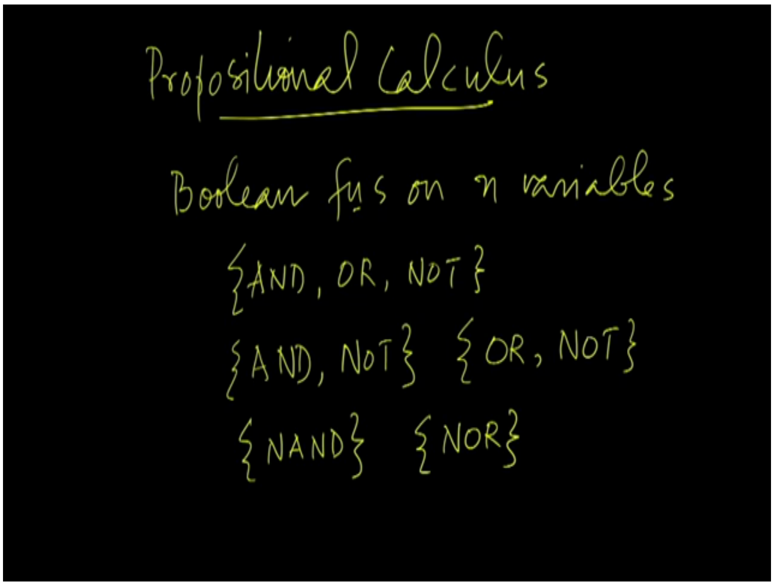
Discrete Mathematics
Professor Sajith Gopalan, Professor Benny George
Department of Computer Science & Engineering
Indian Institute of Technology, Guwahati
Lecture 2
Propositional Calculus

Welcome to the MOOC on discrete mathematics, this is the second lecture on mathematical logic. We will continue with our discussion of propositional calculus that we started in the last class.

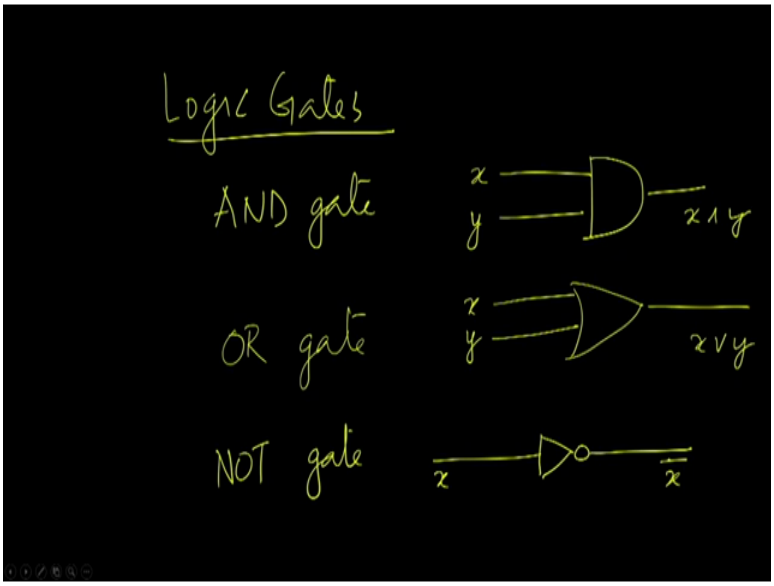
(Refer Slide Time: 00:42)



In the last class, we talked about the Boolean functions on n variables and we saw that such functions can be synthesized using AND, OR, NOT gates, these are logical connectives using these logical connectives we can synthesize any Boolean function on n variables. Therefore, this set of logical connectives is called a complete set of connectives. We found that, this is by no means the only complete set of connectives.

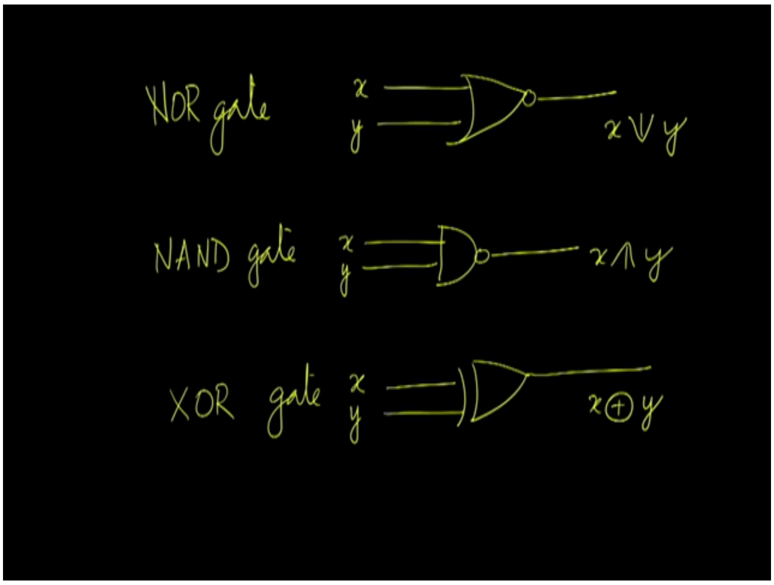
We find that AND and NOT also form a complete set of connectives. Similarly, OR and NOT also form a complete set of connectives, NAND is also a complete set of connectives on its own, NOR also forms a complete set of connectives on its own. So, a Boolean function on n variables can be synthesized using any of these sets.

(Refer Slide Time: 02:02)



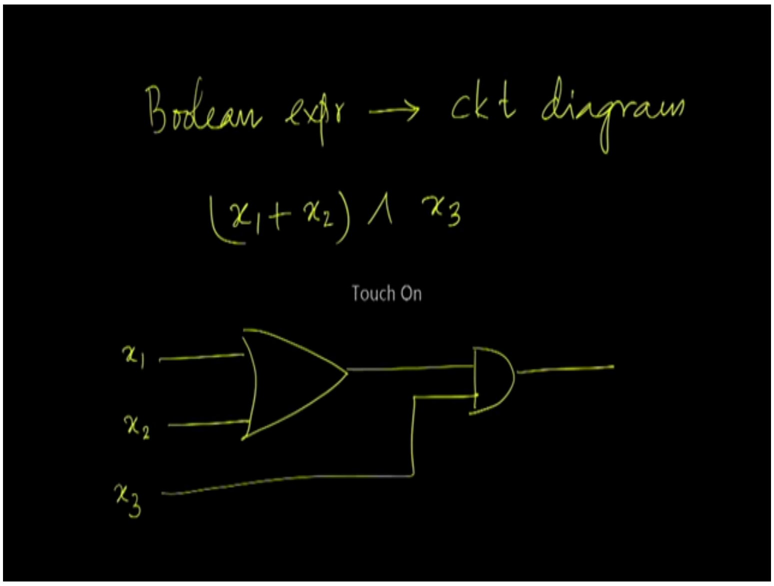
Some of these logical connectives can be expressed as logic gates and, AND gate is drawn like this in a circuit diagram. So, an AND gate has 2 inputs x and y and the output is x and y, the output is 1 if and only if both the inputs are 1. An OR gate is represented in a diagram in this manner where the inputs are x and y and NOT gate has only 1 input, let us call it x and its output is the negation of x.

(Refer Slide Time: 03:01)



A NOR gate, which is a negation of OR is drawn in this manner, x and y are the inputs the output is x NOR y and NAND gate, is an AND gate followed by a negation, an XOR gate, which produces an output of 1 if and only if the inputs are not the same, it is denote it like this. Therefore, using these logical gates you can convert a Boolean expression into a circuit.

