Discrete Mathematics 🔢

Dr. Aamir Alaud Din

Topic 4: Boolean Algebra

Semester: Spring 2026

🎯 Objectives

By the end of this course you will be able to:

- Explain the basics of boolean algebra.
- Use rules of boolean algebra in computer applications.

Boolean Algebra: Introduction

- The circuits in computers and electronic devices have inputs, each of which is either a 0 or a 1, and produce outputs that are also 0s and 1s.
- Circuits can be constructed using any basic element that has two different states.
- Such elements include switches that can be in either the on or the off position and optical devices that can be either lit or unlit.
- In 1938 Claude Shannon showed how the basic rules of logic, first given by George Boole in 1854 in his ***The Laws of Thought***, could be used to design circuits.
- These rules form the basis for Boolean algebra.

Boolean Algebra: Introduction

- In this topic we develop the basic properties of Boolean algebra.
- The operation of a circuit is defined by a Boolean function that specifies the value of an output for each set of inputs.
- The first step in constructing a circuit is to represent its Boolean function by an expression built up using the basic operations of ***Boolean Algebra***.
- We will provide an algorithm for producing such expressions.
- The expression that we obtain may contain many more operations than are necessary to represent the function.
- Later in the chapter we will describe methods for finding an expression with the minimum number of sums and products that represents a Boolean function.

Boolean Functions: Introduction

- Boolean algebra provides the operations and the rules for working with the set {0, 1}.
- Electronic and optical switches can be studied using this set and the rules of Boolean algebra.
- The three operations in Boolean algebra that we will use most are complementation, the Boolean sum, and the Boolean product.
- The complement of an element, denoted with a bar, is defined by $\bar{0} = 1$ and $\bar{1} = 0$.

Boolean Functions: Introduction

- The Boolean sum, denoted by + or by OR, has the following values:

$$1 + 1 = 1 \qquad 1 + 0 = 1 \qquad 0 + 1 = 1 \qquad 0 + 0 = 0$$

- The Boolean product, denoted by $\cdot$ or by AND, has the following values:

$$1.\,1 = 1 \qquad 1.\,0 = 0 \qquad 0.\,1 = 0 \qquad 0.\,0 = 0$$

- When there is no danger of confusion, the symbol $\cdot$ can be deleted, just as in writing algebraic products.
- Unless parentheses are used, the rules of precedence for Boolean operators are: first, all complements are computed, followed by all Boolean products, followed by all Boolean sums.

Boolean Functions: Introduction

Example

Find the value of $1.0 + \overline{(0 + 1)}$

Solution

$$
\begin{aligned}
1.\,0 + \overline{(0 + 1)} &= 0 + \bar{1} \\
&= 0 + 0 \\
&= 0
\end{aligned}
$$

Boolean Functions: Introduction

- The complement, Boolean sum, and Boolean product correspond to the logical operators, $\neg$, $\vee$, and $\wedge$, respectively, where 0 corresponds to F (false) and 1 corresponds to T (true).
- Equalities in Boolean algebra can be directly translated into equivalences of compound propositions.
- Conversely, equivalences of compound propositions can be translated into equalities in Boolean algebra.
- We will see later in this section why these translations yield valid logical equivalences and identities in Boolean algebra.
- Example below illustrates the translation from Boolean algebra to propositional logic.

Boolean Functions: Introduction

Example

Translate $1.0 + \overline{(0+1)} = 0$, into a logical equivalence.

Solution

We obtain a logical equivalence when we translate each 1 into a T, each 0 into an F, each Boolean sum into a disjunction, each Boolean product into a conjunction, and each complementation into a negation. We obtain

$$(T \wedge F) \vee \neg(T \vee F) \equiv F$$

🎉 Thanks!

Questions?