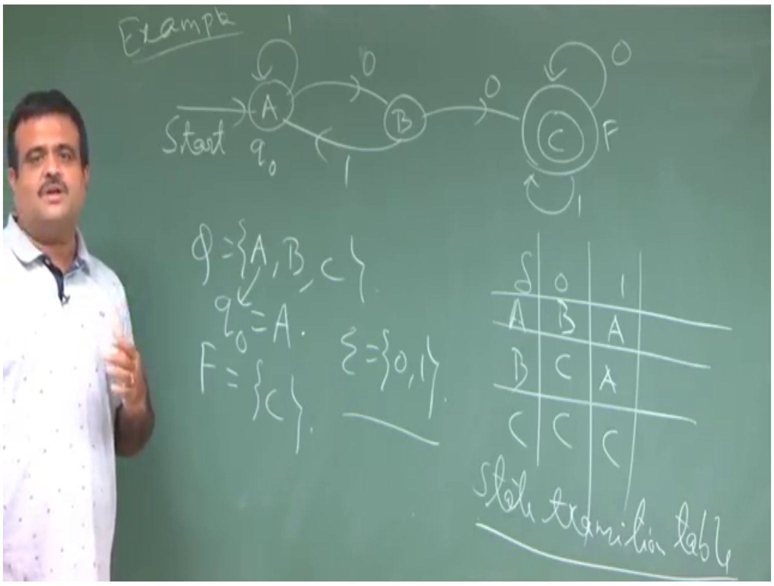
So, we basically have two states; one is off another one is on. And we have only one input which is push. We have a switch if it is on, if we push it, it will be off. If it is off, if we push it, it will be on like this. So, this is push button. So, we are at this stage, where the switch is off. Now, we will put it on, push, then it will be on, switch is on. We will push; it will be off. So, this is our, this is the starting step if we can say we start and this is the ending state or accepting state.

So, for accepting state, we will use this type of double circle which are belongs to F. So, this is there two state off and on. This is the q 0 starting state or the initial state. And this is our this is F. F consists of this step on. We will accept it if it is on like that I mean just the convention. And what is the delta. So, the states are this and sigma. So, sigma is the input, input alphabet which is nothing but push that is our sigma. We just push, switch push. We have a switch you just press the button push that is the sigma. I am accepting it as is this. So, what is the transition function? Transition function we can write in the table. So, this is delta. So, we have two state, one is this on; one is off. And we have two only one input which is push.

So, if we are at off, on, if we take the input push, it will be off. If we are at off, we take the input it will be very simple model of this switching system. So, this is the on-off switch. So, this is an example, where we are modeling a on-off switch using a finite automata. If finite automata your finite automata modeling an on-off switch, this is an example a finite automata modeling and on-off switch ok.

So, this space if we are on state, and if we take the input, there is only input, this will be off. And if we at off state, and if we take the input, it will be on very simple example ok. We can take another example which is not this much simple, you may have many other state. And instead of one input alphabet, we can have more than one like if it is binary 0, 1.

(Refer Slide Time: 12:21)



So, we can have another example like this. So, we have few state A, B, C. And we referred as this is the starting state or the initial state start starting state ok. And this we refer as final state or the accepting state. So, what is Q? Q is A, B, C. And the q 0 is A this is our q 0 starting state. Now, we have to and what is F? The accepted state or the final state is C. It could be many but here in this example we have one.

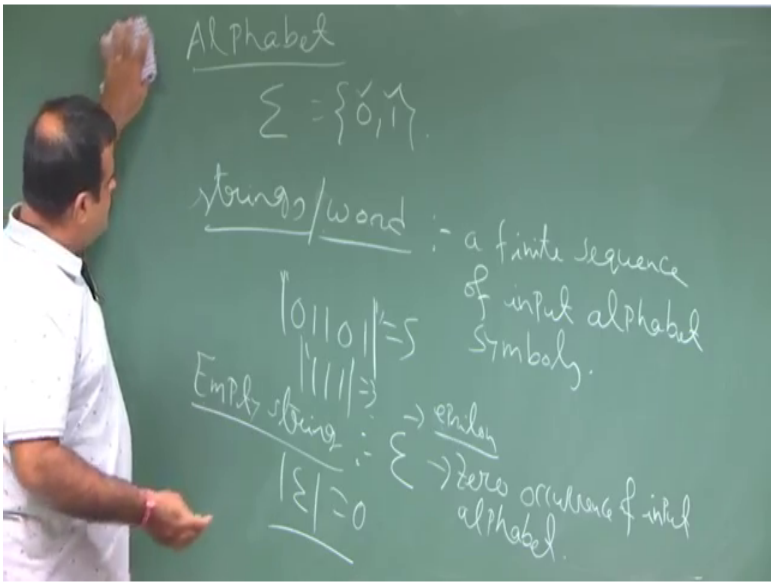
So, now, we need to define the rule the delta. So, delta means so we are at suppose here at this state. So, let us define delta. So, these are the state A, B, C, and there are two possible inputs 0 and 1. So, now, if we had a we can write this in the table if we are this that what this is called a state transition table, state transition table ok. We are at A.