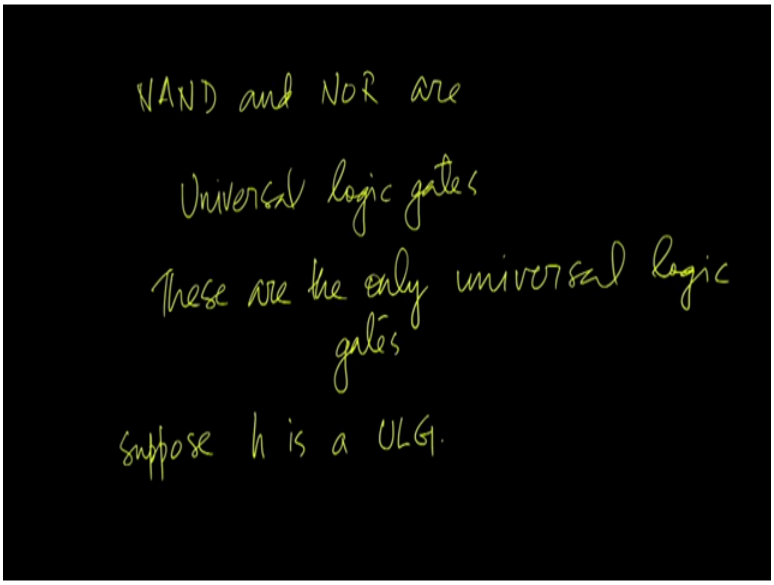
(Refer Slide Time: 04:03)



For example, if you have a Boolean expression of this form, we have the signals x 1 and x 2 they can be odd using an OR gate, the output of which can be ANDed with x 3. So, this is the circuit diagram corresponding to this Boolean expression. So, what we know is that any Boolean expression can be converted into a circuit diagram using AND, OR, NOT gates, using either the sum of products form or the product of sums form.

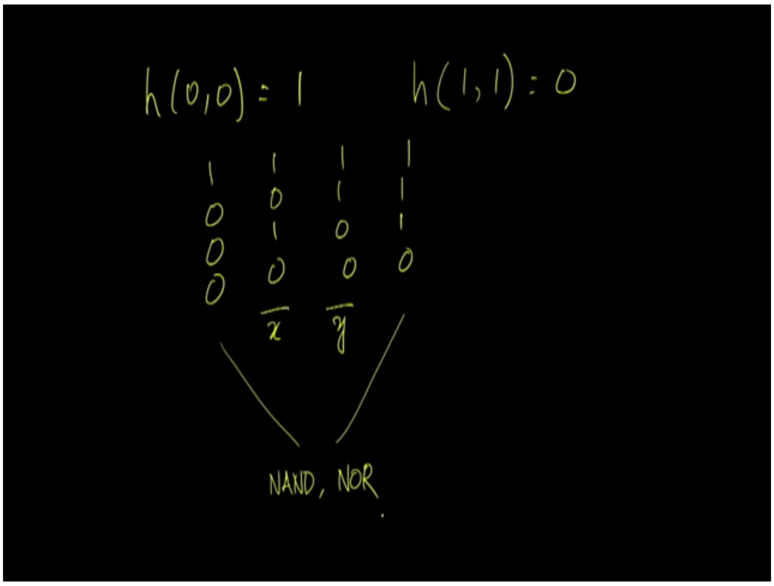
You can have an equivalent circuit diagram created using only NAND gates too or NOR gates.

(Refer Slide Time: 05:03)



NAND and NOR are universal logic gates because every circuit can be synthesized using only 1 of these and these are the only universal logic gates, this we proved using the following argument. Suppose, h is a universal logic gate, a Boolean function h with 2 variables.

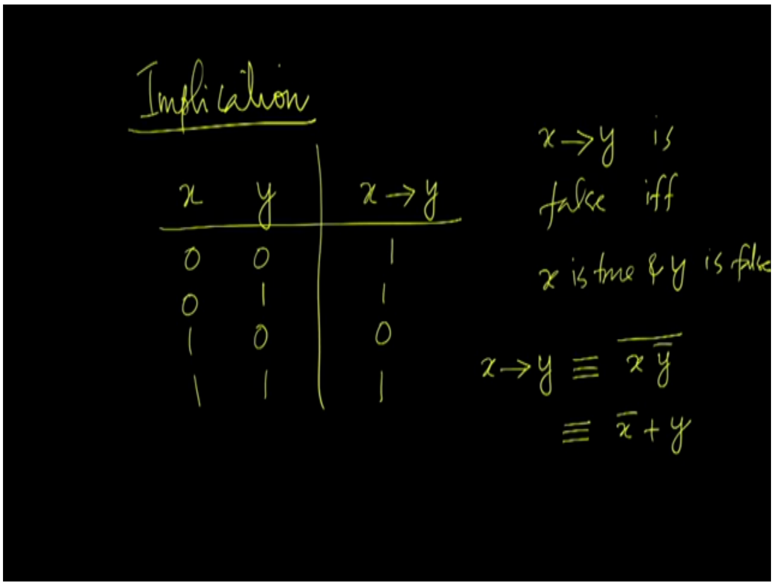
(Refer Slide Time: 06:09)



If this is a universal logic gate then when both the inputs are 0, the output of h will have to be 1 otherwise we will not be able to use h to synthesize a function like NAND that is because when you have a circuit made up only of h gates and when both the inputs to the circuit are 0 then the circuit will have only 0s in the inside, if h of 0, 0 were 0. Therefore, to be able to synthesize every function which produces 1 when both the inputs are 0 h of 0, 0 will have to be 1.

Similarly, h of 1, 1 will have to be 0. Therefore, we have only four possible functions of these 2 are the negations of the inputs. Therefore, the 2 remaining functions are NAND and NOR, these are the only universal logic gates.

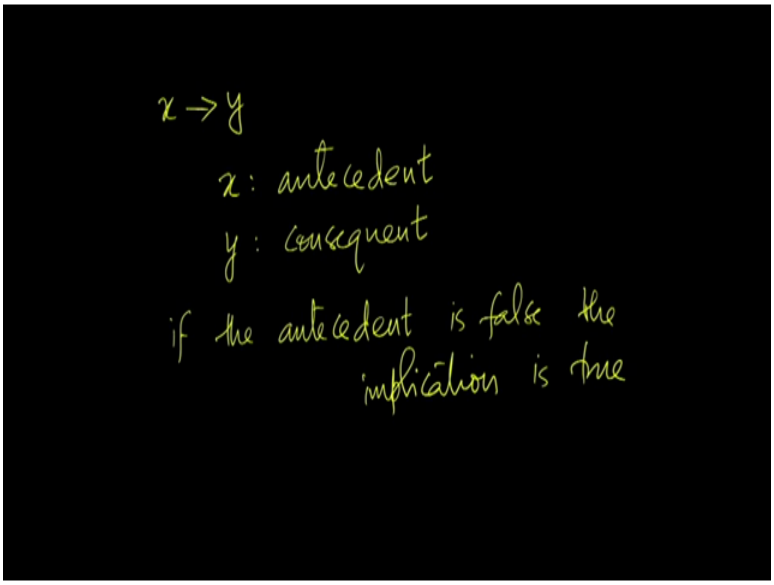
(Refer Slide Time: 07:17)

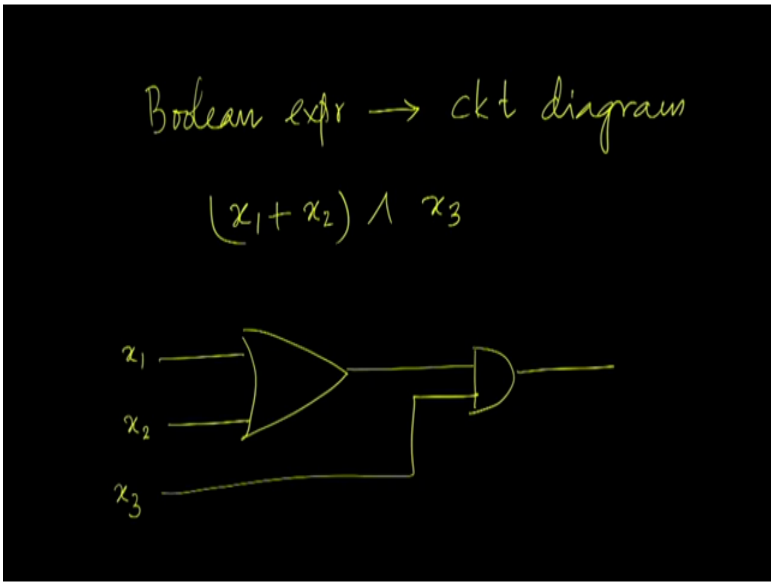


Now, let us consider implication. The implication is a 2 variable Boolean function with this truth table, when both the inputs are 0 the output is 1, when x is 0 and y is 1 the output is 1, when x is 1 and y is 0 the output is 0 and when both the inputs are 1 the output is 1. So, what it means is that, x implies y is false if and only if x is true and y is false or we can say, x implies y is logically equivalent to the complement of x y bar which by De Morgan's law is x bar plus y.

So, we find that x implies y is true when x is false or y is true.

