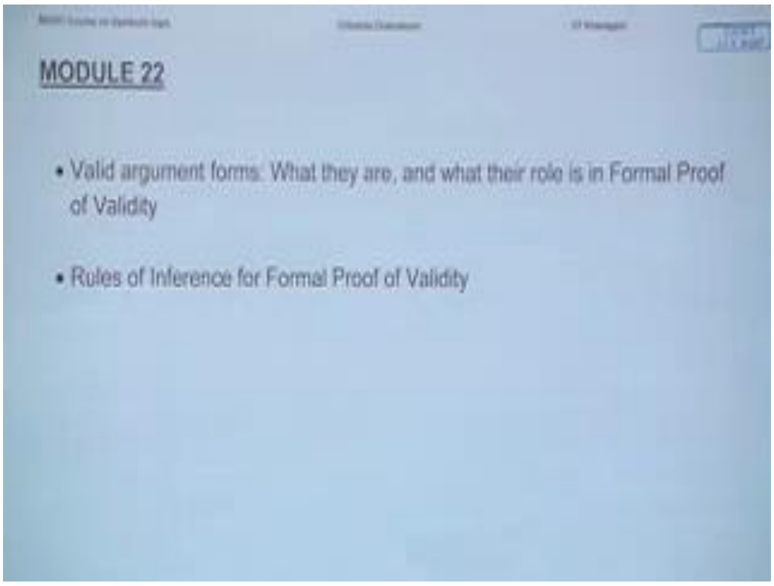
Symbolic Logic
Prof. Chhanda Chakraborti
Department of Humanities and Social Sciences
Indian Institute of Technology, Kharagpur

Lecture- 22
Valid Argument Forms:
What They are, and What their Role is in Formal Proof of Validity
Rules of Inference for Formal Proof of Validity

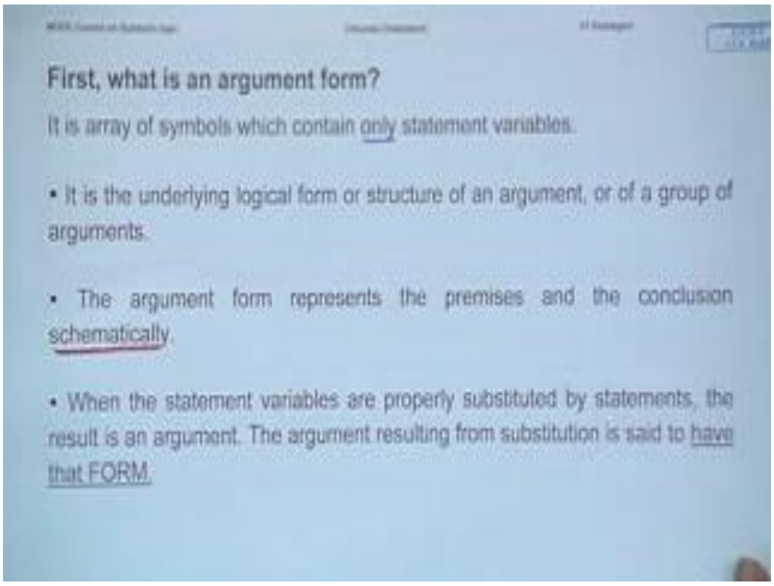
Hello! And welcome back to our session on the valid argument forms and the formal proof of validity. So, this is our module 22, where we are going to now learn how to do this formal proof of validity. And I have introduced you ~~the~~ …I have mentioned about the valid argument forms.

(Refer Slide Time: 00:42)



But this is where we are going to talk about the valid argument forms in detail and to what their functions are, and what the roles are, and then finally, we need to learn about the rules of inference. So, this is where our module 22 is ~~module 22 is.~~

(Refer Slide Time: 01:01)

