

UNIT-3 : Propositional Logic & Finite State Machines

Discrete Structure (CSIT-302) Notes

Easy Language + RGPV Exam Oriented Notes

PART-1 : PROPOSITIONAL LOGIC

1. Proposition

Definition

A **proposition** is a statement that is either:

- True (T)
- or
- False (F)

but not both.

Examples

True Proposition

- “ $2 + 3 = 5$ ”

False Proposition

- “10 is a prime number”
-

Not a Proposition

Questions or commands are not propositions.

Examples:

- “How are you?”
 - “Close the door.”
-

Types of Propositions

1. Simple Proposition

Contains single statement.

Example:

p : “It is raining.”
 p : \text{“It is raining.”}
 p : “It is raining.”

2. Compound Proposition

Made using logical operators.

Example:

$p \wedge q$ and $q \wedge p$

2. First Order Logic (Predicate Logic)

Definition

First order logic deals with:

- objects
 - predicates
 - quantifiers
-

Example

“Every student studies.”

Here:

- Student → object
 - studies → predicate
-

Predicate

A statement containing variable.

Example:

$P(x): x > 5$

Truth depends on value of xxx

3. Basic Logical Operations

1. AND Operation (Conjunction)

Symbol:

\wedge \and \wedge

Meaning:

Both statements must be true.

Truth Table

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

2. OR Operation (Disjunction)

Symbol:

\vee or \vee

At least one statement true.

Truth Table

p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

3. NOT Operation (Negation)

Symbol:

$$\neg p \text{ \neg } p \neg p$$

Reverses truth value.

Truth Table

p	$\neg p$
T	F
F	T

4. Implication

Symbol:

$$p \rightarrow q$$

Meaning:

“If p then q”

Truth Table

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

5. Biconditional

Symbol:

$$p \leftrightarrow q$$

True when both have same truth value.

4. Tautology

Definition

A proposition always true.

Example

$$p \vee \neg p$$

Truth Table

p	$\neg p$	$p \vee \neg p$
T	F	T
F	T	T

Hence tautology.

5. Contradiction

Definition

A proposition always false.

Example

$$p \wedge \neg p$$

Truth Table

P	¬P	p ∧ ¬p
T	F	F
F	T	F

Hence contradiction.

6. Algebra of Propositions

Logical algebra laws.

Important Laws

1. Commutative Law

$$p \wedge q = q \wedge p \quad p \vee q = q \vee p$$

$$p \wedge (q \vee r) = (p \wedge q) \vee (p \wedge r)$$

2. Associative Law

$$(p \wedge q) \wedge r = p \wedge (q \wedge r) \quad (p \vee q) \vee r = p \vee (q \vee r)$$

$$p \wedge (q \vee r) = (p \wedge q) \vee (p \wedge r)$$

3. Distributive Law

$$p \wedge (q \vee r) = (p \wedge q) \vee (p \wedge r)$$

$$p \wedge (q \vee r) = (p \wedge q) \vee (p \wedge r)$$

4. DeMorgan's Law

$$\neg(p \wedge q) = \neg p \vee \neg q$$

$$\neg(p \vee q) = \neg p \wedge \neg q$$

7. Logical Implication

Definition

A statement P logically implies Q if:

whenever P true, Q also true.

Symbol

$$P \models Q \text{ models } Q$$

Example

$$p \wedge q \models p \wedge q \text{ models } p \wedge q \models p$$

8. Logical Equivalence

Definition

Two statements having same truth values.

Symbol

$$P \equiv Q \text{ equiv } Q \equiv P$$

Example

$$p \rightarrow q \equiv \neg p \vee qp \rightarrow q \equiv \neg p \vee q$$

$$qp \rightarrow q \equiv \neg p \vee q$$

9. Predicates

Definition

Predicate is statement depending on variable.

Example

$P(x):x>10$ $P(x): x>10$ $P(x):x>10$

Domain

Set of values of variable.

Example:

Natural numbers.

10. Normal Forms

Used to simplify logical expressions.

1. Disjunctive Normal Form (DNF)

Expression written as OR of AND terms.

Example

$$(p \wedge q) \vee (\neg p \wedge r)$$
$$(p \wedge q) \vee (\neg p \wedge r)$$

2. Conjunctive Normal Form (CNF)

Expression written as AND of OR terms.

Example

$$(p \vee q) \wedge (\neg p \vee r)$$
$$(p \vee q) \wedge (\neg p \vee r)$$

11. Universal Quantifier

Symbol

\forall for all \forall

Meaning:

“For all”

Example

$\forall x(x+0=x)$ \forallall x(x+0=x) $\forall x(x+0=x)$

means every number added with 0 remains same.

12. Existential Quantifier

Symbol

\exists \existsists \exists

Meaning:

“There exists”

Example

$\exists x(x^2=4)$ \existsists x(x^2=4) $\exists x(x^2=4)$

There exists number whose square is 4.

PART-2 : FINITE STATE MACHINES

(FSM)

1. Introduction to Finite State Machine

Definition

A finite state machine is mathematical model having:

- finite number of states
 - input
 - output
 - transitions
-

Components of FSM

1. States
 2. Input symbols
 3. Transition function
 4. Initial state
 5. Final state
-

Example

Traffic light system:

- Red
- Yellow
- Green

Each color is state.

2. FSM as Model of Physical System

FSM models real-world systems.

Examples

- Traffic light controller
 - ATM machine
 - Elevator system
 - Washing machine
-

Traffic Light Example

States:

- Red
- Yellow
- Green

Transitions:

Red → Green → Yellow → Red

3. Equivalence Machines

Definition

Two FSMs are equivalent if they produce same output for same input sequence.

Meaning

Even if structure different,

behavior/output same.

Example

Two vending machines accepting same coins and giving same product.

4. FSM as Language Recognizer

FSM can recognize strings/languages.

Example

Recognize binary strings ending with 1.

States

- q0
 - q1
-

Working

If string ends with 1,

FSM accepts string.

Applications of FSM

1. Compiler design
2. Pattern matching
3. ATM systems
4. Network protocols
5. Digital electronics

MOST IMPORTANT TOPICS FOR RGPV EXAM

- ★ Truth Tables
- ★ Tautology & Contradiction
- ★ Logical Equivalence
- ★ DeMorgan's Laws
- ★ Quantifiers
- ★ CNF & DNF
- ★ Predicate Logic
- ★ Finite State Machine
- ★ FSM as Language Recognizer
- ★ Equivalence Machines
