ADVANCED DISCRETE MATHEMATICS

M.Sc., MATHEMATICS First Year

Semester – I, Paper-V

Lesson Writers

Prof. Dr. Kuncham Syam Prasad
B.Ed., M.Sc., Ph.D.,
Department of Mathematics
Manipal Institute of Technology
Manipal Academy of Higher Education
Manipal-576104, Karnataka

Prof. Dr. S. Srinadh
M.Sc., Ph.D.,
Department of Mathematics (Rtd)
Sri Venkateswara University
Tirupathi, Andhra Pradesh

Prof. Dr. Kedukodi Babushri Srinivas
M.Sc., Ph.D.,
Department of Mathematics
Manipal Institute of Technology
Manipal Academy of Higher Education
Manipal-576104, Karnataka

Prof. Dr. Harikrishnan Panackal
M.Sc., Ph.D.,
Department of Mathematics
Manipal Institute of Technology
Manipal Academy of Higher Education
Manipal-576104, Karnataka

Editor & Lesson Writer
Prof. Dr. Bhavanari Satyanarayana
M.Sc., M.Tech., B.Ed., Ph.D.,
Department of Mathematics
University College of Sciences
Acharya Nagarjuna University.

Director, I/c
Prof. V.VENKATESWARLU
MA., M.P.S., M.S.W., M.Phil., Ph.D.

CENTRE FOR DISTANCE EDUCATION
ACHARAYANAGARJUNAUNIVERSITY
NAGARJUNANAGAR – 522510
Ph:0863-2346222,2346208,
0863-2346259(Study Material)
Website: www.anucde.info
e-mail:anucdedirector@gmail.com

M.Sc., MATHEMATICS - ADVANCED DISCRETE MATHEMATICS

First Edition 2025

No. of Copies :

Published by:
Prof. V.VENKATESWARLU,
Director, I/C
Centre for Distance Education,
Acharya Nagarjuna University

Printed at:

FOREWORD

Since its establishment in 1976, Acharya Nagarjuna University has been forging ahead in the path of progress and dynamism, offering a variety of courses and research contributions. I am extremely happy that by gaining 'A+' grade from the NAAC in the year 2024, Acharya Nagarjuna University is offering educational opportunities at the UG, PG levels apart from research degrees to students from over 221 affiliated colleges spread over the two districts of Guntur and Prakasam.

The University has also started the Centre for Distance Education in 2003-04 with the aim of taking higher education to the doorstep of all the sectors of the society. The centre will be a great help to those who cannot join in colleges, those who cannot afford the exorbitant fees as regular students, and even to housewives desirous of pursuing higher studies. Acharya Nagarjuna University has started offering B.Sc., B.A., B.B.A., and B.Com courses at the Degree level and M.A., M.Com., M.Sc., M.B.A., and L.L.M., courses at the PG level from the academic year 2003-2004 onwards.

To facilitate easier understanding by students studying through the distance mode, these self-instruction materials have been prepared by eminent and experienced teachers. The lessons have been drafted with great care and expertise in the stipulated time by these teachers. Constructive ideas and scholarly suggestions are welcome from students and teachers involved respectively. Such ideas will be incorporated for the greater efficacy of this distance mode of education. For clarification of doubts and feedback, weekly classes and contact classes will be arranged at the UG and PG levels respectively.

It is my aim that students getting higher education through the Centre for Distance Education should improve their qualification, have better employment opportunities and in turn be part of country's progress. It is my fond desire that in the years to come, the Centre for Distance Education will go from strength to strength in the form of new courses and by catering to larger number of people. My congratulations to all the Directors, Academic Coordinators, Editors and Lesson-writers of the Centre who have helped in these endeavors.

Prof. K.Gangadhara Rao
M.Tech.,Ph.D.,
Vice-Chancellor I/c
Acharya Nagarjuna University

M.Sc. – Mathematics Syllabus

SEMESTER-I

105MA24: ADVANCED DISCRETE MATHEMATICS

Unit-I: Propositional Calculus: Statements and Notations- Connectives and Truth Tables – Tautology and Contradiction – Equivalence of Statement/Formulas – Duality Law and Tautological Implication – Normal Forms. (Chapter -I of the reference [3]).

Unit-II: The theory of Inference for Statement Calculus – Consistency of Premises and Indirect Method of Proof. (Chapter -I of the reference [3]).
Predicate Calculus: Predicate Logic – Statement Functions, Variables and Quantifiers – Free and Bound Variable- Inference Theory for the Predicate Calculus (Chapter -II of the reference [3]).

Unit-III: Finite Machines: Introduction, State tables and State diagrams, Simple Properties, Dynamics and Behavior.(refer Chapter 5 of the reference book [1]).

Unit-IV: Properties and Examples of Lattices, Distributive Lattices, Boolean Polynomials.(Section 1 to 4 of Chapter -I of [2]).

Unit-V: Ideals, filters and equations, minimal forms of Boolean Polynomials, Application of Lattices: Application of Switching circuits, (Section 5,6 of Chapter-I and Sections 7 and 8 of Chapter- II of [2]).

Note: For units- III and IV the material of Pages 1 to 66 of [2] is to be Covered.

REFERENCE BOOKS:

[1] "Application Oriented Algebra" JAMES FISHER, IEP, Dun-Danplay Pub.1977.

[2] "Applied Abstract Algebra", Second Edition, R.LIDL AND G.PILZ, Springer, 1998.

[3] "Bhavanari Satyanarayana, Tumurukota Venkata Pradeep Kumar and Shaik Mohnddin Shaw, "Mathematical Foundation of Computer Science" BS Publications,(A Unit of BSP Book Pvt. Ltd.) Hyderabad, India 2016.(ISBN. 978-93-83635-81-8).

[4] Rm. SomaSundaram "Discrete Mathematical Structures" Prentice Hall of India, 2003.

[5] Bhavanari Satyanarayana & Kuncham Syam Prasad, "Discrete Mathematics and Graph Theory" (For b.Tech/B.Sc./ M.Sc.(Maths),Printice Hall of India, New Delhi. April 2014.

CODE:105MA24

M.Sc DEGREE EXAMINATION

First Semester

Mathematics :: Paper V- ADVANCED DISCRETE MATHEMATICS

MODEL PAPER

Time: Three hours Maximum:70 Marks

Answer ONE question from each unit (5x14=70)

UNIT-I

1. **(a).** What do ou mean by Conjunction and Disjunction. Explain by giving one example to each. Also write truth tables for both Conjunction and Disjunction.

(b). If p and q are two statements, then show that the statement (p □ q) □ (p □ q) is equivalent to (p v q) □ (p □ q) .

(OR)

2. **(a).** Write down the Contrapositive of the following statement:
"If Rama have Rs.100/- with him, then he will spend Rs. 50/- for his friend Krishna".

(b). Find the PCNF of the given statement formula $F = X \square \overline{Y}$.

UNIT-II

3. **(a).** Explain the terms: Predicate, 2-place predicate, and m-place predicate. Give one example for each.

(b). Symbolize "All the people respects selfless leaders".

(OR)

4. **(a).** Explain: Universal specification, Universal generalization, Existential specification, Existential generalization. Give one example each.

(b). Prove the validity of the following argument by using the rules of inference.
All men are warriors. (Premise–1)
All Kings are men. (Premise–2)
Therefore All Kings are warriors.

UNIT-III

5.(a). Define input-output Machine. Explain the Parity-Check Machine and write down its State table.

(b). Let f be a state homomorphism from the state machine $M = (\zeta, \mathcal{J}, \delta)$ onto the state machine $M_1 = (\zeta_1, \mathcal{J}, \delta_1)$. Then prove that there is a state machine congruence on M such that $\overline{M}$ is isomorphic to M_1.

(OR)

6. (a). Let $M = (\zeta, \mathcal{J}, O, \delta, \theta)$ be an i/o-machine and let $\square$ an i/o-machine congruence. Then prove that $\overline{\mathrm{M}} = (\square, \mathcal{J}, O, \bar{\delta}, \bar{\theta})$ is an i/o-machine and the function f from ζ onto $\square$ given by $f(s) = [s]$ is an i/o-homomorphim form M onto $\overline{\mathrm{M}}$.

(b). Minimize the number of states for the machine given by the following state table.

Also write down the reduced machine of the given machine.

States	δ		θ	
	0	1	0	1
s_0	s_0	s_2	0	0
s_1	s_2	s_5	1	0
s_2	s_2	s_2	1	1
s_3	s_1	s_1	1	1
s_4	s_2	s_3	0	1
s_5	s_4	s_5	1	1
s_6	s_2	s_6	1	1

UNIT-IV

7. (a). Define Lattice ordered set, and Algebraic Lattice.
Prove that a Lattice ordered set can be turned into an Algebraic Lattice.

(b). (i). Give two examples of lattices with five elements.

(ii). Give two examples of lattices with six elements.

(iii). Define product lattice of a collection of lattices.

(OR)

8. (a). Define modular lattice and distributive lattice.
Prove that every distributive lattice is a modular lattice.

(b). Find the d.n.f of the following function f :

$$f(x_1, x_2, x_3) = [x_1 \square ((x_2 \vee x_3)^1)] \vee \{[(x_1 \square x_2) \vee x_3^{1}] \square x_1\}.$$

UNIT-V

9. (a). Define Boolean Algebra**.**
If B is a finite Boolean algebra, and A denotes the set of all atoms in B, then prove that B is Boolean isomorphic to $P(A)$.

(b). Let B be a Boolean algebra and I a non-empty subset of B. Then prove that the following three conditions are equivalent:

(i) $I \trianglelefteq B$ (That is, I is an ideal of B);

(ii) If $i, j \in I$ and $b \in B$ such that $b \leq i$, then $i + j \in I$ and $b \in I$.

(iii) There exists a Boolean algebra B_1 and a Boolean homomorphism $h : B \rightarrow B_1$ such that $I = \text{Ker } h$.

(OR)

10. (a). (i). What do you mean by Karnaugh diagram. Give an example.
(ii). Simplify the polynomial $p = (x_1 + x_2)(x_1 + x_3) + x_1x_2x_3$ by using its Karnaugh diagram.

(b).(i). Draw switching circuit which represent the Boolean expression:
$x_1 \wedge (x_2 \vee x_3)$.
(ii). Draw NAND gate.
(iii). What do you mean by Half-adder and Full adder. Explain with their diagrams.

CONTENTS

LESSON - 1
STATEMENTS, CONNECTIVES, AND TRUTH TABLES

OBJECTIVE:

- To know Statements
- To understand the Meaning of syllogism
- To identify different types of notations
- To Learn the validity of the arguments
- To have proper understanding of different connectives
- To develop skills to construct the truth tables

STRUCTURE

1.1 Introduction
1.2 Statements.
1.3. Syllogism.
1.4 Notations.
1.5 Connectives and Truth Tables
1.6 Summary
1.7 Technical Terms
1.8 Self Assessment Questions
1.9 Suggested Readings

1.1. INTRODUCTION:

Logic is a form of reasoning. The main object of the logic is to explain the rules by which one can determine the validity or to know the strength of any particular argument or reasoning. Logic deals with all types of reasons like: legal arguments, mathematical proofs, conclusions in a scientific theory based upon a set of given hypothesis. The rules are called "Rules of Inference". The rules should be independent of any particular argument or discipline or language used in the argument.

Logic was discussed by its ancient founder Aristotle (384 BC – 322 BC) from two quite different points of view. On one hand he regarded logic as an instrument or organ for appraising the correctness or strength of the reasoning; On the other hand, he treated the principles and methods of logic as interesting and important topics of the study. According to Charles Pierce "the logic is to classify the arguments, so that all those that are bad are thrown into one basket and those which are good into another". Thus the study of logic, is nothing but the study of the methods and principles to distinguish the correct (good) arguments from incorrect (bad) arguments. The study of logic will provide the reader certain techniques for testing the validity of a given arguments. So the logic is the science of reasoning. Reasoning is a special kind of thinking called inferring, through which conclusions can be drawn.

1.2 STATEMENTS:

In any language, a sentence is constructed by means of some words in that language. So a meaningful sequence of words is called as a sentence. A statement is a sentence for which we can say whether it is true or false.

We need an objective language to frame the rules of inference or theory. The basic unit of our objective language is called a atomic statement or simple statement or primary statement (variable). We assume that these primary statements cannot be broken down further or analyzed into simpler statements. These primary statements have only one of the two possible values TRUE (T) or FALSE (F). These values T or F are referred as truth value of the primary statement. We often denote the truth value TRUTH (T) by '1' and the truth value FALSE (F) by '0'.

1.2.1. Examples:

(i). $2 + 3 = 5$.
(ii). New Delhi is the capital city of HUNGARY.
(iii). Open the door
(iv). $2 + 3 = 6$.

The sentence (iii) is not a primary statement because it has neither the truth value 'T' nor 'F'. The remaining three statements are primary statements. Statements (i) and has the truth value 'T' (or 1), and the statements (ii) and (iv) have the truth value 'F' (or 0).

1.3. SYLLOGISM:

We shall mean, by formal logic, a system of rules and procedures used to decide whether or not a statement follows from some given set of statements.

1.3.1. Note: A familiar example from Aristotelian logic is:

(i). All men are mortal
(ii). Socrates is a man
Therefore (iii). Socrates is mortal.
} Example 1.1

In order to have better understanding , we use symbols. The symbols are easy to manipulate. Hence, the logic we study is also named as "Symbolic logic".

According to the logic, if any three statements have the following form
(i) All M are P
(ii) S is M
Therefore (iii) S is P
} Example 1.2

then (iii) follows from (i) and (ii).
The argument is correct, no matter whether the meanings of statements (i), (ii), and (iii) are correct. All that required is that they have the forms (i), (ii), and (iii). In Aristotelian logic, an argument of this type is called syllogism.

The formulation of the syllogism is contained in Aristotle's organon. It had a great fascination for medieval logicians, for almost all their work centered about ascertaining its valid moods. The three characteristic properties of a syllogism are as follows:

(i). It consists of three statements. The first two statements are called as premises, and the third statement is called as conclusion. The third one (conclusion) being a logical consequence of the first two (the premises).

(ii). Each of the three sentences has one of the four forms given in the Table -1:

Classification	Examples
Universal and affirmative judgment	All X is Y. All monkeys are tree climbers. All integers are real numbers. All men are mortal.
Universal and negative judgment	No X is Y. No man is mortal. No monkey is a tree climber. No negative number is a positive number.
Particular and affirmative judgment	Some X is Y. Some men are mortal. Some monkeys are tree climbers. Some real numbers are integers.
Particular and negative judgment	Some X is not Y. Some men are not mortal. Some monkeys are not tree climbers. Some real numbers are not integers.

Table -1

1.3.2. Note:
Consider Example -1.2. The first two propositions are premises and the third is the conclusion. Here the subject of the conclusion is "S"; and the predicate of the conclusion is "P" and the term to which they are both compared is called the middle term and is denoted by "M".

Consider Example -1.1. The first two propositions

"All men are mortal"

"Socrates is a man"

are premises. The third proposition

"Socrates is mortal"

is the conclusion. The subject of the conclusion is "Socrates" and the predicate of conclusion is "mortal". The middle term is "men".

1.3.3 Example: (i). All fishes are mammals
(ii). All mammals have wings
Therefore (iii). All fishes have wings.
This argument is valid. Note that both the premises are false, and the conclusion also false. The argument of this example 1.3 is valid because if its premises were true then its conclusion would have to be true. Thus the validity of an argument does not guarantee the truth of the conclusion.

1.3.4 Example: (i). No Professors are rich
(ii). All handsome men are Professors
Therefore (iii). No handsome men are rich.
This argument/syllogism is valid (it is similar to Example 1.3). Note that both the premises are false, and the conclusion also false. So we may conclude that the validity of a syllogism is independent of the truth or falsity of its conclusion.

1.3.5 Example: (i). The denominator of $\frac{14}{18}$ is even.
(ii). $\frac{7}{9}$ is another name for $\frac{14}{18}$.
Therefore (iii). The denominator of $\frac{7}{9}$ is even.

This is an invalid argument. The subject in (i) relates to a part (called denominator) of $\frac{14}{18}$. The subject in (ii) is not related to the denominator. So, in this case, (iii) cannot get from (i) and (ii). That is, we cannot get the conclusion from the premises.

1.3.6 Example: (i). If I am President then, I am famous.
(ii). I am not President
Therefore (iii). I am not famous.

Here the argument is clearly invalid because 'one may be famous even though he/she is not a President'.

1.4. NOTATIONS, CONNECTIVES AND TRUTH TABLES:

1.4.1. Some Examples: (i) "$x > 3$" is a statement. This statement is neither true nor false because the value of the variable x is not specified. Therefore "$x > 3$" is not a proposition.
"$x + y + 4 = 7$" is a statement but it is not a proposition.

(ii). "$10 > 3$" is a statement. This statement is true. Therefore "$10 > 3$" is a proposition.

(iii). "$10 < 3$" is a statement. This statement is false (or not true). Therefore "$10 < 3$" is a proposition.
(iv). "$x \geq 3$ for all x such that $x \geq 5$" is a statement. This statement is true.
Therefore, it is a proposition.
(v). "Guntur is the capital of Andhra Pradesh" is a statement which is false. Therefore it is a proposition.
(vi). "What is the time now ?". This is not a statement. So this is not a proposition.
(vii). "$2 + 2 = 3$" is a statement which is false. Therefore it is a proposition.

1.4.2 Subject and Predicate: Consider the statement "Ravana is a King". In this statement "Ravana" is the subject of the statement. The other part "is a King" is called predicate.

1.4.3. Notation

Observe the following statements:

(i) p: Socrates is a man.

(ii) q: $2 + 3 = 6$.

(iii) In statements (i), (ii), p and q are the symbols used. Here "p" is a statement in symbolic logic that corresponds to the English statement "Socrates is a man".

As we know, "Socrates" is the subject and "is a man" is the predicate. The statement P (that is, "Socrates is a man") contains only one subject and only one predicate. So this is a primary statement.

Similarly the statement "q" that represents $2+ 3 = 6$ is also a primary statement. Note that the symbols "p" and "q" were used to represent the names of the statements. We may use this symbolic notation for statements throughout.

From the above discussions, one can understand that the basic unit of our objective language is called as primary or atomic statement (or variable). We assume that these primary (or atomic) statements cannot be further broken down into simple statements.

1.5. CONNECTIVES AND TRUTH TABLES:

By using connectives "not', "or", "and", etc., we may combine two or more primary statements. The words like "or", "and" are called as connectives.

1.5.1. Example: Consider the two statements given by

(i). p: Rama is a King

(u). q: Sita is a Queen.

We know that the two statements (i) and (ii) are primary statements. By using the connective "and" we can combine these two statements to get the third statement:

(iii). Rama is a King and Sita is a Queen.

The statement (iii) is called as compound statement.

The sentences constructed by using two or more primary (or simple or atomic) statements and certain sentential connectives are called as compound statements. The simple statements used to form compound statements are named as the components of the compound statement.

To form compound statements we use simple sentences and the connectives "and", "or", "if....then....", "if and only if", etc.

1.5.2. Example

(i) p and q: Rama is a King and Sita is a Queen.

(ii) p or q: Rama is a King or Sita is a Queen.

(iii) If p then q: If Rama is a King, then Sita is a Queen.

(iv) p if and only if q: "Rama is a King" if and only if "Sita is a Queen".

1.5.3: Negation:

Associated with every given statement 'p' there corresponds an another statement called its negation. The negation of a statement is formed by using/adding the word "not".
If "p" is a statement, then the negation of p is "not P"(denoted by "~p", and called as negation of P). The symbol "~" is called "curl" or "twiddle" or "tilde". The notation "~p" is false if "p" is true. If "p" is false, then "~p" is true. The symbol $\neg p$ (or $\bar{p}$) is also used to represent the negation of p.

1.5.4: Examples:

(i). Let p be the statement "New York is a city". Now ~p is the statement "Not, New York is a city" (equivalently, "New York is not a city").
(ii) If "p : Rama is a King", then "~ p: Rama is not a King".
(iii). If " U : No angle can be trisected by suing straightedge and compass alone", then " ~U: Some angles can be trisected by using straightedge and compass alone".

1.5.5. The truth Table for the negation of a statement

P	~P
T	F
F	T

P	~P
1	0
0	1

As we know, here T (or 1) stands for "True" and F (or 0)stands for "False".

1.5.6. Conjunction:

The conjunction (in symbol, $\wedge$ (read as meet or and)) is commonly used to combine sentences / statements larger ones. The symbol ampersand ("&") also used for "and". The statement "A $\wedge$ B" or "A & B" will be read as the "conjunction of A and B". Let P and Q be statements. The conjunction of P and Q (denoted by P $\wedge$ Q) is true when both P and Q are true; and is false otherwise. In other words, P $\wedge$ Q is true only if both P is true and Q is true.

1.5.7: Examples:

(i). If P : Rama is a King ; Q : Sita is a Queen , then .
P $\wedge$ Q : Rama is a King and Sita is a Queen.

(ii). If P : Two is an even number ; Q : Two is a positive number , then .
P $\wedge$ Q : Two is an even number and a positive number.

1.5.8: Truth Table for conjunction

P	Q	P $\wedge$ Q
T	T	T
T	F	F
F	T	F
T	F	F

1.5.9: Disjunction:

The disjunction ("or", in symbol $\vee$) is used to connect two sentences / statements to form a combined sentence/ compound statement. The symbol $\vee$ is also called as join. "P $\vee$ Q" is called as disjunction of P, Q. If "P" and "Q" are statements, then "P $\vee$ Q" is a statement that is true when "P" is true or "Q" is true or both are true.
In other words, "P $\vee$ Q" is false only when both "P" and "Q" are false.

1.5.10. Examples: (i). If P : Rama is a King ;
Q : Sita is a Queen , then .
$P \vee Q$: Rama is a King or Sita is a Queen.

(ii). If P : Two is an even number ;
Q : Two is a positive number , then .
$P \vee Q$: Two is an even number or a positive number.

1.5.11: Truth Table for disjunction

P	Q	$P \vee Q$
T	T	T
F	T	T
T	F	T
F	F	F

1.5.12. Example: Write the following statements in symbolic form

(i) Rama and Bhima are rich
(ii) Neither Rama nor Sita is poor

Solution: Suppose that

p: Rama is rich
q: Bhima is Rich
Then the given statement can be written in the symbolic form as $p \wedge q$.

(i) write
p: Rama is poor
q: Sita is poor.

Then
~p: Rama is not poor
~q: Sita is not poor

The other form of the given statement is "Rama is not poor" and "Sita is not poor", hence the required answer is $(\sim p) \wedge (\sim q)$.

1.5.13 Example

Obtain the compound statement that is true if exactly two of the three statements p, q and r are true?

Solution: We know that

(i). $(p \wedge q) \wedge (\rceil r)$ is true if "p and q are true" and "r is false".

(ii). $(p \wedge r) \wedge (\rceil q)$ is true if "p and r are true" and "q is false".

(iii). $(q \wedge r) \wedge (\rceil p)$ is true if "q and r are true" and "p is false".

Hence $[(p \wedge q) \wedge (7r)] \vee [(p \wedge r) \wedge (7q)] \vee [(q \wedge r) \wedge (7p)]$ is the compound statement that is true when exactly two of the given three statements p, q and r are true.

1.5.14. Notation: For any two given statements p and q

(i) The compound statement $\rceil(p \wedge q)$ is denoted by "$p \uparrow q$"

(ii) The compound statement is $\rceil(p \vee q)$ denoted by "$p \downarrow q$"

(iii) Note that the symbols "↑" and "↓" are also connectives.

(iv). The truth table for "$p \uparrow q$" is given by

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

Truth table for "$p \uparrow q$"

(v). The truth table for "$p \downarrow q$" is given by

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

Truth table for "$p \downarrow q$"

1.5.15 Statement Formulas and Truth Tables

We know that the statements those contains one or more simple statements and some connectives are called as compound (or composite or molecular) statements.

For example, if p and q are two simple statements,

then $\rceil p$, $p \wedge q$, $p \vee q$, $p \wedge (\rceil q)$, $(\rceil p) \vee (\rceil q)$ are some composite statements.

Such statements are also called as statement formulas derived from the simple statements p and q. In this situation, p and q are called as the components of the statement formulas. The truth value of a statement formulas depends on the truth value of the primary statements involved in it.

As we already know, $\rceil p$ means negation of p: and $\rceil(p \wedge q)$ means negation of $(p \wedge q)$

1.5.16. Example: Construct the truth tables for the following statement formulas

(i) $\sim(\sim p)$ [that is $\neg(\neg p)$]

(ii) $q \wedge (\neg q)$

(iii) $p \vee (\neg q)$

Solution:

(i).

p	~p	~(~p)
1	0	1
0	1	0

Truth Table for ~(~p)

Note that the truth value of both p and ~(~p) are same in all cases

(ii).

q	7q	$q\wedge(\neg q)$
1	0	0
0	1	0

Truth table for q∧(7q)

Note that the truth value of $q\wedge(\neg q)$ is always zero (0 or F).

(iii).

p	q	7q	$p\vee(\neg q)$
T	T	F	T
T	F	T	T
F	T	F	F
F	F	T	T

Truth Table for p∨ (7q)

1.5.17. Implication (or Conditional Statement):

The implication of the statements p and q is a statement that has the form "if p, then q" (denoted by $p \to q$). The truth value of "$p \to q$" is false only if "p is true" and "q is false". In all other cases, the statement "$p \to q$" has truth value "true" (or T). In this implication p is called the hypothesis (or antecedent or premise) and q is called the conclusion (or consequence).

"$P \to Q$" can be read in any one of the following different ways:

(i). P implies Q;
(ii). Q is a (logical) consequence of P;
(iii). P is a sufficient condition for Q;
(iv). Q is a necessary condition for P;
(v). If P then Q;

(vi). P only if Q.
We may also denote "$P \rightarrow Q$" by "$P \Rightarrow Q$".

Truth Table for "Implication" is given below

P	Q	$P \rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

1.5.18. Examples:

(i). "If x > 10, then x > 2" (or "$x > 10 \Rightarrow x > 2$") is a true statement (because if "x > 10" is true, then "x > 2" is also true)

(ii) If "today is a Sunday, then tomorrow is a Monday" (In other words, today is a Sunday $\Rightarrow$ tomorrow is a Monday) is true.

(iii). If "today is a Sunday, then tomorrow is a Saturday" is not true.

(iiii). If "x = 5", then "2x = 10" is a true statement.

In other words, "x = 5" $\Rightarrow$ "2x = 10"

1.5.19. Example: Construct the truth table for $p \rightarrow p \vee q$

Solution:

p	q	$p \vee q$	$p \rightarrow p \vee q$
1	1	1	1
1	1	1	1
0	0	1	1
0	0	0	1

Truth Table for $p \rightarrow p \vee q$

Note that the truth value of the statement $p \rightarrow p \vee q$ is always 1.

1.5.20 Biconditional (or Double implication):

If p , q are statements, then the double implication of the statements p, q is a statement "p if and only if q" (denoted by $p \Leftrightarrow q$).

The truth value of $p \wedge q$ is true if "p" and "q" have the same truth values and is false if they have opposite truth values. The symbol "$\Leftrightarrow$" may be read as "if and only if". Note that $p \Leftrightarrow q$ is nothing but the statement:

"$(p \Rightarrow q)$ and $(q \Rightarrow p)$" (or $(p \Rightarrow q) \wedge (q \Rightarrow p)$)

The truth table for "Double implication" is given below

p	q	$p \Leftrightarrow q$
1	1	1
0	1	0
1	0	0
0	0	1

Truth table for $p \Leftrightarrow q$

1.5.21. Well formed formulas:

As we know a "statement formula" is an expression which is a sequence of variables, parentheses and connectives. Now we present a recursive definition for "statement formula". It may be often called as a well-formed formula.

A well-formed formula can be generated by the following way:

Rule-1: A statement symbol (or variable) is a well-formed formula.

Rule-2: If X is a well-formed formula then ~X is also a well formed formula.

Rule-3: If X and Y are well formed formulas, then $(x \vee y)$, $(x \wedge y)$, $(x \rightarrow y)$, $(x \Leftrightarrow y)$ are also well formed formulas.

We conclude that a string consisting of statement symbols, parenthesis, connectives is a well formed formula if it can be obtained by finitely many applications of the rules 1, 2 and 3 mentioned above.

Note that p, $\rceil p$, $p \vee q$, $p \wedge q$, $(7p) \vee q$, $p \rightarrow q$, $(p \rightarrow q) \rightarrow r$, $(p \rightarrow q) \vee (r \rightarrow q)$ are some well formed formulas.

1.5.22. The operation $\oplus$ or Δ :

There is an operation (which was not yet discussed, and used often) on statements (the operation is denoted by $\oplus$ or Δ).

This operation is called as "ring sum" (or "exclusive or"). $P \oplus Q$ is the exclusive or of the statements P and Q.

The rule is that $P \oplus Q$ is the proposition that is true when exactly one of P and Q is true, and is false otherwise.

The statement $P \oplus Q$ is also denoted by $P \;\Delta\; Q$.

The truth table for $P \oplus Q$ is given by

P	Q	$P \oplus Q$
1	0	1
0	1	1
1	1	0
0	0	0

1.5.23. Examples:

(i). Let P: $5 > 3$ and Q: $8 > 4$. Since both P and Q are true (1), we have that $P \oplus Q$ is false (0).

(ii). Let P: "$5 > 3$" and Q: "$3 > 5$". Since P is true and Q is false, we have that $P \oplus Q$ is true.

1.5.24 Example:

If p and q are two statements, then show that the statement $(p \uparrow q) \oplus (p \wedge q)$ is equivalent to $(p \vee q) \wedge (p \downarrow q)$.

Solution: In the following, we present the truth table for the given two compound statements.

Truth table

p (1)	q (2)	$p \leftarrow q$ (3)	$(p \rightarrow q) \oplus (p \wedge q)$ (4)	$p \vee q$ (5)	$p \leftarrow q$ (6)	$(p \vee q) \wedge (p \leftarrow q)$ (7)
T	T	F	F	T	F	F
T	F	T	F	T	F	F
F	T	T	F	T	F	F
F	F	T	F	F	T	F

Since the values in columns (4) and (7) are same, we have that the two given statements are equivalent.

1.6 SUMMARY:

Logic is a form of reasoning. The main object of the logic is to explain the rules by which one can determine the validity or to know the strength of any particular argument or reasoning. The rules are called rule of inference. We learnt basic terms of logic such as syllogism and truth values of the argument, etc.

1.7 TECHNICAL TERMS:

Truth values:

The primary statements have only one of the two possible values TRUE (T) or FALSE (F). TRUTH (T) by '1' and the truth value of FALSE (F) by '0'. T or F are called as truth values of the statement.

Syllogism:

It is an argument consisting of two propositions called premises and a third proposition called the conclusion.