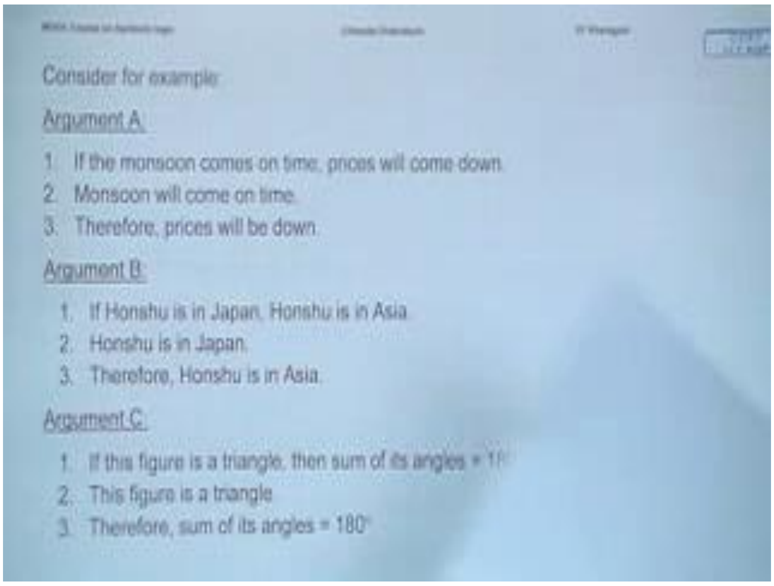


And this is what we are going to do now.

First of all what is this argument form ~~are~~ that we are talking about? Now I have to remind you that I have already talked to you about what is known as statement form. Earlier we have talked about the statement form. And, where we said that the statement form is just a structure, the underlying logical structure. So, similarly, argument form is the underlying logical structure of arguments. So, if you want syntactically what it looks like, then the answer is that it is just an array of symbols which only contain statement variables. So, array made by statement variables, no constants whatsoever. So you are not talking about actual statements at any point. And then as I said that it is the underlying logical structure. Think about the skeleton. If there were something called the skeleton of an argument, the logical skeleton that is, then that is what this argument form is all about. We'll see some examples very soon.

What it does is~~?~~ that it schematically represents the premises and the conclusion, and you get to see the structure of the argument clearly. So, when… as is the case with statement form, in argument form also, if you replace the statement variables with actual statements and properly if you substitute it, what you are going to have in your hand is an actual argument. So, that argument would be the *substitution instance* of the argument form. This is the understanding of formal logic. So, once more the argument forms visually will be an array of statement variables only. And what it represents is the argument's structure, the underlying structure. And schematically, it will represent the premises and the conclusion and any proper substitution of the form will result into an actual argument. So, you can say some arguments have these form, the argument form.

(Refer Slide Time: 03:26)

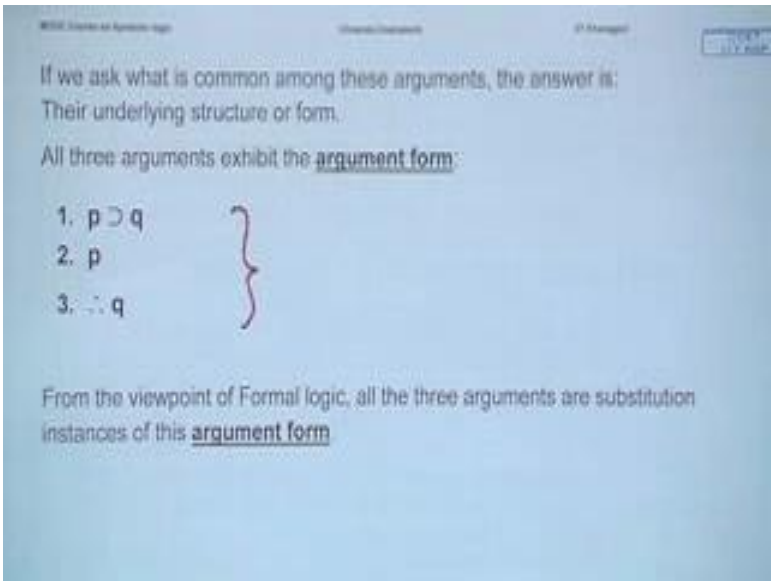


So, let's take a look, actual look into some of the examples. For example, let us consider these three arguments. So, argument A says 'if the monsoon comes on time, prices will be come down. Monsoon will come on time. Therefore, prices will be down'. Argument B says 'if Honshu is in Japan, Honshu is in Asia. Honshu is in Japan. Therefore, Honshu is in Asia'. 'If this figure is a triangle', that is your argument C, 'if this figure is a triangle, sum of its angles is one eighty degrees. This figure is a triangle. Therefore, sum of its angles is one eighty degrees'. If I ask you what is common in that? Because it seems like they are talking about three different things.