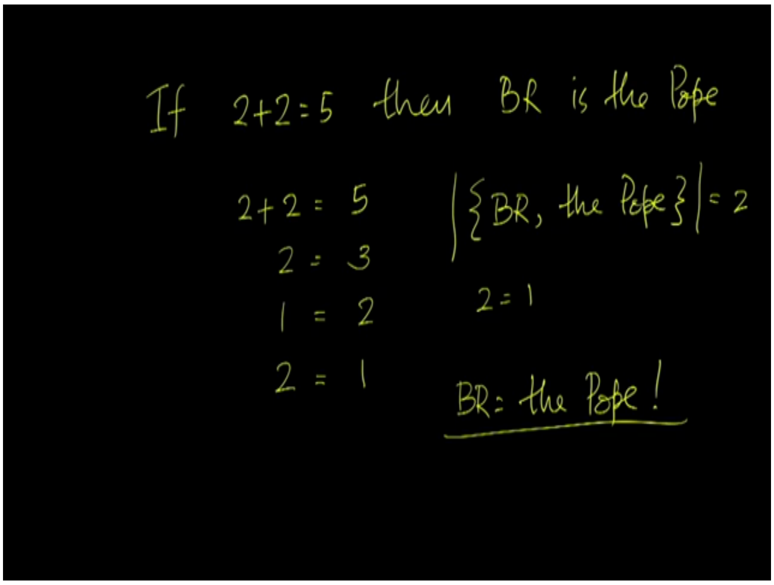
(Refer Slide Time: 08:33)





In the case of an implication x is called the antecedent and y is called the consequent, so what we have seenI is that, if the antecedent is false the implication is true, on the other hand if the antecedent is true then the implication is true only if the consequent is true, this has interesting consequences, we can derive essentially any statement from an antecedent which is wrong.

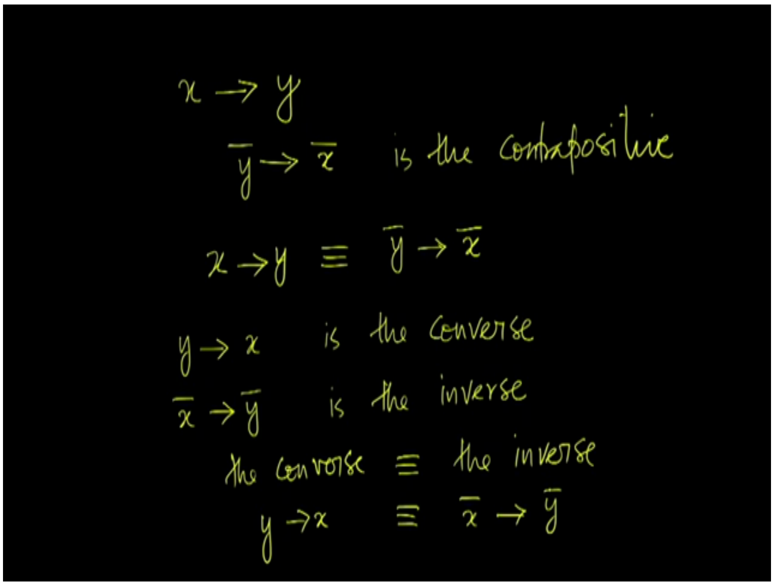
(Refer Slide Time: 09:36)



The legend has it that Bertrand Russell famously showed that, if 2 plus 2 equal to 5 then Bertrand Russell is the pope. His argument was this, if 2 plus 2 equal to 5 then we can subtract 2 from both sides then we have 2 equal to 3, if we subtract a further 1 from both sides we have 1 equal to 2. Therefore, we have 2 equal to 1 then consider the set containing Bertrand Russell and the pope, this set has 2 people namely Bertrand Russell and the pope.

Therefore, the cardinality of the set is 2 but then we know that 2 equal to 1 therefore the cardinality of this set is 1 but if the cardinality of a set is 1 that is it, this is a singleton set and it has only 1 member, therefore, Bertrand Russell is the pope. In other words, if the antecedent is false then practically you can prove anything as the consequent. Therefore, we can judge an implication only when the antecedent happens to be true.

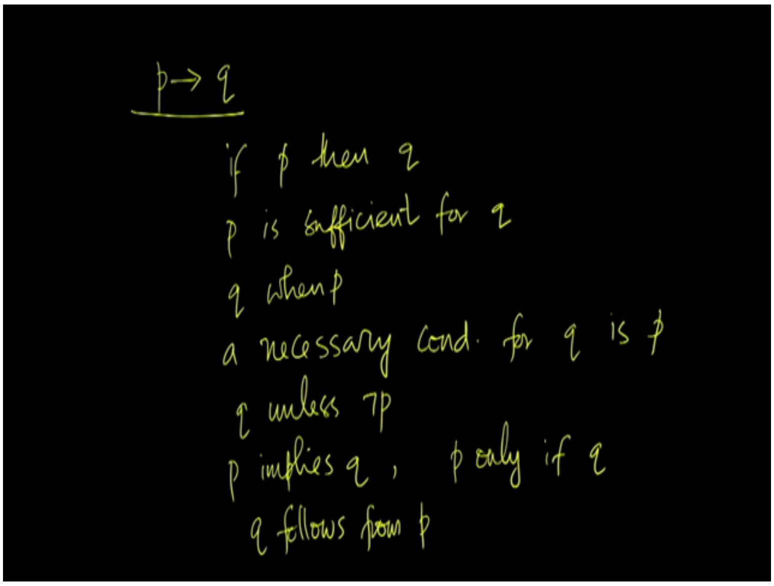
(Refer Slide Time: 11:02)



Consider the implication, x implies y, the implication (y implies) y bar implies x bar is called the contrapositive of x implies y. Using a truth table you can readily verify that, x implies y is equivalent to y bar implies x bar. Every logical every implication is logically equivalent to its contrapositive, y implies x is the converse of x implies y and x bar implies y bar is the inverse of x implies y.

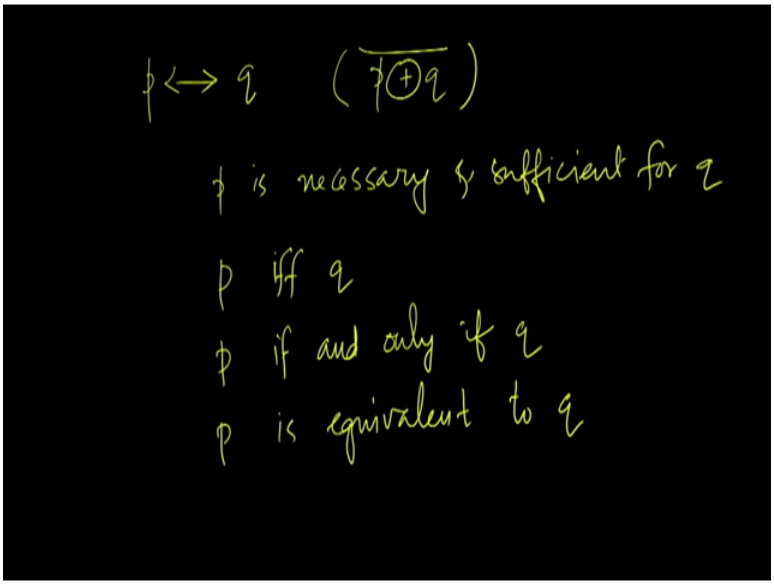
So, by what we have seen just now, the converse and the inverse are logically equivalent namely, y implies x is logically equivalent to x bar implies y bar.

(Refer Slide Time: 12:21)



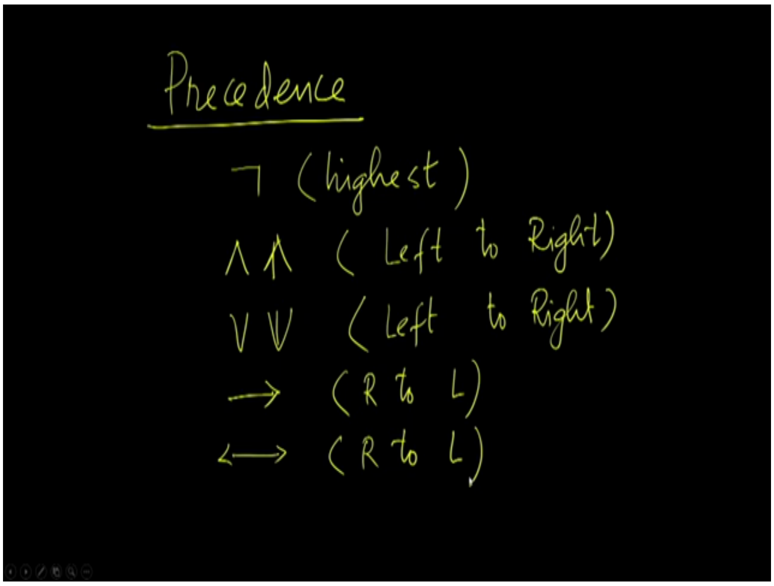
The implication p implies q will be written in English, as if p then q, p is sufficient for q, q when p, a necessary condition for q is p, q unless not of p, p implies q, p only if q, q follows from p. So, these are all essentially the same thing, they all denote the implication p to q.