1.8 SELF ASSESSMENT QUESTIONS:

1. Check whether the following arguments/statements are valid. If necessary, support your answer with reasons.
   (a). If 18 is divided by 3, the result is 5 (Ans: FALSE)
   (b). We use 3, 5, 6 to write 315. (Ans: FALSE)
   (c). The denominator of 5/9 is even (Ans: FALSE)

(d). The numerator of 5/9 is odd (Ans: TRUE)

2. Observe the argument.
   The denominator of 5/9 is odd
   10/8 is another name for 5/9.
   Therefore the denominator of 10/18 is odd. (Ans: Not valid)

3. All who ride by airplane were born after A.D.1800.
   Socrates rode by air plane.
   Therefore Socrates was born after A.D.1800.
   (Ans: argument is valid, but conclusion is false)

4. All monkeys are tree climbers
   All marmosets are monkeys
   Therefore all marmosets are tree climbers.
   (Ans: argument is valid)

5. (i). All bats are mammals
   (ii). All mammals have lungs
   Therefore (iii). All bats have lungs
   (Ans: That is a valid argument. But this argument contains false statement)

6. (i). If Einstein were the president then he would be famous
   (ii). Einstein is not the president
   Therefore (iii). Einstein is not famous.
   (Ans: This argument is clearly invalid. Here premises are true, but the conclusion is false)

7. (i). All whales are heavy
   (ii). All elephants are heavy
   Therefore (iii). All whales are elephants
   (Ans: This not a valid argument).

1.9 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.
2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.
3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237
4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Bhavanari Satyanarayana

LESSON -2

TYPES, EQUIVALENCES, IMPLICATIONS OF STATEMENTS

OBJECTIVE:

- To know types of Statements.
- To understand the Equivalence of Statements.
- To identify different types of notations
- To calculate truth values of the given statements.
- To have proper understanding of different implications.
- To develop skills in constructing the truth tables

STRUCTURE:

2.1. INTRODUCTION:

In this Lesson, we study the important types of Statements namely Tautology, Contradiction, and Contingency. Later, we study the Equivalences and Implications of different Statements.

2.2 SOME TYPES OF STATEMENTS:

Tautology and contradiction are two different types of statements, and are important concepts in the study of logic.

2.2.1 Tautology

Tautology is a statement expression which has truth value 'T' for all possible values of the statement variables involved in the expression.

2.2.2 Examples: (i). Show that $p \vee (\neg p)$ is a tautology.

(ii). Form the truth table for the statement $[p \wedge (p \vee q)] \vee \bar{p}$. Is it a tautology.

***Solution**: (i).*

p	~p	$p \vee (\neg p)$
1	0	1
0	1	1

Table for $p \vee (\neg p)$

Observe the above Table for $p \vee (7p)$. In all cases, the truth value of $p \vee (\neg p)$ is true, and so the statement $p \vee (\neg p)$ is a tautology.

(ii). The required truth table is given below:

p	q	$p \vee q$	$p \wedge (p \vee q)$	$\bar{p}$	$[p \wedge (p \vee q)] \vee \bar{p}$
0	0	0	0	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	0	1

Observing the above table we can conclude that the statement $[p \wedge (p \vee q)] \vee \bar{p}$ is always true, and so it is a tautology.

2.2.3. Contradiction.

A contradiction (or absurdity or Fallacy) is a statement expression whose truth value is always false.

2.2.4. Examples: (i). Show that the statement $p \wedge \sim p$ is a contradiction.

(ii). Show that $[p \wedge (p \vee q)] \wedge \bar{p}$ is a contradiction.

***Solution**: (i).*

p	~p	$p \wedge \sim p$
1	0	0
0	1	0

Truth table for $p \wedge \sim p$

Observe the above truth table for the statement $p \wedge \sim p$. It is clear that in all cases, the truth value of the statement $p \wedge \sim p$ is '0' (false), and so $p \wedge \sim p$ is a contradiction.

(ii). Now we write down the truth table

P	q	$p \vee q$	$p \wedge (p \vee q)$	$\bar{p}$	$[p \wedge (p \vee q)] \wedge \bar{p}$
0	0	0	0	1	0
0	1	1	0	1	0
1	0	1	1	0	0
1	1	1	1	0	0

Observing the table, we can conclude that $[p \wedge (p \vee q)] \wedge \bar{p}$ is always false. Hence $[p \wedge (p \vee q)] \wedge \bar{p}$ is a contradiction.

2.2.5. Contingency

A statement expression that is neither a tautology nor a contradiction is called a contingency.

2.2.6. Example: (i). Show that the statement $p \Leftrightarrow q$ is a contingency.

(ii). Prove that the statement "$(p \to q) \to (p \wedge q)$" is a contingency.

Solution: (i). Consider the following truth table for the given statement $p \Leftrightarrow q$.

p	q	$p \Leftrightarrow q$
1	1	1
0	1	0
1	0	0
0	0	1

Truth table for $p \Leftrightarrow q$

Since $p \Leftrightarrow q$ is not a tautology, and not a contradiction, we conclude that it is a contingency.

(ii). Let us observe the following truth table of given statement "$(p \to q) \to (p \wedge q)$".

p	q	$p \to q$	$(p \wedge q)$	$(p \to q) \to (p \wedge q)$
T	T	T	T	T
T	F	F	F	T
F	T	T	F	F
F	F	T	F	F

Truth table for $(p \to q) \to (p \wedge q)$

Clearly all the truth values of the given statement is neither "T" nor "F". Therefore the given statement is neither a tautology nor a contradiction, and so it is a contingency.

2.3 EQUIVALENCE OF STATEMENTS / FORMULAS:

In this section, we study the equivalence of the statements in the theory of logic.

2.3.1 Equivalent Statements:

Let n be a positive integer and $p_1, p_2, ...p_n...$ are n variables. Let A and B be two statements involving the n variables $p_1, p_2, ...p_n...$ We say A and B are equivalent if the truth values of A is equal to the corresponding truth values of B for every 2^n possible sets of truth values assigned to $p_1, p_2, ...p_n$. If A is equal to B, then this fact is denoted by $A \Leftrightarrow B$. In other words $A \Leftrightarrow B$ is a tautology.

2.3.2. Examples: (i). Prove that the statements $p \Rightarrow q$ and $\wr p \vee q$ are equivalent.

(ii). Show that $\sim(\sim p)$ is equivalent to P.

***Solution**: (i). Write* $A:(p \Rightarrow q)$, and B: $\wr p \vee q$. Now we have to verify that $A \Leftrightarrow B$.

Observe the truth table for the statements A and B which is given below.

p	q	~ p	$p \Rightarrow q$	$\sim p \vee q$
T	T	F	T	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Truth table for A & B

We observe that the truth values of A and B are equal in all cases. So the statement A is equivalent to statement B. In other words, $A \Leftrightarrow B$ is a tautology.

(ii). Let us consider the two statements A: p, B: ~ (~p) .

Observe the truth table for the statements A & B which is given below

p	~ p	~ (~ p)
T	F	T
T	F	T
F	T	F
F	T	F

Truth table for statements p and ~ (~p).

Observe the truth table. We understand that A: p, B: ~ (~p) are equivalent.

2.3.3. Equivalent formulas (One can verify the following statements through truth tables).

1. $\left.\begin{array}{l} p \vee p \Leftrightarrow p \\ p \wedge p \Leftrightarrow p \end{array}\right\}$ The **Idempotent** laws (with respect to $\vee$ and $\wedge$).

2. $\left.\begin{array}{l} p \vee (q \vee r) \Leftrightarrow (p \vee q) \vee r \\ p \wedge (q \wedge r) \Leftrightarrow p \wedge (q \wedge r) \end{array}\right\}$ The **Associative** laws (with respect to $\vee$ and $\wedge$).

3. $\left.\begin{array}{l} p \vee q \Leftrightarrow q \vee p \\ p \wedge q \Leftrightarrow q \wedge p \end{array}\right\}$ The **Commutative** laws (with respect to $\vee$ and $\wedge$).

4. $\left.\begin{array}{l} p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r) \\ p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \wedge r) \end{array}\right\}$ The **Distributive** laws (with respect to $\vee$ and $\wedge$).

5. $\begin{array}{l} P \vee F \Leftrightarrow P \\ P \wedge F \Leftrightarrow F \end{array}$

6. $\begin{array}{l} P \vee T \Leftrightarrow T \\ P \wedge T \Leftrightarrow P \end{array}$

7. $P \vee (\sim P) \Leftrightarrow T.$

$P \wedge (\sim P) \Leftrightarrow F$ The **Complement** laws (with respect to $\vee$ and $\wedge$).

8. $\left.\begin{array}{l} p \vee (p \wedge q) \Leftrightarrow p \\ p \wedge (p \vee q) \Leftrightarrow p \end{array}\right\}$ The **Absorption** laws (with respect to $\vee$ and $\wedge$).

9. $\sim (P \vee Q) \Leftrightarrow (\sim P \wedge \sim Q)$

$\sim (P \wedge Q) \Leftrightarrow (\sim P \vee \sim Q)$ **De Morgan's** laws (with respect to $\vee$ and $\wedge$).

2.3.4. Example: (i). Show that (not through the truth tables) the two statements

$[(p \Rightarrow q) \wedge (p \Rightarrow r)]$ and $[p \Rightarrow (q \wedge r)]$ are equivalent. In other words prove that

$$[(p \Rightarrow q) \wedge (p \Rightarrow r)] \text{ if and only if } [p \Rightarrow (q \wedge r)]$$

Solution: $[(p \Rightarrow q) \wedge (p \Rightarrow r)] \Leftrightarrow$

$(\sim p \vee q) \wedge (p \Rightarrow r)$ [since $(a \Rightarrow b) \Leftrightarrow (\sim a \vee b)$ (refer Example 1.3.2. (i))] .

$\Leftrightarrow (\sim p \vee q) \wedge (\sim p \vee r)$ [since $(a \Rightarrow b) \Leftrightarrow (\sim a \vee b)$ (refer Example 1.3.2. (i))]

$\Leftrightarrow \sim p \vee (q \wedge r)$ (by Distributive Law)

$\Leftrightarrow p \Rightarrow (q \wedge r)$ [since $(a \Rightarrow b) \Leftrightarrow (\sim a \vee b)$ (refer Example 1.3.2. (i))].

This completes the solution.

2.3.5. Example:

Let n be a fixed positive integer. Consider the two statements:

p: n is an even number

q: n + 1 is an odd number

Show that p and q are equivalent

Solution: *Let* p be a true statement. Then n is an even number. Since n is even, it is clear that (n + 1) is an odd number. Hence q is true.

Similarly if q is true, then (n + 1) is an odd number and so n = (n + 1) –1 is even. Hence p is true. So we conclude that $p \Leftrightarrow q$ is true. In other words, p and q are two different and equivalent statements.

2.4 DUALITY AND TAUTOLOGICAL IMPLICATION:

2.4.1. Dual Statement:

Let A and B be any two formulas. Then A is said to be the dual of B, if A be obtained from B by replacing "$\wedge$" by "$\vee$" and "$\vee$" by "$\wedge$". It is clear that if A is the dual of B, then B is the dual of A. Note that (i). the connectives "$\wedge$" and "$\vee$" are dual each other; and the dual of the value "T" is "F", and the dual of F is T.

2.4.2. Examples:

(i) "$(p \wedge q) \vee r$" is the dual of "$(p \vee q) \wedge r$"

(ii) $(p \vee q) \wedge T$ is the dual of $(p \wedge q) \vee T$.

2.4.3. Tautological Implications

A statement A is said to tautologically imply a statement B if and only if $A \to B$ is a tautology.

This fact is denote by (same as) $A \Rightarrow B$, we read it as "A implies B". In other words, $A \Rightarrow B$ states that "$A \to B$ is a tautology" or "A tautologically implies B".

2.4.4. Note: The connectivities $\wedge$, $\vee$ are symmetric in the sense that

$p \wedge q \Leftrightarrow q \wedge p$

$p \vee q \Leftrightarrow q \vee p$

2.4.5. Converse: For any statement formula $p \to q$, the statement formula $q \to p$ is called as the converse of the statement $p \to q$, .

2.4.6. Inverse (or opposite): For any statement formula $p \to q$, the statement formula $\sim p \to \sim q$ is called the inverse (or opposite) of $p \to q$.

2.4.7. Contrapositive

For any statement formula $p \to q$, the statement formula $\sim q \to \sim p$ is called as the Contrapositive of $p \to q$.

2.4.8. Note: For the convenience of the reader we provide the concepts: Converse, inverse, and contrapositive, in one table.

Given statement $p \to q$	Converse $q \to p$
Inverse (or opposite) $\sim p \to \sim q$ (equivalent to the converse)	Contrapositive $\sim q \to \sim p$ (equivalent to the implication)

2.4.9. Example: Write down the Contrapositive of the following statement.

"If Rama have Rs.100/- with him, then he will spend Rs. 50/- for his friend Krishna".

***Solution*:** write p: "Rama have Rs. 100/- with him"

q: "Rama spend Rs. 50/- for his friend krishna"

Given statement is “ $p \to q$ ”.

We know that the Contrapositive of “ $p \to q$ ” is $\sim q \to \sim p$.

It is clear that $\sim q$: “Rama does not spend Rs.50/- for his friend Krishna”

$\sim p$: Rama does not have Rs. 100/- with him.

So the required statement is as follows:

If “Rama does not spend Rs. 50/- for his friend Krishna” then “Rama does not have Rs. 100/-with him”.

2.4.10. Some Implications

The following implications have importance in proving further statements. All of them can be proved by using truth tables or by any other methods in study.

$p \wedge q \Rightarrow p$ .....(1.1)

$(p \vee q) \Rightarrow q$ .....(1.2)

$p \Rightarrow (p \vee q)$ .....(1.3)

$\sim p \Rightarrow (p \to q)$ .....(1.4)

$q \Rightarrow (p \to q)$ .....(1.5)

$\sim (p \to q) \Rightarrow p$ .....(1.6)

$\sim (p \to q) \Rightarrow \sim q$ .....(1.7)

$p \wedge (p \to q) \Rightarrow \sim p$ .....(1.8)

$\sim p \wedge (p \vee q) \Rightarrow q$ .....(1.9)

$(p \to q) \wedge (q \to r) \Rightarrow (p \to r)$ .....(1.10)

$(p \vee q) \wedge (p \to r) \wedge (q \to r) \Rightarrow r$ .....(1.11)

$\sim q \wedge (p \to q) \Rightarrow \sim p$ .....(1.12)

2.4.11 Example:

Construct the truth tables for converse, inverse and Contrapositive of the statement $(p \to q)$.

Solution: Given statement is “ $p \to q$ ”.

(i) The truth table of the converse $(q \to p)$of the proposition “$p \to q$” is as follows:

p	Q	(p→q)	q→p
1	1	1	1
1	0	0	1
0	0	1	0
0	0	1	1

Truth table for proposition "$q \rightarrow p$" (Converse of proposition "$p \rightarrow q$")

(ii) The Truth table for the inverse ($\sim p \rightarrow \sim q$) of the proposition "$p \rightarrow q$" is as follows:

p	q	~ p	~ q	p→q	$\wr p \uparrow \wr q$
1	1	0	0	1	1
1	0	0	1	0	1
0	1	1	0	1	0
0	0	1	1	1	1

Truth table for the proposition "$\sim p \rightarrow \sim q$"
(Inverse proposition of proposition "$p \rightarrow q$")

(iii) The truth table for the Contrapositive ($\sim q \rightarrow \sim p$) of the proposition "$p \rightarrow q$" is as follows:

p	q	~ p	~ q	p→q	$\wr q \uparrow \wr p$
1	1	0	0	1	1
1	0	0	1	0	0
0	1	1	0	1	1
0	0	1	1	1	1

2.5 SUMMARY:

In this lesson some types of Statements namely: Tautology, contradiction and Contingency were studied along with some examples. Later equivalence of Statements or Formulas were introduced. For better understanding of the reader some examples were provided. Finally, the concepts Converse, inverse, and contrapositive of the given statements were introduced and the truth tables related to these concepts were calculated for the convenience of the reader.

2.6 TECHNICAL TERMS:

Tautology
(Tautology is a statement expression which has truth value 'T' for all possible values of the statement variables involved in the expression.)

Contradiction (A contradiction (or absurdity or Fallacy) is a statement expression whose truth value is always false).

Contingency
(A statement expression that is neither a tautology nor a contradiction is called a contingency).

Converse:
(For any statement formula $p \rightarrow q$, the statement formula $q \rightarrow p$ is called as the converse of the statement $p \rightarrow q$). .

Inverse (or opposite):
(For any statement formula $p \rightarrow q$, the statement formula $\sim p \rightarrow \sim q$ is called the inverse (or opposite) of $p \rightarrow q$).

Contrapositive

(For any statement formula $p \rightarrow q$, the statement formula ~q → ~p is called as the Contrapositive of $p \rightarrow q$).

2.7 SELF ASSESSMENT QUESTIONS:

(i). Write down the state table for the statement $[p \wedge (p \vee q)] \vee \overline{p}$ and find out whether it is tautology or contradiction or contingency (Ans: tatualogy)

(ii). Let m be a fixed positive integer. Consider the two statements:

a: m is an odd number

b: m + 1 is an even number

Show that a and b are equivalent

(iii). Write down the Contrapositive of the following statement.

"If Lakshmana have Rs.200/- with him, then he will spend Rs. 150/- for his brother Bharatha".

(iv). Construct the truth tables for converse, inverse and Contrapositive of the statement (p→q).

2.8 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd. New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.
2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.
3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237
4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Bhavanari Satyanarayana

LESSON - 3
NORMAL FORMS

OBJECTIVE:

- ❖ To understand Normal Forms.
- ❖ To obtain Normal Form of a given expression.
- ❖ To identify different types of notations
- ❖ To understand some technique in forming the truth tables.

STRUCTURE

3.1. INTRODUCTION

In this Lesson, we study some important concepts: Normal Forms, Disjunctive Normal forms, Conjunctive Normal Forms, Principle disjunctive Normal Forms, Principle Conjunctive Normal Forms. For the convenience of readers we included necessary examples.

3.2. NORMAL FORMS

Suppose that n is a positive integer, $P_1, P_2,...P_n$ are the atomic statements (or variables) and A $(P_1, P_2,...P_n)$ is a statement formula. We know that each P_i have truth value T (or 1) or F (or 0). Hence the truth table for $(P_1, P_2,...P_n)$ have 2^n values. So we can form the truth table for $A(P_1, P_2,...P_n)$ with 2^n rows.

If for all 2^n values of $(P_1, P_2,...P_n)$ the truth value of $A(P_1, P_2,...P_n)$ is T (or 1) then the statement formula is said to be identically true. In other words, we say that $A(P_1, P_2,...P_n)$ is a tautology.

If for all 2^n values of $(P_1, P_2,...P_n)$ the truth value of $A(P_1, P_2,...P_n)$ is F (or 0) then the statement formula $A(P_1, P_2,...P_n)$ is said to be identically false. In other words, we, we say that $A(P_1, P_2,...P_n)$ is a contradiction.

If the truth value of $A(P_1, P_2,...P_n)$ is True (T or 1) for atleast one of the truth values of $(P_1, P_2,...P_n)$ then $A(P_1, P_2,...P_n)$ is said to be satisfiable.

3.2.1. Decision Problem

Suppose a statement formula is given, and we have to find whether it is a tautology or contradiction of satisfiable. This problem of determining (in a finite number of steps) whether the given statement formula is a tautology (or) a contradiction (or) satisfiable is named as a decision problem.

So every decision problem in the statement calculus has a solution because we can decide this by forming a truth table for the given statement formula.

Now we study different forms (of a given statement formula) called as normal forms

(i). Disjunctive Normal Form (in short, DNF)

(ii). Conjunctive Normal Form (in short, CNF)

(iii). Principal Disjunctive Normal Form (in short, PDNF)

(iv). Principal Conjunctive Normal Form (in short, PCNF)

3.2.2. Disjunctive Normal Form (DNF or D.N.F or dnf)

Let $X_1, X_2,...X_n$ be n given atomic variables and $\overline{X}_1,\overline{X}_2,...\overline{X}_n$ (or $\sim X_1, \sim X_2,... \sim X_n$) are the negations of $X_1, X_2,...X_n$ respectively.

Product (or meet, $\wedge$) of some elements from $\{X_1, X_2...X_n, \overline{X}_1, \overline{X}_2...\overline{X}_n\}$ is called as an elementary product.

Sum (or join, $\vee$) of some elements from $\{X_1, X_2...X_n, \overline{X}_1, \overline{X}_2...\overline{X}_n\}$ is called as elementary sum.

For example, $X_i, \overline{X}_i \wedge X_j, \overline{X}_1 \wedge X_3X_4, X_1 \wedge X_2 \wedge ... \wedge X_n, X_1 \wedge X_2 \wedge \overline{X}_3 \wedge X_4 \wedge \overline{X}_5$ are some elementary products.

$\overline{X}_1, X_1 \vee X_2, \overline{X}_3 \vee X_4 \vee X_5, X_1 \vee X_2 \vee ... \vee X_n, \overline{X}_1 \vee \overline{X}_2 \vee X_3 \vee \overline{X}_4 \vee X_5$ are some elementary sums.

A statement formula which is equivalent to a given statement formula and which is of the form "a sum of elementary products" is called as Disjunctive Normal Form (DNF) of the given statement formula.

3.2.3 How to find DNF

Suppose the given statement formula is $A(P_1, P_2,...P_n)$, and we wish to find its DNF.

If it is already in the form: sum of elementary products then it is already in DNF.

If it is not in the form of DNF then we use some known results or formulas or axioms step by step to get DNF. Most of the cases when '$\rightarrow$' presents, we use the known formula / result:

$P \rightarrow Q \Leftrightarrow 7\ P \vee Q$. [that is $\overline{P} \vee Q$].

Also we use known laws like: distributive laws, demorgan laws, Commutative and associative laws, and so on. One can observe in the next coming example.

3.2.4. Example: Find DNF of X ∧ (X→Y)

Solution: $X \wedge (X \rightarrow Y)$

$\Leftrightarrow X \wedge (7X \vee Y)$ [Since P→Q ⇔ 7 P ∨ Q, a known result]

$\Leftrightarrow [X \wedge (7X)] \vee [X \wedge Y]$ [by distributive law]

Now $[X \wedge 7X] \vee [X \wedge Y]$ is the sum of two elementary product terms $X \wedge 7X$ and $X \wedge Y$. Hence $[X \wedge 7X] \vee [X \wedge Y]$ is the DNF of the given statement formula $X \wedge (X \rightarrow Y)$.

3.2.5. Conjunctive Normal Form (CNF or C.N.F. or c.n.f. or cnf):

A statement formula which is equivalent to a given statement formula and which is of the form "a product of elementary sums" is called as Conjunctive Normal Form (CNF) of the given statement formula.

3.2.6. Example: Find CNF for the statement formula given by X ∧ (X→Y)

Solution:

$X \wedge (X \rightarrow Y)$

$\Leftrightarrow X \wedge (\overline{X} \vee Y)$ [Since P→Q ⇔ 7 P ∨ Q, a known result].

The obtained form $X \wedge (\overline{X} \vee Y)$ is a product of two sums: X and $\overline{X} \vee Y$.

So it is in CNF. Hence $X \wedge (\overline{X} \vee Y)$ is a CNF for the given statement $X \wedge (X \rightarrow Y)$.

3.2.7. Principal Disjunctive Normal Form (PDNF) (or sum of products canonical form).

Suppose $P_1, P_2, ..., P_n$ are n statement variables. The expression $P_1^* \wedge P_2^* \wedge ... \wedge P_n^*$ where P_i^* is either P_i or $\sim P_i$ is called a minterm. It is clear that there exist 2^n minterms.

The expression $P_1^* \vee P_2^* \vee ... \vee P_n^*$, where P_i^* is either P_i or $\sim P_i$ is called a maxterm. It is clear that there exist 2^n maxterms.

Let P, Q, R be the three variables.

Then the minterms are:

$P \wedge Q \wedge R$, $P \wedge Q \wedge \sim R$, $P \wedge \sim Q \wedge R$, $P \wedge \sim Q \wedge \sim R$,

$\sim P \wedge Q \wedge R$, $\sim P \wedge Q \wedge \sim R$, $\sim P \wedge \sim Q \wedge R$, $\sim P \wedge \sim Q \wedge \sim R$,

For a given statement formula, an equivalent statement formula which is in the form "disjunction (or sum or join) of minterms" is known as its Principal Disjunctive Normal Form (PDNF) (or sum of products canonical form).

3.2.8. Example: Find PDNF for $(\overline{X} \vee Y)$.

Solution: $\overline{X} \vee Y \Leftrightarrow (\overline{X} \wedge 1) \vee (Y \wedge 1)$ (since A $\wedge$ 1 = A for all A)

$\Leftrightarrow [\overline{X} \wedge (Y \vee \overline{Y})] \vee [Y \wedge (X \vee \overline{X})]$ (since $A \vee \overline{A} = 1$)

$\Leftrightarrow [(\overline{X} \wedge Y) \vee (\overline{X} \wedge \overline{Y})] \vee [(Y \wedge X) \vee (Y \wedge \overline{X})]$ (by distributive law)

$\Leftrightarrow (\overline{X} \wedge Y) \vee (\overline{X} \wedge \overline{Y}) \vee (X \wedge Y) \vee (\overline{X} \wedge Y)$ [By commutative law].

$\Leftrightarrow (\overline{X} \wedge Y) \vee (\overline{X} \wedge \overline{Y}) \vee (X \wedge Y)$ (since $A \vee A = A$, idempotent law)

Now we got the form $(\overline{X} \wedge Y) \vee (\overline{X} \wedge \overline{Y}) \vee (X \wedge Y)$ which is the PDNF for $(\overline{X} \wedge Y)$.

3.2.9. Note: (i). To find the PDNF of the given statement formula, there is a method named as 'Black box method'.
For convenience we use ~p or $\overline{P}$ to denote negation of the statement p·

(ii). If there are three atomic variables p, q, r, then we use the notation given by the following table:

Binary Notation			Expression		
0	0	0	$\overline{p}$	$\overline{q}$	$\overline{r}$
0	0	1	$\overline{p}$	$\overline{q}$	r
0	1	0	$\overline{p}$	q	$\overline{r}$
0	1	1	$\overline{p}$	q	r
1	0	0	p	$\overline{q}$	$\overline{r}$
1	0	1	p	$\overline{q}$	r
1	1	0	p	q	$\overline{r}$
1	1	1	p	q	r

We understand that $\overline{p}\,\overline{q}\,\overline{r}$ is the product term (or related expression) for 000; $pq\overline{r}$ is the related expression for 110.

(iii). Statements and related equivalent binary forms for three statement variables P, Q, R given below.

P $\wedge$ Q $\wedge$ R (or PQR, the product of P, Q, R) (equivalent binary notation is 111),

P $\wedge$ Q $\wedge$ ~ R (binary notation is 110), P $\wedge$ ~ Q $\wedge$ R (binary notation is 101),

P $\wedge$ ~ Q $\wedge$ ~ R (binary notation is 100), ~ P $\wedge$ Q $\wedge$ R (011),

~ P $\wedge$ Q $\wedge$ ~ R (010), ~ P $\wedge$ ~ Q $\wedge$ R (001), ~ P $\wedge$ ~ Q $\wedge$ ~ R (000),

3.2.10. Black Box Method (to find PDNF)

Suppose that $A(X_1, X_2, ...X_n)$ is the given statement formula where $X_1, X_2, ...X_n$ are atomic statement variables and each atomic statement variable may attain its value either 0 or 1 (that is, False or True).

Form the truth table for $A(X_1, X_2, ...X_n)$ which contains 2^n rows.

This truth table determines the PDNF, simply by adding all the product terms that occurs when $A(X_1, X_2, ...X_n)$ takes Value 1.

3.2.11. Example:

Find PDNF for the given statement "$P \to Q$" by Black Box Method (or by using truth tables)

Solution: First we form the truth table for $P \to Q$.

P	Q	$P \to Q$
T	T	T
T	F	F
F	T	T
F	F	T

Or

P	Q	$P \to Q$
1	1	1
1	0	0
0	1	1
0	0	1

Observe the column under $P \to Q$, there are three 1's in the column.

The 1's are in 1st row, 3rd row and forth row only.

Consider the 1st row. In this first row P & Q have truth values 1 and 1 respectively. So the related product term is PQ (or 11).

Consider the 3rd row. In the third row, the truth values of P and Q are 0 and 1 respectively.

So the related product term is $\overline{P}.Q$ (or 01).

Consider the 4th row. In this forth row, the truth values of P and Q are 0 and 0 respectively.

So the related product term is $\overline{P}\,\overline{Q}$ (or 00).

The sum of these three terms, that is "11v 01v 00" is the PDNF.

So the required PDNF is $PQ \vee \overline{P}Q \vee \overline{P}\,\overline{Q}$ (in detail $(P \wedge Q) \vee (\overline{P} \wedge Q) \vee (\overline{P} \wedge \overline{Q})$).

3.2.12. Example: Find PDNF for the given statement $(\overline{X} \vee Y)$ (by Black Box Method)

(Compare this problem with example 1.6.8)

Solution: First we form truth table for the given statement $(\overline{X} \vee Y)$.

X	Y	$\overline{X}$	$\overline{X} \vee Y$
0	0	1	1
0	1	1	1
1	0	0	0
1	1	0	1

Truth table for $(\overline{X} \vee Y)$

Observe the column for $\left(\overline{X} \vee Y\right)$. There are three 1's in first row, second row and forth row.

The product term related to row–1 is $\overline{X}\ \overline{Y}$.

Consider row–2. The product term related to this row–2 is $\overline{X}\ Y$ (because the truth values of x and y are 0 and 1 respectively).

Consider row–4. The product term related to row–4 is X Y (because the truth values of x and y are 1 and 1 respectively)

Therefore the PDNF is

$$\overline{X}\ \overline{Y} \vee \overline{X}\ Y \vee X\ Y \text{ or } \left(\overline{X} \wedge \overline{Y}\right) \vee \left(\overline{X} \wedge Y\right) \vee \left(X \wedge Y\right)$$

3.2.13. Principal Conjunctive Normal Form (PCNF)

For a given statement formula, an equivalent formula consisting of the conjunction (product) of maxterms is known as the Principal Conjunctive Normal Form (PCNF) (or product of sums canonical form).

To find PCNF (by using truth table or through PDNF) of the given statement formula p.

Step 1: Suppose the given statement formula is p.

Step 2: Find the complement (that is the negation) of p.

Step 3: Find the PDNF for the complement of p (we may use black box method or some other method)

Step 4: Required PCNF of p = complement of (PDNF of (complement of p))

$$= \sim (\text{PDNF}(\sim p)).$$

3.2.14. Example: Find the PCNF of the given statement formula $F = X \wedge \overline{Y}$.

Solution: We follow the method given in 3.2.13.

Step 1: The given statement formula is

$$F = \left(X \wedge \overline{Y}\right)$$

Step 2: Now we have to find $\overline{F}$ (the complement (or the negation) of F).

$\sim F = \sim (X \wedge \overline{Y}) = (\sim X) \vee Y.$

Step 3: In this step we find the PDNF for $\overline{F} = \overline{X} \vee Y$

X	Y	$\overline{X}$	$\overline{X} \vee \overline{Y}$
0	0	1	1

0	1	1	1
1	0	0	0
1	1	0	1

Truth table for $(\overline{X} \vee Y)$

As in the above example 3.2.11, we get that

PDNF of $(\overline{F}) = (\overline{X} \wedge \overline{Y}) \vee (\overline{X} \wedge Y) \vee (X \wedge Y)$

Step 4: $\text{PCNF}(F) = \overline{\text{PDNF}(\overline{F})}$

$$= \overline{(\overline{X} \wedge \overline{Y}) \vee (\overline{X} \wedge Y) \vee (X \wedge Y)}$$

$$= \overline{(\overline{X} \wedge \overline{Y})} \wedge \overline{(\overline{X} \wedge Y)} \wedge \overline{(X \wedge Y)}$$

(by demorgan laws)

$$= (X \vee \overline{Y}) \wedge (X \vee \overline{Y}) \wedge (\overline{X} \vee \overline{Y})$$

$$= (X \vee \overline{Y}) \wedge (\overline{X} \vee \overline{Y})$$

(since $A \wedge A = A$, called as idempotent law).