So, this one for example, is about weather and the connection to market economy. This is about geography, location of Honshu and so on. This is about geometry. But what is common in this? And, you might be able to see that first of all not only there are three statements in each, two premises and a conclusion. But there is a certain pattern that you can recognize; namely, that the first statement is a conditional statement or an 'if then' statement and the second in every case is the assertion of the antecedent of that conditional. Right? The first clause in that conditional. And then the conclusion is the consequent of that condition.

Now, what is that that you capturing? Your bare eyes can capture that there is some sort of a pattern, though it is not overtly stated. So, that is… that is what we call the underlying logical structure, ok? Which we may represent in terms of statement variables in certain ways. So, keeping that in mind, this is what you are going to gain like this.

(Refer Slide Time: 05:20)

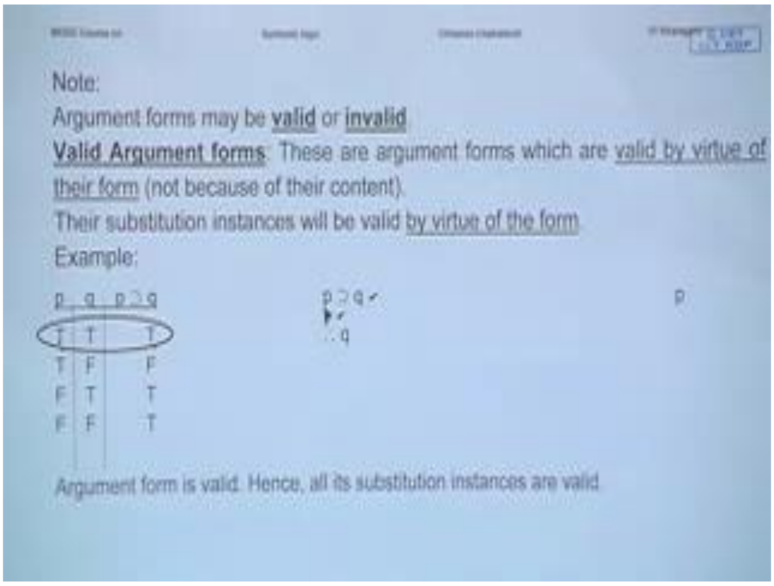


If you try to capture what is that argument form that we just saw, then probably you will pick up something like this. Let us remind ourselves that the p, q, these are statement variables. They are like holds, moulds where we can actually plug in, substitute actual statements. So, these may though look like statements, but they are not. They are propositional variables. Right? So, statement variables. So, together what they represent is the structure that all these three argument exhibit. And that is what we are going to call an argument form. Now, why are we looking into that? The answer will come clearer as we go along. But at the outset please note that formal logic's approach to this would be that what you saw earlier as three arguments are actually substitution instances of this form. Once more, we are not really trying to encapsulate the form from the examples; rather, the form precedes the substitution instances.

So, the form is first. What you do is you replace this with any choice of your statement; actual statement and you are going to get an actual argument which is a substitution instance of this form. So, if you want it speak about Honshu and its location, fine. If you want to talk about geometry, that’s fine. This is the form will, whichever~~,~~ way you want to go, that you will exhibit in the substitution instances.

If you have understood what argument forms are, then let me further this point by saying that argument forms themselves can be valid or invalid. You may remember what validity invalidity is.

(Refer Slide Time: 07:15)

