Discrete Mathematics

&

Graph Theory

For

Computer Science

&

Information Technology

By

THE GATE ACADEMY

A Forum of IIT / IISc Graduates

www.thegateacademy.com

Syllabus

for

Discrete Mathematics & Graph Theory

Propositional and First Order Logic, Sets, Relations, Functions, Partial Orders and Lattices, Groups, Graphs, Connectivity, Matching, Coloring, Combinatorics, Counting, Recurrence Relations, Generating Functions

Analysis of GATE Papers

Year	Percentage of Marks	Overall Percentage
2015	11.00	12.23 %
2014	12.7	
2013	9.00	
2012	10.00	
2011	10.00	
2010	7.00	
2009	10.66	
2008	18.00	
2007	17.33	
2006	16.67	

Contents

"Give me a stock clerk with a goal and I'll give you a man who will make history"
....J.C. Penney

CHAPTER 1

Mathematical Logic

Learning Objectives

After reading this chapter, you will know

1. Syntax
2. Propositional Logic and First Order Logic
3. List of Important Equivalences
4. Simplification Method and Duality Law
5. Equivalence of Well-Formed Formulas
6. Disjunction Normal Form
7. Quantifiers
8. Predicate Calculus

Logic is a formal language. It has syntax, semantics and a way of manipulating expressions in the language.

Syntax

Set of rules that define the combination of symbols that are considered to be correctly structured.

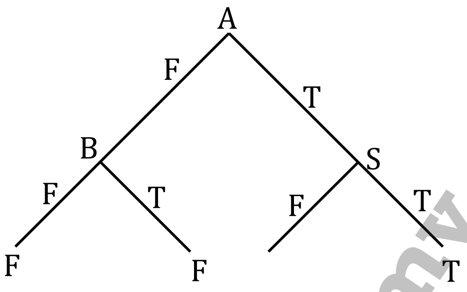
- Semantics give meaning to legal expressions.
- A language is used to describe about a set
- Logic usually comes with a proof system which is a way of manipulating syntactic expressions which will give you new syntactic expressions
- The new syntactic expressions will have semantics which tell us new information about sets.
- In the next 2 topics we will discuss 2 forms of logic
  1. Propositional logic
  2. First order logic

Propositional Logic and First Order Logic

- Sentences are usually classified as declarative, exclamatory interrogative, or imperative
- Proposition is a declarative sentence to which we can assign one and only one of the truth values "true" or "false" and called as zeroth-order-logic.
- Propositions can be combined to yield new propositions

Logic: In general logic is about reasoning. It is about the validity of arguments consistency among statements and matters of truth and falsehood. In a formal sense logic is concerned only with the form of truth in an abstract sense.

Truth Tables: Logic is mainly concerned with valid deductions. The basic ingredients of logic are logical connectives, and or, not if then if and only if etc. we are concerned with expressions involving these connectives. We want to know how the truth of a compound sentence like "x = 1 and y = 2" is affected by, or determined by the truth of the separate simple sentences "x = 1" "y = 2"
Truth tables present an exhaustive enumeration of the truth values of the component propositions of a logical expression, as a function of the truth values of the simple propositions contained in them. The information embodied in them can also be usefully presented in tree from.



The branches descending from the node A are labelled with the two possible truth values for A. the branches emerging from the nodes marked B give the two possible values for B for each value of A. the leaf nodes at the bottom of the tree are marked with the values of A B for each truth combination of A and B.

Logical Connectives or Operators

The following symbols are used to represent the logical connectives or operators.

And	∧ (Conjunction)
Or	∨ (Disjunction)
Not	¬ (Not)
Ex – or	⊕
Nand	↑
Nor	↓
If……then	→ (Implication)
If and only if	↔ (Bidirectional)
Ex-or	⊕

1. **∧ (And/Conjunction)**

We use the letters F and T to stand for false and true respectively

A	B	A ∧ B
F	F	F
F	T	F
T	F	F
T	T	T

It tells us that the conductive operation A is being treated as a binary logical connective-it operates on two logical statements. The letters A and B are "propositional variables"
The table tells us that the compound proposition A ∧ B is true only when both A and B are true separately. The truth table tells us how to do this for the operator. A ∧ B is called a truth function of A and B as its value is dependent on and determined by the truth values of A and B.
A & B can be made to stand for the truth values of propositions as follows:

A: The cat sat on the mat
B: The dog barked
Each of which may be true or false. Then A ∧ B would represent the compound proposition "The cat sat on the mat and the dog barked" A ∧ B is written as A.B in Boolean Algebra.

2. **∨ (Disjunction):** The truth table for the disjunctive binary operation ∨ tells us that the compound proposition A ∨ B is false only if A and B are both false, otherwise it is true.

A	B	A ∨ B
F	F	F
F	T	T
F	F	T
T	T	T

This is inclusive use of the operator 'or'
In Boolean Algebra A ∨ B is written as A + B

3. ¬ (**Not**):
The negation operator is a "unary operator" rather than a binary operator like A and its truth table is.

A	¬ A
F	T
T	F

The table presents ¬ in its role i.e., negation of true is false, and the negation of false is true.

4. **⊕(Exclusive OR of Ex-OR)**
A ⊕ B is true only when either A or B is true but not when both are true or when both are false.