Now, if you see A 1, we will have a cell loop. So, A with and if we see a 0, it will come here. So, A, A with 0, will be B, A with sorry 0 will B, B and A with is A. So, now, B now from B if we see A 0, it will come here. And if you see a 1e, it will come here. This is we are defining a finite state machine I mean. So, now, so this will be from B, if we see A 0, it will C from B. If we see A 1, it is coming back to A. And from C, we can have if you see a 0, we are here; if we see 1, then also here. So, C, if we see a 0, C 1, C, so this is the way we have defined this.

So, this is also the example of DFA, so where A is the starting state. And here sigma is the binary input 0, 1 binary alphabet. And this is the final state, we do the double circle for the final state. And for starting state, we will just use this as we start this is our q 0 starting state and this is our F final state. So, this is another example of DFA - deterministic finite automata, where we have more than two steps as two state and we have two input 0 and 1.

Now, we will define more on alphabet that is the string we will see how we can put the string, run the DFA, and whether we can reach to a final state or not. So, that will be the accepting string rejecting string. So, those we will discuss now. So, now, we concentrate on the that sigma the input alphabet. And we defined the string, and then we will see what do you mean by the string accepted by a DFA ok.

(Refer Slide Time: 16:12)

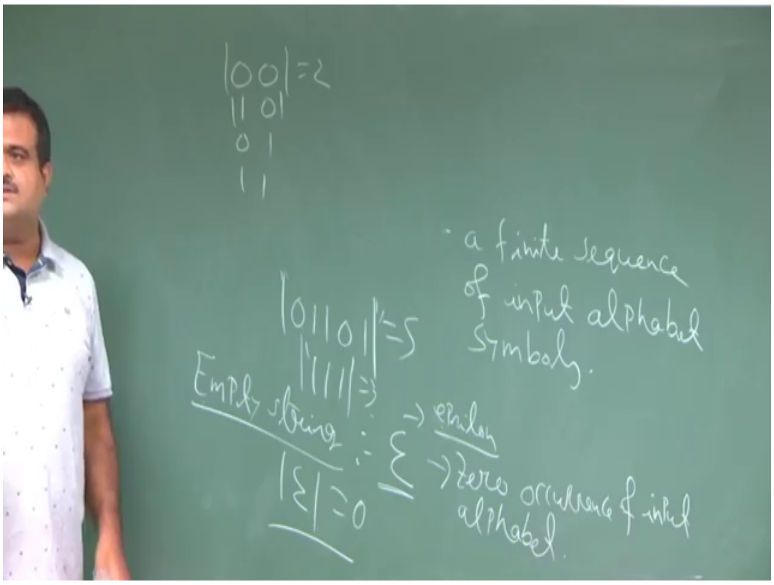


Let us talk more about this alphabet. So, alphabet is sigma. So, for binary it is just 0 and 1 for example, ok. It is a finite input we can say ok. It would be 3 also like this. Now, we define a string with this alphabet, string or it is called word also. This is basically a finite sequence of input alphabet this is defined as a finite sequence of input alphabet symbols. This is how we define a string. For example, if our alphabet is this two symbols 0 and 1 binary alphabet, then any sequence of 0 1 is the string like 0 1 1 0 1 this is a string then we can have 1 1 1, this is your string.

So, all such example is a string, because this is a sequence of these alphabets. Any sequence of alphabet is called a string ok. String this is a string of length 5; this is a string of length 3 like this. And there is a empty string which is defined to be a as epsilon this symbol, this symbol is called epsilon. So, this is referred to be a empty string.

