Propositional Logic – Part 3 – Algebraic Properties

Slide 1

In part 3 on propositional logic, we will examine a number of algebraic properties of the logical operators. Many of these properties should be familiar from properties of the arithmetic operators.

Slide 2

We begin with the commutative property.

Being commutative means that we can interchange the operands without altering the result. Recall that the arithmetic operations of addition and multiplication are both commutative, but subtraction and division are not.

First we see by comparing the truth tables for *and* with the operands in both orders that the result is the same in each row, which means *and* is a commutative operation.

We repeat the process with *or* and observe the same thing, leading us to conclude that *or* is also a commutative operation.

Slide 3

Next we discuss the associative property. Being associative means that in a compound proposition that involves two adjacent occurrences of the operation, the result is the same regardless of how they are grouped. Recall that the arithmetic operations of addition and multiplication are both commutative, but subtraction and division are not.

First we notice by comparing the truth tables for *and* with the compound proposition grouped in both orders that the result is the same in each row, which means *and* is an associative operation.

Examining the truth tables with *or*, we observe the same thing, leading us to conclude that *or* is also an associative operation.

Slide 4

The third algebraic property we consider is the distributive property. The distributive property involves two operations. Recall that the arithmetic operation of multiplication distributes across addition but the reverse is not true.

First we notice that *and* distributes across *or*, so p and the quantity q or r is the same as the quantity p and r ored with the quantity p and q. As in every previous case we draw this conclusion based on the fact that the final rows in both truth tablesm are identical in each row.

Unlike the behavior of addition and multiplication, in propositional logic the distributive property holds in both directions. As we see from the second pair of truth tables, *or* also distributes across *and*.

Slide 5

Next we define two categories of propositions, which are referred to as tautologies and contradictions.

A tautology is a proposition which is always true, meaning that each row in the final column of its truth table is a true. The simplest example of a tautology is the compound proposition p or not p.

By contrast, a contradiction is a proposition which is always false. The simplest example of a contradiction is the compound proposition p and not p.

The letter t is customarily used to represent an elementary tautology and the letter c is used to represent an elementary contradiction.

Slide 6

Next we examine the role of the tautology and contradiction as identity elements for the two most fundamental logical operators.

Recall that in arithmetic 0 is the additive identity and 1 the multiplicative identity. That means that adding 0 to any number results in the same number. Similarly multiplying any number by 1 does not change it.

Examining the truth table on the left, we see that the tautology is the identity element for the *and* operation. Anding any proposition with the tautology does not change it.

The truth table on the right illustrates that the contraction is the identity element for the *or* operation. In programming, it is important to understand the importance of identity elements. When performing successive additions, we always initialize the variable we are using to the identity 0. Similarly if we are performing successive *and*s, we must initialize it to true.

Slide 7

Next we consider the algebraic property that involve double negation.

Again we should recall a similar property from arithmetic, which is that when applying the unary minus operation twice to any value, the result is the same as the original value.

As the truth table above illustrates double logical negation has a similar effect. The result is the same as the original.

For this reason, English teachers often caution against using double negatives when we mean single negatives because a double negative is really an affirmative.

Slide 8

The final algebraic property that we consider is the property referred to as DeMorgan’s laws. Of all the properties that we have discussed, it is the least intuitive. It is another distributive property, but in this case, it involves the effect of distributing *not* across both *and* and *or*.

Notice that when *not* distributes across a conjunction, a proposition containing an *and*, the *and* becomes an *or*.

Similarly, when *not* distributes across a disjunction, a proposition containing an *or*, the *or* becomes an *and*. Understanding these properties is extremely important in programming. Requirements can often contain a hidden *not*. For example, the requirements might state that the user should be required to enter a value until a *y* or *n* is entered. Because many languages contain loops that use *while* rather than *until*, this requirement must be translated into *while the user does not enter a y* ***and*** *the user does not enter an n*. Notice that because *until* means *while not*, DeMorgan's law turns the original *or* into an *and*.

Slide 9

Next we define the term logical equivalence.

We say that two propositions are logically equivalent when each row in the final column of their truth tables are identical.

Another way to demonstrate that two propositions are logically equivalent is to examine the compound proposition formed by conjoining the two propositions with the biconditional. If this compound proposition is a tautology, then we can also conclude that the two propositions are logically equivalent.

Every algebraic property that we have discussed so far in this presentation is an example of a logical equivalence.

Slide 10

Finally we define three terms that involve transformations of implications.

Initially we begin with the implication p implies q and then define terms that characterize three transformations of that implication.

The first involves transforming the original implication by reversing the supposition p and conclusion q. The transformed implication q implies p is called the converse of the original.

The next transformation involves negating both the supposition and the conclusion, which produces not p implies not q. This implication is the inverse of the original.

The final transformation involves negating both sides and also reversing them. The resulting implication is not q implies not p. It is the contrapositive of the original.

It is important to understand that the contrapositive is logically equivalent to the original implication. In addition the inverse and converse are logically equivalent to each other, but neither is logically equivalent to the original.

The logical equivalence of the original and its contrapositive is the basis for a proof technique called proof by contraposition.