Contents

1 Propositional Logic

1.1 Proposition: DEFINITION

- A proposition is a **declarative sentence** (that is, a sentence that **declares a fact**) that is **either true** or **false**, **but not both**.

Example

- All the following declarative sentences are propositions.

  1. Washington, D.C., is the capital of the United States of America.
  2. Alexandria is the capital of Egypt.
  3. $1 + 1 = 2$
  4. $2 + 2 = 3$

- Propositions 1 and 3 are true, whereas 2 and 4 are false

Example

- All the following sentences are not Propositions.

  1. What time is it?
  2. Read this carefully.
  3. $x + 1 = 2$
  4. $x + y = z$

- Sentences 1 and 2 are not propositions because they are not declarative sentences

- Sentences 3 and 4 are not propositions because they are neither true nor false.

1.2 Propositional Variables

- We use **letters** to denote **propositional variables** (or **sentential variables**).
- That is, variables that **represent propositions**, just as letters are used to denote numerical variables.
- The **conventional letters** used for propositional variables are $p, q, r, s, \ldots$
- The **truth value** of a proposition is **true**, denoted by **T**, if it is a **true proposition**, and the **truth value** of a proposition is **false**, denoted by **F**, if it is a **false proposition**
- The area of logic that deals with propositions is called the **propositional calculus** or **propositional logic**.
- It was first developed systematically by the Greek philosopher **Aristotle** more than 2300 years ago.

1.3 Logical Operators

- We now turn our attention to methods for **producing new propositions** from those that we already have.
- These methods were discussed by the English mathematician **George Boole** in 1854 in his book **The Laws of Thought**.
- Many mathematical statements are constructed by **combining one or more** propositions.
- New propositions, called **compound propositions**, are formed from **existing propositions** using **logical operators**.
- These logical operators are also called **connectives**.

1.3.1 DEFINTION 1 : Negation

- Let p be a proposition.
- The negation of p, denoted by $\neg p$ (also denoted by $\bar{p}$), is the statement It is not the case that p.
- The proposition $\neg p$ is read not p.
- The truth value of the negation of p, $\neg p$, is the opposite of the truth value of p.
- The negation operator **constructs a new proposition** from a **single** existing proposition
- **Truth Table**

Example

- Find the negation of the proposition
- Michaels PC runs Linux
- Michaels PC does not run Linux.

1.3.2 DEFINITION 2 : Conjunction

- Let p and q be propositions.
- The conjunction of p and q, denoted by $p \wedge q$, is the proposition p and q.
- The conjunction $p \wedge q$ is true when both p and q are true and is false otherwise.
- **Truth Table**

1.3.3 DEFINITION 3 : Disjunction

- Let p and q be propositions.
- The disjunction of p and q, denoted by $p \vee q$, is the proposition p or q.
- The disjunction $p \vee q$ is false when both p and q are false and is true otherwise.

Note

- The use of the connective or in a disjunction corresponds to one of the two ways the word or is used in English, namely, as an **inclusive or**.
- Students who have taken calculus or computer science can take this class.
- Here, we mean that students who have taken both calculus and computer science can take the class, as well as the students who have taken only one of the two subjects.
- On the other hand, we are using the **exclusive or** when we say Students who have taken calculus or computer science, **but not both**, can enroll in this class.
- Here, we mean that students who have taken both calculus and a computer science course cannot take the class.
- Similarly, when a menu at a restaurant states, Soup or salad comes with an entre, the restaurant almost always means that customers can have either soup or salad, but not both.
- If $ab = 0$, then $a = 0$ **or** $b = 0$

1.3.4 DEFINITION 4

- Let p and q be propositions.
- The exclusive or of p and q, denoted by p q, is the proposition that is true when exactly one of p and q is true and is false otherwise.

1.3.5 DEFINITION 5 : Conditional Statement

- Let p and q be propositions.
- The conditional statement $p \to q$ is the proposition **if** p**, then** q**.**
- The conditional statement $p \to q$ is false when p is true and q is false, and true otherwise.
- In the conditional statement $p \to q$, p is called the **hypothesis** (or **antecedent** or **premise**) and q is called the **conclusion** (or **consequence**).

Example

- If it is raining, then there are clouds in the sky.
- $P =$ It is raining.
- $Q =$ There are clouds in the sky.

Notes

- p is sufficient to conclude q, and q is necessary for p.
- However, p is not necessary for q, and q is not sufficient to conclude p

1.3.6 DEFINITION 6 : Biconditional Statement

- Let p and q be propositions.
- The biconditional statement $p \Leftrightarrow q$ is the proposition p if and only if q.
- The biconditional statement $p \Leftrightarrow q$ is true when p and q have the same truth values, and is false otherwise.
- Biconditional statements are also called **bi-implications**.

Notes

- Note that the statement $p \Leftrightarrow q$ is true when both the conditional statements $p \to q$ and $q \to p$ are true and is false otherwise.
- That is why we use the words if and only if to express this logical connective and why it is symbolically written by combining the symbols $\to$ and $\leftarrow$

Example

- Let p be the statement You can take the flight, and let q be the statement You buy a ticket.
- Then $p \Leftrightarrow q$ is the statement "You can take the flight if and only if you buy a ticket".
- This statement is true if p and q are either both true or both false.
- It is false when p and q have opposite truth values, that is, when you do not buy a ticket, but you can take the flight (such as when you get a free trip) and when you buy a ticket but you cannot take the flight (such as when the airline bumps you).

1.3.7 Precedence of Logical Operators

- However, to reduce the number of parentheses, we specify that the negation operator is applied before all other logical operators.
- The conjunction operator takes precedence over the disjunction operator.
- The conditional and biconditional operators $\rightarrow$ and $\Leftrightarrow$ have lower precedence that the conjunction and disjunction operators, $\wedge$ and $\vee$
- 1. $\neg$
  2. $\wedge$
  3. $\vee$
  4. $\rightarrow$
  5. $\Leftrightarrow$

1.4 Logic and Bit Operations

- Computers represent information using bits
- A bit is a symbol with two possible values, namely, 0 (zero) and 1 (one).
- This meaning of the word bit comes from binary digit, because zeros and ones are the digits used in binary representations of numbers.
- The well-known statistician **John Tukey** introduced this terminology in 1946.
- A bit can be used to represent a truth value, because there are two truth values, namely, true and false.

2 Applications of Propositional Logic

2.1 Translating English Sentences

- English (and every other human language) is often ambiguous.
- Translating sentences into compound statements removes the ambiguity.
- Moreover, once we have translated sentences from English into logical expressions we can analyze these logical expressions to determine their truth values, we can manipulate them, and we can use rules of inference to reason about them.

Example

- How can this English sentence be translated into a logical expression?
- You can access the Internet from campus only if you are a computer science major or you are not a freshman.
- $p \to (q \vee \neg r)$

2.2 Logic Circuits

- Propositional logic can be applied to the design of computer hardware.
- This was first observed in 1938 by **Claude Shannon** in his MIT masters thesis.