3.2.15. Example

Show that the principal conjunctive normal form (PCNF) of the formula $[p \rightarrow (q \wedge r)] \wedge [\overline{p} \rightarrow (\overline{q} \wedge \overline{r})]$ is π (1, 2, 3, 4, 5, 6) (Here we use standard notation π for product).

Solution: Given formula is

$$[p \rightarrow (q \wedge r)] \wedge [\overline{p} \rightarrow (\overline{q} \wedge \overline{r})]$$

$$\Leftrightarrow [\overline{p} \vee (q \wedge r)] \wedge [\overline{p} \rightarrow (\overline{q} \wedge \overline{r})] \quad \text{[Since } p \rightarrow q \Leftrightarrow \sim p \vee q]$$

$$\Leftrightarrow [\overline{p} \vee (q \wedge r)] \wedge [p \vee (\overline{q} \wedge \overline{r})] \quad \text{[Since } p \rightarrow q \Leftrightarrow \sim p \vee q]$$

$$\Leftrightarrow [(\overline{p} \vee q) \wedge (\overline{p} \vee r)] \wedge [(p \vee \overline{q}) \wedge (p \vee \overline{r})] \quad \text{[By Demorgan laws]}$$

$$\Leftrightarrow [\overline{p} \vee q \vee (r \wedge \overline{r})] \wedge [\overline{p} \vee r \vee (q \wedge \overline{q})] \wedge [p \vee \overline{q} \vee (r \wedge \overline{r})] \wedge [p \vee \overline{r} \vee (q \wedge \overline{q})]$$

[since $x \wedge (\sim x) = o$, and $x \vee o = x$]

$$\Leftrightarrow [\overline{p} \vee q \vee r] \wedge [\overline{p} \vee q \vee \overline{r}] \wedge [\overline{p} \vee r \vee q] \wedge [\overline{p} \vee r \vee \overline{q}] \wedge [p \vee \overline{q} \vee r] \wedge [p \vee \overline{q} \vee \overline{r}]$$

$$\wedge [p \vee \overline{r} \vee q] \wedge [\overline{p} \vee \overline{r} \vee \overline{q}]$$

$\Leftrightarrow (\overline{p} \vee q \vee r) \wedge (\overline{p} \vee q \vee \overline{r}) \wedge (\overline{p} \vee \overline{q} \vee r) \wedge (p \vee \overline{q} \vee r) \wedge (p \vee q \vee \overline{r}) \wedge (p \vee \overline{q} \vee \overline{r})$ is the principal conjunctive normal form of given statement formula.

Now by the known representation, we have

1. Can be represented as $\bar{p}\,\bar{q}\,r$ (001)
2. Can be represented as $\bar{p}\,q\,\bar{r}$ (010)
3. Can be represented as $\bar{p}\,q\,r$ (011)
4. Can be represented as $p\,\bar{q}\,\bar{r}$ (100)
5. Can be represented as $p\,\bar{q}\,r$ (101)
6. Can be represented as $p\,q\,\bar{r}$ (110).

In Binary notation, 001 stands for 1 (because if xyz is a binary form, then the equivalent number (with respect to 10) is [(x multiplied by 4) + (y multiplied by 2) + (z multiplied by 1)].

For binary number 010, is equal to 2; 011 is equal to 3; 100 is equal to 4; 101 is equal to 5; 110 is equal to 6.

Therefore, the principal conjunctive normal form of the given formula can be represented as π (1, 2, 3, 4, 5, 6).

The Solution is complete.

3.3 SUMMARY:

In this lesson, we studied some important forms of statement formulas namely Disjunctive Normal form (DNF), Conjunctive Normal form (CNF), Principal Disjunctive Normal form (PDNF), Principal Conjunctive Normal form (PCNF).We included sufficient number of examples for the training of the readers.

3.4 TECHNICAL TERMS:

Disjunctive Normal Form (DNF):

A statement formula which is equivalent to a given statement formula and which is of the form "a sum of elementary products" is called as Disjunctive Normal Form (DNF) of the given statement formula.

Conjunctive Normal Form (in short, CNF):

A statement formula which is equivalent to a given statement formula and which is of the form "a product of elementary sums" is called as Conjunctive Normal Form (CNF) of the given statement formula.

Principal Disjunctive Normal Form (in short, PDNF):

For a given statement formula, an equivalent statement formula which is in the form "disjunction (or sum or join) of minterms" is known as its Principal Disjunctive Normal Form (PDNF) (or sum of products canonical form).

Principal Conjunctive Normal Form (in short, PCNF):

For a given statement formula, an equivalent formula consisting of the conjunction (product) of maxterms is known as the Principal Conjunctive Normal Form (PCNF) (or product of sums canonical form).

3.5 SELF ASSESSMENT QUESTIONS:

(i). Find the principal conjunctive normal Form of the statement formula $(\sim p) \wedge q$.

[Ans: $(\sim p \vee q) \wedge (\sim p \vee q) \wedge (p \vee q)$]

(ii). Find the principal conjunctive normal Form of the statement formula $\sim (p \vee q)$

. [Ans: $(p \vee \sim q) \wedge (\sim p \vee q) \wedge (\sim p \vee \sim q)$] of

3.6 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd, New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237 .

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Bhavanari Satyanarayana

LESSON- 4

THEORY OF INFERENCE, AND PREDICATE LOGIC

OBJECTIVE:

- ❖ To know the concept Inference Theory.
- ❖ To understand the Meaning of Predicate, 2-place predicate.
- ❖ To identify different types of Predicates
- ❖ To have proper understanding of different connectives.
- ❖ To develop skills in solving the problems
- ❖ To Learn the consistency, validity of the statements.

STRUCTURE

4.1 INTRODUCTION:

In Lessons 1,2 and 3, we studied atomic statements and statement formulas. In this Lesson, we study: Theory of Inference for Statement Calculus; Consistency of premises and indirect method of proof; Predicates; m-place Predicates; and Connectives, Statement Functions, and Variables. In the inference theory, all the premises and conclusions are statements. If any two statements have common feature, then we are unable to express the common feature. In order to study the common feature statements, the concept “predicate” is useful. The logic related to the analysis of predicates is called as predicate logic.

4.2. THEORY OF INFERENCE FOR STATEMENT CALCULUS:

Logic gives the rules of inference, or principles of reasoning. The theory deal with these rules is called as inference theory. This theory is concerned with the inferring of a a statement (called as conclusion) from the given hypothesis (or certain statements, called as premises)

The process of deriving the conclusion from the set of given statements (or premises) by using the accepted rules, and known results, is known as deduction or a formal proof. In the formal proof, at any stage, the rule of inference used in the derivation may be acknowledged.

The conclusion obtained by using the rules of inference is named as Valid Conclusion; and the argument involved is named as Valid Argument.

4.2.1. Tautology

If A and B are two statement formulas, then we say that "B logically follows from A" (or "B is a valid conclusion (or consequence) of the premise A") if and only if $A \to B$ is a tautology (that is, $A \Rightarrow B$).

4.2.2. Validity using truth table

Let m, n be positive integers. Suppose $P_1, P_2, \ldots, P_n$ are n variables appearing in the m premises $H_1, H_2, \ldots, H_m$ and in the conclusion C.

Suppose that all the possible combinations of truth values are assigned to $P_1, P_2, \ldots, P_n$ and also suppose that the truth values of $H_1, H_2, \ldots, H_m$ and C are entered in the truth table.

We say that C follows logically from the premises $H_1, H_2, \ldots, H_m$ if and only if

$H_1 \wedge H_2 \wedge \ldots H_m \Rightarrow C$.

This can be checked from the truth table using the following procedure:

1. Observe the rows in which C has the truth value F.
2. In every such row (that is the row for which the value under C is F) if at least one of the values of $H_1, H_2, \ldots, H_m$ is F then the conclusion is valid.

4.2.3. Example: Show that the conclusion C: ~P follows from the premises

H_1: ~ P ∨ Q, H_2: ~ (Q ∧ ~ R) and H_3: ~ R.

Solution: Given Conclusion and premises are C: ~ P, H_1: ~ P ∨ Q, H_2: ~ (Q ∧ ~ R)

and H_3: ~ R.

P	**Q**	**R**	**H_1**	**H_2**	**H_3**	**C**
1	1	1	1	1	0	0
1	1	0	1	0	1	0
1	0	1	0	1	0	0
1	0	0	0	1	1	0
0	1	1	1	1	0	1
0	1	0	1	0	1	1
0	0	1	1	1	0	1
0	0	0	1	1	1	1

The rows (1, 2, 3, and 4) in which C has the truth values 0 (that is, F) has the situation that at least one of H_1, H_2, H_3 has truth value F. Thus C logically follows form the premises H_1, H_2, and H_3.

4.2.4. Rules of Inference

In the following, we mention the three rules of inference.

Rule P: A premise may be introduced at any point in the derivation.

Rule T: A formula S may be introduced in a derivation if S is tautologically implied by any one or more of the preceding formulas in the derivation.

Rule CP: If we can derive S and R and a set of premises then we can derive $R \to S$ from the set of premises alone.

4.2.5. Some Implications were listed in the following:

I_1 : $p \wedge q \Rightarrow p$ (Simplification)

I_2 : $p \wedge q \Rightarrow q$

I_3 : $p \Rightarrow p \vee q$ (addition)

I_4 : $q \Rightarrow p \vee q$

I_5 : $\bar{p} \Rightarrow p \to q$

I_6 : $q \Rightarrow p \to q$

I_7 : $\overline{p \to q} \Rightarrow p$

I_8 : $\overline{p \to q} \Rightarrow \bar{q}$

I_9 : $p, q \Rightarrow p \wedge q$

I_{10} : $\bar{p},\ p \vee q \Rightarrow q$ [disjunctive syllogism]

I_{11} : $p, p \to q \Rightarrow q$ [modus ponens]

I_{12} : $\bar{q},\ p \to q \Rightarrow \bar{p}$ [modus tollens]

I_{13} : $p \to q\ ,\ q \to r \Rightarrow p \to q$ [hypothetical syllogism]

I_{14} : $p \vee q, p \to r, q \to r \Rightarrow r$ [dilemma]

4.2.6. Some Equivalences

E_1 : $\bar{\bar{p}} \Leftrightarrow p$ [double negation]

E_2 : $p \wedge q \Leftrightarrow q \wedge p$

E_3 : $p \vee q \Leftrightarrow q \vee p$

E_4 : $(p \wedge q) \wedge r \Leftrightarrow p \wedge (q \wedge r)$

E_5 : $(p \vee q) \vee r \Leftrightarrow p \vee (q \vee r)$

E_6 : $p \wedge (q \vee r) \Leftrightarrow (p \wedge q) \vee (p \wedge r)$

E_7 : $p \vee (q \wedge r) \Leftrightarrow (p \vee q) \wedge (p \vee r)$

E_8 : $\overline{p \wedge q} \Leftrightarrow \bar{p} \vee \bar{q}$

E_9 : $\overline{p \vee q} \Leftrightarrow \bar{p} \wedge \bar{q}$

E_{10} : $p \vee p \Leftrightarrow p$

E_{11} : $p \wedge p \Leftrightarrow p$

E_{12} : $r \vee (p \wedge \bar{p}) \Leftrightarrow r$

E_{13} : $r \wedge (p \vee \bar{p}) \Leftrightarrow r$

E_{14}	:	$r \vee (p \vee \bar{p}) \Leftrightarrow T$
E_{15}	:	$r \wedge (p \wedge \bar{p}) \Leftrightarrow F$
E_{16}	:	$p \to q \Leftrightarrow \bar{p} \vee q$
E_{17}	:	$\overline{p \to q} \Leftrightarrow p \wedge \bar{q}$
E_{18}	:	$p \to q \Leftrightarrow \bar{q} \to \bar{p}$
E_{19}	:	$p \to (q \to r) \Leftrightarrow (p \wedge q) \to r$

4.2.7. Example: **Prove** that $r \wedge (p \vee q)$ is a valid conclusion from the given premises $p \vee q$, $q \to r$, $p \to m$ and $\bar{m}$.

Solution: The given four premises (P) are $p \vee q$, $q \to r$, $p \to m$ and $\bar{m}$.

{1}	(1) $p \to m$	Rule P
{2}	(2) $\bar{m}$	Rule P
{1,2}	(3) $\bar{p}$	Rule T, (1), (2) and I_{12}.
{4}	(4) $p \vee q$	Rule P
{1,2,4}	(5) q	Rule T, (3), (4) and I_{10}.
{6}	(6) $q \to r$	Rule P
{1,2,4,6}	(7) r	T, (5), (6) and I_{11}.
{1,2,4,6}	(8) $r \wedge (p \vee q)$	T, (4), (7) and I_9.

We arrived to the conclusion that $r \wedge (p \vee q)$.

4.2.8. Example: Show that the conclusion C: ~ P follows from the premises H_1: ~ P ∨ Q, H_2: ~ (Q ∧ ~ R) and H_3: ~ R.

Solution: We get

	(1) ~ R	Rule P (assumed premise)
	(2) ~ (Q ∧ ~ R)	Rule P
{2}	(3) ~ Q ∨ R	Rule T
{3}	(4) R ∧ ~ Q	Rule T
{4}	(5) ~ R → ~ Q	Rule T
{1, 5}	(6) ~ Q	Rule T
	(7) ~ P ∨ Q	Rule P
{7}	(8) ~ Q → ~ P	Rule T

{6, 8} (9) ~ P Rule T

Hence C logically follows from H_1, H_2 and H_3.

4.2.9. Example: Prove or disprove the conclusion given under from the following axioms.
"If Socrates is a man, Socrates is mortal". Socrates is a man.
Therefore Socrates is mortal.

Solution: The argument is valid because the argument follows the pattern of Modus ponens.
Consider the argument

p: Socrates is a man.

q: Socrates is mortal.

$p \rightarrow q$: If Socrates is a man, then Socrates is mortal.

The modus ponens is

$$\begin{array}{l} p \rightarrow q \\ \underline{p \qquad} \\ \therefore q \end{array}$$

Hence the conclusion q : "Socrates is mortal" is true.

4.3. CONSISTENCY OF PREMISES AND INDIRECT METHOD OF PROOF:

A set of m statement formulas H_1, H_2, ...H_m is called as consistent if the conjunction ($H_1\wedge$, $H_2\wedge...\wedge H_m$) has truth value "T" (or 1) for some assignment of the truth values to the atomic variables appearing in the statement formulas H_1, H_2, ...H_m. In other words, in the truth table, there exist at least one 1 under the column for ($H_1\wedge$, $H_2\wedge...\wedge H_m$).

If $H_1\wedge$, $H_2\wedge...\wedge H_m$ is false for every assignment of the truth values of the automic variables (that is ($H_1\wedge$, $H_2\wedge...\wedge H_m$) is a contradiction) appearing in the statement formulas H_1, H_2, ...H_m then we say that H_1, H_2, ...H_m are inconsistent.
We may also say that a set of formulas H_1, H_2, ...H_m are inconsistent if their conjunction ($H_1\wedge$, $H_2\wedge...\wedge H_m$) implies a contradiction, that is,

$$H_1\wedge, H_2\wedge...\wedge H_m \Rightarrow R \wedge \bar{R}$$

where R is any statement formula. Note that $R \wedge \bar{R}$ is a contradiction for any formula R.
We use this concept a procedure called "proof by contradiction" (or indirect method of proof).

4.3.1. Indirect Method of Proof

In order to prove that a conclusion C follows logically from the given statements (that is, premises) H_1, H_2, ..., H_m , we assume that C is FALSE and consider ~C as an additional premise. If $H_1 \wedge H_2 \wedge ... \wedge H_m \wedge$~C is a contradiction, then we conclude logically that "C follows logically from the premises H_1, H_2, ..., H_m.".

4.3.2. Example Show that $\sim (P \wedge Q)$ follows from $\sim P \wedge \sim Q$.

Solution: For this problem, the conclusion is $\sim (P \wedge Q)$. We have to consider the negation of this conclusion as an additional premise.

So we assume that $\sim (\sim (P \wedge Q))$ as an additional premise. Then

	(1) $\sim (\sim (P \wedge Q))$	Rule P
{1}	(2) $P \wedge Q$	Rule T
	(3) P	Rule T
	(4) $\sim P \wedge \sim Q$	Rule P
{4}	(5) $\sim P$	Rule T
{3, 5}	(6) $P \wedge \sim P$	Rule T

We know that $P \wedge \sim P$ is a contradiction. Hence by the indirect method of proof $\sim(P \wedge Q)$ follows logically from $\sim P \wedge \sim Q$.

4.3.3. Example:

"If there was a party, then catching the train was difficult. If they arrived on time then catching the train was not difficult. They arrived on time. Therefore there was no party." Show that the statement constitutes a valid argument.

Solution: Suppose that

p: There was a party

q: Catching the train was difficult.

r: They arrived on time.

Here, the conclusion is "there was no party" (that is, $\bar{p}$).

So we have to prove that $\bar{p}$ follows from the premises $p \to q$, $r \to \bar{q}$ and r.

	(1) r	Rule P
	(2) $r \to \bar{q}$	Rule P
{1, 2}	(3) $\bar{q}$	Rule T
	(4) $p \to q$	Rule P
{4}	(5) $\bar{q} \to \bar{p}$	Rule T
{3, 5}	(6) $\bar{p}$	Rule T

4.3.4 Example:

Using indirect method (or proof by contradiction) show that $\sqrt{2}$ is not a rational number.

Solution: The conclusion is that p: "$\sqrt{2}$ is not a rational number".

We have to consider not p, that is "7p : $\sqrt{2}$ is a rational number".

So suppose that $\sqrt{2}$ is a rational number.

Since $\sqrt{2}$ is a rational number, we have that $\sqrt{2} = \frac{a}{b}$ where a, b are two integers with gcd (a, b) = 1 and b ≠ 0.

Squaring on both sides, we get $\left(\sqrt{2}\right)^2 = \frac{a^2}{b^2}$

$$\Rightarrow 2 = \frac{a^2}{b^2}$$

$$\Rightarrow 2b^2 = a^2 \qquad \text{.....(1)}$$

[2 divides the left hand side, and so the right hand side also]

$\Rightarrow$ 2 divides a^2

$$\Rightarrow 2 \text{ divides } a \qquad \text{.....(2)}$$

$\Rightarrow a = 2k$ for some integer k

$$\Rightarrow a^2 = 4k^2 \qquad \text{.....(3)}$$

By (1) & (3) we get $2b^2 = 4k^2$

$$\Rightarrow 2b^2 = 4k^2$$

$$\Rightarrow b^2 = 2k^2$$

$$\Rightarrow 2 \text{ divides } b \qquad \text{.....(4)}$$

Now 2 divides 'a' and 'b' [from (2), (4)]

This is a contradiction to the fact gcd (a, b) = 1.

Hence the conclusion is true (that is $\sqrt{2}$ is not a rational number).

4.3.5. Example:

Prove that the following set of premises are not inconsistent (that is, consistent).

$p \to q,\ q \to r,\ \sim(p \wedge r),\ p \vee r \Rightarrow r$

Solution: We have to prove that the given set of premises (four statements:

$p \to q,\ q \to r,\ \sim(p \wedge r),\ p \vee r \Rightarrow r$) is not inconsistent.

This is equal to say that the meet ($\wedge$) (or the product) of all these four premises has truth value T in at least one case.

Consider the case when (p, q, r) = (F, T, T). That is, the truth values of p, q, r are equal to F, T, T respectively. In this case, the truth values of $p \to q$, $q \to r$, $\sim(p \wedge r)$ and $p \vee r \to r$ are all equal to T. Hence the product (that is, meet) of all the four premises also have truth value T. This says that the set of four premises is consistant. The proof is complete.

Hence the given set of premises $p \to q$, $q \to r$, $\sim(p \wedge r)$, $p \vee r \to r$ will not form a set of inconsistent formula.

4.4 PREDICATE LOGIC:

Logic that deals with predicates is named as Predicate Logic.

4.4.1 Predicate

In the statement “Satya is beautiful”, the part “is beautiful” is called a predicate. The part “Satya” is a noun or subject or object. Every predicate describes some property of one or more objects.

In symbolizing the statements, in general, we use capital letters for predicates, and small letters for individuals or objects.

Let us consider the following two atomic statements:

1. Satya is beautiful.
2. Lakshmi is beautiful.

If we express these two statements by symbols, we need to have two different symbols. We introduce some symbol to denote “is beautiful”. Also a method to join it with symbols that denotes the names of individuals.

4.4.2. Examples:

(i). Consider the statements:

1. Satya is beautiful.
2. Lakshmi is beautiful.

We denote the predicate “is beautiful” by the capital letter B (here, B first letter of the word (predicate) beautiful). We use symbol “s” for “satya” and “l” for “Lakshmi”.

In symbolic form, the statements (1) and (2) will be written as B(s) and B(l) respectively.

In general, any statement of the form “p is Q” where Q is the predicate and p is noun (or subject) is denoted by Q (p).

Thus B(s) denotes the statement “Satya is beautiful”.

B(l) denotes the statement “Lakshmi is beautiful”.

(ii). Consider the statements:

1. Mallikarjun is a student.
2. Gnyana is a student.

Observe the given statements. The predicate involved in these statements I “is a student”. We denote this predicate “is a student” by the capital letter “S”, and the nouns (or subjects) Mallikarjun by small letter “m” and Gnyana by the small letter “g”.

Now S(m) means “m is S” (that is, Mallikarjun is a student); S(g) means “Gnyana is a student”.

4.4.3. 2–place predicate

In the Example 4.4.2, we considered the atomic statement “Mallikarjun is a student”. Here the predicate “is a student” have one and only one noun (or subject) namely “Mallikarjun”. So it is named as 1–place predicate.

Now consider the statement “Mallikarjun is taller than Gnyana”. In this statement, “is taller than” is a predicate and it deals with two names (or nouns or individuals). This predicate is called as 2–place predicate.

A predicate associated with two names (or nouns) is called as 2–place predicate.

In symbolic form we write T(m, g) where T denotes the predicate "is taller than", m denotes Mallikarjun, and g denotes Gnyana.

4.4.4. Example:

Consider the following atomic statement:

Andhra Pradesh is to the north of Tamilnadu.

In the given statement, "is to the north of" is the predicate, we denoted by the capital letter "N".

We denote Andhra Pradesh is by "a", and the Tamilnadu is denoted by "t".

So N(a, t) means "Andhra Pradesh is to the north of Tamilnadu".

Thus this predicate is a 2–place predicate.

4.5. m-PLACE PREDICATE:

A predicate associated with m names (or nouns) (where m is a positive integer) is called an m–place predicate. In order to extent this definition to m = 0, we say that a predicate is a 0–place predicate if no names are associated with the predicate.

4.5.1. Example:

Consider the following statement: "Satya sits between Mallikarjun and Gnyana".

In the given statement, "Sits between" is the predicate, we denote this predicate by B.

We denote Satya, Mallikarjun and Gnyana by s, m, g respectively. Then B(s, m, g) denotes the given statement. In the given statement, the predicate is associated with three names (or individuals). Hence this predicate is a 3–place predicate.

4.6. CONNECTIVES:

The known connectives ($\wedge, \vee$, 7) that were used in statement logic, can be used to form compound statements.

"Satya is beautiful" and "Lakshmi is beautiful".

"Satya is beautiful" or "Lakshmi is beautiful".

These sentences were were represented by

$B(s) \wedge B(l)$

$B(s) \vee B(l)$

"The painting is Red" is denoted by R(p).

"The painting is not Red" is denoted by $\rceil R(p)$ or $\sim R(p)$ or $\overline{R(p)}$

4.6.1. Example:

Represent the statement "Rama is handsome and Sita is beautiful" by predicate logic.

Solution: Suppose "H" denotes the predicate "is handsome"; and "B" denotes the predicate "is beautiful". Suppose the symbols "r" and "s" denotes the subjects Rama and Sita, respectively.

Then H(r) denotes the statement "Rama is handsome", and B(s) denotes "Sita is beautiful".

So H(r) $\wedge$ B(s) denotes the statement "Rama is handsome and Sita is beautiful".

In terms of predicate logic, the given statement is represented by

H(r) $\wedge$ B(s).

4.7. STATEMENT FUNCTIONS, AND VARIABLES:

An expression consisting of a predicate symbol and an individual variable is said to be a simple statement function of one variable.

Such a statement function becomes a statement when the variable is replaced by the name of the object.

As an illustration, take the predicate B ("is beautiful") and "s" (the name Satya). As we know, B(s) means "s is beautiful" (that is, Satya is beautiful).

In place of Satya let us use a variable x. We write B(x), the notation for "x is beautiful". In place of x we may substitute "Lakshmi" (or l), then we get B(l) which means "Lakshmi is beautiful". Now we consider B(x) as a statement and x is a place holder (or variable) B(x) is a statement function.

4.7.1. Example

In the statement function (here a 2-place predicate) T(x, y) (means x is taller than y), where T denotes the predicate "is taller than", x and y are place holders in the 2–place predicate T(x, y).

If we replace, x by Mallikarjun, and y by Gnyana then T (m, g) denotes the statement.

Mallikarjun is taller than Gnyana.

Note that T(x, y) is a statement function and x, y are variables.

4.7.2. Note

Let B be the predicate "is beautiful".

Consider the following three statements.

B(s): Satya is beautiful.

B(g): Gnyana is beautiful.

B(l): Lakshmi is beautiful.

Here B(s), B(g), B(l) all denote statements, but they have common form (feature, beautiful).

If we write B(x) for "x is beautiful", then B(s), B(g), B(l) and others with same form can be obtained from B(x) by replacing x by the suitable name s, g, l, …

We note that B(x) is not a statement but it result in many statements when we replace the variable x by appropriate names (or subjects or nouns).

4.7.3. Combined Statements and Connectives:

Consider the given two statement functions in one variable x.

B(x): x is beautiful.

M(x): x is mortal.

Now

$B(x) \wedge M(x)$ denotes "x is beautiful and x is mortal".

$B(x) \vee M(x)$ denotes "x is beautiful or x is mortal".

7B(x) (or $\overline{B(x)}$) denotes "x" is not beautiful".

Suppose T(x, y) denotes "x" is taller than y" .

This is a predicate in two variables x and y.

Then $\rceil$T(x, y) (or $\overline{T(x,y)}$) denotes that "x" is not taller than y".

4.7.4. Example:

Construct the statement function in predicate calculus for the given statement "x" is rich and y is tall".

Solution: We know that "x is rich" is denoted by R(x); and "y is tall" is denoted by T(y).

So "$R(x) \wedge T(y)$" denotes the statement "x is rich and y is tall", where R and T are predicates "is rich" and "is tall" respectively; and x and y are variables.

4.8 SUMMARY:

In Lessons 1,2 and 3, we studied atomic statements and statement formulas. In this Lesson, we studied: Theory of Inference for Statement Calculus; Consistency of premises and indirect method of proof; Predicates; m-place Predicates; and Connectives, Statement Functions, and Variables. In the inference theory, all the premises and conclusions are statements. If any two statements have common feature, then we are unable to express the common feature. In order to study the common feature statements, the concept "predicate" is useful. The logic related to the analysis of predicates is called as predicate logic.

In this Lesson, we also studied the concepts: 2-place predicate, m-place predicate. A predicate associated with m names (or nouns) (where m is a positive integer) is called as m–place predicate. Some examples related to 2-place predicate, and 3-place predicate were included. Connectives used in predicate logic were introduced and explained in detail for the better understanding of the reader. Statement functions and variables in predicate logic were explained.

4.9 TECHNICAL TERMS:

Predicate

[In the statement "Satya is beautiful", the part "is beautiful" is called a predicate].

Predicate Logic

[Logic that deals with predicates is named as Predicate Logic].

2- place predicate

[A predicate associated with two names (or nouns) is called as 2–place predicate].

3-place predicate

[A predicate associated with three names (or nouns) is called as 3–place predicate].

m-place predicate

[A predicate associated with m names (or nouns) (where m is a positive integer) is called as m–place predicate].

Simple statement function of one variable.

[An expression consisting of a predicate symbol and an individual variable is said to be a simple statement function of one variable].

4.10 SELF ASSESSMENT QUESTIONS:

(i). Show that the following set of premises is inconsistent.

$$p \to (q \to r), s \to (q \wedge \sim r), p \wedge s$$

(ii). Prove that $S \vee R$ is tautologically implied by $(P \vee Q)$, $(P \to R)$, $(Q \to S)$.

(iii). Represent the statement “Rama is King” by predicate logic.

[Ans: K(r) represents the given statement. Here K denotes “is a king”, and r denotes Rama].

(iv). Represent the statement “Rama is a brother of Lakhmana” by predicate logic.

[Ans: B(r,l), where B denotes the predicate “a brother of”, r,l denotes Rama, Lkhmana, repectively].

(v). Represent the statement “Rama is boy and Sita is girl” by predicate logic.

[Ans: $B(r) \wedge G(s)$, where B and G are predicates “is a boy”, and “is a girl” respectively].

(vi). Represent the statement “Rama is standing between Bhima and Krishna” by predicate logic.

[Ans: S(r,b,k) where S denotes the predicate “is standing between”, r, b, k denotes the nouns Rama, Bhima, Krishna respectively].

(vii). Represent the statement “Rama is sitting between Bhima and Krishna” by predicate logic.

[Ans: S(r,b,k) where S denotes the predicate “is sitting between”, r, b, k denotes the nouns Rama, Bhima, Krishna respectively].

4.11. SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Bhavanari Satyanarayana

LESSON - 5
QUANTIFIERS

OBJECTIVE:

- ❖ To know the concept Quantifier.
- ❖ To identify different types of Quantifiers.
- ❖ To Learn the validity of the Statements
- ❖ To have proper understanding of different Quantifiers.
- ❖ To develop skills in solving the problems.

STRUCTURE:

5.1 Introduction
5.2 Quantifier
5.3 The Universe of discourse
5.4 Free and bounded variables
5.5 Summary
5.6 Technical Terms
5.7 Self Assessment Questions
5.8 Suggested Readings

5.1 INTRODUCTION:

In the previous Lessons, we discussed regarding the Statements, connectives, tautology, contradiction, truth tables, etc. In this Lesson we added some more knowledge by presenting new concept "Quantifier". In this lesson finally we presented the concepts: free variable and bounded variable. We included some examples that are needed to understand the new concepts.

5.2. QUANTIFIERS:

We know about the atomic statements and predicates. In this Lesson, we introduce the notion quantifiers ("all" and "some"). These concepts provide some extension to the earlier knowledge. The word "all" is said to be "universal quantifier"; and the word "some" is said to be "existential quantifier". We use the words "all" and "some" in several sentences such as "All men are mortal", "Some men are not professors".

5.2.1. Universal Quantifier

The quantifier "all" (which is the universal quantifier) is denoted by (x) or $\forall x$

We place this symbol just before the statement function. For example, consider the statement functions:

B(x): x is beautiful

W(x): x is a woman

Then $(x)\ (W(x) \rightarrow B(x))$ denotes "for all x, if x is a women then x is beautiful"(In other words, All women are beautiful).