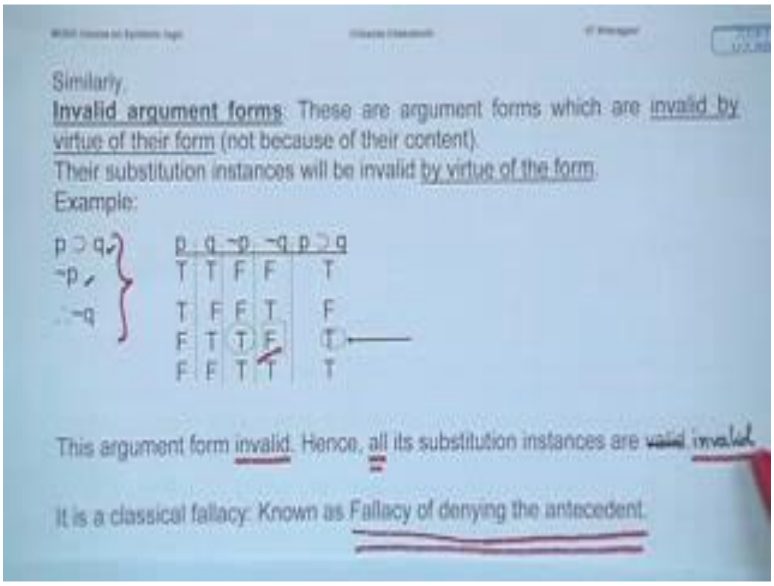


Now, I am taking it to the formal plane to claim that argument forms themselves can be valid or invalid. Valid argument forms would be those argument forms which are valid *by virtue of its form*. So, not because of the content, but there is something structural property is there which makes them valid. This is how formal logic would understand it. And therefore, no matter what you substitute them with, no matter which proposition you substitute ~~and~~ them with, if the substitution is proper, the substitution instance is going to be valid just by having the form. The form will ensure that you have a valid argument in your hand.

For example, let’s take the argument form that we gleaned from the earlier three arguments A, B, C. This was your form. So, this is $p \supset q$, p, therefore, q. If you do a quick truth table on this, then you will see that this is… we have laid down the truth table. Now we see when is… when the premises are all true, what is happening with the conclusion? This is your first premise, this is your second premise. So, we look under $p \supset q$ and we find this is when its true and we look under p and this is when it’s true. So, together when they are true, please see that q is then true. In other cases, you have one of them false here, here and here. So, do not be worry about that. The only time we need to worry is when the premises are all true, what is happening under the conclusion? And the answer is then conclusion got to be true.

So, this is a valid argument form. If this is a valid argument form, please note, what you have gained is that just by calling or proving the argument form as valid what we have gained is a say over the entire class of arguments that would be the substitution instance of this form. So, all the three arguments that you saw, remember A B C and so on? They are… you don't have to do separate truth tables for any of them anymore. All three of them now you can call them valid. How? *By virtue of the form* that they represent. This is our answer. So, you don't have to do separate truth table for each case to establish validity. You can now refer to the very fact that they have a certain form and if you know that the form, argument form is valid, then by virtue of the form you can invoke that they must be valid also.

(Refer Slide Time: 10:29)



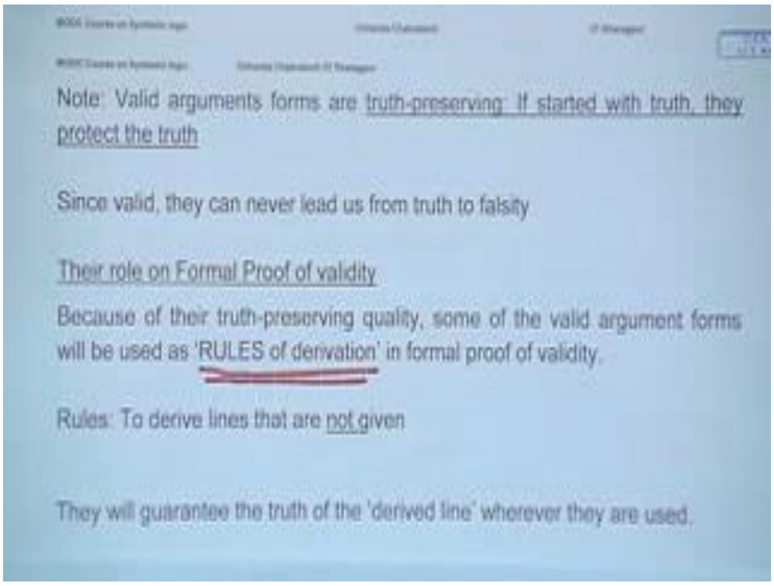
So, this is valid argument form and similarly you should know that there can be *invalid* argument forms also. Argument forms which are invalid because again of their, because of their form. So, certain property, structural property is there in them which makes them invalid and we will try to show that also. But the point to note is that if you found such invalid argument forms, all their substitution instances would be also invalid *by* *virtue of having that form*. That is the classic point. So, just by having the same form they would be automatically invalid. It doesn't matter what their content is, what you want to substitute them with, doesn't matter. The property of invalidity will stick to them by virtue of having this form and that's what we have to understand from the formal logic point of view.