A	B	A⊕B
F	F	F
F	T	T
T	F	T
T	T	T

5. **↑ NAND**
$P \uparrow Q \equiv \neg (P \land Q)$

6. **↓ NOR**
$P \downarrow Q \equiv \neg (P \lor Q)$
Note: $P \uparrow P \equiv \neg P$
$P \downarrow P \equiv \neg P$
$(P \downarrow Q) \downarrow (P \downarrow Q) \equiv P \lor Q$
$(P \uparrow Q) \uparrow (P \uparrow Q) \equiv P \land Q$
$(P \uparrow P) \uparrow (Q \uparrow Q) \equiv P \lor Q$

7. → (Implication):

A	B	A→ B
F	F	T
F	T	T
T	F	F
T	T	T

Note that A →B is false only when A is true B is false. Also note that A →B is true, whenever A is false, irrespective of the truth value of B.

8. ↔(If and only if):

The truth table is

A	B	A↔B
F	F	T
F	T	F
T	F	F
T	T	T

Note that Bi-conditional (if and only if) is true only when both A & B have the same truth values.

Assumptions about Propositions

- For every proposition p, either p is true or p is false
- For every proposition p, it is not the case that p is both true and false.
- Propositions may be connected by logical connective to form compound proposition. The truth value of the compound proposition is uniquely determined by the truth values of simple propositions.
- An algebraic system ({F, T}, ∨, ∧, -) where the definitions of components are.
  A tautology corresponds to the constant T and a contradiction corresponds to constant F.

The definitions of ∧, ∨ and ¬ are given below

∨	F	T
F	F	T
T	T	T

∧	F	T
F	F	F
T	F	T

	¬
F	T
T	F

- The negation of a propositions p can be represented by the algebraic expression ¬p
- The conjunction of two propositions p and q can be represented as an algebraic expression "p∧q"

Truth Table

P	q	P ∧ q
T	T	T
T	F	F
F	T	F
F	F	F

P	q	p ∨ q
T	T	T
T	F	T
F	T	T
F	F	F

- The disjunction of two propositions p and q can be represented as an algebraic expression "p∨q"
- Compound propositions can be represented by a Boolean expression.
  The truth table of a compound proposition is exactly a tabular description of the value of the corresponding Boolean expression for all possible combinations of the values of the atomic propositions.

Example: I will go to the match either if there is no examination tomorrow or if there is an examination tomorrow and the match is a championship tournament

Solution: p = "there is an examination tomorrow"
q = "the match is a championship tournament"
∴ I will go to the match if the proposition $\overline{p} \vee (p \wedge q)$ is true.

- A proposition obtained from the combination of other propositions is referred to as a compound propositions.
- Let p and q be two propositions, then define the propositions as "p then q", denoted as (p→q) is ("p implies "q)
- (p → q) is true, if both p and q are true or if p is false.
- It is false if p is true and q is false specified as in the table

Truth Table

P	**q**	**p → q**
T	T	T
T	F	F
F	T	T
F	F	T

Example: The temperature exceeds 70°C" and "The alarm will be sounded" denoted as p and q respectively.

- "If the temperature exceeds 70°C then the alarm will be sounded " = r i.e. it is true if the alarm is sounded when the temperature exceeds 70°C (p & q are true) and is false if the alarm is not sounded when the temperature exceeds 70°C. (p is true & q is false).
- p ↔ q is true if both p and q are true or If both p and q are false
- It is false if p is true while q is false and if p is false while q is true.
- If p and q are 2 propositions then "p bi implication q" is a compound proposition denoted by p ⟷ q

Truth Table

P	q	p → q
T	T	T
T	F	F
F	T	T
F	F	T

Example: p = "A new computer will be acquired"

q = "Additional funding is available"

Consider the proposition, "A new computer will be acquired if and only if additional funding is available" = r.

- "r" is true if a new computer is indeed acquired when additional funding is available (p and q are true)
- the proposition r is also true if no new computer is acquired when additional funding is not available (p and q are false)
- The r is false if a new computer acquired. Although no additional funding is available. (P is true & q is false)
- "r" is false if no new computer is acquired although additional funding is available (p is false & q is true)
- Two compound propositions are said to be equivalent if they have the same truth tables.
- Replacement of an algebraic expressions involving the operations ⇢ and ↔ by equivalent algebraic expressions involves only the operations ∧, V and

Equation:

p	q	p→ q	~p	~p V q
T	T	T	F	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

$\therefore p \rightarrow q = \overline{p} \vee q$

Similarly $p \leftrightarrow q = (p \wedge q) \vee (\sim p \wedge \sim q)$

Example: $p \rightarrow q = (\sim p \vee q)$

$= \sim\sim (q) \vee \sim p$

$= \sim q \rightarrow \sim p$

Given below are some of the equivalence propositions.

$\sim(\sim(p)) \equiv p$

$\sim(p \vee q) \equiv \sim p \wedge \sim q$

$\sim(p \wedge q) \equiv \sim p \vee \sim q$

$p \longrightarrow q \equiv \sim p \vee q$

$p \leftrightarrow q \equiv (\sim p \vee q) \wedge (\sim q \vee p) \equiv (p \wedge q) \vee (\sim p \wedge \sim q) \equiv (p \rightarrow q) \wedge (q \rightarrow p)$

- A propositional function is a function whose variables are propositions
- A propositional function p is called tautology if the truth table of p contains all the entries as true
- A propositional function p is called contradiction if the truth table of p contains all the entries as false
- A propositional function p which is neither tautology nor contradiction is called contingency.
- A proposition p logically implies proposition q. If p ⟶ q is a tautology.