Now we can understand that the statement "for all x, if x is a women then x is beautiful" (In other words, All women are beautiful) is denoted by "(x) (W(x) → B(x))".

Also we have to note that x is arbitrary. In place of the variable, we may use any other variable such as y or z. We get the same meaning.

So (x) (W(x) → B(x); and (y) (W(y) → B(y)) are having equivalent meaning (because x, y are just variables) (x) (W(x) → B(x)) may be denoted by ∀x (W(x) → B(x)).

5.2.2. Example:

Represent the statement "For any x and for any y, if x is richer than y, then x is not poorer than y" by predicate logic.

Solution: Consider the following 2–place predicates (two in number):

R(x, y): x is richer than y.

P(y, x): y is poorer than x.

From the second statement we get that "⌉P(x, y): x is not poorer than y".

Now we got the third statement.

⌉P(x, y): x is not poorer than y.

By using the Universal quantifiers (x) and (y), we can write the forth statement:

"(x)(y) (R(x, y) → ⌉P(x, y))".

This forth statement means that

"For any x and for any y, if x is richer than y, then x is not poorer than y".

This statement may also be denote as follows:

∀x ∀y (R(x, y) → ⌉P(x, y)).

Note that in the above, we obtained representation of the statement "For any x and for any y, if x is richer than y, then x is not poorer than y" in terms of predicate logic.

5.2.3. Existential Quantifier

We know that the word "some" is called as the existential quantifier. This existential quantifier is denoted by "∃". This also have the meaning "for some" or "there exists at least one".

If we write "∃x" , this have the meaning "for some x" or "there exists at least one x".

The symbol ∃!x (or ∃ unique x) is used for "there is a unique x", or "there exists unique x").

As in case of for all, we place this symbol also, just before the statement functions.

5.2.4. Example:

Consider the following five statement functions:

M(x): x is a man, where M denotes the predicate "is a man".

C(x): x is clever, where C denotes the predicate "is clever".

I(x): x is an integer, where I denotes the predicate "is an integer".

E(x): x is even, where E is the predicate “is even”.

P(x): x is prime, where P is the predicate “is prime”.

Then

$\exists$ x M(x) symbolizes “There exists a man”

$\exists$ x (M(x) $\wedge$ C(x)) symbolizes “There are some men who are clever”

$\exists$ x (I(x) $\wedge$ E(x)) symbolizes “Some integers are even” or “There are some integers which are even”

$\exists$!x (E(x) $\wedge$ P(x)) symbolizes “There exists unique even number which is a prime number”.

5.2.5. Example:

Represent the statement “There is a man who is clever” by predicate logic.

Solution: Consider the two statements given below;

M(x): x is a man, where M denotes the predicate “is a man”.

C(x): x is clever, where C denotes the predicate “is clever”.

M(x) $\wedge$ C(x) represents “x is a man, and x is clever”.

Therefore $\exists$ x (M(x) $\wedge$ C(x)) is the symbolic form of the statement “There is a man who is clever”

5.2.6. Example:

Represent the statement “there is only one prime number which is also an even number” by predicate logic.

Solution: Consider the two statements given below;

P(x): x is a prime number, where P is the predicate “is aprime number”.

.

E(x): x is an even number, where E is the predicate “is an even number”.

.

(E(x) $\wedge$ P(x)) symbolizes “There exists a prime number which is an even number”.

$\exists$!x (E(x) $\wedge$ P(x)) symbolizes “There exists unique (or only one) prime number which is also an even number”.

5.3 THE UNIVERSE OF DISCOURSE:

Variables that were quantified may belong to certain sets.

That particular set is called as the universe of discourse or the domain or simply universe.

In the statement “M(x) : x is a man”, the variable x relates to the set of all men. Here the set of all men is the universe of discourse.

In the statement “E(x) : x is an even number”, then x relates to all the even numbers. Here the universe is the set of integers.

So, the universe may be, the class of human beings, or numbers (real, complex, and rational) or some other objects. The truth value of a statement function containing quantifier depends upon the universe.

5.3.1. Example

Suppose Q(x) is the predicate that

Q(x): x is less than 10.

Consider the statements (x) Q(x) and ∃ x Q(x).

Let us define the sets U_1, U_2 and U_3 as follows:

U_1: {–1, 0, 1, 2, 4, 6, 8};

U_2: {3, –2, 12, 14, 10} and

U_3: {10, 20, 30, 40}.

By considering different cases, let us observe whether the statements are true / false with respect to U_1, U_2 and U_3 ,treating as universes.

(i) The statement (x) Q(x) is true in U_1 because the statement function Q(x)

[that is, $x < 10$] is true for every x in U_1. In this case (x) Q(x) is True.

(ii) The statement (x) Q(x) is not true in U_2 [because there is the element 12 in U_2 such that 12 is not less than 10]. Hence, in this case (x) Q(x) is False.

(iii) The statement (x) Q(x) is not true in U_3., because 20 is not less than 10.

(iv) The statement "∃ x Q(x)" is true in U_1 and U_2 [because there exist atleast one element in U_1 (also in U_2) which is less than 10].

(v) The statement "∃ x Q(x)" is not true (that is, false) in U_3 [because there is no element in U_3 which is less than 10].

5.3.2. Example

Suppose that "the set of integers" is the universe of discourse.

Determine the truth values of the following sentences:

1. (x) ($x^2 \geq 0$)
2. (x) ($x^2 - 5x + 6 = 0$)
3. ∃(x) ($x^2 - 5x + 6 = 0$)
4. (y) (∃ x ($x^2 = y$))

Solution: 1. For any integer x, we know that $x^2 \geq 0$. Hence the statement (x) ($x^2 \geq 0$) is true when "the set of integers" is the universe of discourse.

2. Consider the integer 1. If we substitute x = 1, then $x^2 - 5x + 6 = 2$ which is not equals to 0. So the statement $x^2 - 5x + 6 = 0$ is not true with x = 1.

Hence the given statement (x) ($x^2 - 5x + 6 = 0$) is not true if we consider "the set of integers" as universe of discourse.

3. Consider the integer 2. If we substitute $x = 2$, then $x^2–5x + 6 = 0$. So the statement $x^2–5x + 6 = 0$ is true with $x = 2$, and 2 is an integer.

Hence the given statement $(x) (x^2–5x + 6 = 0)$ is true if we consider "the set of integers" as universe of discourse.

4. Consider the integer $y = 2$. We know that there is no integer x such that $x^2 = y$.

So the statement $x^2 = y$ is not true if $y = 2$ and x is an integer.

Hence the given statement $(y) (\exists x (x^2 = y))$ is not true if we consider "the set of integers" as the universe of discourse.

5.3.3. Example:

Find out the quantifiers for the following statements where predicate symbols denote.

K(x): x is two-wheeler

L(x): x is a scooter

M(x): x is manufactured by Bajaj

(a) Every two wheeler is a scooter.
(b) There is a two wheeler that is not manufactured by Bajaj.
(c) There is no two wheeler manufactured by Bajaj that is not a scooter.
(d) Every two wheeler that is a scooter is manufactured by Bajaj.

Solution*:* Given that

K(x): x is a two-wheeler

L(x): x is a scooter

M(x): x is manufactured by Bajaj

(a). We have to find out quantifier for the statement:

"Every two wheeler is a scooter".

The expression $(K (x) \rightarrow L(x))$ denotes the statement that "two wheeler is a scooter".

Therefore

$$(x) (K (x) \rightarrow L(x))$$

represents the expression that "Every two wheeler is a scooter".

(b). We have to find out the quantifier for the statement:

"There is a two wheeler that is not manufactured by Bajaj".

The expression 7M(x) denotes the statement that "x in not manufactured by Bajaj".

The expression $K(x) \wedge 7M(x)$ denotes the statement that "x is a two wheeler and not manufactured by Bajaj".

Hence "$\exists x (K(x) \wedge 7M(x))$" is the expression that states that "there exists x which is a two wheeler and not manufactured by Bajaj".

(c). We have to find out the quantifier for the statement

"There is no two wheeler manufactured by Bajaj that is not a scooter".

The expression " $\neg$L (x)" denotes the statement that "x is not a scooter".

The expression "K(x) $\wedge$ M(x) $\wedge$ $\neg$L (x)" denotes the statement that "x is a two wheeler manufactured by Bajaj that is not a scooter".

Therefore the quantifier expression $\neg$(∃x (K(x) $\wedge$ M(x) $\wedge$ 7 L(x)) denotes the statement that "There is no two wheeler manufactured by Bajaj that is not a scooter".

(d). We have to find out the quantifier for the statement "Every two wheeler that is a scooter is manufactured by Bajaj".

The expression "K(x) $\wedge$ L(x)" denotes the statement that "x is a two wheeler that is a scooter".

So the expression (x) ((K(x) $\wedge$ L(x) $\rightarrow$ M(x)) denotes the statement that "Every two wheeler that is a scooter is manufactured by Bajaj".

5.4 FREE AND BOUNDED VARIABLES:

Suppose that (x) p(x) or ∃ x p(x) is a part of a given formula. Such a part of the form (either (x) p(x) or ∃ x p(x)) is called as x–bound part of that given formula.

The formula p(x) either in "(x) p(x)" or in "∃ x p(x)" is called as the scope of the quantifier.

5.4.1. Example

Suppose the universe of discourse is the set of integers.

Consider the statement that

p(x) : $x^2 \geq 0$

We know that $x^2 \geq 0$ for all integers

So p(x) is true for all x in the universe of discourse.

We know that we write this fact as (x) p(x).

This (x) p(x) is a x–bound part.

5.4.2. Example:

Suppose the universe of discourse is the set of all complex numbers.

Consider the statement that

"If y is a complex number, then there exist a complex number x such that $x^2 = y$.

The expression (y) (∃ x ($x^2 = y$)) denotes the statement that "If y is a complex number, then there exist a complex number x such that $x^2 = y$.

Now the expression ∃ x ($x^2 = y$) is x–bound part of "(y) (∃ x ($x^2 = y$))".

5.4.3. Definitions

(i) Any occurrence of in an x–bound part of a formula is called as bound occurrence of x.

(ii) Any occurrence of x (or a variable) which is not a bound occurrence is called a free occurrence.

5.4.4. Example

(i). Consider the formula

$$\exists x\ (p(x) \wedge q(x))$$

Here the scope of ($\exists$ x) is p(x) $\wedge$ q(x).

Hence in "$\exists$ x (p(x) $\wedge$ q(x))", all the occurrences of x are bound occurrences.

(ii) If we consider a statement r(x), then the occurrence of x in r(x) is a free occurrence.

5.4.5. Examples:

(i). Consider the statement:

Lakshmi is beautiful.

The symbolic representation is B(l). It is clear that the statement formula B(x) means x is beautiful, where x is a variable. Note that in "B(x)" there is no quantifier. Hence the occurrence of the variable x in "B(x)" is a free occurrence.

(ii). Consider the statement:

"All birds can fly".

Now we symbolize this statement. Write

B(x): x is a bird

F(x): x can fly

It is clear that "(x) (B(x) $\rightarrow$ F(x))" denotes the statement "All birds can fly".

In this "(x) (B(x) $\rightarrow$ F(x))", all occurrences of x are bound occurrences.

5.4.6. Example:

Symbolize "All the people respects selfless leaders".

Solution: Let us consider the following three statements

P(x): x is a person

S(x): x is a selfless leader

R(x, y): x respects y

Now the required symbol

The expression S(y) $\rightarrow$ R(x, y) denotes the statement that "If y is a selfless leader then x respects y".

The expression "(y) (S(y) $\rightarrow$ R(x, y))" denotes the statement that "x respects every selfless leader y".

The expression "p(x) → (y) (S(y) → R(x, y))" denotes the statement that "person x respects every selfless leader y".

The expression "(x) [p(x) → (y) (S(y) → R(x, y))" denotes the statement that "All the people respects selfless leaders".

5.4.7. Note:

The negations of some frequently used, important statement functions were presented in the following table.

Statement function	Negation
∃ x F(x)	(x) (~F(x))
(x) F(x)	∃ x (~F(x))
∃ x (~ F(x))	(x) F(x)
(x) (~F(x))	∃ x F(x)

5.4.8. Example:

Find the negation of the given expression: "(x) (E(x) → S(x))

Solution: Suppose F(x): "E(x) → S(x)".

Now the given expression is of the form "(x) F(x)", where F(x): "E(x) → S(x)" .

From the above table, the negation of "(x) F(x)" is ∃ x (~F(x)).

It is clear that ~F(x) is the negation of E(x) → S(x).

We know that the negation of (E(x) → S(x)) is "E(x) → ~S(x)".

Now observe that ∃ x (~F(x)) is equivalent to ∃ x (E(x) → ~S(x)).

This states that "∃ x (E(x) → ~S(x))" is the negation of the given expression

"(x) (E(x) → S(x))".

5.4.9. Example

Find out the quantifiers of the following statements where predicate symbols denotes,

F(x): x is fruit

V(x): x is vegetable and

S(x, y): x is sweeter than y

(a) Some vegetables are sweeter than all fruits
(b) Every fruit is sweeter than all vegetables
(c) Every fruit is sweeter than some vegetables
(d) Only fruits are sweeter than vegetables

Solution:

(a). Consider the given statement "Some vegetables are sweeter than all fruits".

The expression $F(y) \rightarrow S(x, y)$ denotes that "x is sweeter than the fruit y".

The expression "$(y)\ (F(y) \rightarrow S(x, y))$" denotes that "x is sweeter than y for all fruits y".

So $\exists\ x\ [V(x) \rightarrow ((y)\ (F(y) \rightarrow S(x, y))]$ is the required predicate formula.

(b). We have to symbolize "Every fruit is sweeter than all vegetables".

The expression $V(y) \rightarrow S(x, y)$ denotes the statement "x is sweeter than the vegetable y".

The expression $(y)\ (V(y) \rightarrow S(x, y))$ denotes the statement
"x is sweeter than all vegetables y".

Therefore the required predicate formula is "$(x)\ [F(x) \rightarrow (y)\ (V(y) \rightarrow S(x, y))]$".

(c). We have to symbolize "Every fruit is sweeter than some vegetables".
The expression $\exists\ y\ (V(y) \wedge F(x, y))$ denotes "there exists a vegetable y such that x in sweeter than y".

Hence the required predicate formula is "$(x)\ [F(x) \rightarrow \exists\ y\ (V(y) \wedge F(x, y))]$".

(d). We have to symbolize the given statement "only fruits are sweeter than vegetables".

In other words, this statement can be written as "if x is sweeter than all vegetables, then x is a fruit".

The expression "$(y)(V(y) \rightarrow S(x, y))$" denotes the statement "x is sweeter than all vegetables".

Hence the required predicate formula is $[(y)\ (V(y) \rightarrow S(x, y)\ )] \rightarrow F(x)$.

5.5 SUMMARY:

In the previous Lessons, we discussed regarding the Statements, connectives, tautology, contradiction, truth tables, etc. In this Lesson we added some more knowledge by presenting new concept "Quantifier". The concepts Universal Quantifier, Existential Quantifier and Universe of Discourse were explained and some related examples were presented. Finally we presented the concepts: free variable and bounded variable, and included sufficient number of examples that are needed for clear understanding of the new concepts.

5.6 TECHNICAL TERMS:

Quantifiers
("all" and "some").

Universal Quantifier
The quantifier "all" is called as the universal quantifier.

Existantial Quantifier
The quantifier "some" is called as the universal quantifier.

The Universe of discourse.

A variable that was quantified may belongs to certain set, called as the universe of discourse or the domain or simply universe.

Bound Variable.

In the expressions "(x) p(x)" and "$\exists$ x p(x)", the variable x is called as bound variable.

Free Variable

In the expression like "p(x)", there is no bound such as "for all", or "there exists", such a variable is called as free variable.

5.7 SELF ASSESSMENT QUESTIONS:

1. Find out the quantifiers for the following statement

"Every integer is a rational number".

Ans: (x) (I(x) $\rightarrow$ Q(x)) is the required expression where I(x) denotes "x is an integer", and Q(x) denotes "x is a rational number".

2. Find out the quantifiers for the following statement

"Every integer is not an even integer".

Ans: "(x) (I(x) $\rightarrow$ ~E(x))" is the required expression where I(x) denotes "x is an integer", and E(x) denotes "x is an even integer".

3. Find out the quantifier for the statement:

"Every two wheeler is a scooter".

The expression (x) (K (x) $\rightarrow$ L(x)) denotes the statement that "every two wheeler is a scooter".

4. Find out the quantifier for the statement:

"All dogs are not cats".

"(x) (D(x) $\rightarrow$ ~C(x))" is the required expression where D(x) denotes "x is a dog", and C(x) denotes "x is a cat".

5. Symbolize "All Dogs are Animals" using the quantifier.

Ans: The expression (x) (D(x) $\rightarrow$ A(x)) denotes "All dogs are animals", where D(x): x is a dog and A(x): x is an animal.

.

6. Symbolize "Some horses are black"

Ans: The expression $\exists$ x (H(x) $\rightarrow$ B(x)) denotes "there exists a horse which is black (or) some horses are black", where H(x): x is a horse, and B(x): x is black.

5.8 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd. , New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Bhavanari Satyanarayana

LESSON - 6

INFERENCE THEORY FOR PREDICATE CALCULUS

OBJECTIVE:

- ❖ To know more about predicate calculus.
- ❖ To understand the Rules of Inference.
- ❖ To identify the validity of arguments.
- ❖ To Learn the additional rules of inference.
- ❖ To develop skills in solving the problems by using rules of inference**.**

STRUCTURE:

6.1 Introduction
6.2 Universal Specification
6.3. Universal Generalization
6.4 Existential Specification
6.5 Existential Generalization
6.6. Formulas with more than one Quantifier
6.7 Summary
6.8 Technical Terms
6.9 Self Assessment Questions
6.10 Suggested Readings

6.1 INTRODUCTION:

In earlier lessons, we have already discussed the "inference theory for the statement calculus". We know that the method of derivation related to the predicate formulas uses the rules of inference that discussed for the statement calculus. In addition to the rules of inference discussed pfor the statement calculus, in derivations, we also use certain additional rules (or principles) that are given below: Universal Specification (US, in short), Universal Generalization (UG, in short). Existential Specification (ES, in short). Existential Generalization (EG, in short).

6.2. UNIVERSAL SPECIFICATION (US):

If (x) p(x) is true, then the universal quantifier can be dropped to obtain "p(c) is true", where c is an arbitrary object in the universe of discourse.

6.2.1. Example:

Consider the following statements.
All women are mortal.
Lakshmi is a woman.
Here the universe is the set of all women.

M(x): x is mortal
(x) M(x): all women are mortal (as x is in the universe).

Since "Lakshmi is a woman", we have that "Lakshmi" is in universe of discourse.
So by using Universal Specification we may replace x by "Lakshmi",.
If we replace x by Lakshmi in "x is mortal", we get that "Lakshmi is mortal".
Note that in this example, we used US.

6.3. UNIVERSAL GENERALIZATION (UG):

If P(c) is true for all c in the universe of discourse, then the universal quantifier may be prefixed to obtain (x) P(x).

6.3.1. Example:

Suppose that U = {1, 2, 3, 4} is the universe of discourse.
Suppose that p(x) : "$x^2 \leq 50$".

It is clear that for every $x \in U$ the statement $x^2 \leq 50$ is true.

If x = 1 then $x^2 = 1 \leq 50$, and so p(1) is true

If x = 2 then $x^2 = 4 \leq 50$, and so p(2) is true

If x = 3 then $x^2 = 9 \leq 50$, and so p(3) is true

If x = 4 then $x^2 = 16 \leq 50$, and so p(4) is true

Now we verified that P(c) is true for all c in the universe U of discourse.
Hence, by Universal Generalization, we can write "(x) P(x)".

6.3.2. Example:

Prove the following statement (transitivity) by using the rules of Inference:

$(x)\ (P(x) \rightarrow Q(x)) \wedge (x)\ (Q(x) \rightarrow R(x))$

$\Rightarrow (x)\ (P(x) \rightarrow R(x))$

***Solution**:* Given statements (premises) are:

$(x)\ (P(x) \rightarrow Q(x))$ Premise–1

and

$(x)\ (Q(x) \rightarrow R(x))$ Premise–2

Assuming premise–1 and premise–2 we have to obtain the conclusion "$(x)\ (P(x) \rightarrow R(x))$".

Derivation:

$(x)\ (P(x) \rightarrow Q(x))$	P (Premise–1)
$P(c) \rightarrow Q(c)$	US and (1)
$(x)\ (Q(x) \rightarrow R(x))$	P (Premise–2)
$Q(c) \rightarrow R(c)$	US and (3)
$P(c) \rightarrow R(c)$	[(2), (4) and Inference Rule (hypothetical Syllogism)]

$(x)\ (p(x) \rightarrow R(x)$ UG and (5)

Hence we get the conclusion that $(x)\ (p(x) \rightarrow R(x)$.

So we proved the given statement that

$[\ (x)\ (P(x) \rightarrow Q(x)) \wedge (x)\ (Q(x) \rightarrow R(x))] \Rightarrow (x)\ (P(x) \rightarrow R(x))$.

6.3.3. Example

Consider the following statements.

All men are selfish. (Premise–1)
All kings are men. (Premise–2)
Prove that all kings are selfish.

***Solution**:* Suppose that

M(x): x is man.
K(x): x is King.
S(x): x is selfish.

$(x)\ (M(x) \rightarrow S(x))$ is Premise–1; and

$(x)\ (K(x) \rightarrow M(x))$ is Premise–2.

The derivation is as follows.

$(x)\ (M(x) \rightarrow S(x))$ P (Premise–1)

$M(c) \rightarrow S(c)$ US, (1)

$(x)\ (K(x) \rightarrow M(x))$ P (Premise–2)

$K(c) \rightarrow M(c)$ US, (3)

$K(c) \rightarrow S(c)$ [(2), (4) and Inference Rule hypothetical syllogism]

$(x)\ (K(x) \rightarrow S(x))$ UG and (5)

Hence we get that "All Kings are selfish".

6.3.4. Example

Prove or disprove the validity of the following argument by using the rules of inference.

All men are warriors. (Premise–1)
All Kings are men. (Premise–2)
Therefore All Kings are warriors.

***Solution**:* Let

M(x): x is a man.
K(x): x is a king
W(x): x is a warrior

$(x)\ (M(x) \rightarrow W(x))$ (Premise–1),

$(x)\ (K(x) \rightarrow M(x)$ (Premise–2).

Now the derivation is as follows:

$(x)\ (M(x) \rightarrow W(x))$ P (Premise–1)

M(c) → W(c) US and (1)

(x) (K(x) → M(x)) P (Premise–2)

K(c) → M(c) US and (3)

K(c) → W(c) [(2), (4), and Inference rule: hypothetic syllogism]

(x) (K(x) → W(x)) UG and (5)

Now we got the conclusion that "All kings are warriors".

6.3.5. Example

Using predicate logic, Rules of Inference, show that the following argument is valid.
Every wife argues with her husband.
X is a wife.
Therefore, X argues with her husband.

***Solution**:* Write

W(x): x is a wife.

A(x, h): x argues with her husband, where h denotes husband.

(x) (W(x) → A(x, h)): Every wife x argues with her husband h.

(x) (W(x) → A(x, h)) Premise–1

W(x): x is a wife Premise–2

Derivation:

W(x) P (Premise–2)

(x) (W(x) → A(x, h)) P (Premise–1)

W(x) → A(x, h) US and (2)

A(x, h) [(1), (3) and modus ponens]

Therefore, A(x, h): x argues with her husband.

So we conclude that if X is a wife, then x argues with her husband.

6.4. EXISTENTIAL SPECIFICATION (ES):

If ∃ x P(x) is assumed to be true, then P(c) is true for some element c in the universe of discourse.

6.5. EXISTENTIAL GENERALIZATION (EG):

If P(c) is true for some element c in the universe of discourse, then we can write "∃ x P(x)" is true.

6.5.1. Example

Prove that $\exists x\,(r(x) \wedge q(x)) \Rightarrow (\exists x\, r(x)) \wedge (\exists x\, q(x))$ by using the rules of inference.

***Solution**:* The given premise is $\exists x\,(r(x) \wedge q(x))$.

We have to prove the conclusion that $(\exists x\, r(x)) \wedge (\exists x\, q(x))$.

The derivation is as follows:

$\exists x\,(r(x) \wedge q(x))$	P (Premise)
$r(y) \wedge q(y)$	ES and (1)
$r(y)$	[(2) and Inference Rule (Simplification)]
$q(y)$	[(2) and Inference Rule (Simplification)]
$\exists x\, r(x)$	EG and (3)
$\exists x\, q(x)$	EG and (4)
$\exists x\, r(x) \wedge \exists x\, q(x)$	[(5), (6), and Inference Rule:I_9]

6.5.2. Example

Prove that " $\exists x\,(M(x))$" follows logically from the premises.

$(x)\,(A(x) \rightarrow M(x))$ and $\exists x\, A(x)$

***Solution**:* The given two premises are

$(x)\,(A(x) \rightarrow M(x)$	Premise–1
$\exists x\, A(x)$	Premise–2

We have to get the conclusion: $\exists x\,(M(x))$.

Derivation:

$\exists x\, A(x)$	P (Premise–2)
$A(c)$	ES and (1)
$(x)\,(A(x) \rightarrow M(x))$	P (Premise–1)
$A(c) \rightarrow M(c)$	US and (3)
$M(c)$	[(2), (4) and Inference rule (Modus Ponens)]
$\exists x\, M(x)$	EG and (5)

6.5.3. Example

Explain with an example:

$(x)\,[E(x) \wedge B(x)]$ need not be a conclusion form $\exists x\, E(x)$ and $(\exists x)\, B(x)$

Solution*:*

Let $U = \{1, 2\}$ be the universe of discourse.

Write:

E(x): x is even
B(x): x is odd

Since 1 is an element of U such that 1 is odd, it is true that ∃ x B(x) (by EG).

Since 2 is an element of U such that 2 is even, it is true that ∃ x E(x) (by EG).

E(x) ∧ B(x): means x is both even and odd.

If x = 1 then x is not both even and odd.

If x = 2 then x is not both even and odd.

So there is no element in the universe U which is both even and odd.

So E(x) ∧ B(x) is False for any x in the universe.

Therefore (x) (E(x) ∧ B(x)) is False.

Hence we got that "(x) (E(x) ∧ B(x)) need not be a conclusion from ∃ x E(x) and ∃ x B(x)".

6.6. FORMULAS WITH MORE THAN ONE QUANTIFIER:

In the above parts, we studied the formulas with one quantifier. One may consider the formulas with more than one quantifier.
If we consider a 2–place predicate formula "P(x, y)" where x, y are variables, then the following different cases may exists.

(x) (y) P(x, y)

(x) (∃y) P(x, y)

(∃x) (y) P(x, y)

(∃x) (∃y) P(x, y)

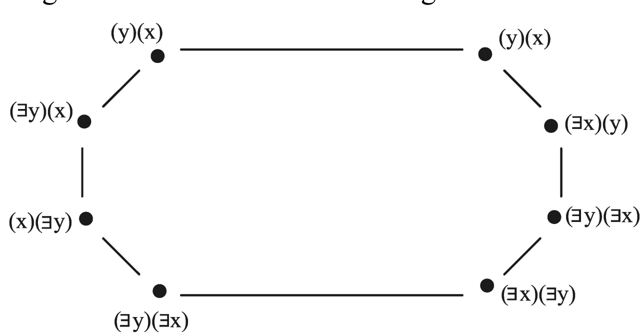
(y) (x) P(x, y)

(y) (∃x) P(x, y)

(∃y) (x) P(x, y)

(∃y) (∃x) P(x, y)

The logical relation of the above mentioned predicate formulas was presented in diagrammatic form in the following:



6.7 SUMMARY:

In earlier lessons, we have already discussed the statement calculus, and inference theory for the statement calculus. We know that the method of derivation is solving the problems related to the predicate formulas uses the rules of inference that discussed for the statement calculus. In addition to those rules of inference discussed for the statement calculus, in derivations, we also use certain rules (or principles). The names of the rules are: Universal Specification, Universal Generalization, Existential Specification, and Existential Generalization. These four additional rules were discussed and related examples were presented for better understanding of the reader.

6.8 TECHNICAL TERMS:

Universal Specification (US)
If (x) p(x) is true, then the universal quantifier can be dropped to obtain "p(c) is true", where c is an arbitrary object in the universe of discourse.

Universal Generalization (UG)
If P(c) is true for all c in the universe of discourse, then the universal quantifier may be prefixed to obtain (x) P(x).

Existential Specification (ES)
If ∃ x P(x) is assumed to be true, then P(c) is true for some element c in the universe of discourse.

Existential Generalization (EG)
If P(c) is true for some element c in the universe of discourse, then we can write "∃ x P(x)" is true.

6.9 SELF ASSESSMENT QUESTIONS:

1. Prove the following statement (transitivity) by using the rules of Inference:

$$(x)\,(R(x) \rightarrow S(x)) \wedge (x)\,(S(x) \rightarrow T(x))$$
$$\Rightarrow (x)\,(R(x) \rightarrow T(x))$$

2. Prove the validity of the following argument by using the rules of inference.
   All birds do have wings. (Premise–1)
   All eagles are birds. (Premise–2)
   Therefore eagles do have wings.

3. Prove that "(x) (M(x))" follows logically from the premises.

$$(x)\,(A(x) \rightarrow M(x)) \text{ and } (x)\,A(x)$$

4. Using predicate logic, Rules of Inference, show that the following argument is valid.
Every husband argues with his wife.
X is a husband.
Therefore, X argues with his wife.

5. Prove that $\exists$ x [p(x) $\vee$ q(x)] $\Rightarrow$ [$\exists$ x p(x)] $\vee$ [$\exists$ x q(x)]

6. Prove that r(x) $\wedge$ (x) q(x) $\Rightarrow$ ($\exists$ x) [r(x) $\wedge$ q(x)]

7. Test the validity of the argument
If a person is rich, he is happy.
If a person is happy, he lives long.
Therefore, Rich persons live long.
Ans: Valid

8. Test the validity of the following argument:
If there is a quarrel by students, the examinations will be postponed.
There was no quarrel by students
Therefore, the examination was not postponed.
Ans: Not Valid

6.10 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd, New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Bhavanari Satyanarayana

LESSON -7

STATE TABLES AND DIAGRAMS

OBJECTIVES:

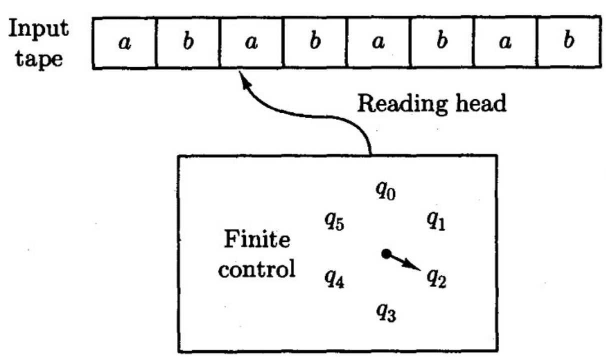
- To understand the finite state machine.
- To know how to construct state tables.
- To draw the state diagrams of finite state machine.
- To learn concepts related to finite state machine.