(Refer Slide Time: 13:32)



The logical connective p is equivalent to q which is the negation of the exclusive OR, is usually written in English as p is necessary and sufficient for q, p if and only if q or p is equivalent to q, equivalence is true if and only if p and q have the same logical value either both are true or both are false.

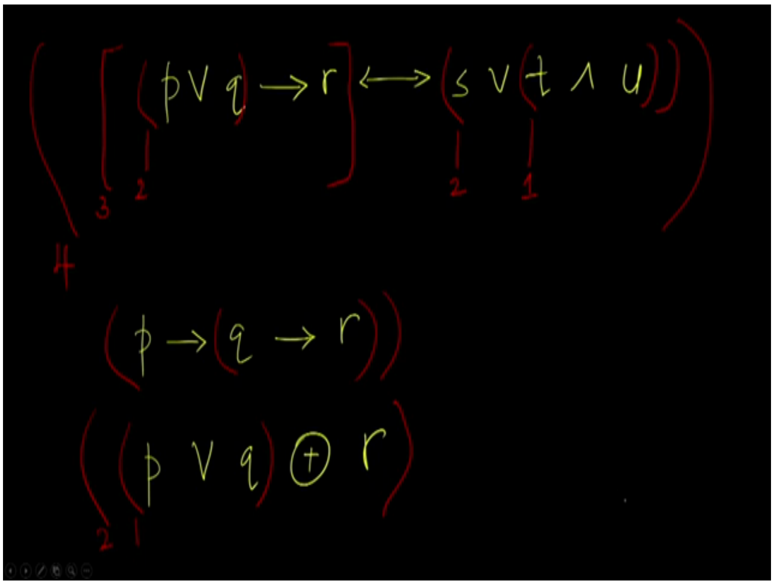
(Refer Slide Time: 14:28)



So, we have seen several logical connectives now. In an expression that uses many of these logical connectives, how would you parenthesize the expression if the parentheses are not already placed in it? For this, we have to use the precedence's rules. The precedence's rule, commonly used are these, negation has the highest precedence which means you have to associate the negation symbol to the nearest variable first AND and NAND have the next preference, they associate from left to right. OR and NOR have the next precedence, again they are from left to right.

Single implication comes next but this associates R to left, right to left and double implication has the least preference which again has right to left association.

(Refer Slide Time: 15:37)

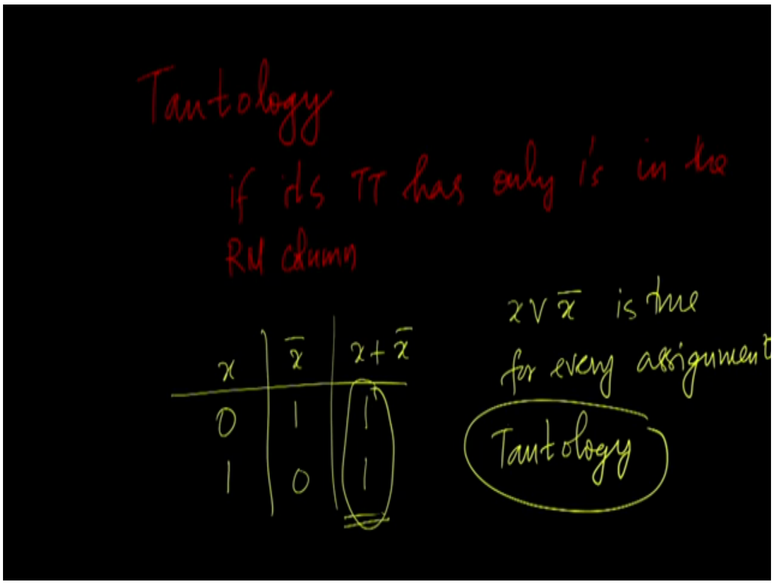


So, with these precedences in mind, let us work out an example. Let us see how parentheses can be inserted in this expression. In an expression is given like this, first you should insert parentheses here because AND has the highest precedence, after that the precedence is for OR therefore you should parenthesize them in this manner. So, this is the first parentheses, this is the second one, this is also at the second level then at the third level we have implication.

So, this is the third parenthesization and then the double implication, the 2-way implication has the least precedence, so this is the last one. So, this is how you would parenthesize an expression containing many of these connectives. Let us consider another expression, p implies q implies r, when we discuss precedences we said that implications have a right to left associativity, therefore you will have to associate q to r first and then an outer parentheses.

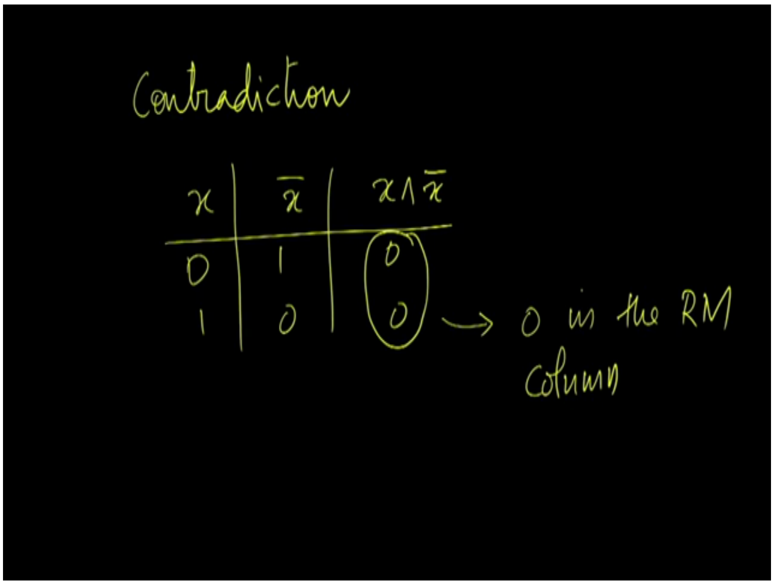
Similarly, when I have an expression of this form p or q exclusive OR r, we find that the precedences for this therefore this is where you put the parentheses first and then the whole of the expression is put in a parentheses of a lower precedence. So, this is how you would parenthesize Boolean expressions.

(Refer Slide Time: 17:43)



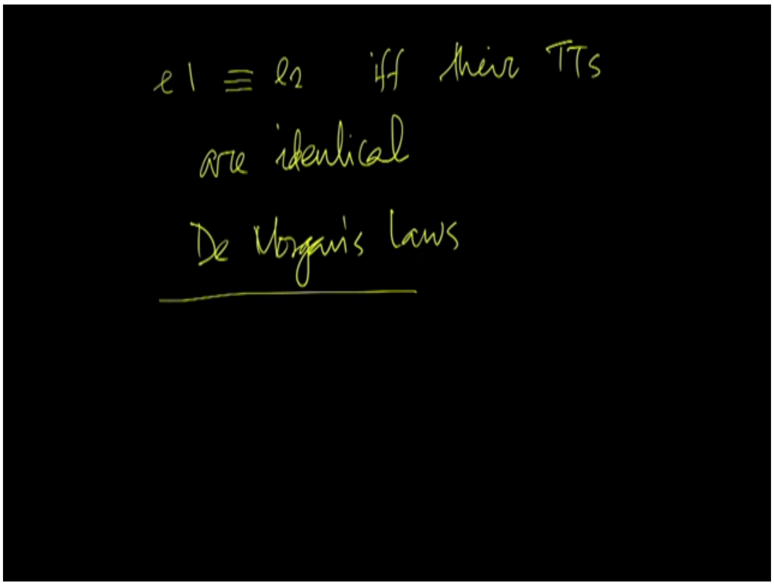
An expression is called a tautology, if its truth table has only 1s in the rightmost column. For example, let us consider x OR x NOT, this evaluates to 1 for every assignment to x which means x OR x bar is true for every imaginable assignment. If you drop the truth table of it in the rightmost column, we have only 1s therefore this is a tautology. A tautology, is a statement which is always true, whatever be the truth values that you assigned to the variables of the formula. The formula will always evaluate to true.