So, let's take a look at what would be an example of an invalid argument form. For example, see this one. This may look like similar to what we have earlier said. But suppose there is a form which says that the premise form is like this. If p then q, and then not-p (~p). You are saying if p then q, and then you are are also see saying not-p, and from that you concluding therefore, not-q. This is an argument form. Let's see whether it's valid or invalid. So, we can lay out the truth table like so. Remember we are separating the ~p, ~q, because they are all appearing in the argument form. So, we place them like this. Again what are we looking for? Is when the premises are all true, what is happening with the conclusion? So, this is your first premise and this is your second premise. So, we look under each of them. Here is p ⊃ q true and here is not-p. One of them is false. So, we do not need to see that row. This is where it's already false. This is also false. We don't need to see this row. But this row has p ⊃ q as true and not-p also as true. This is when you have all the premises true at the same time. What is happening in the conclusion here? And you find that it is false. Right? So, when the premises are true, it is possible for the conclusion to be false. that makes the argument form invalid. Ok? So, this is an invalid argument form. And what you have claimed, therefore, is that every substitution instance, even the ones that you are not aware of, that are not in front of your eyes, that you have not yet thought of; every single substitution instance of this form is going to be invalid. That's the point. And just because you are meeting this, there are many such argument forms by the way which are invalid. And this one, the one that you just saw is a classical invalid argument form. It is called the *fallacy of denying the antecedent*. The fallacy of denying the antecedent.

Fallacy means they are logical errors, logical mistakes. But people keep on doing them. So, one needs to identify them and the medieval logicians have done an excellent work of classifying many such fallacies. This is one of them, which is known and routinely people do this, but it's nonetheless a mistake, an error and it's called fallacy of denying the antecedent. But our point sticks ~~to~~ that this is an example of an invalid argument form and if you encounter this in any argument, you can safely call that argument invalid. Even without… just by referring to the argument form you can call it invalid.

Now, why are we discussing this argument form, valid invalid etcetera? The answer is here is that you need to understand that valid argument forms have a very important property for formal validity. What is that property? The property is that valid argument forms are *truth-preserving*. Truth-preserving. They have a quality that they preserve truth. As in, if you start with truth,

(Refer Slide Time: 14:44)

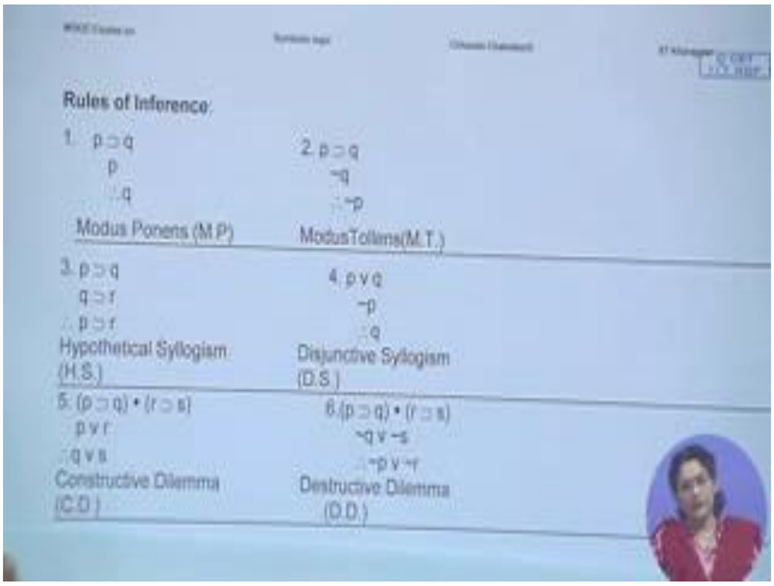


and if you use this valid argument forms, they will protect the truth. They will preserve it. So, there is no way that you can start with truth and use the valid argument forms and still land into falsity. Get it? So, ~~if you~~ remember in formal proof of validity you have step by step progress. So, if you start with a step that is true and then apply valid argument forms to that line, there is no way you can land into falsity. Now there is no way you can derive falsity out of that. So, that is what they do. Therefore, now you know that the role of valid argument forms in formal proof of validity is important.