STRUCTURE:

7.1 INTRODUCTION:

A finite-state machine (FSM) or finite-state automaton, or finite automaton, or simply a state machine, is a mathematical model of computation. It is an abstract machine that can be in exactly one of a finite number of *states* at any given time. The FSM can change from one state to another in response to some inputs; the change from one state to another is called a *transition*. An FSM is defined by a list of its states, its initial state, and the inputs that trigger each transition. The behavior of state machines can be observed in many devices in modern society that perform a predetermined sequence of actions depending on a sequence of events with which they are presented. Simple examples are vending machines, which dispense products when the proper combination of coins is deposited, elevators, whose sequence of steps is determined by the floors requested by riders, traffic lights,

A study of finite automaton is their applicability to the design of several common types of computer algorithms and programs.



Let us now describe the operation of a finite automaton in more detail.

Strings are fed into the device by means of an *input tape*, which is divided into squares, with one symbol inscribed in each tape square (see figure). The main part of the machine itself is a "black box" with innards that can be, at any specified moment, in one of a finite number of distinct internal *states*. This black box - called the *finite control* - can sense what symbol is written at any position on the input tape by means of a movable *reading bead*. Initially, the reading head is placed at the leftmost square of the tape and the finite control is set in a designated *initial state*.

7.2 FINITE STATE MACHINES:

Input: The various inputs applied at the input side of the model are the elements of an input set, $\mathcal{I}$, also called the input alphabet.

Output: The various outputs generated at the output side of the model are the elements of an output set *O*, also called the output alphabet.

Next state function $\delta : \zeta \times \mathcal{I} \rightarrow \zeta$ is a function and
Output function $\theta : \zeta \times \mathcal{I} \rightarrow O$ is a function.

7.2.1 Definition: An input-output machine is a system $M = (\zeta, \mathcal{I}, O, \delta, \theta)$

where ζ is a finite set (called the set of states of the machine), $\mathcal{I}$ is a finite set (called the set of inputs (or input alphabet) of the machine), O is a finite set (called the output alphabet),

$\delta : \zeta \times \mathcal{I} \rightarrow \zeta$ is a function (called the next state function) and $\theta : \zeta \times \mathcal{I} \rightarrow O$ is a function (called the output function).

7.2.2 Notation: (*i*) The non-negative integers denote successive instances of time;
(ii) a_t = the input to the *i*/o – machine (that is, input to the machine) at time t;
(iii) $s(t)$ = state of the machine at time t;
(iv) $s(t+1) = \delta(s(t), a_t)$;
(v) $w(t)$ = output at time t;
(vi) $w(t) = \theta(s(t), a_t)$ [Here θ gives the current output].

7.2.3 Note: If we are not concerned about output only, then we may omit O and θ. In this case, we may define a machine as follows:

7.2.4 Definition: *A* state machine M is $(\zeta, \mathcal{I}, \delta)$, where ζ is a finite set, $\mathcal{I}$ is a finite set and δ is a function from $\zeta \times \mathcal{I}$ to ζ. Here ζ, $\mathcal{I}$ and δ are called the set of states, the set of inputs and the next state function, respectively.

7.2.5 Note: We use term machine to refer either an *i*/o-machine or state machine.

7.2.6 Example: (Parity-check machine): This machine is designed to show whether the total number of 1's in a finite sequence of 0's and 1's is whether even or odd (for example, in the sequence 1100110010, the number of 1's is 5 which is an odd number). Now we define this machine mathematically as follows:

This machine has an input 0 or 1. So $\mathcal{I} = \{0, 1\}$.
States correspond to 'even' or 'odd'. So $\zeta = \{\text{Even, odd}\}$.
We define δ and θ as follows:

$\delta(\text{Even}, 1) = \text{odd}$, $\delta(\text{Even}, 0) = \text{Even}$,
$\delta(\text{odd}, 1) = \text{Even}$, $\delta(\text{odd}, 0) = \text{odd}$,

$\theta(\text{Even}, 1) = 0$, $\theta(\text{Even}, 0) = E$,
$\theta(\text{odd}, 1) = E$, $\theta(\text{odd}, 0) = 0$.

Here we use the symbol '0' for 'odd', and the symbol 'E' for 'even' and so $O = \{0, E\}$.

Note that the last output gives the result.
Table for party check machine:

	$\delta(s, a)$		$\theta(s, a)$	
Input	0	1	0	1
States				
EVEN	EVEN	ODD	*E*	O
ODD	ODD	EVEN	O	*E*

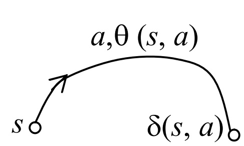
7.3 STATE TABLES AND DIAGRAMS:

In this section, we come to know how to form a table; and how to draw a diagram representing a given input/output machine.

7.3.1 How to draw the directed graph for a given finite machine:

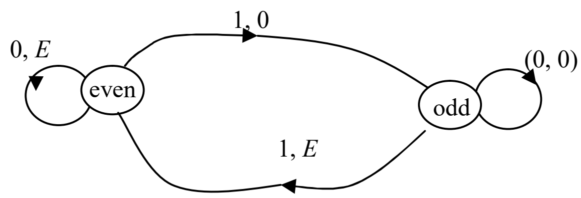
(i) The nodes of the graph are the states of the machine;
(ii) For every input '*a*' and state *s*; define an arc that originates at the node 's' and terminates at the node $\delta(s, a)$.

(iii) Label the arc (described in (ii)) with input "*a*" followed by the output $\theta(s, a)$. The arc is illustrated in the diagram.



7.3.2 Problem: Draw the directed graph for the parity-check machine.

Solution: Following the procedure given in 7.3.1, we get the following graph.



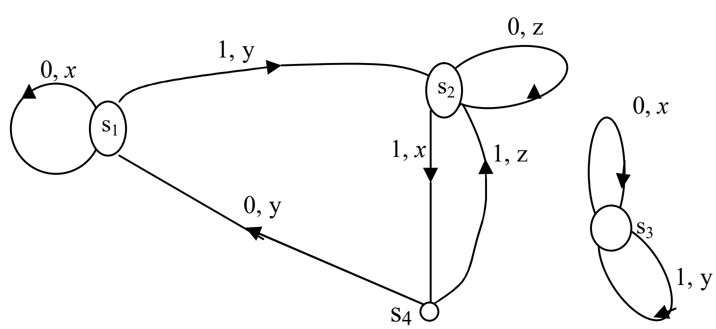
7.3.3 Note: (*i*) The directed graph obtained in the above problem (following the procedure given in 7.3.1) is called the state-diagram of the given machine.
(ii) If the state diagram is known, then we can write the state table and vice versa.

7.3.4 Problem: Draw the state diagram for the machine given by the table.

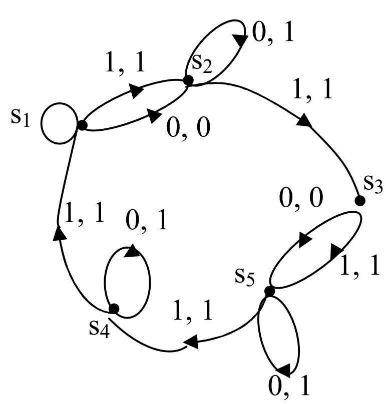
	δ		θ	
	0	1	0	1
s_1	s_1	s_2	x	y
s_2	s_2	s_4	z	x
s_3	s_3	s_3	x	y
s_4	s_1	s_2	y	z

Solution: Here the nodes are s_1, s_2, s_3 and s_4,

The state diagram is given by



7.3.5 Problem: Write the state table for the machine given by the state diagram.



Solution:

	δ		θ	
	0	1	0	1
s_1	s_2	s_2	0	1
s_2	s_2	s_3	1	1
s_3	s_5	s_5	0	1
s_4	s_4	s_1	1	1
s_5	s_5	s_4	1	1

7.3.6 Problem: Construct a machine to add two given binary digits and draw its state diagram.

Solution: (*i*) In the computation for the addition of two binary integers, corresponding digits of the two integers are operated. For this computation, we start with the right most pair of digits. So the corresponding digits of the given two binary integers are fed into the machine simultaneously.

(ii) If the number of the significant digits of the two given numbers are not equal, then we use the symbol *b* (for blank) to fed into the machine. For example, suppose the given numbers are 101 and 11. Then we fed (1, 1) at first step, (0, 1) at second step and (1, *b*) at the third step.

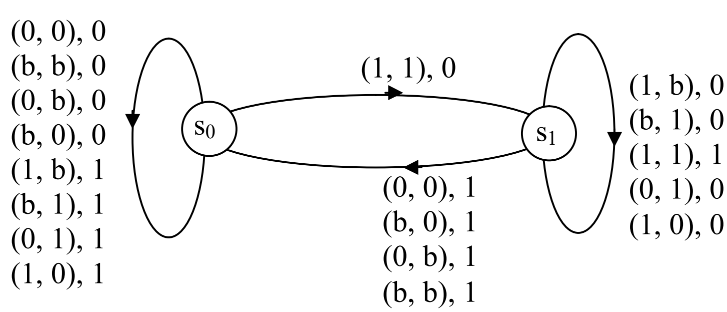
(iii) From the above, we can understand that the input alphabet is $\mathcal{I} = \{(x, y) / x, y \in \{0, 1, b\}\}$.

(iv) The output alphabet is $O = \{0, 1\}$.

(v) The states of the machine correspond to carry.

Therefore $\zeta = \{s_0, s_1\}$, where s_0 stands for the carry "0" and s_1 stands for the carry "1".

(vi) The state diagram to add two binary integers is given in the figure. In this diagram, we eliminated numerous arrows by simply labeling a single arrow with various possible inputs and corresponding outputs.



7.3.7 Example: How to get the sum of 101 and 11 from the above machine?

Step-(*i*): In the beginning, we suppose that the machine is at starting state s_0. Now let us start the procedure. Take the first right most digits of both the given numbers. They are 1, 1. So we use (1, 1).

Step-(ii): Fed the input (1, 1). Since the machine is at the state s_0, the output is "0" and the next state is s_1 (this means, for the next step, the carry is "1").

Step-(iii): Consider the second digits (from right) of the given two numbers. They arc 0, 1. So fed (0, 1) into the machine. Then the output is "0" and the machine still lies in state s_1 (see the diagram).

Step-(iv): Now we have to consider the third digits (from right) of the given numbers. The given second number has only two digits. So we use *b* (blank) for the third place which is not significant. So the input is (1, b). Since the machine is at state s_1 when we fed input (1, *b*), the output is "0" and machine is still in state s_1.

Step-(v): Since the number of digits in the given numbers is not more than 3, the process is completed here. To get the answer, consider the outputs in the order.

The answer is

1	0	0	0	
Final carry	3rd output	2nd output	1st output	

Therefore 101 + b11 = 1000

7.4 SUMMARY:

A finite-state machine (FSM) or finite-state automaton, finite automaton, or simply a state machine, is a mathematical model of computation. It is an abstract machine that can be in exactly one of a finite number of *states* at any given time. The FSM can change from one state to another in response to some inputs; the change from one state to another is called a *transition*. An FSM is defined by a list of its states, its initial state, and the inputs that trigger each transition. In this lesson we have learned basic terminologies of a finite machine and how to draw a state table corresponding to a digraph and vice-versa.

7.5 TECHNICAL TERMS:

Input:
Inputs applied at the input side of the model.

Output:
Outputs generated at the output side of the model and we denote output alphabet as *O*.

Next state function

A function which provides next state ($\delta : \zeta \times \mathcal{I} \rightarrow \zeta$ is a function) is named as next state function.

Output function

A function which provides the out put ($\theta : \zeta \times \mathcal{I} \rightarrow O$ is a function) is named as out put function.

Input-output machine

An input-output machine is a system $M = (\zeta, \mathcal{I}, O, \delta, \theta)$

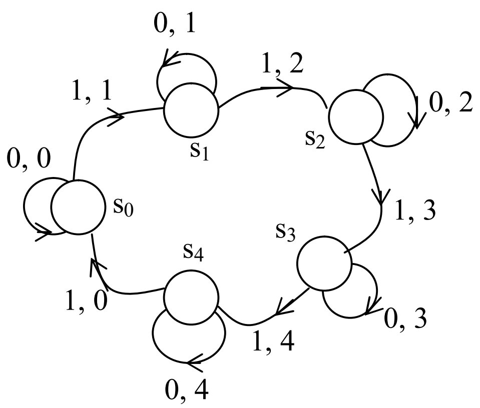
where ζ is a finite set (called the set of states of the machine), $\mathcal{I}$ is a finite set (called the set of inputs (or input alphabet) of the machine), O is a finite set (called the output alphabet),

$\delta : \zeta \times \mathcal{I} \to \zeta$ is a function (called the next state function) and $\theta : \zeta \times \mathcal{I} \to O$ is a function (called the output function).

7.6 SELF ASSESSMENT QUESTIONS:

1. Determine a state diagram for a machine that has input 0 or 1 and outputs the remainder when the number of received 1's is divisible by 5.

Ans:

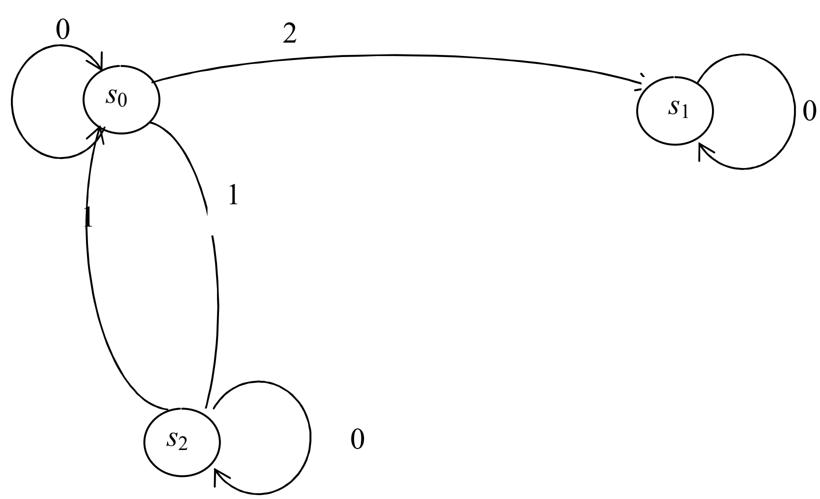


2. Draw the labeled digraphs for the machines given by the State table.

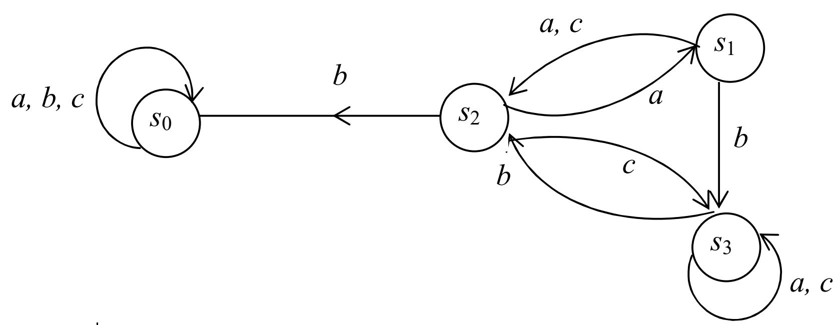
	0	1
s_0	s_0	s_1
s_1	s_1	s_2
s_2	s_2	s_0

.

Ans:



3. Draw the state table for the Finite machine represented by the digraph given below.



Ans:

	a	b	c
s_0	s_0	s_0	s_0
s_1	s_2	s_3	s_2
s_2	s_1	s_0	s_3
s_3	s_3	s_2	s_3

7.7 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998. UTM Springer, 1998.

Prof. Dr. S. Srinadh

LESSON - 8
STATE HOMOMORPHISMS

OBJECTIVES:

- To understand state homomorphism
- To know state machine congruence

STRUCTURE:

8.1 Introduction
8.2 Dynamics
8.3. State homomorphisms
8.4 State Machine Congruence
8.5 Summary
8.6 Technical Terms
8.7 Self Assessment Questions
8.8 Suggested Readings

8.1. INTRODUCTION:

In this lesson, we define the notions: state homomorphism and state machine congruence and prove some important theorems.

8.2. DYNAMICS

8.2.1 Note: Let X be a set.

(i) We define X^* = the set of all finite sequences of elements from X.

The elements of X^* are called **strings.**

(ii) The binary operation "**concatenation**" on X^* is defined as

$(x_1x_2 \ldots x_n).(y_1y_2 \ldots y_k) = z_1z_2 \ldots z_{n+k}$, where $z_i = x_i$ for $1 \leq I \leq n$; and $z_{n+j} = y_j$ for $1 \leq j \leq k$ (that is, $(x_1x_2 \ldots x_n).(y_1y_2 \ldots y_k) = x_1x_2 \ldots x_ny_1y_2 \ldots y_k$).

(iii) The **length** $1(w)$ of the string $w = x_1x_2 \ldots x_n$ is defined to be n.

It is clear that $1(w_1 o w_2) = 1(w_1) + 1(w_2)$.

(iv) The operation "concatenation" is associative.

(v) **Empty sequence** is denoted by "e", and we assume that $e \in X^*$ with

$w o e = w = e o w$ for all $w \in X^*$.

So X^* has an identity element. Thus X^* becomes a monoid.

8.2.2 Note: Consider the state machine $M = (\zeta, \mathcal{I}, \delta)$.

(i) We define $\delta^*: \zeta \times \mathcal{I}^* \to \zeta$ as $\delta^*(s, a) = \delta(s, a)$ for all $s \in \zeta$ and $a \in \mathcal{I}$. If $a_1 a_2 \ldots a_{k-1} a_k$ is in $\mathcal{I}^*$, then we define

$\delta^*(s, a_1 a_2 \ldots a_{k-1} a_k) = \delta(\delta^*(s, a_1 a_2 \ldots a_{k-1}), a_k)$. Here $\delta^*(s, e) = s$ is a convention.

By the definition, it is clear that $\delta^*: \zeta \times \mathcal{I}^* \to \zeta$ is an extension of δ.

(ii) Sometimes we write $(s)w$ to mean $\delta^*(s, w)$ for $w \in \mathcal{I}^*$.

8.2.3 Example: Consider the machine given by the table.

Sates	δ	
	0	1
1	3	1
2	2	3
3	2	1

Here $\delta^*(1, 00) = \delta(\delta^*(1, 0), 0)$ (by definition of δ^*)

$= \delta(\delta(1, 0), 0)$

$= \delta(3, 0) = 2$.

Similarly, $\delta^*(2, 00) = 2$, and $\delta^*(3, 00) = 2$.

8.3 STATE HOMOMORPHISMS:

In this section we define state machine homomorphism and state machine congruence and prove some important theorems.

8.3.1. Definition: Let $M_1 = (\zeta_1, \mathcal{I}, \delta_1)$ and $M_2 = (\zeta_2, \mathcal{I}, \delta_2)$ be state machines.

A function $f: \zeta_1 \to \zeta_2$ is called a **state homomrphism** of M_1 into M_2 if

$f(\delta_1(s_1, a)) = \delta_2(f(s_1), a)$ for all $s_1 \in \zeta_1$, and $a \in \mathcal{I}$.

If f is a bijection, then f is called a **state isomorphism**.

8.3.2. Problem: If f is a state homomorphism from $M_1 = (\zeta_1, \mathcal{I}, \delta_1)$ into $M_2 = (\zeta_2, \mathcal{I}, \delta_2)$, then $f(\delta_1^*(s, w)) = \delta_2^*(f(s), w)$ for all $w \in \mathcal{I}^*$.

Proof: (The proof is by induction on the length of w, where $w \in \mathcal{I}^*$).

Suppose $w \in \mathcal{I}^*$ with $1(w) = 1$.

Then $f(\delta_1^*(s, w)) = f(\delta_1(s, w))$ (by the definition of δ^*)

$= \delta_2(f(s), w)$ (since f is a state homomorphism)

$= \delta_2^*(f(s), w)$ (by the definition of δ^*).

Now we suppose the induction hypothesis.

That is, we suppose the result for all $w \in \mathcal{I}^*$ with $1(w) = k$-1.

Suppose $w \in \mathcal{I}^*$ with $1(w) = k$.

Then $w = w_0w_1$ with $1(w_0) = k - 1$ and $1(w_1) = 1$.

Consider $f(\delta_1^*(s, w)) = f(\delta_1^*(s, w_0w_1))$

$= f(\delta_1(\delta_1^*(s, w_0), w_1))$ (by the definition of δ^*)

$= \delta_2(f(\delta_1^*(s, w_0)), w_1)$ (since f is a state homomorphism)

$= \delta_2(\delta_2^*(f(s), w_0), w_1)$ (by the induction hypothesis)

$= \delta_2^*(f(s), w_0w_1)$ (by the definition of δ^*)

$= \delta_2^*(f(s), w)$ (since $w = w_0w_1$).

Hence the result is true for all $w \in \mathcal{I}^*$.

8.4 STATE MACHINE CONGRUENCE:

8.4.1. Note: Let $M = (\zeta, \mathcal{I}, \delta)$ be a state machine.

(*i*). For any subset P of ζ and $a \in \mathcal{I}$, we define $\delta(P, a) := \{\delta(s, a) \ / \ s \in P\}$.

(ii) If $\wp$ is a partition of ζ, then we denote the class of the partition containing s by $[s]$.

(iii) *A* partition $\wp$ of ζ is said to be a **state machine congruence** if for each subset P in $\wp$ and each input $a \in \mathcal{I}$, we have that the set $\delta(P, a)$ is contained in a unique class of the partition $\wp$. The class containing $\delta(P, a)$ is denoted by $[\delta(P, a)]$.

(iv) If $\wp$ is a state machine congruence on $M = (\zeta, \mathcal{I}, \delta)$, then

$\overline{M} = (\wp, \mathcal{I}, \overline{\delta})$, where $\overline{\delta}(P, a) = [\delta(P, a)]$, is a machine.

8.4.2. Theorem: Let $\wp$ be a state machine congruence on $M = (\zeta, \mathcal{I}, \delta)$. Then there exists a state homomorphism f from M onto $\overline{M} = (\wp, \mathcal{I}, \overline{\delta})$ given by $f(s) = [s]$.

Proof: Since $\wp$ is a partition of ζ, we have that [s] is a unique class containing s.

Therefore $f : \zeta \to \wp$ defined by $f(s) = [s]$ is well defined.

Now it remains to show that $f(\delta(s, a)) = \overline{\delta}(f(s), a)$ (That is, $[\delta(s, a)] = \overline{\delta}([s], a)$).

Since $\wp$ is a machine congruence, by the definition of $\overline{\delta}$ we have that $\overline{\delta}([s], a) = [\delta([s], a)]$ …. (1).

Since $\delta([s], a) = \{\delta(x, a) \ / \ x \in [s]\}$, we have that

$\delta(s, a) \in \delta([s], a)$

$\Rightarrow [\delta(s, a)] = [\delta([s], a)]$

$\Rightarrow [\delta(s, a)] = \overline{\delta}([s], a)$ (from (1))

So we have that $[\delta(s, a)] = \overline{\delta}([s], a)$ …… (2).

Therefore $f(\delta(s, a)) = [\delta(s, a)]$ (by the definition of f)

$= \overline{\delta}([s], a)$ (from (2))

$= \overline{\delta}(f(s), a)$ (by the definition of f)

Now we have to show that f is onto.

For this, take $P \in \wp$.

Since P is an equivalence class, it is non-empty.

Let $s \in P$. Since s is in the equivalence class P, we have that $[s] = P$.

Now $f(s) = [s] = P$.

This shows that f is onto. The proof is complete.

8.4.3. Theorem: Let f be a state homomorphism from the state machine $M = (\zeta, \mathcal{I}, \delta)$ onto the state machine $M_1 = (\zeta_1, \mathcal{I}, \delta_1)$. Then there is a state machine congruence on M such that $\overline{M}$ is isomorphic to M_1.

Proof: **Step-(*i*)**: In this step, we find out a partition of ζ.

Define $x \sim y \Leftrightarrow f(x) = f(y)$ for all $x, y \in \zeta$.

Then $\sim$ is an equivalence relation on ζ and the set $\wp$ of all equivalence classes form a partition for ζ.

Step-(ii): Now we show that $\wp$ is a state machine congruence.

For this, take $P \in \wp$ and 'a' be an input symbol.

Any two elements of $\delta(P, a)$ are of the form $\delta(s, a), \delta(s^1, a)$ where $s, s^1 \in P$.

Now $s, s^1 \in P \Rightarrow f(s) = f(s^1)$.

Then, $f(\delta(s, a)) = \delta_1(f(s), a) = \delta_1(f(s^1), a) = f(\delta(s^1, a))$.

Therefore $\delta(s, a) \sim \delta(s^1, a)$, implies that $\delta(s, a)$ and $\delta(s^1, a)$ belongs to the *s*ame equivalence class.

Hence $\delta(P, a)$ is contained in one equivalence class.

This shows that $\wp$ is a state machine congruence.

Step-(iii): Now we define mapping $g : \overline{M} \rightarrow M_1$.

For this, consider the machine $\overline{M} = (\wp, \mathcal{I}, \overline{\delta})$.

Here the definition of $\overline{\delta}$ is $\overline{\delta}(p, a) = [\delta(p, a)]$.

Define $g : \wp \rightarrow \zeta_1$, by $g([s]) = f(s)$ for each class $[s] \in \wp$.

Step-(iv): Now we show that the mapping g is well defined 1-1, and onto.

Let $[s_1], [s_2] \in \wp$.

Now $[s_1] = [s_2] \Leftrightarrow s_1 \sim s_2$

$\Leftrightarrow f(s_1) = f(s_2)$

$\Leftrightarrow g([s_1]) = g([s_2])$.

This shows that g is well defined and 1-1.

To show that g is onto, let $s^* \in \zeta_1$.

Since f is onto, $f(s) = s^*$ for some s.

Now $g([s]) = f(s) = s^*$.

Hence g is onto.

Step-(v): Now we show that g is a state homomorphism.

Let $[s] \in \wp$ and $a \in \mathcal{I}$.

Now $g(\overline{\delta}[s], a)) = g([\delta(s, a)])$ (by the definition of $\overline{\delta}$)

$= f(\delta(s, a))$ (by the definition of g)

$= \delta_1(f(s), a)$ (since f is a homomorphism)

$= \delta_1(g([s]), a)$ (by the definition of g).

Hence g is a state homomorhism.

Now we proved that g is a bijection and state homomorphism.

Hence $\overline{M} \cong M$, (that is, $\overline{M}$ is isomorphic to M).

8.5 SUMMARY:

In this lesson we have discussed the concepts state homomorphisms, state ismorphisms and state machine congruences. Some theorems also included related to these concepts.

8.6 TECHNICAL TERMS:

1. State homomorphism

A function $f : \zeta_1 \rightarrow \zeta_2$ such that $f(\delta_1(s_1, a)) = \delta_2(f(s_1), a)$ for all $s_1 \in \zeta_1$, and $a \in \mathcal{I}$.

2. State isomorphism

If f is a bijection and a state homomorphism, then it is called as State ismorphism.

*3. S***tate machine congruence**

A partition $\wp$ of ζ is *sa*id to be a state machine congruence if for each subset P in $\wp$ and each input $a \in \mathcal{I}$, we have that the set $\delta(P, a)$ is contained in a unique class of the partition $\wp$.

8.7 SELF ASSESSMENT QUESTIONS:

1. Define state homomorphism.
2. Define state isomorphism.
3. Define state machine congruence

8.8 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998. UTM Springer, 1998.

Prof. Dr.S. Srinadh

LESSON - 9

INPUT / OUTPUT (I/O) – HOMOMORPHISMS

OBJECTIVES:

- To know i/o-homomorphism
- To understand distinguishable states
- To know the concept of reduced machine

Structure:

9.1 INTRODUCTION:

In the beginning of this lesson we explain the concept behavior of the machine with respect to a starting state. We explain the state output machine, input-output homomorphism. Few theorems on these concepts were included.

9.2. BEHAVIOUR OF THE MACHINE:

9.2.1. Definition: (*i*). Let X and Y be two sets. A **behavior** from X to Y is a function $\beta : X^* \setminus \{e\} \to Y$ where X and Y are sets and e is the empty string in X^*.

(ii) For a machine, by the term **behavior** from $\mathcal{I}$ to O, we mean a function

$\beta : \mathcal{I}^* \setminus \{e\} \to O$.

9.2.2. Definition: Let $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ is an *i*/o–machine and $s \in \zeta$ be a fixed state (call it as **starting state**).

Define inductively *a* function β as follows:

$\beta_s : \mathcal{I}^* \setminus \{e\} \to O$ by $\beta_s (a) = \theta(s, a)$ for all $a \in \mathcal{I}$.

$\beta_s (w, a) = \theta(\delta^*(s_0, w), a)$ for all $wa \in \mathcal{I}^* \setminus \{e\}$.

Then β_s is a behavior from $\mathcal{I}$ to O.

9.2.3. Note: Note that $\beta_s(wa)$ is the last output when the input sequence wa is fed to the machine with starting state s.

9.2.4. Definition: A **state output machine** is an *i*/o-machine $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ such that $\theta(s, a) = \rho(\delta(s, a))$ for a function ρ form ζ to O.

The state output machine defined here is denoted by $M = (\zeta, \mathcal{I}, O, \delta, \rho)$.

9.2.5. Note: If M is a state output machine, then

$\delta(s_1, a_1) = \delta(s_2, a_2) \Rightarrow \theta(s_1, a_1) = \rho(\delta(s_1, a_1) = \rho(\delta(s_2, a_2)) = \theta(s_2, a_2)$.

9.2.6. Theorem: Let $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ be an *i*/o-machine. Then there exists a state output machine $M_1 = (\zeta_1, \mathcal{I}, O, \delta_1, \rho)$ and a one-one function f from ζ into ζ_1 such that $\beta_s = \beta_{f(s)}$ for all $s \in \zeta$.

Proof: Step-(i): Write

$\zeta_1 = \{ \frac{s}{z} \ / \ s \in \zeta$ and there exists $t \in \zeta$, $a \in \mathcal{I}$ such that $\delta(t, a) = s$ and

$\theta(t, a) = z\} \cup \{ \frac{s}{\phi} \ / \ s \in \zeta$ and there is no $t \in \zeta$ and $a \in \mathcal{I}$ such that $\delta(t, a) = s\}$.

Define $\delta_1(\frac{s}{z}, a) = \frac{\delta(s, a)}{\theta(s, a)}$,

$\theta_1(\frac{s}{z}, a) = \theta(s, a)$.

Then $M_1 = (\zeta_1, \mathcal{I}, O, \delta_1, \theta_1)$ is an i/o-machine.

Step-(ii): Fix some $z_0 \in O$ and define

$\rho : \zeta \to O$ by $\rho(\frac{s}{z}) = z$ and $\rho(\frac{s}{\phi}) = z_0$.

Now $\theta_1(\frac{s}{z}, a) = \theta(s, a)$ (by definition of θ).

$= \rho(\frac{\delta(s, a)}{\theta(s, a)})$ (by the definition of ρ)

$= \rho(\delta_1(\frac{s}{z}, a))$ (by the definition of δ_1)

Therefore $\theta_1(\frac{s}{z}, a) = \rho(\delta_1(\frac{s}{z}, a))$ which implies that $M_1 = (\zeta_1, \mathcal{I}, O, \delta_1, \rho)$ is a state output machine.

Step-(iii): Define $f : \zeta \to \zeta_1$ as follows:

For $s \in \zeta$, choose some output z such that there exists

$t \in \zeta$ and $a \in \mathcal{I}$ with $\delta(t, a) = s$ and $\theta(t, a) = z$.