(Refer Slide Time: 19:08)



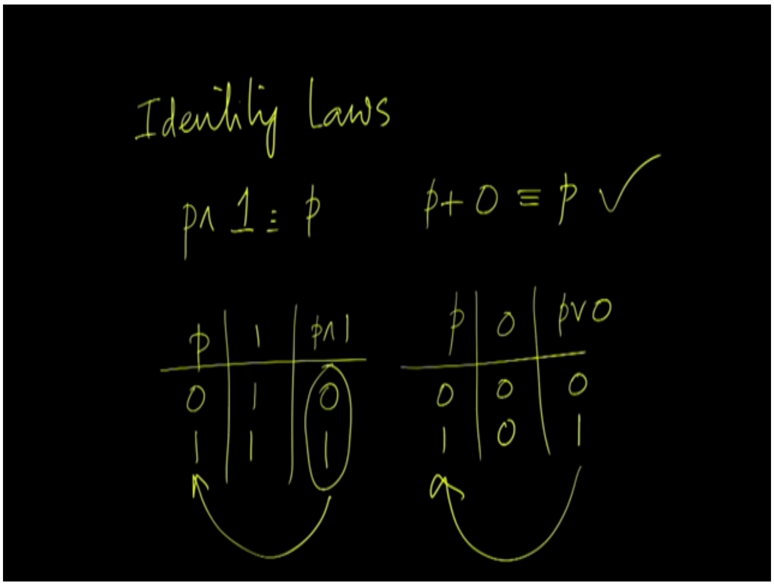
The complement of a tautology is called a contradiction. For example, if you have x AND x bar, you find that it evaluates to true for every possible assignment, it has only 0s in the rightmost column such a Boolean expression is called a contradiction, a statement which is always false.

(Refer Slide Time: 19:47)



In the last class, we saw the 2 expressions e 1 and e 2 are logically equivalent if and only if their truth tables are identical, in particular we saw De Morgan's laws, let us see several more equivalences which will be useful for us to prove various statements.

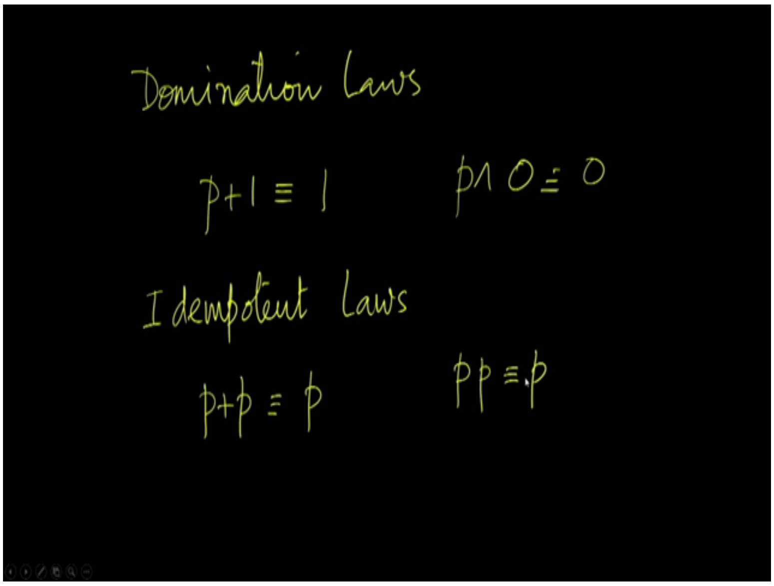
(Refer Slide Time: 20:18)



One is the identity laws, the identity laws states that p AND 1 equal to p and p OR 0 is equivalent to p, you can verify this using the truth table. For example, consider variable p and the logical function 1, p can take on values 0 and 1 and for any assignment to p, 1 takes on values 1 and 1 therefore p AND 1 here would be 0 and 1, this seems to be identical to the first column which is p, so p AND 1 is the same as p.

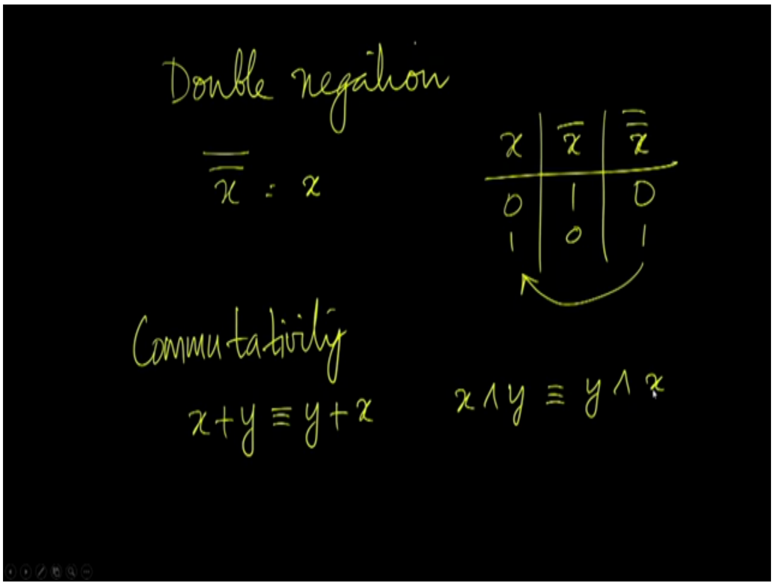
Similarly, p OR 0, you find as identical to p again, proving the identity laws. So, when you take the AND of p with 1 you get p itself, AND of anything with 0 will give you the same thing.

(Refer Slide Time: 21:38)



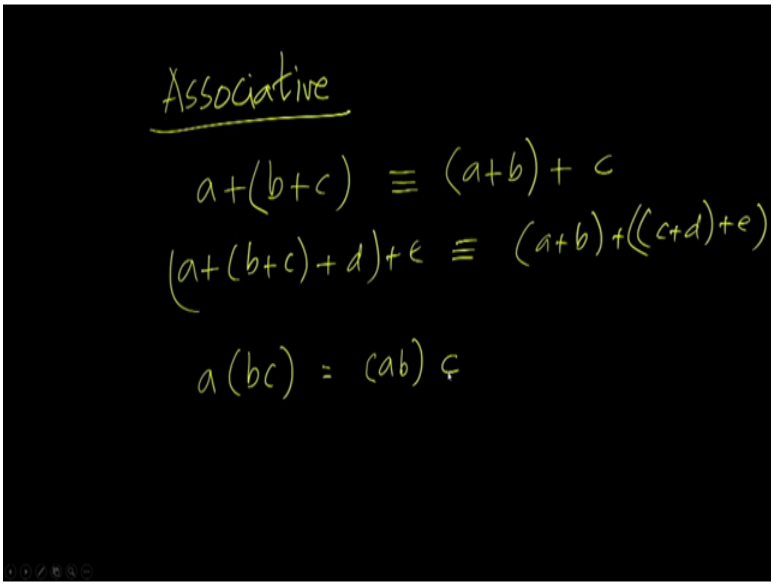
The next are the domination laws, domination laws say that p plus 1 is equivalent to 1 anything OR with 1 gives us 1 that is 1 is a certainty so something OR a certainty is always a certainty. Similarly, p AND 0 is 0, you can verify these using the truth tables. Then idempotent laws, idempotent laws says that p OR p is p and p AND p is also p.

(Refer Slide Time: 22:31)



Double negation, double negation asserts that the negation of a negation is the same as x again you can verify this using a truth table. The negation of 1 is 0, the negation of 0 is 1 so this is identical to x, establishing the double negation law. Then the law of commutativity, which says that AND and OR are commutative operations, namely x OR y is the same as y OR x, similarly x AND y is the same as y AND x.