What they serve as: Some of them are going to be used as the *rules of derivation*. Remember when I defined the formal proof of validity we said that this would be sequential process of deriving lines from previous line according to certain rules. And I did not explain what those rules are. ~~The~~ Time has come now to talk about it. But understand that these rules are nothing~~,~~ but what we would call valid argument forms.

So, some valid argument forms would be identified as rules of derivation to serve this purpose. These rules what they do is to allow us to derive new lines in a proof that are not given. Some lines are given and then, you need to derive new line. What would protect the truth of those new lines? The answer is these rules of derivation or valid argument forms will guarantee the truth of the derived line, wherever they are used. This is our introduction to your… whole mechanism of the proof.

(Refer Slide Time: 16:39)



Let us now look into something very important. So, we are going to now talk about what is known as the rules of inference. That you are going to use in the formal proof of validity. These are all valid argument forms. So, let us get acquainted with it. This is what we have seen earlier : If p then q; and p, and therefore, q. Once more, so, you have if p then q, and p. So, there is an assertion. These are 2 separate lines and you need each of these lines before you can pull the consequent out. Just from p alone you cannot pull q out. Just as if~~,~~ you have only if p then~~,~~ q, you cannot pull q out. So, you need two lines to pull this q out. This is what we encountered earlier as the common structure argument form in all the three arguments that we saw. There is a name for it and we are going to call it *Modus Ponens*. That is a medieval name. We will shorten it and call it the MP rule. MP stands for *modus ponens*. If you want to know then this is *by affirming we affirm*. That's Latin. So, what you are affirming is that if p then q. You are further affirming that p has happened. Therefore, q has happened.

Let's come to its corollary. This is the second rule which says that you have to have the premise if p then q, and not-q. So, together from these two lines, you can say therefore, p is not the case. If you are stumped by this, then let me remind you that q is the necessary condition, right? So, if p and if it is if it is true that if p then q, and if it is also true that q has not happened, then you can safely infer p also has not happened. And that is a valid argument form. It is called *Modus Tollens*. Shortened form is MT and we are going to refer to it as MT.

Few more. This line number… this rule number 3 is an ancient rule. Starting from the Greek times we have had it. It is colloquially called the chain rule, but the full name is *Hypothetical Syllogism*. *Hypothetical* means 'if-then' and *syllogism* is an Aristotelian term. So, the name is hypothetical syllogism. We are going to use the abbreviation HS. What does it say? ~~is it said?~~ That if you have two premises like this, 'if p then q', 'if q then r'. See the chain? Then, if p then r. Please note, that you are not saying p has happened, you are not also saying r has happened. The entire argument form places into a hypothetical or conditional mode. That's the name. If p then q, if q then r. If that's the case then, if p then r. Nothing more is being stated. But that helps us to get rid of the q. Ok? If you, if you need it so in a proof, you will see that the rule has certain advantage.

This one is simply saying, this is called *Disjunctive Syllogism*. Because it uses the wedge, the disjunction. And in short its called DS. What am I doing? The first premise is p wedge q, and then says not-p, alright? So, it follows therefore, q.

Now I have to remind you, or sort of emphasize on this that this rule of inference is very rather formal and rigid. So, when you have such a situation, please note the sequence. What the rule allows is that you have a wedge and then you have the negation of the first disjunct. The position of the disjunct also matters. If you have so, the first disjunct is negated, then you can pull the second disjunct out. Ok? What you cannot do is to change or swap the situation just by this rule. You need a different rule to do that and we'll see that in the rule of equivalence. But right now please note the disjunctive syllogism only allows you in this sequence.