Then define $f(s) = \frac{s}{z}$.

If no such output z exists, then define $f(s) = \frac{s}{\phi}$.

Now $f(s_1) = f(s_2) \Rightarrow \frac{s_1}{z_1} = \frac{s_2}{z_2}$

$\Rightarrow s_1 = s_2$. Therefore f is one-one.

Now it remains to show that $\beta_s = \beta_{f(s)}$.

We prove this in the following steps 4, 5 and 6.

Step-(iv): To prove $\beta_s = \beta_{f(s)}$, first we prove that

$$\delta_1^*(f(s), wa) = \frac{\delta^*(s, wa)}{\theta(\delta^*(s, w), a)} \quad \ldots.. \quad (1)$$

for all $w \in \mathcal{I}^{-*}$, $a \notin \mathcal{I}$. This proof is by induction on k, the length of wa.

If $k = 1$, then $1(wa) = 1$

$\Rightarrow w = e \Rightarrow wa = a$.

Also $\delta(s, w) = s$ (since $w = e$).

Now $\delta_1^*(f(s), wa) = \delta_1^*(f(s), a)$ (since $w = e$)

$= \delta_1(f(s), a)$ (since $a \in \mathcal{I}$)

$= \delta_1(\frac{s}{z}, a)$ (by definition of f)

$= \frac{\delta(s, a)}{\theta(s, a)}$ (by the definition of δ_1)

$= \frac{\delta^*(s, wa)}{\theta(\delta^*(s, w), a)}$ (since $w = e$).

Therefore equation (1) is true if $k = 1 = 1(wa)$.

Step-(v): Suppose $k > 1$, and equation (1) is true for all strings wa such that $l(wa) \leq k$. Now suppose wa is of length $k + 1$.

Since $k > 1$, $l(w) > 1$ and so we can write $w = w_1a_1$ for some $w_1 \in \vartheta^*$, $a_1 \in \vartheta$.

Now $l(w) = l(w_1 a_1) = k$.

By induction hypothesis, we have that

$$\delta_1^* (f(s), w) = \delta_1^* (f(s), w_1a_1) = \frac{\delta^*(s, w_1a_1)}{\theta(\delta^*(s, w_1), a_1)}.$$

Now $\delta_1^* (f(s), wa) = \delta_1 (\delta_1^* (f(s), w)), a)$

(by the definition of δ^*)

$= \delta_1 (\delta_1^* (f(s), w_1a_1), a)$ (since $w = w_1a_1$)

$= \delta_1 (\frac{\delta^*(s, w_1a_1)}{\theta(\delta^*(s, w_1), a_1)}, a)$ (by induction hypothesis)

$= \frac{\delta(\delta^*(s, w_1a_1), a)}{\theta(\delta^*(s, w_1a_1), a)}$ (by the definition of δ_1)

$= \frac{\delta^*(s, w_1a_1a)}{\theta(\delta^*(s, w_1a_1), a)}$ (by the definition of δ^*)

$= \frac{\delta^*(s, wa)}{\theta(\delta^*(s, w), a)}$ (since $w_1a_1 = w$)

Hence the equation (1) is true for all sequences wa with $wa \in \vartheta^*$ and $a \in \vartheta$.

Step-(vi): Now $\beta_{f(s)} (wa) = \theta (\delta_1^* (f(s), w), a)$ (by the definition β_s)

$= \rho (\delta_1 (\delta_1^* (f(s), w), a)$ [by the condition $\rho (\delta (s, a) = \theta (s, a)$]

$= \rho(\delta_1^*(f(s), wa))$ (by the definition of δ^*)

$= \rho(\frac{\delta^*(s, wa)}{\theta(\delta^*(s, w), a)})$ (by (1))

$= \theta (\delta^* (s, w), a)$ (by the definition ρ)

$= \beta_s (wa)$.

Hence $\beta_{f(s)} = \beta_s$. The Proof is complete.

9.3. INPUT-OUTPUT HOMOMORPHISM (OR I/O-HOMOMORPHISM):

9.3.1. Definition: Let $M = (\zeta, \mathcal{J}, O, \delta, \theta)$ and $M^1 = (\zeta,^1, \mathcal{J}, O, \delta^1, \theta^1)$ be *i*/o - machines. *A* function $f: \zeta \to \zeta^1$ is *s*aid to be an ***i*/o-homomorphism** if $f(\delta (s, a)) = \delta^1 (f(s), a)$ and $\theta (s, a) = \theta^1 (f(s), a)$.

If f is a bijection, then we *s*ay that f is an ***i*/o-isomorphism**.

9.3.2. Result: Consider the machines $M_1 = (\zeta_1, \vartheta, O, \delta_1, \theta_1)$ and

$M = (\zeta, \mathcal{I}, O, \delta, \theta)$ given in Theorem 9.2.6. Define $g : \zeta_1 \to \zeta$ by $g(\frac{s}{z}) = s$.

Show that g is a *i*/o-homomorphism.

Proof: To show this, we have to show that $g(\delta_1(\frac{s}{z}, a)) = \delta(g(\frac{s}{z}), a)$ …. (*i*) and

$\theta_1(\frac{s}{z}, a) = \theta(g(\frac{s}{z}), a)$ ….. (ii).

Now we prove (*i*),

$g(\delta_1(\frac{s}{z}, a) = g(\frac{\delta(s,a)}{\theta(s,a)})$ (by definition δ_1)

$= \delta(s, a)$ (by the definition of g)

$= \delta(g(\frac{s}{z}), a)$ (since $g(\frac{s}{z}) = s$)

To prove (ii), $\theta_1(\frac{s}{z}, a) = \theta(s, a)$ (by definition of θ_1)

$= \theta(g(\frac{s}{z}), a)$ (by the definition of g).

Hence g in an *i*/o- homomorphism.

9.3.3. Example: Consider the machines $M^1 = (\zeta,^1, \mathcal{I}, O, \delta^1, \theta^1)$, and

$M = (\zeta, \mathcal{I}, O, \delta, \theta)$ given in the tables.

Machine M^1

	δ^1		θ^1	
STATES	0	1	0	1
a	b	a	1	0
b	b	a	1	1
c	c	b	1	1

Machine M

	δ		θ	
STATES	0	1	0	1
0	2	1	1	0
1	3	0	1	0
2	2	1	1	1
3	2	0	1	1
4	4	2	1	1

If we define $f : \zeta \to \zeta^1$ as $f(0) = f(1) = a$, $f(2) = f(3) = \text{b}$, $f(4) = \text{c}$, then f is an *i*/o-homomorphism from M to M^1

9.3.4. Theorem: Let $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ and $M_1 = (\zeta_1, \mathcal{I}, O, \delta_1, \theta_1)$ be *i*/o-machines and let f be an *i*/o-homomorphism from M to M_1.
If s is a state of M, then $\beta_s = \beta_{f(s)}$.

Proof: Now $\beta_s(wa) = \theta(\delta^*(s, w), a)$ (by the definition of β_s)

$= \theta_1(f(\delta^*(s, w)), a)$ (since f is an *i*/o- homomorphism)

$= \theta_1(\delta_1^*(f(s), w), a)$ (since f is a homomorphism)

$= \beta_{f(s)}(wa)$ (by the definition of β_s).

Hence $\beta_s = \beta_{f(s)}$.

9.3.5. Definition: A partition $\wp$ of ζ where $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ is an ***i*/o-machine congruence** if it satisfies the following two conditions:

(*i*) $\delta(P, a)$ is contained in some subset in $\wp$ for each $P \in \wp$ and $a \in \mathcal{I}$, and (ii) $\theta(s, a) = \theta(t, a)$ for all $a \in \mathcal{I}$ and $s, t \in P$.

9.3.6. Theorem: Let $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ be an *i*/o-machine and let $\wp$ an *i*/o-machine congruence. Then $\overline{M} = (\wp, \mathcal{I}, O, \bar{\delta}, \bar{\theta})$ is an *i*/o-machine and the function f from ζ onto $\wp$ given by $f(s) = [s]$ is an *i*/o-homomorphim form M onto $\overline{M}$.

Proof: Since an i/o-machine congruence is also a state machine congruence, we have that any i/o-machine M with an i/o-machine congruence $\wp$ satisfies the hypothesis of the Theorem 8.4.2.

By Theorem 8.4.2., f is a state homomorphism from M onto $\overline{M} = (\wp, \mathcal{I}, \bar{\delta})$.

Define $\bar{\theta}([s], a) = \theta(s, a)$. Now we show that $\bar{\theta}$ is well defined.

Suppose $[s] = [t]$

$\Rightarrow$ t, s $\in$ P for some p $\in \wp$

$\Rightarrow \theta(t, a) = \theta(s, a)$

(by the definition of i/o-machine congruence)

$\Rightarrow \bar{\theta}([t], a) = \bar{\theta}([s], a)$ (by the definition of $\bar{\theta}$).

Now we show that f is an i/o-homomorphism.

Since f is a state homomorphism, by Theorem 8.4.2., it remains to show that $\theta(s, a) = \bar{\theta}(f(s), a)$.

Now $\theta(s, a) = \bar{\theta}([s], a)$ (by the definition of $\bar{\theta}$).

$= \bar{\theta}(f(s), a)$ (by the definition of f).

Hence f is an i/o-homomorphism. The proof is complete.

9.3.7. If f is an i/o- homomorphism from $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ onto $M_1 = (\zeta_1, \mathcal{I}, O, \delta_1, \theta_1)$, then there exists an i/o-machine congruence $\wp$ of ζ such that the mapping $g([s]) = f(s)$ is an i/o-isomorphism from $\overline{M}$ onto M_1.

Proof: Since f is an i/o- homomorphism, it is also *a* state homomorphism.

So by the Theorem 8.4.3., g is a state isomorphism from $\overline{M}$ onto M_1.

It remains to show that $\bar{\theta}([s], a) = \theta_1(g[s], a)$, where $\bar{\theta}$ is defined in Theorem 9.3.6. Now

$\theta_1(g[s], a) = \theta_1(f(s), a)$ (by the definition of g)

$= \theta(s, a)$ (since f is an i/o-homomorphism)

$= \bar{\theta}([s], a)$ (by the definition of $\bar{\theta}$).

Therefore $\bar{\theta}([s], a) = \theta_1(g[s], a)$.

Hence g is an i/o-homomorphism from $\overline{M}$ onto M_1.

9.3.8. Note: Let $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ be a machine. Define $s \sim t \Leftrightarrow \beta_s = \beta_t$.

This is an equivalence relation on ζ.

We write ζ_R = the set of all equivalence classes.

The states in any class have identical behavior. Now let δ_R be $\bar{\delta}$, and θ_R be $\bar{\theta}$. Now $\zeta_{\cdot R}$ is an *i*/o-machine congruence .

The system $M_R = (\zeta_R, \mathcal{I}, O, \delta_R, \theta_R)$ is an *i*/o-machine.

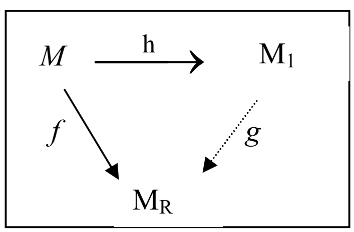
This M_R is called the **reduced machine** of the given machine *M*.

Here $\delta_R(P, a) = \bar{\delta}(P, a) = [\delta\ (P, a)]$ which is defined in Note 8.4.1 and

$\theta_R([s], a) = \bar{\theta}([s], a) = \theta(s, a)$ which is defined in Theorem 9.3.6.

If we define $f : \zeta \to \zeta_R$ by $f(s) = [s]$, then by the Theorem 9.3.6., f is an *i*/o-homomorphism. This f is called the **natural *i*/o- homomorphism**.

9.3.9. Theorem: Let $M = (\zeta, \mathcal{I}, O, \delta, \theta)$ be an *i*/o-machine and M_R is its reduced machine. If h is an *i*/o-homomorphism from M onto M_1, then there exists an *i*/o-homomorphism g from M_1 onto M_R such that $f = g \circ h$, where f is the natural *i*/o-homomorphism form M onto M_R.



Proof: Let $M_1 = (\zeta_1, \mathcal{I}, O, \delta_1, \theta_1)$. We define $g : \zeta_1 \to \zeta_R$ as follows:

Let $s_1 \in \zeta_1$. Since h is onto, there exists $s \in \zeta$ such that $h(s) = s_1$.

Now we define $g(s_1) = g(h(s)) = [s]$.

Now we show that g is well defined. Let $s^1, t^1 \in \zeta_1$ such that $s^1 = t^1$.

Suppose $s^1 = h(s)$ and $t^1 = h(t)$. Now $s^1 = h(s) = h(t) = t^1$.

Since h is an *i*/o-homomorphism, by Theorem 10.2.7, $\beta_s = \beta_{f(s)}$ and $\beta_t = \beta_{h(t)}$. Since $h(s) = h(t)$, we have $\beta_s = \beta_{f(s)} = \beta_{h(t)} = \beta_t$

$$\Rightarrow s \sim t \Rightarrow s, t \in P \text{ for some } P \in \zeta_R$$

$$\Rightarrow [s] = P = [t]$$

$\Rightarrow g(h(s)) = g(h(t))$

$\Rightarrow g(s^1) = g(t^1)$.

Now we show that $f = g \circ h$. By the definition of g, we have that

$g(h(s)) = [s] = f(s)$ (by the definition of natural homomorphism)

$\Rightarrow g \circ h = f$.

Now we show that g is an i/o-homomorphism.

For this we have to show that $g(\delta_1(s_1, a)) = \delta_R(g(s_1), a)$

and $\theta_1(s_1, a) = \theta_R(g(s_1), a)$.

Since h is onto, there exists $s \in \zeta$ such that $h(s) = s_1$.

Now $g(\delta_1(s_1, a)) = g(\delta_1(h(s), a)$

$= g(h(\delta(s, a)))$ (since h is an i/o-homomorphism)

$= [\delta(s, a)]$ (by the definition of g)

$= \bar{\delta}([s], a)$ (by the definition of $\bar{\delta}$)

$= \delta_R([s], a)$ (since $\delta_R = \bar{\delta}$)

$= \delta_R(g(h(s)), a)$ (by the definition of g)

$= \delta_R(g(s_1), a)$ (since $h(s) = s_1$)

Now $\theta_1(s_1, a) = \theta_1(h(s), a)$ (since $h(s) = s_1$)

$= \theta(s, a)$ (since h is an i/o-homomorphism)

$= \bar{\theta}([s], a)$ (by the definition of $\bar{\theta}$)

$= \theta_R([s], a)$ (since $\theta_R = \bar{\theta}$)

$= \theta_R(g(h(s)), a)$ (by the definition of g)

$= \theta_R(g(s_1), a)$ (since $h(s) = s_1$)

Hence g is an i/o-homomorphism such that $f = g \circ h$.

9.4 SUMMARY:

In the beginning of this lesson we explained the concept behavior of the machine with respect to a starting state. We explained the state output machine, input-output homomorphism, reduced machine. Few theorems on these concepts were included.

9.5 TECHNICAL TERMS:

1. **Behavior**

   Behavior from $\mathcal{I}$ to O, we mean a function $\beta : \mathcal{I}^* \setminus \{e\} \to O$.

2. **State output machine**

A State output machine is an *i*/o-machine $M = (\zeta, \mathcal{J}, O, \delta, \theta)$ such that

$\theta(s, a) = \rho(\delta(s, a))$ for a function ρ form ζ to O, it is denoted by

$M = (\zeta, \mathcal{J}, O, \delta, \rho)$.

3. ***i*/o-homomorphism**

$M = (\zeta, \mathcal{J}, O, \delta, \theta)$ and $M^1 = (\zeta,^1, \mathcal{J}, O, \delta^1, \theta^1)$ be *i*/o - machines. $f : \zeta \to \zeta^1$ is an *i*/o-homomorphism if $f(\delta(s, a)) = \delta^1(f(s), a)$ and $\theta(s, a) = \theta^1(f(s), a)$.

9.6 SELF ASSESSMENT QUESTIONS:

1. Define i/o homomorphism between two finite state machines.
2. Give an example of i/o homomorphism.
3. Define reduced machine.

9.7 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998. UTM Springer, 1998.

Prof. S. Srinadh

LESSON -10

REDUCED MACHINE AND ALGORITHM

OBJECTIVES:

- To calculate equivalence classes of states
- To understand the algorithm for a reduced machine
- To know the concept reduced machine

STRUCTURE:

10.1 INTRODUCTION:

In this section we provide algorithm for reduced machine and provide some illustration.

10.2 DISTINGUISHABLE STATES:

10.2.1 Note: (*i*) States s and t of an *i*/o–machine are said to be **distinguishable** if $\beta_s \neq \beta_t$.

(ii) An input sequence $w \in \mathscr{I}^*$ is said to **distinguish** the states s and t if $\beta_s(w) \neq \beta_t(w)$.

10.2.2. Example: Consider the machine given by the table. Now we recollect the definition of β_s.

$\beta_s(\text{a}) = \theta(s, a)$ for all $a \in \mathscr{I}$ and $\beta_s(wa) = \theta(\delta^*(s, w), a)$.

States	δ		θ	
	0	1	0	1
0	2	1	1	0
1	3	0	1	0
2	2	1	1	1

3	2	0	1	1
4	4	2	1	1

(i) Observe that $\beta_0(1) = \theta(0, 1)$

$$= 0 \neq 1 = \theta(2, 1) = \beta_2(1).$$

Therefore 0, 2 are distinguishable.

The input "1" distinguishes the states "0" and "2".

(ii) $\beta_2(a) = \theta(2, a)$

$$= \theta(4, a) = \beta_4(a), \text{ for all } a \in \mathcal{I}.$$

(iii) $\beta_2(11) = \theta(\delta^*(2, 1), 1)$

$$= \theta(\delta(2, 1), 1) = \theta(1, 1) = 0,$$

$\beta_4(11) = \theta(\delta^*(4, 1), 1)$

$$= \theta(\delta(4, 1), 1) = \theta(2, 1) = 1.$$

Hence the input string "11" distinguishes the states 2 and 4.

10.2.3. Note: (*i*) Let k be a non-negative integer. Two states s and t said to be ***k*-equivalent** if $\beta_s(w) = \beta_t(w)$ for all strings w of length k;

(ii) k-equivalence is an equivalence relation;

(iii) $\beta_s = \beta_t \Leftrightarrow s$ and t are k-equivalent for all k..

10.3 ALGORITHM TO FIND Z_R OF A GIVEN MACHINE:

10.3.1 Algorithm:

Step-(*i*): Determine 1-equivalent classes. (Here, s is 1-equivalent to t if $\theta(s, a) = \theta(t, a)$ for all $a \in \mathcal{I}$).

Step-(ii): Set $k = 1$.

Step-(iii): Set $k =$ (previous value of k) + 1.

Step-(iv): By using the $(k - 1)$–equivalence classes find k-equivalence classes.

[How to find this ? Suppose $k \geq 2$. Also suppose s is $(k - 1)$–equivalent to t.

If $s_i = \delta(s, a_i)$, and $t_i = \delta(t, a_i)$ are $(k-1)$-equivalent for all $a_i \in \mathcal{I}$, then s and t are k-equivalent. Otherwise s and t are not k-equivalent].

Step-(v): If (k-1)-equivalent classes are not identical with k-equivalent classes, then go to step-(iii).

Step-(vi): Write ζ_R : = the set of all k-equivalence classes.

Step-(vii): Stop.

Step-(viii): End.

10.3.2 Problem: Find ζ_R and hence the reduced machine M_R for the machine given by the table.

States	δ		θ	
	0	1	0	1
s_1	s_2	s_5	1	0
s_2	s_5	s_5	1	1
s_3	s_1	s_8	1	1
s_4	s_8	s_2	1	0
s_5	s_6	s_5	1	1
s_6	s_1	s_5	1	1
s_7	s_2	s_3	1	0
s_8	s_3	s_5	1	1

Solution: The input 0 does not distinguish any two states.

The input "1" distinguishes the states s_1, s_4, s_7 from s_2, s_3, s_5, s_6, s_8.

Therefore 1-equivalnet classes are $\{s_1, s_4, s_7\}$ and $\{s_2, s_3, s_5, s_6, s_8\}$.

Now we check whether s_1 and s_4 2-equivalent.

$s_2 = \delta(s_1, 0)$, $s_8 = \delta(s_4, 0)$ are 1-equivalent, and

$s_5 = \delta(s_1, 1)$, $s_2 = \delta(s_4, 1)$ are 1-equivalent

$\Rightarrow$ s_1 and s_4 are 2-equivalent.

Similarly, s_1, s_7 are 2-equivalent. Hence $\{s_1, s_4, s_7\}$ is a 2-equivalence class.

Now we check whether s_2 and s_3 2-equivalent. $s_5 = \delta(s_2, 0)$, and $s_1 = (s_3, 0)$ are not 1-equivalnet. Therefore s_2 and s_3 are not 2-equivalent.

Similarly, we observe that s_2, s_5 are 2-equivalent; s_2, s_8 are 2-equivalent; and s_3, s_6 are 2-equivalent.

Therefore, the 2-equivalent classes are $\{s_1, s_4, s_7\}$, $\{s_2, s_5, s_8\}$, $\{s_3, s_6\}$.

In the same way, we find the 3-equivalent classes.

The 3-equivalent classes are $\{s_1, s_4\}$, $\{s_7\}$, $\{s_2\}$, $\{s_5, s_8\}$, $\{s_3, s_6\}$.

The 4-equivalent classes are $s_1\}$, $\{s_2\}$, $\{s_4\}$, $\{s_5, s_8\}$, $\{s_7\}$, $\{s_3, s_6\}$.

The 5-equivalent classes are $\{s_1\}$, $\{s_2\}$, $\{s_3, s_6\}$, $\{s_4\}$, $\{s_5, s_8\}$, $\{s_7\}$.

Since the 4-equivalent classes are identical to the 5-equivlent classes, the process will terminate here.

Therefore $\zeta_R = \{ \{s_1\}, \{s_2\}, \{s_3, s_6\}, \{s_4\}, \{s_5, s_8\}, \{s_7\}\}$.

Now the reduced machine $M_R = \{\zeta_R, \mathcal{I}, O, \delta_R, \theta_R)$ is given by

States	$\delta_R = \overline{\delta}$		$\theta_R = \overline{\theta}$	
	0	1	0	1
$\{s_1\}$	$\{s_2\}$	$\{s_5, s_8\}$	1	0
$\{s_2\}$	$\{s_5, s_8\}$	$\{s_5, s_8\}$	1	1
$\{s_3, s_6\}$	$\{s_1\}$	$\{s_5, s_8\}$	1	1
$\{s_4\}$	$\{s_5, s_8\}$	$\{s_2\}$	1	0
$\{s_5, s_8\}$	$\{s_3, s_6\}$	$\{s_5, s_8\}$	1	1
$\{s_7\}$	$\{s_2\}$	$\{s_3, s_6\}$	1	0

10.3.3. Problem: Minimize the number of states for the machine given by the following state table.

States	δ		θ	
	0	1	0	1
s_0	s_0	s_2	0	0
s_1	s_2	s_5	1	0
s_2	s_2	s_2	1	1
s_3	s_1	s_1	1	1
s_4	s_2	s_3	0	1
s_5	s_4	s_5	1	1
s_6	s_2	s_6	1	1

Solution: We know that 1-equivalence is defined as follows: s is 1-equivelnce to t if $\theta(s, a) = \theta(t, a)$ for all $a \in I = \{0, 1\}$.

Now 1-equivalent classes for the given machine are

$\{s_0\}$, $\{s_1\}$, $\{s_2, s_3, s_5, s_6\}$, $\{s_4\}$.

Now we find the 2-equivalent classes.

Clearly $\{s_0\}$, $\{s_1\}$, $\{s_4\}$ are 2-equivalent class.

Now we check whether s_2, s_3 2-equivalent.

$\delta(s_2, 0) = s_2$, $\delta(s_3, 0) = s_1$, $\delta(s_2, 1) = s_2$, $\delta(s_3, 1) = s_1$, and s_2, s_1 are not 1-equivalent.

Therefore s_2, s_3 are not 2-equivalent.

Now we check whether s_2, s_5 2-equivalent.

$\delta(s_2, 0) = s_2$, $\delta(s_5, 0) = s_4$, $\delta(s_2, 1) = s_2$, $\delta(s_5, 1) = s_5$, and s_2, s_4 are not 1-equivalent.

Therefore s_2, s_5 are not 2-equivalent.

Next we verify whether s_2, s_6 2-equivalent.

$\delta(s_2, 0) = s_2$, $\delta(s_2, 1) = s_2$, $\delta(s_6, 0) = s_2$, $\delta(s_6, 1) = s_6$, and s_2, s_6 are 1-equivalent.

Therefore s_2, s_6 are 2-equivalent. Hence the 2-equivalent classes are

$\{s_0\}$, $\{s_1\}$, $\{s_2, s_6\}$, $\{s_3\}$, $\{s_4\}$, $\{s_5\}$.

Now we find the 3-equivalence classes.

We check whether s_2, s_6 3-equivalent.

$\delta(s_2, 0) = s_2$, $\delta(s_6, 0) = s_2$, $\delta(s_2, 1) = s_2$, $\delta(s_6, 1) = s_6$. Therefore s_2, s_6 are 3-equivalent.

Hence the 3-equivalent classes are $\{s_0\}$, $\{s_1\}$, $\{s_2, s_6\}$, $\{s_3\}$, $\{s_4\}$, $\{s_5\}$.

Observe that the 2-equivalent classes and 3-equivalent classes are identical.

The reduced machine is given in the table.

States	$\bar{\delta}$		$\bar{\theta}$	
	0	1	0	1
$\{s_0\}$	$\{s_0\}$	$\{s_2, s_6\}$	0	0
$\{s_1\}$	$\{s_2, s_6\}$	$\{s_5\}$	1	0
$\{s_2, s_6\}$	$\{s_2, s_6\}$	$\{s_2, s_6\}$	1	1
$\{s_3\}$	$\{s_1\}$	$\{s_1\}$	1	1
$\{s_4\}$	$\{s_2, s_6\}$	$\{s_3\}$	0	1
$\{s_5\}$	$\{s_4\}$	$\{s_5\}$	1	1

10.4. Construction

10.4.1. Note: To give an electronic construction of an *i*/o-machine, the state table must be described in terms of Boolean functions. To do this

(i) Code the input and output alphabets in binary

(ii) Code the set of states in binary

(iii) Describe the output and next state functions as Boolean functions.

This procedure was illustrated in the following problem.

10.4.2. Problem: Describe the given machine in terms of Boolean functions.

States	δ		θ	
	0	1	0	1
0	1	2	c	d
1	2	0	a	b
2	2	3	a	b
3	1	0	c	d

Solution: We solve this in three parts (*i*), (ii), and (iii).

(*i*) First we code the input and output alphabet:

Output	z_1	z_2
a	0	0
b	0	1
c	1	0
d	1	1

Table 1

State	x_1	x_2
0	0	0
1	0	1
2	1	0
3	1	1

Table 2

For this example $\mathcal{I}$ = {0, 1} and O = {a, b, c, d}.

The input alphabet is already in binary.

Since O contains four elements, we may use 00, 01, 10, 11.

So label a, b, c, d with 00, 01, 10, 11 respectively.

Call the two's digit z_1 and the units digit z_2. Observe table-*I*.

(ii) Now we code the set of states. Here ζ = {0, 1, 2, 3}.

Label 0, 1, 2, 3 with 00, 01, 10, 11.

Call the two's digit of this representation as x_1 and units digit as x_2.

Observe the table-II.

(iii) Now we describe the output and next state functions as Boolean functions. Suppose x_1^1 represents the two's digit and x_2^1 represents the

States		Input	δ		θ	
x_1	x_2	y	x_1^1	x_2^1	z_1	z_2
0	0	0	0	1	1	0
0	0	1	1	0	1	1
0	1	0	1	0	0	0
0	1	1	0	0	0	1
1	0	0	1	0	0	0
1	0	1	1	1	0	1
1	1	0	0	1	1	0
1	1	1	0	0	1	1

Table III

units digit of the next state.

The symbol y represents the input. Then we have table-III.

From the table, we can observe that x_1^1, x_2^1, z_1 and z_2 are functions of x_1, x_2 and y. Now it is clear that

$$x_1^1 = \overline{x_1}\,\overline{x_2}\,y \vee \overline{x_1}\,x_2\,\overline{y} \vee x_1\,\overline{x_2}\,\overline{y} \vee x_1\,\overline{x_2}\,y$$

$$x_2^1 = \overline{x_1}\,\overline{x_2}\,\overline{y} \vee x_1\,\overline{x_2}\,y \vee x_1 x_2\,\overline{y}.$$

$$z_1 = \overline{x_1}\,\overline{x_2}\,\overline{y} \vee \overline{x_1}\,\overline{x_2}\,y \vee x_1 x_2\,\overline{y} \vee x_1 x_2 y$$

$$z_2 = \overline{x_1}\,\overline{x_2}\,y \vee \overline{x_1}\,x_2 y \vee x_1\,\overline{x_2}\,y \vee x_1 x_2 y.$$

10.4.3. Note: Consider the Problem 10.4.2. A gating network can represent the functions x_1^1, x_2^1, z_1 and z_2 described. Now we draw the gating network.

(i) The functions z_1 and z_2 are available as out put from getting network.

(ii) Since the next state is a function of its previous state the functions x_1^1 and x_2^1 are stored in a delay and are fed along with the input into the machine at the next time period.

10.4.4. Problem: Draw the getting network for the machine given in the Problem 10.4.2.

Solution: In the Problem 10.4.2., we obtained representation of x_1^1, x_2^1, z_1 and z_2 in terms of x_1, x_2 and y.

The first input to the getting network is x_1, x_2, y.

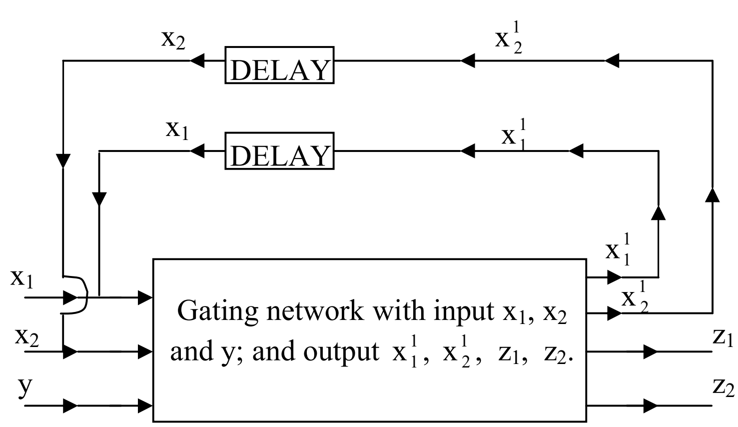
The second input to the getting network is x_1^1, x_2^1, y.

(Here x_1^1, x_2^1 are first outputs).

The outputs x_1^1, x_2^1 are stored in a delay and fed into the machine as input for the next period.

The outline of the getting network is given below.

We call this getting network as '**realization**' of the given machine.



10.4.5. Note: In the above 10.4.4., we obtained a realization for the given machine. If one wants to find the further simplified electronic getting network (that is, a more **simplified realization**), first we have to use Quine–Mc Cluskey minimization procedure for the functions x_1^1, x_2^1, z_1 and z_2 and then we have to draw the realization.