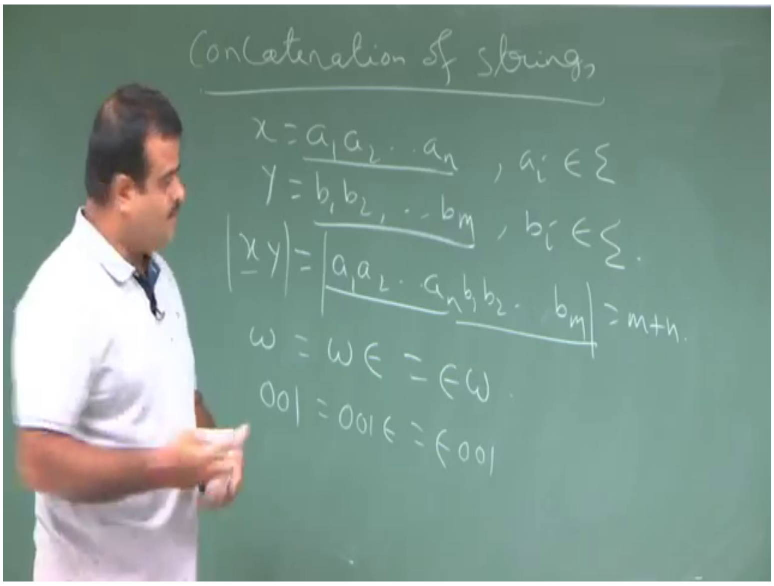
Empty string means zero occurrence of the symbol, zero occurrence of the symbol of input alphabet, so that means zero occur, so that means, length of this empty string is 0, because there is no alphabet. So, zero occurrence of the alphabet is and length of this is length, we define is 5, length of this is 3. So, this is the way we define the length of a string ok. String is also called wond. Now, we talk about the concatenation of two strings. This is a string of length 5; we can have string of length 2.

(Refer Slide Time: 19:19)



So, how many are there with string of length 2? This is 0 0 1 0 0 1 1 1, these are all string of length 2, with the alphabet 0 1. String of length 3, similarly, we can define, but this epsilon is the string of length 0 that means, there is no in the 0 occurrence ok. So, these we need, so we define the concatenation of two strings.

(Refer Slide Time: 19:55)

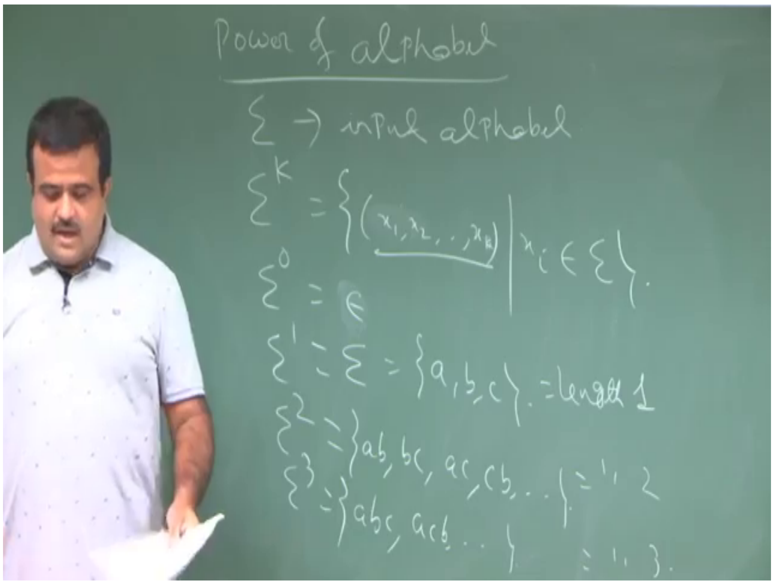


Now, we define how we define the concatenation of strings ok. Suppose, you have two string a 1, a 2, a n. And we have y string beyond b 1 to b n. And here all a i must come from the sigma, because strings is a sequence of the input alphabet. So, they are coming from sigma. Now, what we how we define the concatenation x y, x y is nothing but a 1, a 2, a n this is say a b m.