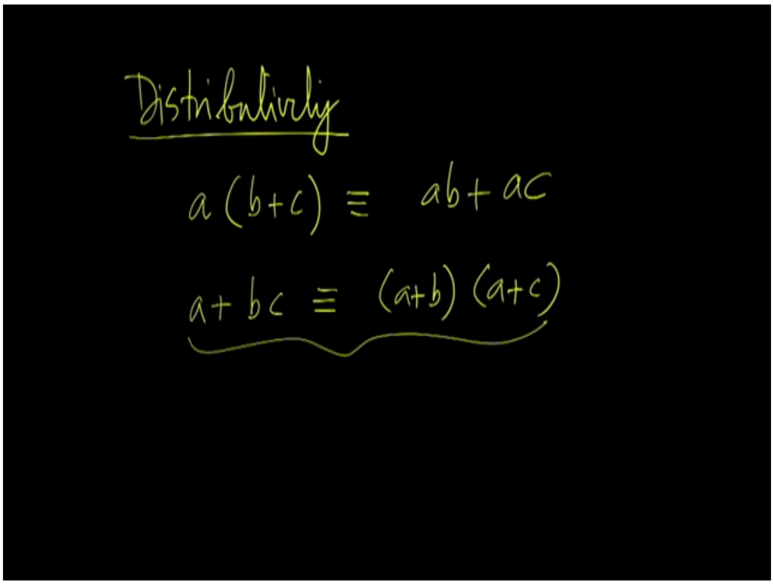
(Refer Slide Time: 23:24)



Then we have the (associate) associative law, the associative law says that the OR of a with the OR of b and c is equivalent to the OR of a OR b with the OR of c. So, by extending this we can take the OR of a chain of formally in whichever order is convenient. For example, this is the same as, so you can parenthesize in whichever way when you have a long sequence of disjunctions.

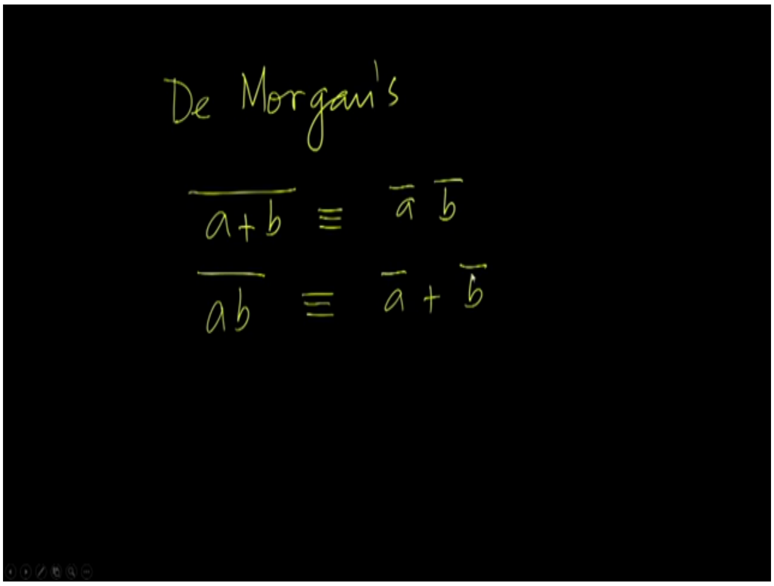
So, you can verify this using the truth table readily. Similarly, for the AND operation a AND b AND c is the same as a AND b AND c, again you can verify this using the truth table.

(Refer Slide Time: 24:38)



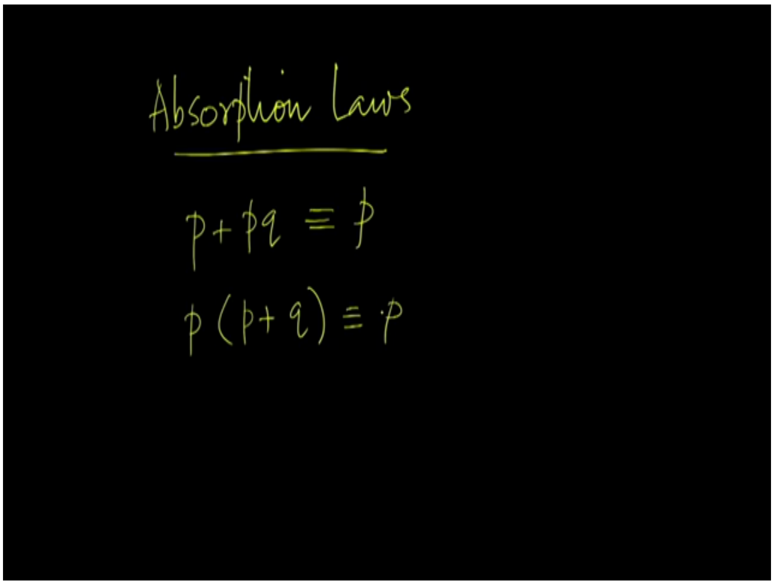
Now, comes the distributivity, the law of distributivity in this form is familiar to all of you from arithmetics, that is when we compare OR with addition and AND with multiplication, this is a familiar form of distributivity which you can easily verify using the truth table. But there is another distributivity law which does not have an equivalent in the algebra of numbers a OR b AND c is a OR b AND a OR c, which means, OR distributes over AND, the way addition does not distribute over multiplication in arithmetics.

(Refer Slide Time: 25:50)



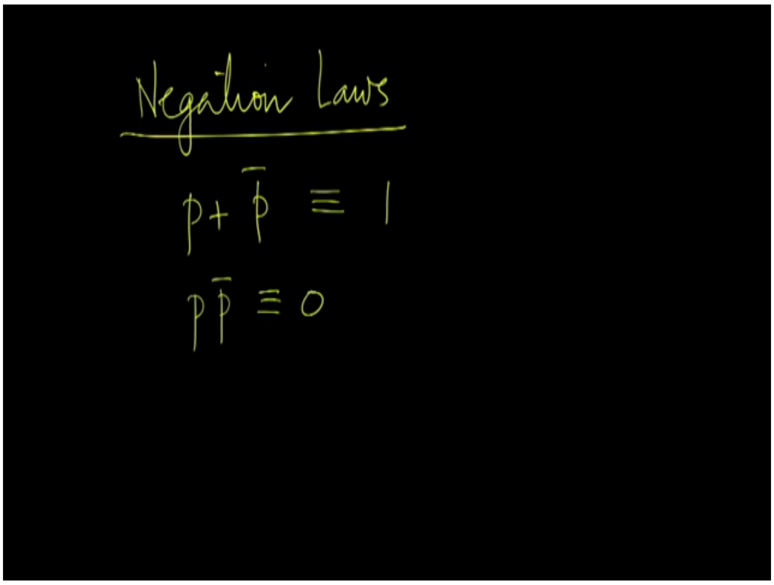
Then of course, we have the familiar De Morgan's laws which says that the complement of a OR b is the complement of a AND the complement of b. similarly, the complement of a b is the complement of a OR the complement of b.

(Refer Slide Time: 26:12)



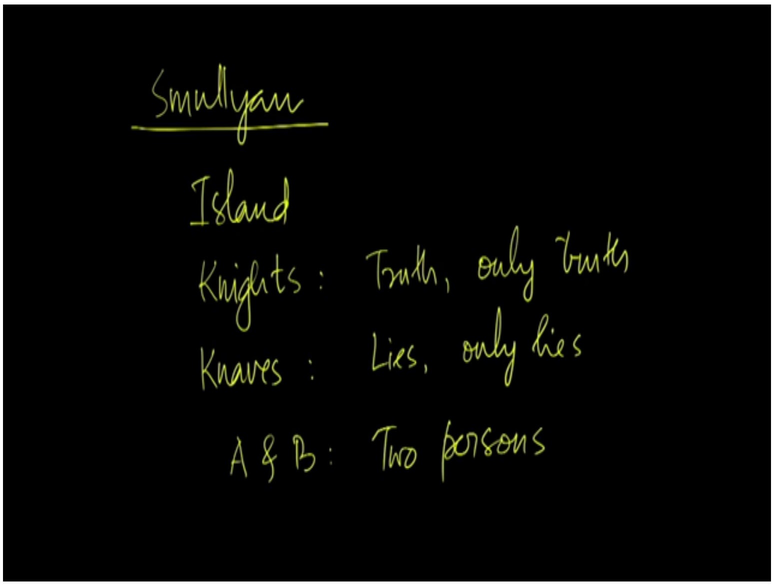
The absorption laws allow you to absorb q in this fashion, p OR p q is the same as p. similarly, p AND p OR q is also the same as p.