10.5 SUMMARY:

In this lesson, we have learned to find the equivalence classes of the set of all states and hence provided an algorithm to find its reduced machine. Also some problems related to the reduced machine were presented.

10.6 TECHNICAL TERMS:

Distinguishable states
States s and t of an *i*/o–machine are said to be **distinguishable** if $\beta_s \neq \beta_t$.

k-equivalent states
Two states s and t said to be ***k*-equivalent** if $\beta_s(w) = \beta_t(w)$ for all strings w of length k;

Algorithms for set of States of the reduced machine
(Algorithm 10.3.1.)

10.7 SELF ASSESSMENT QUESTIONS:

1. Minimize the number of states for the machine given by the following state table.

States	δ		θ	
	0	1	0	1
s_0	s_0	s_2	0	0
s_1	s_2	s_5	1	0
s_2	s_2	s_2	1	1
s_3	s_1	s_1	1	1
s_4	s_2	s_3	0	1
s_5	s_4	s_5	1	1
s_6	s_2	s_6	1	1

Ans: We know that 1-equivalence is defined as follows:

s is 1-equivelnce to t if $\theta(s, a) = \theta(t, a)$ for all $a \in \mathcal{I} = \{0, 1\}$.

Step 1: Now 1-equivalent classes for the given machine are

$\{s_0\}$, $\{s_1\}$, $\{s_2, s_3, s_5, s_6\}$, $\{s_4\}$.

Step 2: Now we find the 2-equivalent classes.

Clearly $\{s_0\}$, $\{s_1\}$, $\{s_4\}$ are 2-equivalent class.

Now we check whether s_2, s_3 2-equivalent.

$\delta(s_2, 0) = s_2$, $\delta(s_3, 0) = s_1$, $\delta(s_2, 1) = s_2$, $\delta(s_3, 1) = s_1$, and

s_2, s_1 are not 1-equivalent.

Therefore s_2, s_3 are not 2-equivalent.

Now we check whether s_2, s_5 2-equivalent.

$(s_2, 0) = s_2$, $\delta(s_5, 0) = s_4$, $\delta(s_2, 1) = s_2$, $\delta(s_5, 1) = s_5$, and

s_2, s_4 are not 1-equivalent.

Therefore s_2, s_5 are not 2-equivalent.

Next we verify whether s_2, s_6 2-equivalent.

$\delta(s_2, 0) = s_2$, $\delta(s_2, 1) = s_2$, $\delta(s_6, 0) = s_2$, $\delta(s_6, 1) = s_6$, and s_2, s_6 are 1-equivalent.

Therefore s_2, s_6 are 2-equivalent.

Hence the 2-equivalent classes are $\{s_0\}, \{s_1\}, \{s_2, s_6\}, \{s_3\}, \{s_4\}, \{s_5\}$.

Step 3: Now we find the 3-equivalence classes.

We check whether s_2, s_6 3-equivalent.

$\delta(s_2, 0) = s_2, \delta(s_6, 0) = s_2, \delta(s_2, 1) = s_2, \delta(s_6, 1) = s_6$.

Therefore s_2, s_6 are 3-equivalent.

Hence the 3-equivalent classes are $\{s_0\}, \{s_1\}, \{s_2, s_6\}, \{s_3\}, \{s_4\}, \{s_5\}$.

Observe that the 2-equivalent classes and 3-equivalent classes are identical.

The reduced machine is given in the table.

States	$\bar{\delta}$		$\bar{\theta}$	
	0	1	0	1
$\{s_0\}$	$\{s_0\}$	$\{s_2, s_6\}$	0	0
$\{s_1\}$	$\{s_2, s_6\}$	$\{s_5\}$	1	0
$\{s_2, s_6\}$	$\{s_2, s_6\}$	$\{s_2, s_6\}$	1	1
$\{s_3\}$	$\{s_1\}$	$\{s_1\}$	1	1
$\{s_4\}$	$\{s_2, s_6\}$	$\{s_3\}$	0	1
$\{s_5\}$	$\{s_4\}$	$\{s_5\}$	1	1

10.8 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.
2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.
3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237
4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998. UTM Springer, 1998.

Prof. Dr. Kuncham Syam Prasad

LESSON -11
SOME PROPERTIES OF LATTICES

OBJECTIVES:

- To know the system Lattice.
- To understand the concepts Algebraic Lattice, Ordered Lattice.
- To identify different types of relations.
- To Learn to draw the diagrams related to lattices.
- To have proper understanding of different properties.
- To develop skills in solving the problems.

STRUCTURE:

11.1. Introduction

The present day concept of lattice was first considered by E. Schroder about the year 1890. At the same time, R. Dedekind developed a similar concept in his work on groups and ideals. Dedekind defined modular and distributive lattices, which are different types of lattices. The lattice theory developed rapidly around 1930, when G. Birkhoff started his contribution to the lattice theory.

11.2. Partial Order relations, PO sets, Hasse Diagrams.

The concept 'relation' plays an important role in algebraic structures. Different types of relations which play a vital role are: equivalence relations, functions, totally ordered relations, partially order relations, etc.

11.2.1. Definitions: Let A and B be sets.

(i). A relation R from A to B is a subset of $A \times B$, the Cartesian product of A and B.

(ii). Relations from A to A are called relations on A.

(iii). If $(a, b) \in R$, we write aRb and say that "a is in relation R to b".

(iv) If we consider a set A together with a relation R, then we write (A, R).

11.2.2. Definitions: Let R be a relation on a set A. Then

(i). R is said to be a **reflexive relation** if
aRa for all $a \in A$.

(ii) R is said to be a **symmetric relation** if
$aRb \Rightarrow bRa$ for all $a, b \in A$.

(iii) R is said to be an **antisymmetric relation** if
aRb and $bRa \Rightarrow a = b$ for all $a, b \in A$.

(iv) R is said to be a **transitive relation** if
aRb and $bRc \Rightarrow aRc$ for all $a, b, c \in A$.

(v) A relation R is said to be an **equivalence relation** on A if it is reflexive, symmetric, and transitive.

In this case, for any $a \in A$, we write $[a] := \{b \in A \,/\, aRb\}$ and this set is called the equivalence class of a.

(vi). A reflexive, antisymmetric, and transitive relation R on a set A is called a **partial order relation**. In this case, (A, R) is called a partially ordered set (or POset, in short).

11.2.3. Note: (i) In case of partially ordered relation, we may write $\leq$ or $\subseteq$ or $\geq$ instead of R.

(ii) Now let us write $\leq$ instead of R.

(iii) Partially ordered finite sets $(A, \leq)$ can be graphically represented by **Hasse diagrams**.

Here the elements of A are represented as points on a plane.

If $b \leq a$ and $b \neq a$, then we write $b < a$.

If $b < a$ and there is no c in A such that $b < c < a$, then we say that a covers b.

If a covers b, then we mark a point representing

a above the point for b, and connect the points of a and b by a line segment.

11.2.4. Examples: (i) The fact a covers b is illustrated in the following Figures: Fig-1 and Fig-2.

(ii) Now consider the Fig - 2.
In this, we can observe the following facts:
D covers E; B covers C; F covers C; A covers F. Also note that B joined to E by a sequence of line segments all going downwards.
So we have $B \geq E$.

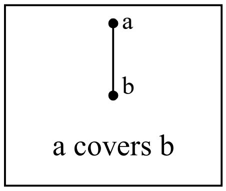


Fig - 1

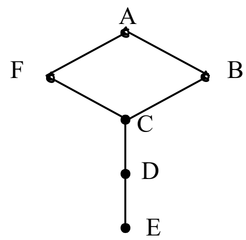


Fig - 2

11.2.5. Examples: (i) The Hasse diagram of the POset

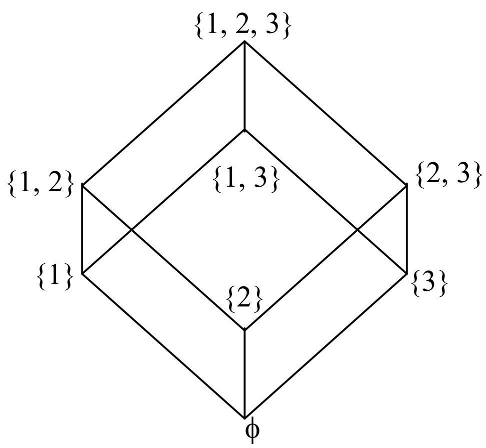


Figure 3

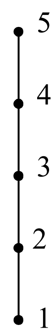


Figure 4

(P({1, 2, 3}), $\subseteq$) is shown in Figure-3, where P(S) denotes the power set of S (that is, the set of all subsets of S).

(ii) The Hasse diagram of ({1, 2, 3, 4, 5}, $\leq$), where $\leq$ means usual "less than or equal to" is shown in figure - 4.

(iii) Write A = {a, b, c, d}, and

R = {(a,a), (b,b), (c,c), (d,d), (b,a), (c,a), (d,a), (d,c)}.

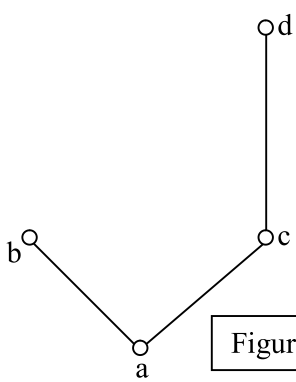


Figure 5

Now it is easy to verify that R is a partial order on the set A. In this example we take aRb as $a \geq b$.

The diagram for this POset is given in figure - 5.

11.2.6. Definition: A partial order relation $\leq$ on A is said to be a linear order if for each $a, b \in A$ either $a \leq b$ or $b \leq a$ holds.

In this case, $(A, \leq)$ is called a linearly ordered set or a chain or a totally ordered set.

11.2.7. Examples: (i) The partially ordered set $(\{1, 2, 3, 4, 5\}, \leq)$ is a chain.

(ii) The partially ordered set $(P(\{1, 2, 3\}), \subseteq)$ is not a chain.

11.2.8. Definitions: (i). Let R be a relation from A to B.

Then we define a relation R^{-1} from B to A by

$$(a, b) \in R^{-1} \Leftrightarrow (b, a) \in R.$$

This relation R^{-1} is called the inverse relation (or transpose relation) of R.

In other words, if $(A, \leq)$ is a partially ordered set, then $(A, \geq)$ is also a partially ordered set, and $\geq$ is the inverse relation to $\leq$..

(ii) Let $(A, \leq)$ be a POset, and $B \subseteq A$. We say that an element a in A is said to be a greatest element if all other elements are smaller than a (that is, $x \leq a$ for all $x \in A$).

(iii) An element b in A is said to be a smallest element of A if $b \leq x$ for all $x \in A$.

(iv) An element c in A is said to be a maximal element of A if "no element is bigger than c " (that is, $c \leq x \Rightarrow c = x$ for all $x \in A$).

(v) An element $d \in A$ is said to be a minimal element of A if $x \leq d \Rightarrow x = d$ for all $x \in A$.

(vi) $a \in A$ is called an upper bound of B if $b \leq a$ for all $b \in B$.

(vii) $a \in A$ is called a lower bound of B if $a \leq b$ for all $b \in B$

(viii) The greatest amongst the lower bounds of B, whenever it exists, is called the infimum of B, and is denoted by inf B.

(ix) The least upper bound of B, whenever it exists, is called the supremum of B, and is denoted by sup B.

(x). We write $\inf(a_1, \ldots, a_n)$ and $\sup(a_1, \ldots, a_n)$ instead of $\inf\{a_1, \ldots, a_n\}$ and $\sup\{a_1, \ldots, a_n\}$, respectively.

11.2.9. Note: Let $(A, \leq)$ be a PO set. Then we have the following:

(i) A has at most one greatest and one smallest element.

(ii) There may be none, one, or several maximal (or minimal) elements in a POset.

(iii) Every greatest element is maximal.

(iv) Every smallest element is minimal.

11.2.10. Examples: Consider the POset $(A, \leq) = (\mathbb{R}, \leq)$ where $\mathbb{R}$ is the set of real numbers and " $\leq$ " is the usual order on the set of all real numbers.

(i) Write $B =$ the interval $[0, 3)$. Then it is clear that $\inf B = 0$ and $\sup B = 3$.

(ii) Write $C =$ the interval $(0, 3]$. Then it is clear that $\inf C = 0$ and $\sup C = 3$.

(iii) From (i) and (ii), we can understand that in general, the infimum (or supremum) of a set X may or may not be in the set X.

(iv) Consider $D = \mathbb{N}$, the set of natural numbers.
It is clear that $\inf D = 1$, but $\sup D$ does not exist.

11.2.11. Zorn's lemma: If $(A, \leq)$ is a poset such that every chain of elements in A has an upper bound in A, then A has at least one maximal element.

11.3. LATTICES:

11.3.1. Definition: (i). A poset $(L, \leq)$ is said to be a lattice (or lattice ordered set) if supremum of x and y; and infimum of x and y exist for every pair $x, y \in L$.

11.3.2. Note: (i) Every chain $(A, \leq)$ is a lattice ordered set [If a, b are in A, then since A is a chain, we have that $a \leq b$ or $b \leq a$. If $a \leq b$ then b is the sup of a, b; and a is the inf. of a,b. If $b \leq a$ then a is the sup of a, b; and b is the inf. of a,b. Hence, $(A, \leq)$ is a lattice ordered set].

(ii). Let $(L, \leq)$ be a lattice ordered set; and
$x, y \in L$. Then we have the following:

$x \leq y \Leftrightarrow \sup(x, y) = y \Leftrightarrow \inf(x, y) = x$.

11.3.3. Definition: An **(algebraic) lattice** $(L, \wedge, \vee)$ is a set L with two binary operations $\wedge$ (called as meet or intersection or product) and $\vee$ (called as join or union or sum) which satisfy the following laws (for all $x, y, z \in L$):

(L1) Commutative laws:
$x \wedge y = y \wedge x$, and $x \vee y = y \vee x$.

(L2) Associative laws:
$x \wedge (y \wedge z) = (x \wedge y) \wedge z$, and $x \vee (y \vee z) = (x \vee y) \vee z$.

(L3) Absorption Laws:
$x \wedge (x \vee y) = x$; and $x \vee (x \wedge y) = x$.

11.3.4. Note: Let $(L, \wedge, \vee)$ be an algebraic lattice and $x \in L$.

(i) $x \wedge x = x \wedge (x \vee (x \wedge x))$ (by absorption law)

$= x \wedge (x \vee (y))$, where $y = x \wedge x$

$= x$ (by absorption law).

(ii) $x \vee x = x \vee (x \wedge (x \vee x))$ (by absorption law)

$= x \vee (x \wedge (y))$ where $y = x \vee x$

$= x$ (by absorption law)

(iii) From (i) and (ii), we got the following axioms:

(L4) Idempotent laws:

$x \wedge x = x$, and $x \vee x = x$.

(iv). Sometimes we read $x \vee y$ and $x \wedge y$ as "x vee y" and "x wedge y".

11.3.5. Remark: Suppose $(L, \leq)$ be an algebraic lattice. Now we verify that

$x \vee y = y \Leftrightarrow x \wedge y = x$ for any $x, y \in L$.

(i) Suppose $x \vee y = y$. Then

$x \wedge y = x \wedge (x \vee y)$ (by the supposition)

$= x$ (by obsorbtion law)

(ii) Suppose $x \wedge y = x$. Then

$x \vee y = (x \wedge y) \vee y$ (by supposition)

$= y \vee (y \wedge x)$ (by commutative law)

$= y$ (by obsorption law).

(iii) By (i) and (ii), we have that $x \vee y = y \Leftrightarrow x \wedge y = x$.

1.3.6. Theorem: (i) Let $(L, \leq)$ be a lattice ordered set. If we define

$x \wedge y := \inf(x, y)$, and $x \vee y := \sup(x, y)$

Then $(L, \wedge, \vee)$ is an algebraic lattice.

(ii) Let $(L, \wedge, \vee)$ be an algebraic lattice. If we define $x \le y \Leftrightarrow x \wedge y = x$, then $(L, \le)$ is a lattice ordered set.

Proof: Part-(i): Let $(L, \le)$ be a lattice ordered set and $x, y, z \in L$.

Now we verify the axioms in (L1). (Commutative laws):

$$x \wedge y = \inf(x, y) = \inf(y, x) = y \wedge x,$$

$$x \vee y = \sup(x, y) = \sup(y, x) = y \vee x.$$

Now we verify the axioms in (L2). (Associative laws):

$$\begin{aligned} x \wedge (y \wedge z) &= x \wedge \inf(y, z) \\ &= \inf(x, \inf(y, z)) = \inf(x, y, z) \\ &= \inf(\inf(x, y), z) = \inf(x, y) \wedge z \\ &= (x \wedge y) \wedge z. \end{aligned}$$

Similarly, we have that

$$x \vee (y \vee z) = (x \vee y) \vee z$$

Now we verify the axioms in (L3). (Absorption laws):

$$\begin{aligned} x \wedge (x \vee y) &= x \wedge \sup(x, y) \\ &= \inf(x, \sup(x, y)) = x \end{aligned}$$

$$\begin{aligned} x \vee (x \wedge y) &= x \vee \inf(x, y) \\ &= \sup(x, \inf(x, y)) = x. \end{aligned}$$

Part-(ii): Let $(L, \wedge, \vee)$ be an algebraic lattice. Let $x, y, z \in L$.

Step-(i): In this step we prove that $(L, \le)$ is a partially ordered set.
By idempotent laws, we have that

$x \wedge x = x$ and $x \vee x = x$ and so $x \le x$.

This shows that $\le$ is reflexive.

Now we verify the antisymmetric property.

For this, suppose $x \le y$ and $y \le x$.

$$\begin{aligned} &\Rightarrow x \wedge y = x \text{ and } y \wedge x = y \\ &\Rightarrow x = x \wedge y = y \wedge x \text{ (by commutative law)} \\ &\qquad\qquad\qquad = y \\ &\Rightarrow x = y. \end{aligned}$$

This shows that $\le$ is antisymmetric.

Now we verify the transitive property.

For this, suppose $x \leq y$ and $y \leq z$.

$\Rightarrow x \wedge y = x$ and $y \wedge z = y$

$\Rightarrow x = x \wedge y = x \wedge (y \wedge z)$

$= (x \wedge y) \wedge z$ (by associative law)

$= x \wedge z.$

$\Rightarrow x = x \wedge z. \Rightarrow x \leq z.$

This shows that $\leq$ is transitive.

So we can conclude that $(L, \leq)$ is a poset.

Step-(ii): In this step we prove that $\sup(x, y) = x \vee y$.

By Remark 11.3.5., we have that

$x \leq y \Leftrightarrow x \vee y = y \Leftrightarrow x \wedge y = x$ .... (i).

Let $x, y \in L$. Then $x \wedge (x \vee y) = x \Rightarrow x \leq x \vee y$.

Similarly $y \leq x \vee y$. Therefore $x \vee y$ is an upper bound for $\{x, y\}$.

Suppose $z \in L$ be an upper bound for $\{x, y\}$. Then $x \leq z$ and $y \leq z$.

By (i), we get that $x \vee z = z$ and $y \vee z = z$.

Now $(x \vee y) \vee z = x \vee (y \vee z)$ (by associative law)

$= x \vee z$ (by (i))

$= z.$

$\Rightarrow x \vee y \leq z.$

This shows that $x \vee y$ is the least upper bound for x and y, and hence $\sup(x, y) = x \vee y$.

Step-(iii): In this step, we prove that $\inf(x, y) = x \wedge y$.

Now $x \vee (x \wedge y) = x \Rightarrow x \wedge y \leq x$.

Similarly $y \vee (x \wedge y) = y \vee (y \wedge x)$ (by commutative law)

$= y$ (by absorption law).

$\Rightarrow x \wedge y \leq y.$

This shows that $x \wedge y$ is a lower bound for $\{x, y\}$.

Suppose $z \in L$ be a lower bound for $\{x, y\}$.

Then $z \leq x$ and $z \leq y$.

By (i), we get that $x \wedge z = z$ and $y \wedge z = z$.

Now $(x \wedge y) \wedge z = x \wedge (y \wedge z)$ (by associative law)

$= x \wedge z = z.$

$\Rightarrow z \leq x \wedge y.$

This shows that $x \wedge y$ is the greatest lower bound for x and y, and hence $\inf(x, y) = x \vee y$.

Step-(iv): From the above steps (i) to (iii), we conclude that $(L, \leq)$ is a lattice ordered set.

11.3.7. Remark: (i) From the Theorem 11.3.6., it is clear that there exists a one-to-one relationship between lattice ordered sets and algebraic lattices. In other words, the concepts "lattice ordered set" and "algebraic lattice" are equivalent. So we can use the term lattice for both concepts: lattice ordered sets and algebraic lattices.

(ii) We write $|L|$ to denote the number of elements of L.

(iii) If N is a subset of a POset, then $V_{x \in N}\, x$ and $\wedge_{x \in N}\, x$ denote the supremum and infimum of N, respectively, whenever they exist.

We also say that the supremum of N is the join of all elements of N and the infimum is the meet of all elements of N.

11.4. SOME MORE CONCEPTS IN LATTICE THEORY:

11.4.1. Duality Principal (or Principle of Duality):
Any "formula" involving the binary operations $\wedge$ and $\vee$ which is valid in any lattice $(L, \wedge, \vee)$ remains valid if we replace $\wedge$ by $\vee$, and $\vee$ by $\wedge$ everywhere in the formula. This process of replacing is called dualyzing.

11.4.2. Definitions: If a lattice L contains a smallest (greatest, respectively) element with respect to $\leq$, then this uniquely determined element is called the zero element (unit element, respectively). The zero element is denoted by 0, and the unit element is denoted by 1. The elements 0 and 1 are called universal bounds. If the elements 0 and 1 exist, then we say that the lattice L is a bounded lattice.

11.4.3. Note: If a lattice L is bounded (by 0 and 1), then every x in L satisfies $0 \leq x \leq 1$, $0 \wedge x = 0$, $0 \vee x = x$, $1 \wedge x = x$, and $1 \vee x = 1$.

11.4.4. Problem: Suppose that L is a lattice. Show that

(i) If $x_1, x_2, \ldots x_n \in L$, then $x_1 \vee x_2 \vee \ldots \vee x_n \in L$.

Also $x_1 \wedge x_2 \wedge \ldots \wedge x_n \in L$.

(ii) If L is a finite lattice, then L is bounded.

Solution: (i) Let $x_1, x_2, \ldots x_n \in L$.

We prove that $x_1 \vee x_2 \vee \ldots \vee x_n \in L$, by using the mathematical induction.

If $n = 2$, then since L is a lattice, we get that $x_1 \vee x_2 \in L$.

Now assume the induction hypothesis that

$x_1, x_2, \dots x_{n-1} \in L \Rightarrow x_1 \vee x_2 \vee \dots \vee x_{n-1} \in L$.

Suppose that $x_1, x_2, \dots x_n \in L$

$\Rightarrow$ $x_1 \vee x_2 \vee \dots x_{n-1} \in L$ and $x_n \in L$ (by induction hypothesis)

$\Rightarrow$ $x_1 \vee x_2 \vee \dots \vee x_{n-1} \vee x_n \in L$ (by the definition of lattice).

By mathematical induction, we conclude that

$x_1 \vee x_2 \vee \dots \vee x_n \in L$ for any integer n and $x_1, x_2, \dots x_n \in L$.

In a similar way, we can prove that $x_1 \wedge x_2 \wedge \dots \wedge x_n \in L$.

(ii). Suppose that L is a finite lattice with m elements.

Then we can take $L = \{x_1, x_2, \dots x_m\}$.

By (i), $x_1 \vee x_2 \vee \dots \vee x_m$, $x_1 \wedge x_2 \wedge \dots \wedge x_m \in L$.

It is clear that $x_1 \wedge x_2 \wedge \dots \wedge x_m \leq x_i$ for $1 \leq i \leq m$ and

$x_i \leq x_1 \vee x_2 \vee \dots \vee x_m$ for $1 \leq i \leq m$.

Therefore $x_1 \vee x_2 \vee \dots \vee x_m$ is an upper bound for L and $x_1 \wedge x_2 \wedge \dots \wedge x_m$ is a lower bound for L. Therefore L is a bounded lattice.

11.4.5. Lemma: In every lattice L the operations $\wedge$ and $\vee$ are isotone (that is, $y \leq z \Rightarrow x \wedge y \leq x \wedge z$, and $x \vee y \leq x \vee z$).

Proof: Suppose that $y \leq z$.

Part-(i): We know that $y \leq z \Rightarrow y \wedge z = y$.

So we have that

$x \wedge y = (x \wedge x) \wedge (y \wedge z)$ (by idempotent law)

$= (x \wedge y) \wedge (x \wedge z)$ (by associative and commutative laws)

$\Rightarrow x \wedge y \leq x \wedge z$.

Part-(ii): We know that $y \leq z \Rightarrow y \vee z = z$.

Now $(x \vee y) \vee (x \vee z) = (x \vee x) \vee (y \vee z)$ (by associative and commutative laws)

$= x \vee (y \vee z)$ (by idempotent law)

$= x \vee z$.

This shows that $x \vee y \leq x \vee z$. The proof is complete.

11.4.6. Theorem: Let L be a lattice, and $x, y, z \in L$.
Then L satisfy the following distributive inequalities:

(i) $x \wedge (y \vee z) \geq (x \wedge y) \vee (x \wedge z)$

(ii) $x \vee (y \wedge z) \leq (x \vee y) \wedge (x \vee z)$

Proof: Part-(i): We know that $x \wedge y \leq x$, and $x \wedge y \leq y \leq y \vee z$.

So $x \wedge y$ is a lower bound for x and $y \vee z \Rightarrow x \wedge y \leq x \wedge (y \vee z)$ … (iii)

Now $x \wedge z \leq x$ and $x \wedge z \leq z \leq y \vee z$

$\Rightarrow$ $x \wedge z$ is a lower bound for x and $y \vee z \Rightarrow x \wedge z \leq x \wedge (y \vee z)$ … (iv)

From (iii) and (iv), we have that $x \wedge (y \vee z)$ is an upper bound for $x \wedge y$ and $x \wedge z$.

$\Rightarrow (x \wedge y) \vee (x \wedge z) \leq x \wedge (y \vee z)$.

The proof is complete for (i).

Part-(ii): We know that $x \leq x \vee y$ and $x \leq x \vee z$

$\Rightarrow$ x is a lower bound for $x \vee y$ and $x \vee z \Rightarrow x \leq (x \vee y) \wedge (x \vee z)$ … (v)

we know that $y \wedge z \leq y \leq x \vee y$ and $y \wedge z \leq y \leq x \vee z$

$\Rightarrow$ $y \wedge z$ is a lower bound for $x \vee y$ and $x \vee z$

$\Rightarrow y \wedge z \leq (x \vee y) \wedge (x \vee z)$ … (vi)

From (v) and (vi), we have that $(x \vee y) \wedge (x \vee z)$ is an upper bound for x and $y \wedge z$.

$\Rightarrow x \vee (y \wedge z) \leq (x \vee y) \wedge (x \vee z)$.

The proof is complete.

11.4.7. Definition: A subset S of a lattice L is called a sublattice of L if S is a lattice with respect to the restriction of $\wedge$ and $\vee$ from L to S.

It is clear that a subset S of L is a sublattice of the lattice L $\Leftrightarrow$ S is "closed" with respect to $\wedge$ and $\vee$ (that is, $s_1, s_2 \in S \Rightarrow s_1 \wedge s_2 \in S$ and $s_1 \vee s_2 \in S$).

11.4.8. Definition: For two elements x, y in a lattice L (with $x \leq y$), we define the interval as follows: $[x, y] := \{a \in L \ / \ x \leq a \leq y\}$.

Note that this interval is a sublattice of L.

11.5 SUMMARY:

Several properties of lattices were presented. The definitions of some important concepts related to Lattice theory namely Partial order relation, partial ordered set, Ordered lattice set, algebraic lattice, sublattice, universal bounds were included. Some Lemmas, and theorems were also proved. It is proved that every ordered lattice may be turned into algebraic lattice, and vice-versa. Some examples were presented to understand the concepts in a better way by the reader.

11.6 TECHNICAL TERMS:

1. Partial order relation.
A reflexive, antisymmetric, and transitive relation R on a set A is called a partial order relation.

2. Partially ordered set (or POset, in short).
(A, R) is called a partially ordered set (or POset) if R is a partial order relation on A.

3. Hasse Diagram
(Refer Note 11.2.3., and Example 11.2.5)

4. Zorn's lemma.
If $(A, \leq)$ is a poset such that every chain of elements in A has an upper bound in A, then A has at least one maximal element.

5. Lattice (or Lattice ordered set)
A poset $(L, \leq)$ is said to be a lattice (or lattice ordered set) if supremum of x and y; and infimum of x and y exist for every pair $x, y \in L$.

6. Duality Principal (or Principle of Duality):
Any "formula" involving the binary operations $\wedge$ and $\vee$ which is valid in any lattice $(L, \wedge, \vee)$ remains valid if we replace $\wedge$ by $\vee$, and $\vee$ by $\wedge$ everywhere in the formula. This process of replacing is called dualyzing.

7. Sublattice
A subset S of a lattice L is called a sublattice of L if S is a lattice with respect to the restriction of $\wedge$ and $\vee$ from L to S.

11.7 SELF ASSESSMENT QUESTIONS:

1. Define partial ordered set, and give an example.

2. What do you mean by Hasse Diagram, and give an example.

3. What do you mean by a chain. Show that every chain is a lattice.

4. Prove that every ordered lattice set can be turned in to an algebraic lattice.

5. Prove that every algebraic lattice can be turned in to an ordered lattice set.

6. Determine all the partial orders and their Hasse diagrams on the set $L = \{a, b\}$. Which of them are chains ?

[Ans: The possible partial orders on $L = \{a, b\}$ are

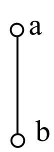


Fig-1

Fig-2

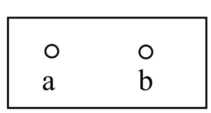


Fig - 3

The POsets in Fig -1 and fig-2 are chains.

The POset in fig - 3 is not a chian (because a ≰ b and b ≰ a).]

7. Determine all the partial orders and their Hasse diagrams on the set L = {a, b, c}. Which of them are chains?

[Ans: The required partial orders are given below.

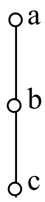


Fig -1

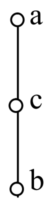


Fig -2

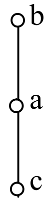


Fig -3

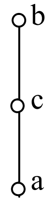


Fig -4

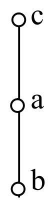


Fig -5

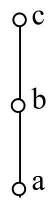


Fig -6

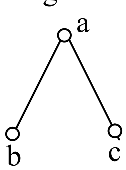


Fig -7

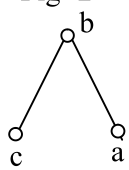


Fig -8

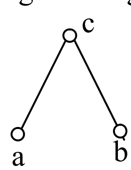


Fig -9

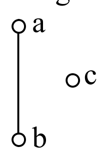


Fig -10

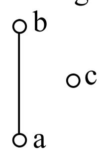


Fig -11

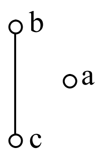


Fig -12

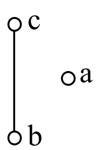


Fig -13

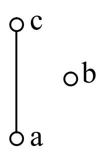


Fig -14

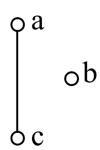


Fig -15

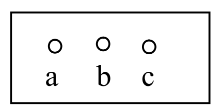


Fig -16

Among all these partial orders, the partial orders given in Figures 1 to 6 are chains.]