So, this is a string of length n, this is a string of length m, then if we concatenate it will be a new string of length m plus n. So, this is nothing but b 1, b 2 b m just a occurrence of the sequence. We write first sequence, then we write followed by the second sequence occurrence of the sequence. Now, what is the length of these 2? This length of this is m plus n, length of x plus length of y ok. Now, for any string w, we can write as w is equal to w epsilon is equal to epsilon w, because this is the zero occurrence of a string. This is called a null string or empty string.

So, we can either have 0 and epsilon, because we can always suppose you have a string say 0 0 1 we can always write this as 0 0 1 epsilon, or epsilon 0 0 1 any side of it, it does not matter even though you can put this epsilon in the middle ok, because length of this is same as w, because length of this is the zero occurrence null string. So, this is the length of this is (Refer Time: 22:11) ok.

(Refer Slide Time: 22:18)

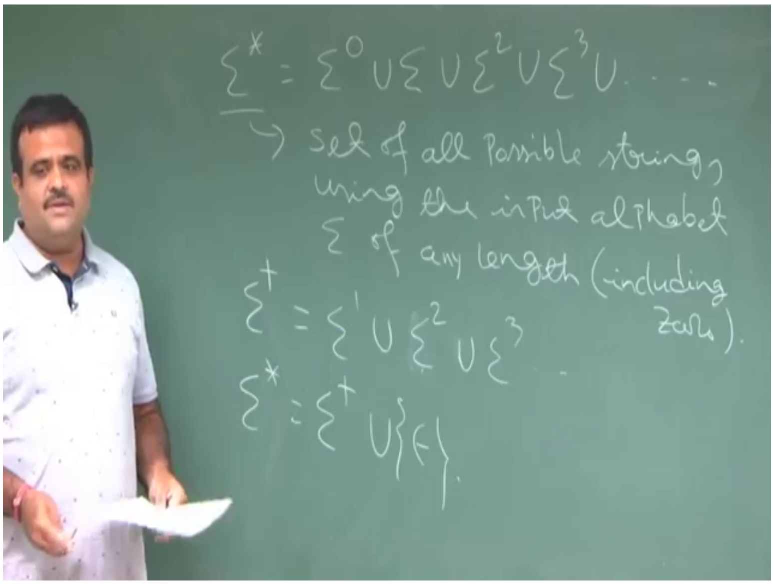


Now, we define power of alphabet, then we define the language. Power of alphabets ok. So, suppose you have a input alphabet sigma. Now, we defined sigma to the power k, it is basically the a, b, c, I mean sorry x 1, x 2 x k, it is all k tuple of alphabet, where x i are coming from sigma. So, it is all the k tuple these are all strings. So, we just forming the string of the alpha, string from the alphabet.

So, for example, if k is equal to 1, so k is equal to 0 means it is the zero occurrence of the string of the symbol. So, it is epsilon; it is epsilon. So, k is equal to 1 is basically the sigma itself. So, if it is 0, 1, or if sigma is say a, b, c it will be just a b c. And k is equal to 2, it is all the string of length 2. So, this is ab, bc then ac, cb like this. So, all the string of length 2 using this same alphabet sigma.

And then we have so this is the all three combination abc, then acb like this, so all three combinations, all the string of length 3 using the same alphabet. So, like this. So, this is of length 1, length 1 string. This is of length 2 string; this is of length 2-3 string ok; this way.

(Refer Slide Time: 25:02)



Now, we define sigma star. Sigma star is basically the string of any length is denoted by sigma star. So, sigma star is sigma 0 which is epsilon null string. Sigma which is the string of length 1 that is the original input alphabet, sigma square string of length 2, then sigma 3 string of length 3 like this. So, all possible string we can made out of this input alphabet, these are all comes under sigma star including the null string including the zero occurrence of the alphabet. So, this is the null input ok. So, this is referred as sigma star.

This is the set of all possible string using the input alphabet sigma, I mean the alphabets are coming from sigma of any length of any length including 0 that means, a null string also part of it including the zero length, that means, the empty string also part of it. And the sigma plus is we are not including the null string, it is basically the sigma 1, sigma 2 like this, sigma 3 like this.

So, this sigma star is nothing but null string. So, this is the sigma star. And the language is any subset of sigma star is called the language. It is a collection of string it could be null string also. It is a collection of string, set of string it is called language. We define the language, it is the collection of string, so that we will discuss in the next class.

Thank you.