(Refer Slide Time: 26:39)



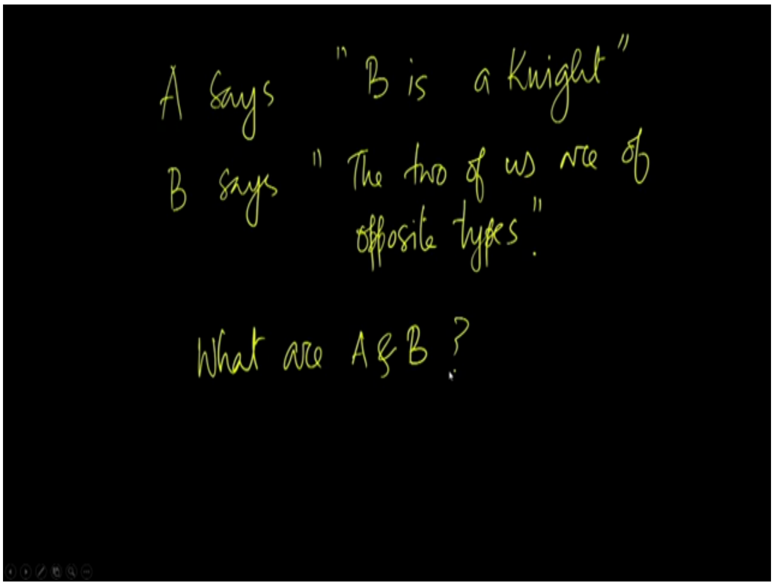
Then, we have the negation laws, the OR of a quantity with a the negation of itself is 1, the AND of a quantity with the negation of it is equivalent to 0, these are the negation laws. So, all these laws can be proved using truth tables.

(Refer Slide Time: 27:12)



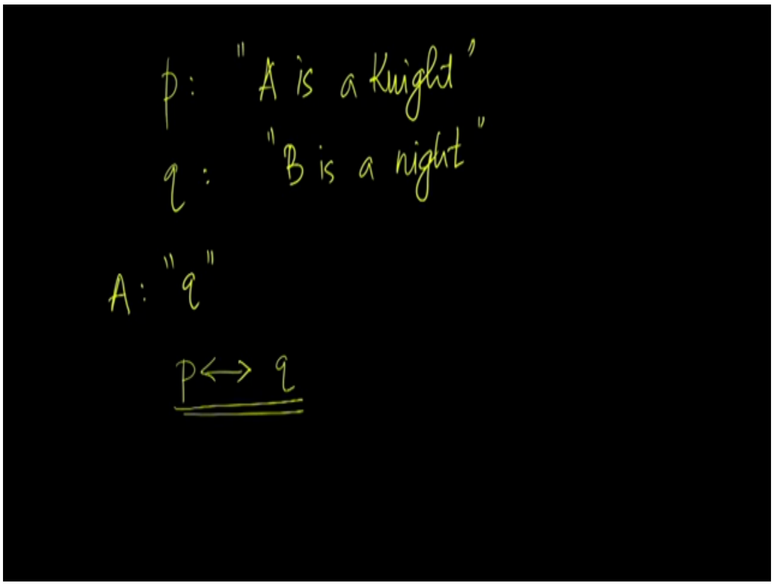
Now, let us consider a problem, a problem that is used to Smullyan and you can find them in various textbooks. Let us say, there is an island, on this island there are only 2 kinds of inhabitants, knights and knaves, (knives) knights speak truth and only truth, knaves tell lies and only lies but you cannot distinguish who is a knight or who is a knave, all of them look alike. Let us say, we encounter 2 persons A and B, so their appearances do not disclose whether they are knightsor knaves, you will have to figure out that from what they say.

(Refer Slide Time: 28:14)



Let us say, A says B is a knight and let us say B says the two of us are of opposite types then the question is, what are A and B? So, let us try to solve this problem by forming propositions.

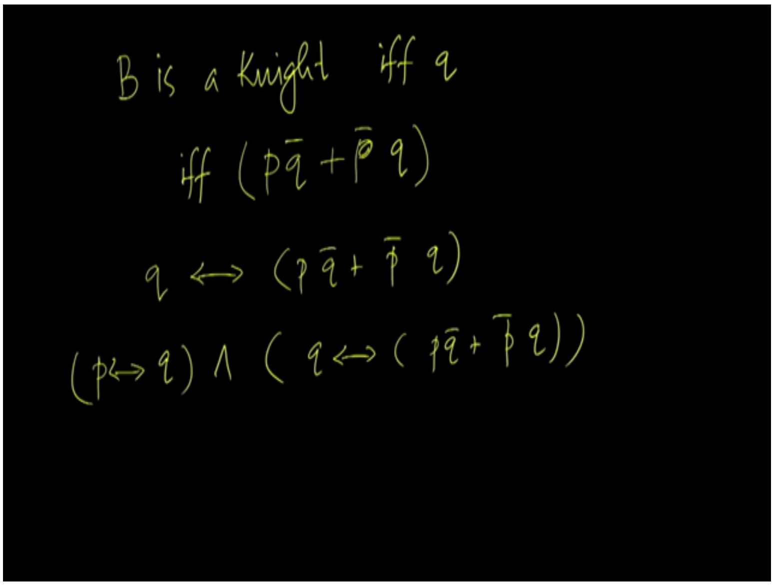
(Refer Slide Time: 28:56)



Let p denote the proposition A is a knight, let q form the proposition B is a knight. Now, what does A say? A says that B is a knight, so A asserts q, so if A is a knight which means if p is true then A would speak only the truth and this would be true and if A is a knave then a would tell only lies and this would be a lie. That is if a is a knight which means if p is true then q also would be true.

On the other hand, if A is a knave which means if p is false then q would be false therefore we have p if and only if q, so this is 1 conclusion we have.

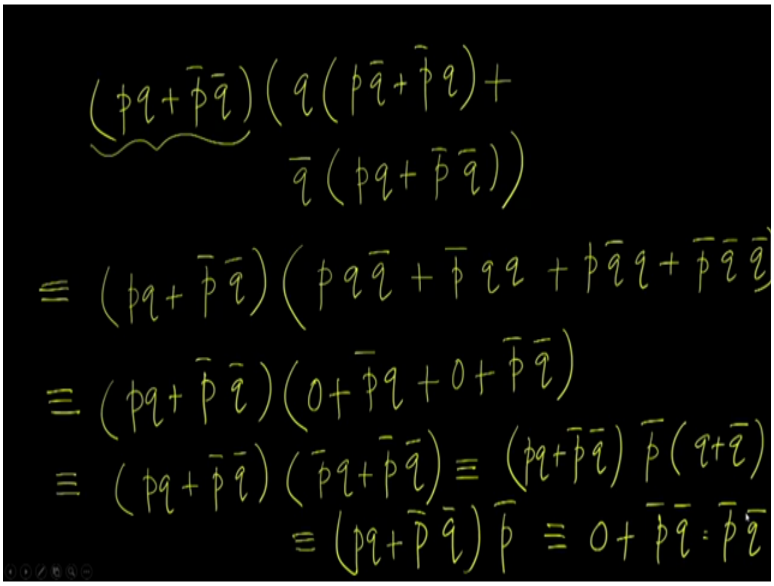
(Refer Slide Time: 29:58)



Now, from what B has said, we can conclude that B is a knight which is if and only if q this is true if and only if what he said is, right. Now, what did he say? He said that both of them are of different types, which would mean? That either p is 1 and q is 0 or that p is 0 and q is 1. In other words, either A is a knight and B is a knave or A is a knave and B is a knight. So, now we have this logical equivalence that is q is true if and only if this is true.

Therefore, the AND of the 2 logical statements is what we have p if and only if q and q if and only if p q bar plus p bar q. So, let us try to simplify this expression, let us see if we can figure out what p and q are from these.

(Refer Slide Time: 31:25)



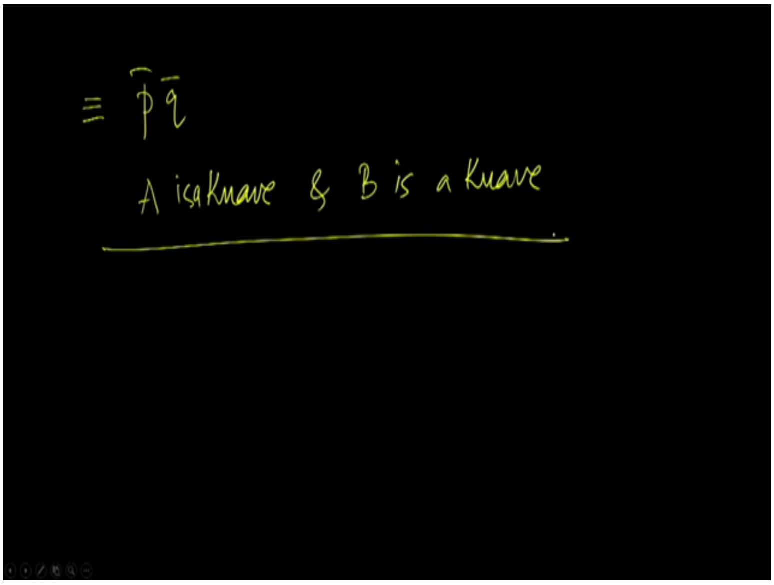
Now, p if and only if q can be written using AND and OR and NOT in this manner, this is true if and only if both have the same logical value, which is possible if both are true or both are false. On the other hand, if q is true then this will have to be true, that is if B is a knight then what he said should be right, which means either A is a knight and B is a knave or A is a knave and B is a knight.