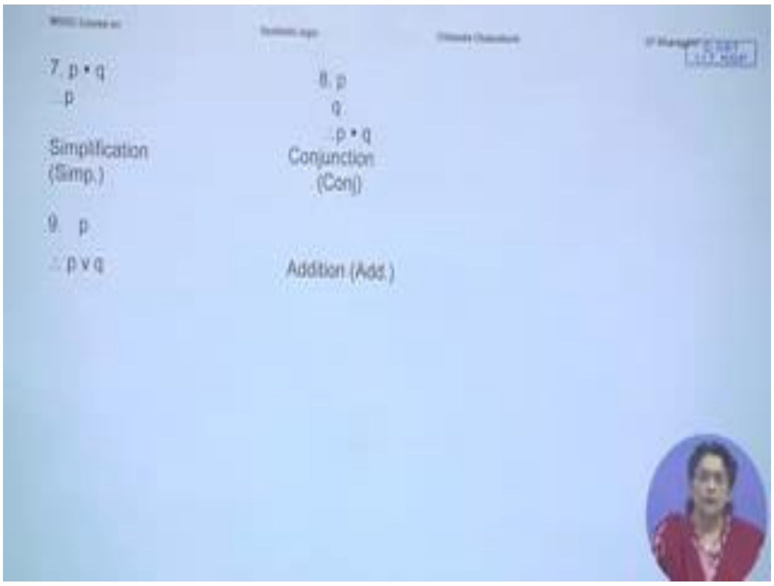
Few more to go along with. This is called *Constructive Dilemma*. Let's take a note of what is what is it saying. We are going to call it the CD abbreviated. If you have a premise that says if p then q, *and* if r then s. So, it's a conjunction of two horseshoe statements. On top of that if you have either p or r, notice the position of p and r. p is the antecedent of this one horseshoe, and r is the antecedent of the second horseshoe statement. So, if you have together this sentence, and p wedge r, then you can derive q or s, or q wedge s. Notice the position of q and s. q is the consequent of the first conditional, s is the consequent of the second conditional. If you have a situation like this along with these premises, together you can derive q wedge s. And this is construct dilemma.

Its corollary is the *Destructive Dilemma* which we are going to refer to as DD. What does it say? It's out of a combination of this earlier rule and the *Modus Tollens*. If you have a conjunction of if p then q, if r then s, you also have not-q wedge not-s.

So, you have a negation of the consequent of the each of these conditionals as a disjunction. Then you can pull out not-p or not-r. Get it? So, if you have the negation of the consequent, then you can derive along with, the both the premises will be necessary , then you can derive either not-p or not-r and that is your *Destructive Dilemma* rule.

These six rules are all valid argument forms and they are also rules of inference. There is more we are going to have total nine rules of inference to start with. This is simple *Simplification* as you can see,

(Refer Slide Time: 23:07)



if you are told p dot q is true, you can infer p must be true. Remember the dot truth table, you will see why this has to be valid always. The rule is called *Simplification*, but we are going to refer to it as Simp. Again I have to remind you or insist upon the fact that look at the position. What does it allow? It says that if you have a conjunction you can chop the second conjunct out, and pull the first conjunct out. What you cannot do is to change the position of p dot q. You need a separate rule for that. Simplification does not allow you to simply change the position of p dot q. Alright? So, it is exactly, follow it exactly that if you what it allows you is to if you have a conjunction statement, you can simply take the first conjunct out by simplification rule.

This is *Conjunction* rule. If you have p as true and q as true, then this rule allows you to join them together. And that is the job of conjunction and it is called conjunction which we will refer to as Conj.

This is a different and should be institutively obvious to you. If it is known that p is true, then this rule called *Addition,* which we are going to refer to as Add., allows you to add a statement with disjunction. So, if p is known to be true, therefore p wedge q, where q can be anything that you need or anything that you want to add. Why is that true? Because we know that with disjunction if one of the disjuncts is true, the disjunction must be true. The whole disjunction must be true. So, this is what it is. We are going to use these rules in our next modules to get into the proof properly. But it is very important at this stage that you go through the rules of inference by yourself. I have tried to explain and we'll see how they apply, but it will take some time. But you need to give yourself that time to get the understanding properly. We are going to refer back again and again to the rules.

But in this module, this is where I will stop. And in the next module show you how to utilize these rules. But, before that gain knowledge of the rules. The more confident you are about the rules, the better you will be performing in the proof. So, that will be our end of our module number 22.

Thank you very much.