Otherwise, q is false in which case, both are of the same type, that is either both are knight OR both are knaves. So, this is the sum total of the two statements, the AND of the conjunction of the two statements is what we have taken, there is this, we concluded from the first statement and this we concluded from the second statement. Therefore, both of these conclusions must be right, what does that entail?

Let us try to simplify these expressions, we have p q q bar by taking the AND of q with p q bar by associativity and commutativity, I can flip q and p and write this as p q q bar and then I have p bar q q that is from the first conjunction. From the second conjunction, we have p q bar q and p bar q bar q bar. Let us simplify this further, so this is logically equivalent to q q bar is 0, so this is 0 here, p and 0 is 0 q q is q therefore we have p bar q here, q bar q is 0 then we have p and 0 which is 0 again and then p bar q bar q bar is q bar, so we have p bar q bar.

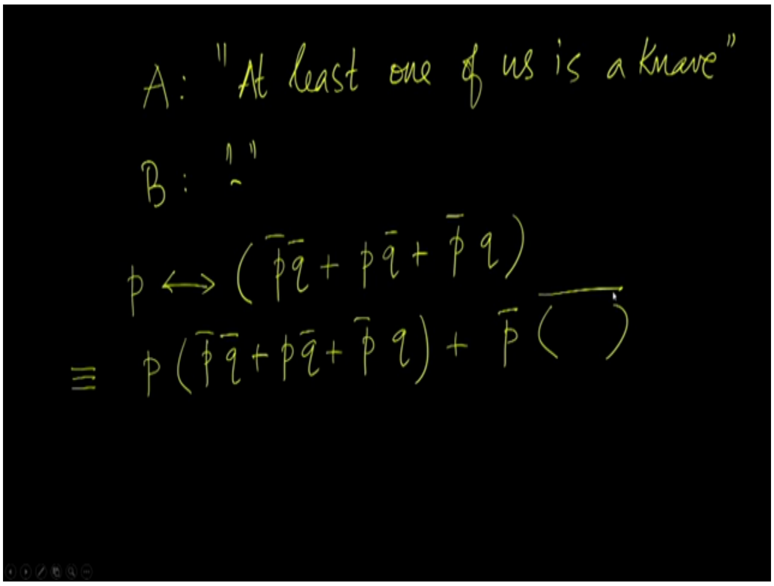
Which is then, equivalent to p q OR p bar q bar AND p bar q and p bar q bar, this can be written as p q or p bar q bar and from the second term, if you take p bar outside we have q plus q bar on the inside which is because q plus q bar is 1 and p bar and 1 is p bar. Now, if you take p bar inside, from the first term we have p bar p q but since p bar p q is 0 we have 0 here and then from the second term we have p bar p bar q bar but p bar p bar is p bar, so we have p bar q bar which is p bar q bar.

(Refer Slide Time: 35:11)



So, we find that this logical expression reduces to p bar q bar, what does that say? It says that A is a knave and b is a knave, both of them are knaves that is what we conclude from the logical expressions.

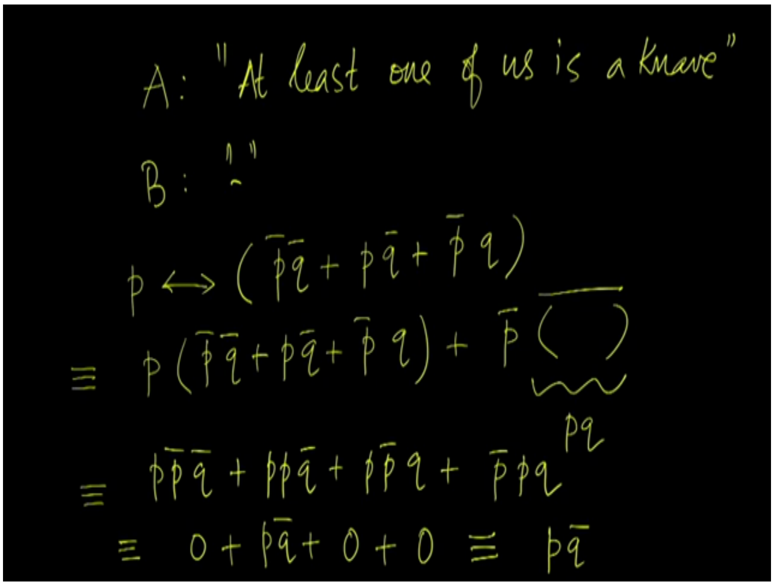
(Refer Slide Time: 35:38)



Let us consider one more such problem, let us say in this case person A says, at least 1 of us is a knave and suppose B does not say anything then let us see what A said. If A is a knight, if p stands for the statement that A is a knight in which case what he said is right. Now, what did he say? He said that, at least 1 of us is a knave then either both of them are knaves or A is a knight and B is a knave, or A is a knave and B is a knight, it is not possible that both of them are knights for at least one of them to be a knave this is precisely how the situation should be.

Now, what does this mean? This is logically equivalent to saying that if p is true then the quantity in the bracket is true or if p is false then this quantity in the bracket is false but what is the negation of the quantity in the bracket?

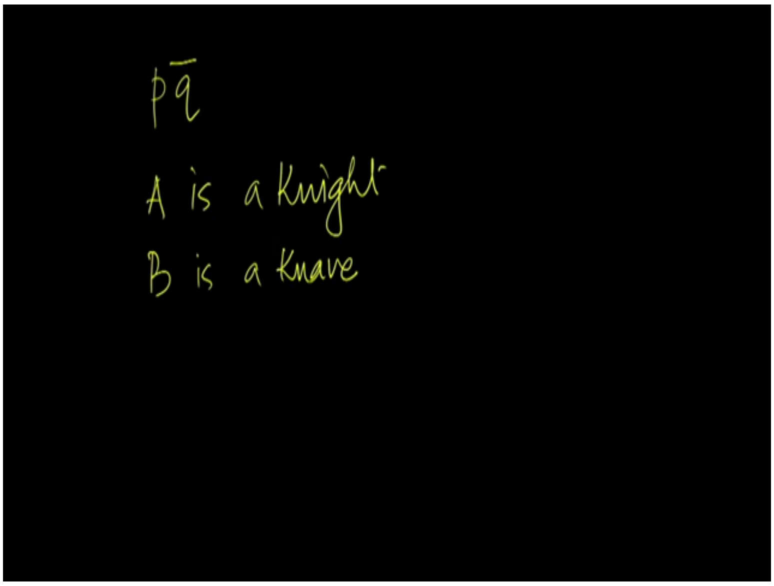
(Refer Slide Time: 37:05)



We want to negate p bar q bar plus p bar q plus p q bar. To compute the negation of this, let us first simplify this expression, we can write this as p bar times q bar plus q and that OR with p q bar this is what we want to negate but then q bar plus q is 1 therefore this is the negation of p bar plus p q bar which by De Morgan's law is the negation of p bar and the negation of p q bar. But double negation of p is p itself and the negation of p q bar by De Morgan's law is p bar plus q.

If you take p inside we have p p bar plus p q which is equivalent to 0 plus p q which means, we have p q. So, the quantity within the bracket when negated will give us p q there is this is p q therefore this expression reduces to p p bar q bar plus p p,q bar plus p p bar q plus p bar p q which is 0 for the first expression p q bar for the second, 0 for the third and 0 for the fourth, the OR of these four quantities would be p q bar.

(Refer Slide Time: 39:12)



So, what we have concluded is that, the logical statement that we had is equivalent to p q bar which means? A is a knight and B is a knave, that is a conclusion we have drawn. So, that is it from this lecture, hope to see you in the next, thank you.