11.8 **SUGGESTED READINGS**:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Kuncham Syam Prasad

LESSON -12

SOME EXAMPLES OF LATTICES, AND HOMOMORPHISMS

OBJECTIVE:

- To know different examples of Lattices.
- To understand the concept: product of lattices.
- To Learn to draw the diagrams of lattices.
- To have proper understanding of different types of homomorphisms.

STRUCTURE

12.1. INTRODUCTION

In the previous lesson, we came to know the fundamental definitions of lattices, and also some important theorems. In this Lesson, we present several important examples for better understanding of the concepts. Later different types of homomorphisms were explained.

12.2. SOME EXAMPLES OF LATTICES:

In this section, we include some important examples of lattices.

12.2.1. Examples: (i). Consider $\mathbb{N}$ = the set of all natural numbers.

Define $a \leq b \Leftrightarrow a$ divides b, for all $a, b \in \mathbb{N}$. Then $(\mathbb{N}, \leq)$ is a POset.

For any $x, y \in \mathbb{N}$, we write $x \wedge y = \gcd\{x, y\}$ and $x \vee y = \text{lcm}\{x, y\}$.

Then $(\mathbb{N}, \leq)$ is a lattice. Here 1 is the zero element.

The greatest element does not exist.

(ii). Let A be a set. Consider $\wp(A)$ = the power set of A, the set of all subsets of A. $(\wp(A), \subseteq)$ is a POset (where $\subseteq$ is the set inclusion).

For any $X, Y \in \wp(A)$, we write $X \wedge Y = X \cap Y$ and $X \vee Y = X \cup Y$.

Then $(\wp(A), \subseteq)$ is a lattice.

In this lattice, ϕ is the smallest element and A is the greatest element.

(iii) Let V be a vector space. Write S(V) = the set of all subspaces of V.

(S(V), $\subseteq$) is a POset where $\subseteq$ is the set inclusion.

For U, W $\in$ S(V), we write $U \wedge W = U \cap W$ and $U \vee W = U + W$.

Then (S(V), $\subseteq$) is a lattice.

In this lattice, the subspace (0) is the smallest element and V is the greatest element.

12.2.2. Definition: Let $\{L_i \,/\, i \in I\}$ be a collection of lattices with 0 and 1.

Write $A = \prod_{i \in I} L_i$, the Cartesian product of sets. Let $\{a_i\}, \{b_i\} \in A$.

Define $\{a_i\} \leq \{b_i\} \Leftrightarrow a_i \leq b_i$ for all $i \in I$.

With this definition, (A, $\leq$) is a POset.

Define $\{a_i\} \vee \{b_i\} = \{c_i\}$ and $\{a_i\} \wedge \{b_i\} = \{d_i\}$ where

$c_i = a_i \vee b_i$ and $d_i = a_i \wedge b_i$ for all $i \in I$.

Then (A, $\leq$) is a lattice.

Consider the elements $\{x_i\}$ where $x_i = 0$ in L_i for all i and $\{y_i\}$ where $y_i = 1$ in L_i for all i.

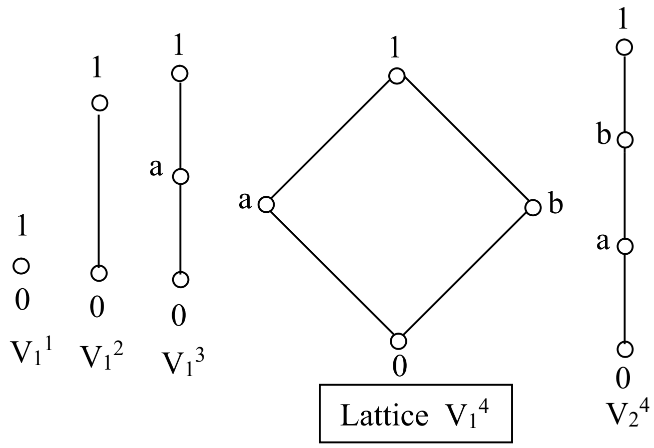
Then $\{x_i\}$ is the smallest element in A, and $\{y_i\}$ is the greatest element in A.

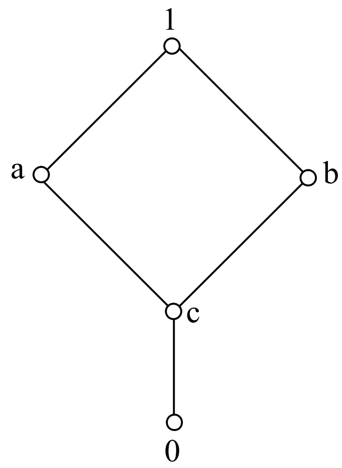
Here we may write $\{x_i\}$ as (0, 0, …0, …) and $\{y_i\}$ as (1, 1, … 1, …).

Hence (A, $\leq$) is a lattice with 0 and 1. This lattice is called as product lattice.

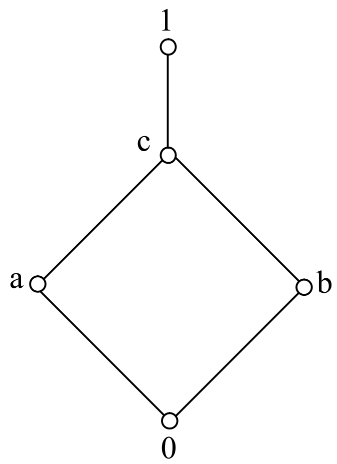
12.2.3. Examples: Now we provide the Hasse diagrams of all the lattices with n elements where $1 \leq n \leq 6$.

The symbol V_i^n denotes the i^{th} lattice with n elements.

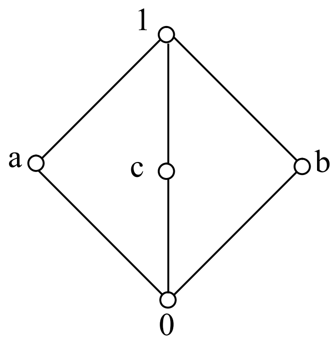




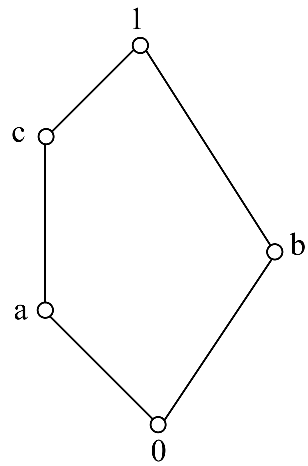
Lattice V_1^5



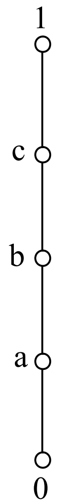
Lattice V_2^5



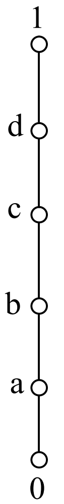
Lattice V_3^5



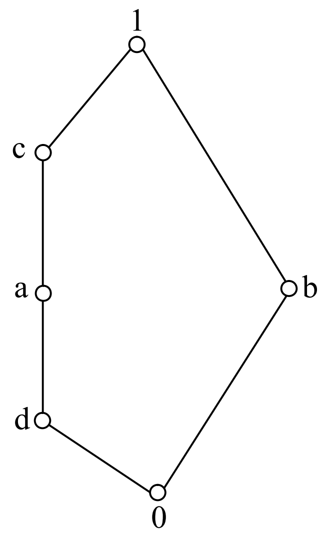
Lattice V_4^5



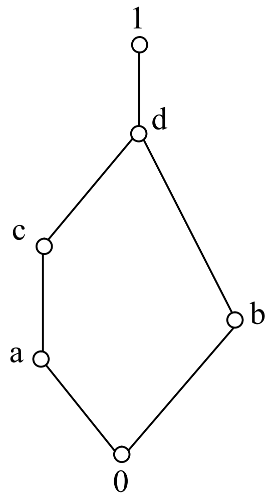
V_5^5



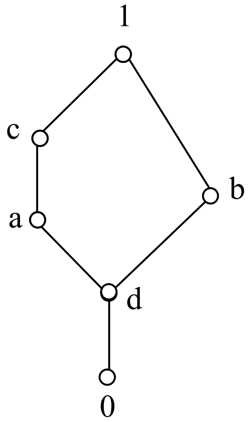
V_1^6



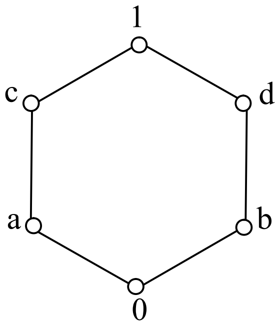
Lattice V_2^6



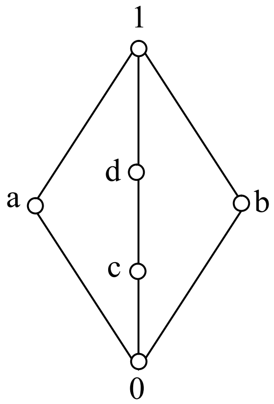
Lattice V_5^6



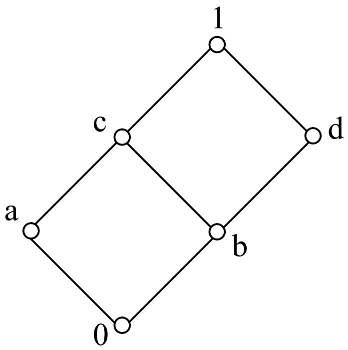
Lattice V_6^6



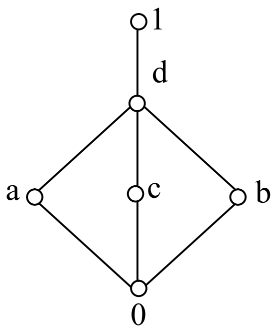
Lattice V_7^6



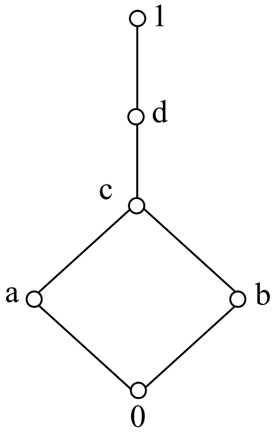
Lattice V_8^6



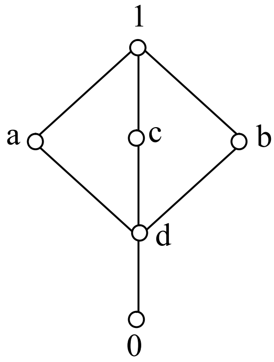
Lattice V_9^6



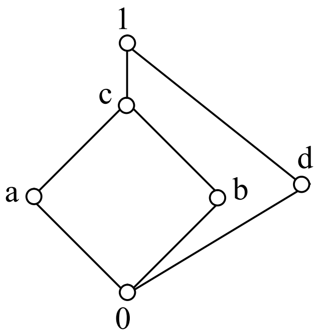
Lattice V_{10}^6



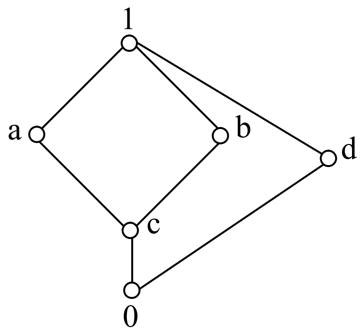
Lattice V_{11}^{6}



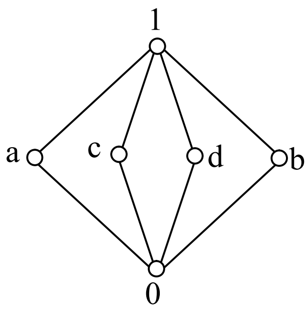
Lattice V_{12}^{6}



Lattice V_{13}^{6}



Lattice V_{14}^{6}



Lattice V_{15}^{6}

12.2.4. Examples: (i). In the following tables, we provide the operation tables for the lattice V_4^5. This table, provides the information regarding $x \wedge y$ and $x \vee y$ for all x and y of the lattice.

$\wedge$	0	a	b	c	1
0	0	0	0	0	0
a	0	a	0	a	a
b	0	0	b	0	b
c	0	a	0	c	c
1	0	a	b	c	1

$\vee$	0	a	b	c	1
0	0	a	b	c	1
a	a	a	1	c	1
b	b	1	b	1	1
c	c	c	1	c	1
1	1	1	1	1	1

The tables given in above are called as operation tables.

(ii). Let V be a vector space.

Write S(V) = the set of all subspaces of V.

L(V) = $\wp$(V) = the set of all subsets of V.

Now $(S(V), \subseteq)$ and $(L(V), \subseteq)$ are two lattices where $\subseteq$ is the set inclusion. (Refer Example 12.2.1 (ii) and (iii)).

(iii). $S(V) \subseteq L(V)$ and $(S(V), \subseteq)$ is a subPOset of $(L(V), \subseteq)$.

(iv). Suppose V is a vector space over the field $\mathbb{R}$ of real numbers with basis $\{v_1, v_2, v_3\}$. Write $V_i = \mathbb{R}v_i$ for $1 \leq i \leq 3$.

Then each V_i is a subspace of V.

Now in S(V), we have that $V_1 \vee V_2 = V_1 + V_2$.

In L(V), we have that $V_1 \vee V_2 = V_1 \cup V_2$.

It is clear that $v_1 + v_2 \in V_1 + V_2$ and $v_1 + v_2 \notin V_1 \cup V_2$.

Hence $V_1 \vee V_2 = \sup\{V_1, V_2\} = V_1 + V_2$ in the lattice S(V), is not same as $V_1 \vee V_2 = \sup\{V_1, V_2\} = V_1 \cup V_2$ in the lattice L(V).

This shows that S(V) cannot be a sublattice of L(V).

(v). Every singleton subset of a lattice L is a sublattice of L.

12.3. HOMOMORPHISMS:

12.3.1. Definitions: Let L and M be lattices. A mapping $f : L \rightarrow M$ is called a

(i) join-homomorphism if $f(x \vee y) = f(x) \vee f(y)$;

(ii) meet-homomorphism if $f(x \wedge y) = f(x) \wedge f(y)$;

(iii) order-homomorphism if $x \leq y \Rightarrow f(x) \leq f(y)$ hold for all $x, y \in L$.

(iv) The function f is said to be a homomorphism (or lattice homomorhism) if it is both a join-homomorphism and a meet-homomorphism.

(v) Injective, surjective, or bijective (lattice) homomorphisms are called (lattice) monomorphisms, epimorphisms, or isomorphisms, respectively.

(vi) If f is a homomorhisms from L to M, then f(L) is called as the homomorphic image of L.

(vii) If there is an isomorphism from L to M, then we say that L and M are isomorphic, and we denote this fact by the symbol ' $L \cong M$ '.

12.3.2. Note: (i) The homomorphic image $f(L)$ is a sublattice of M where $f: L \to M$ is a lattice homomorphism.

[Verification: Let $f: L \to M$ be a lattice homomorphism. Let $x^1, y^1 \in f(L) \Rightarrow$ there exists $x, y \in L$ such that $f(x) = x^1$ and $f(y) = y^1$.

Since L is a lattice, $x \vee y$ and $x \wedge y$ exists.

Now $x^1 \vee y^1 = f(x) \vee f(y) = f(x \vee y) \in f(L)$.

Also $x^1 \wedge y^1 = f(x) \wedge f(y) = f(x \wedge y) \in f(L)$.

Now we have that $x^1 \vee y^1, x^1 \wedge y^1 \in f(L)$.

Hence $f(L)$ is a lattice and so it is a sublattice of M].

(ii) Every join (or meet)-homomorphism is an order-homomorphism.

[Verification: Part-(i): Let $x \leq y$

$\Rightarrow x \vee y = y \Rightarrow f(x \vee y) = f(y)$

$\Rightarrow f(x) \vee f(y) = f(y)$ [if f is a join-homomorphism]

$\Rightarrow f(x) \leq f(y)$

Now we proved that if f is a join-homomorphism, then f is an order homomorphism.

Part-(ii): Let $x \leq y$

$\Rightarrow x = x \wedge y \Rightarrow f(x) = f(x \wedge y)$

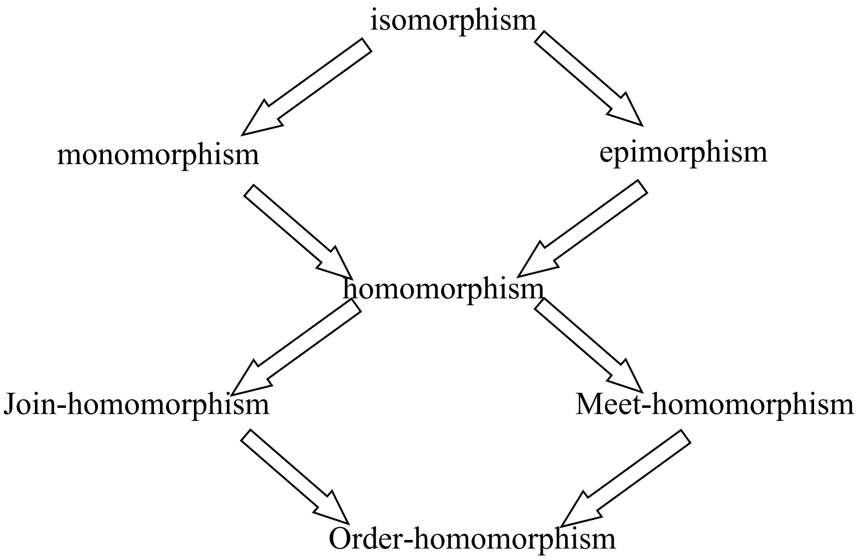
$\Rightarrow f(x) = f(x) \wedge f(y)$ (if f is a meet-homomorphism)

$\Rightarrow f(x) \leq f(y)$.

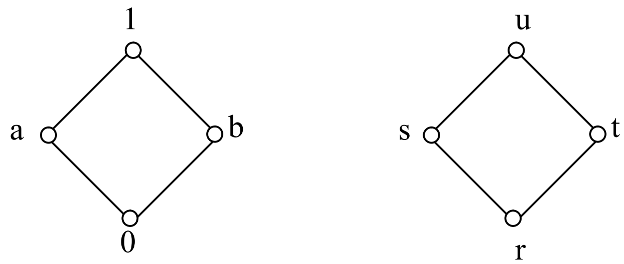
Now we proved that if f is a meet-homomorphism, then f is an order omomorphism.]

(iii) Every order-homomorphism need not be a join (or meet)-homomorphism. [Please refer the mapping h defined in the Example 12.3.4. This h is an order homomorphism, but not either meet homomorphism or join homomorphism].

(iv) The relationship between different types of homomorphisms is presented in a diagramatic form.



12.3.3. Example: Consider the lattices represented by the following diagrams.



These lattices are isomorphic under the isomorphism given by
$0 \mapsto r, \quad a \mapsto s, \quad b \mapsto t, \quad 1 \mapsto u$.

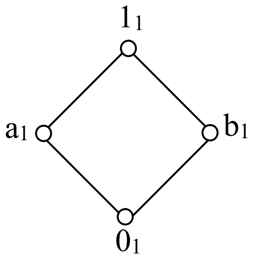
The map $0 \mapsto r, \quad a \mapsto t, \quad b \mapsto s, \quad 1 \mapsto u$ is another isomorphism.

12.3.4. Example: Consider the lattices L_1, L_2 and L_3 given here. Define the functions f, g and h as follows:

$f : L_1 \mapsto L_2$ by $f(0_1) = f(a_1) = f(b_1) = 0_2$, $f(1_1) = 1_2$;

$g : L_1 \mapsto L_2$ by $g(0_1) = 0_2$, $g(1_1) = g(a_1) = g(b_1) = 1_2$;

$h : L_1 \mapsto L_3$ by $h(0_1) = 0_3$, $h(a_1) = a_3$, $h(b_1) = b_3$, $h(1_1) = 1_3$.



Lattice - L_1

Lattice - L_2

Lattice - L_3

(i) These three mappings are order-homomorphisms.

(ii) Here f is a meet-homomorphism (since $f(a_1 \wedge b_1) = f(0_1) = 0_2 = f(a_1) \wedge f(b_1)$).

However, f is not a homomorphism (since $f(a_1 \vee b_1) = f(1_1) = 1_2$ and $f(a_1) \vee f(b_1) = 0_2$).

So f is a meet-homomorphism but not a join-homomorphism.

(iii) We can observe that g is a join-homomorphism, but not a meet-homomorphism.

(iv) We can observe that h is neither a meet-homomorphism nor a join-homorphism.

(Since $h(a_1 \wedge b_1) = h(0_1) = 0_3$ and $h(a_1) \wedge h(b_1) = a_3 \wedge b_3 = a_3$.

Also $h(a_1 \vee b_1) = h(1_1) = 0_3$ and $h(a_1) \vee h(b_1) = a_3 \vee b_3 = b_3$).

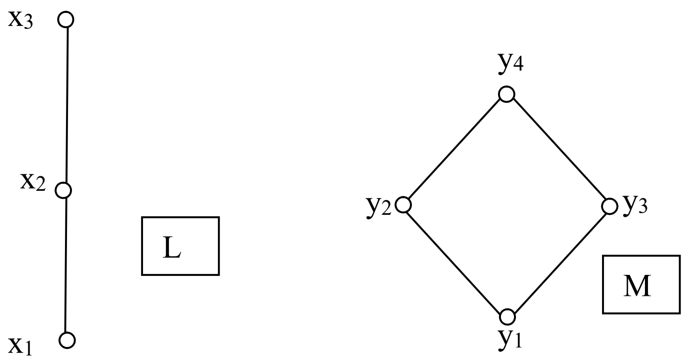
12.3.5. Definition: Let L and M be two lattices. The set of ordered pairs $\{(x, y) \ / \ x \in L, y \in M\}$ (that is, the direct product of L and M (in symbols, we write $L \times M$)) with operations $\vee$ and $\wedge$ defined by

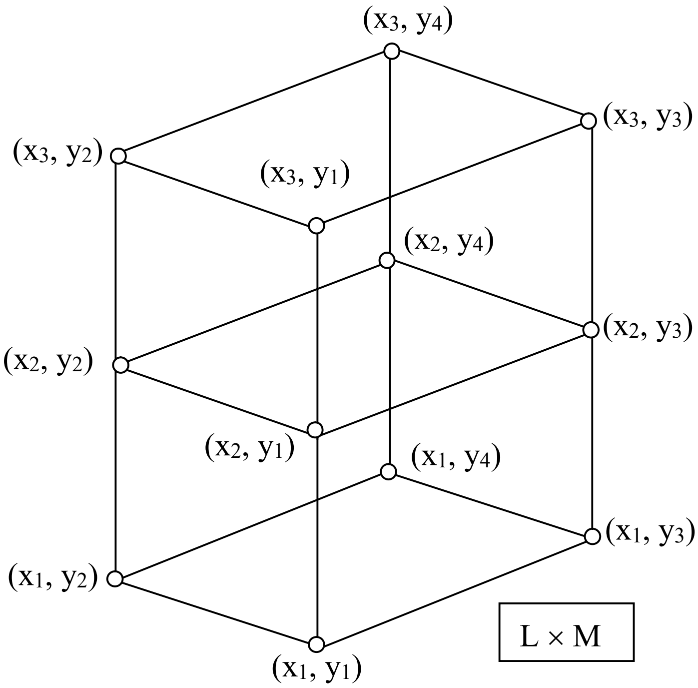
$(x_1, y_1) \vee (x_2, y_2) = (x_1 \vee x_2, y_1 \vee y_2)$, and

$(x_1, y_1) \wedge (x_2, y_2) = (x_1 \wedge x_2, y_1 \wedge y_2)$,

is called the product of two lattices. The product lattice of finite number of lattices will be defined similarly (Refer Definition. 12.2.2.).

12.3.6. Example: Consider the lattices L and M given here. Observe the product lattice of L and M, which is also presented here.





12.4 SUMMARY:

We presented several important examples of lattices for better understanding of the concepts. Later different types of homomorphisms were explained. Hasse diagrams of some lattices, and a product lattice were also included.

12.5 TECHNICAL TERMS:

Product Lattice (Refer 12.2.2., 12.3.5.)

Lattice homomorhism

The function f is said to be a lattice homomorhism if it is both a join-homomorphism and a meet-homomorphism.

12.6 SELF ASSESSMENT QUESTIONS:

1. Define product lattice.

2. Define different types of homomorphisms related to lattices.

3. Give an example of a order homomorphism which is not a meet homomorphism.

4. Give two examples of lattices, and draw Hasse diagrams.

5. Give three Hasse diagrams related to two lattices and their product lattice.

12.7 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Kuncham Syam Prasad.

LESSON-13

MODULAR AND DISTRIBUTIVE LATTICES

OBJECTIVE:

- To know Modular and Distributive Lattices.
- To understand the theorems on Modular and Distributive Lattices.
- To identify the difference between Modular and Distributive Lattices.
- To know diamond and pentagon lattices.

STRUCTURE:

13.1. INTRODUCTION:

In Lessons 11 and 12, we came to know the fundamentals, some examples, and some important results on Lattices. In this lesson, we present two more important concepts: Modular Lattice, and Distributive Lattice which plays vital role in the theory of lattices. Examples to show the difference between these two concepts are also included. Few theorems on this concepts were also presented.

13.2. MODULAR LATTICES:

13.2.1. Definition: (i). A lattice $(L, \vee, \wedge)$ is called a **modular lattice** if it satisfies the following condition: $x \leq z \Rightarrow x \vee (y \wedge z) = (x \vee y) \wedge z$ for all $x, y, z \in L$.

This condition is called as modular identity.

13.2.2. Example: Let $(G, \bullet)$ be a group and L be the set of all subgroups of G.

We define $\vee$, $\wedge$ and on L as follows :

For $N_1, N_2 \in L$, define $N_1 \vee N_2 = N_1 \bullet N_2$ and $N_1 \wedge N_2 = N_1 \cap N_2$.

Then $(L, \vee, \wedge)$ is a lattice.

Now we prove that this lattice L is a modular lattice.

Let $N_1, N_2, N_3 \in L$ and $N_1 \subseteq N_3$.

From the set theory, we have that $N_1 \vee (N_2 \wedge N_3) \subseteq (N_1 \vee N_2) \wedge N_3$.

Now we have to prove that $(N_1 \vee N_2) \wedge N_3 \subseteq N_1 \vee (N_2 \wedge N_3)$.

That is, $(N_1 \bullet N_2) \cap N_3 \subseteq N_1 \bullet (N_1 \cap N_3)$.

Let $x \in (N_1 \bullet N_2) \cap N_3$. Then $x \in N_3$ and $x = y \bullet z$ for some $y \in N_1$ and $z \in N_2$.

Now $x \in N_3$ and $y \in N_1 \Rightarrow x^{-1} y^{-1} x \in N_1 \subseteq N_3$.

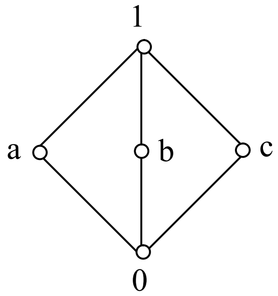
Hence $z = y^{-1} x = e\, y^{-1} x = x \bullet (x^{-1} y^{-1} x) \in N_3$ (since x, $x^{-1} y^{-1} x \in N_3$).

Thus $z \in N_2 \cap N_3$. $\Rightarrow x = yz \in N_1 \bullet (N_2 \cap N_3)$.

Hence $(L, \vee, \wedge)$ is modular lattice.

13.2.3. Definition: Consider the lattice $L_1 = \{0, a, b, c, 1\}$ whose Hasse diagram is given.

Diamond Lattice L_1



This lattice L_1 is a modular lattice.

This lattice is denoted by M_5 (or V_3^5) and it is called as **diamond** lattice.

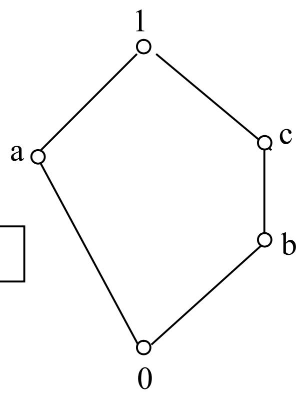
13.2.4. Note: Consider the lattice $L_2 = \{0, a, b, c, 1\}$ whose Hasse diagram (pentagon lattice) is given here. This lattice L_2 is not a modular lattice.

[Verification: In a contrary way, suppose that this lattice L_2 is a modular lattice. Since $b \leq c$, by modular law, we have that $b \vee (a \wedge c) = (b \vee c) \wedge c$

$\Rightarrow b \vee 0 = 1 \wedge c \Rightarrow b = c$, a contradiction.

Hence L_2 is not a modular lattice.]

This lattice is denoted by N_5 (or V_4^5) and it is called as the pentagon lattice.



Pentagon Lattice L_2

13.2.5. Lemma: A lattice $(L, \vee, \wedge)$ is Modular $\Leftrightarrow$
$x \vee \{y \wedge (x \vee z)\} = (x \vee y) \wedge (x \vee z)$ for all $x, y, z \in L$.

Proof: Suppose L is a modular lattice and $x, y, z \in L$.
Since $x \leq x \vee z$, by modular law, we have that $x \vee \{y \wedge (x \vee z)\} = (x \vee y) \wedge (x \vee z)$.

Converse: Suppose $x \vee \{y \wedge (x \vee z)\} = (x \vee y) \wedge (x \vee z)$ for all $x, y, z \in L$.
Let $x, y, z \in L$ and $x \leq z$. Then $x \vee z = z$.
Now $x \vee (y \wedge z) = x \vee \{y \wedge (x \vee z)\}$ (since $z = x \vee z$)
$= (x \vee y) \wedge (x \vee z)$ (by the converse hypothesis)
$= (x \vee y) \wedge z$
This shows that L is a modular lattice.

13.2.6. Theorem: A lattice $(L, \vee, \wedge)$ is a modular lattice $\Leftrightarrow$ L contains no sublattice which is isomorphic to N_5 (the Pentagon lattice).

Proof: Suppose L has a sublattice S which is isomorphic to N_5, the pentagon lattice.
We know that N_5 is not a modular lattice (refer the Note 13.2.4.).
So we get that S is not a modular lattice and so L is not a modular lattice.

Converse: Suppose that L is not a modular lattice. Then there exist elements
$x, y, z \in L$ such that $x \leq z$ and $x \vee (y \wedge z) \neq (x \vee y) \wedge z$.
We know that $x \vee (y \wedge z) \leq (x \vee y) \wedge z$.
So we have that $x \vee (y \wedge z) < (x \vee y) \wedge z$.

Part-(i): Write $S = \{t, a, b, c, s\}$ where $t = y \wedge z$, $a = x \vee (y \wedge z)$, $b = (x \vee y) \wedge z$,
$c = y$ and $s = x \vee y$.
Then we get that $t \leq a < b \leq s$ … (i) and
$t \leq c \leq s$ … (ii)
Now $t \leq a$ and $t \leq c$
$\Rightarrow t \leq (a \wedge c) \leq b \wedge c$ (since $a < b$)
$= (x \vee y) \wedge z \wedge y$
$= y \wedge z = t$. (by commutative, associative and absorption laws)
So we get that $a \wedge c = b \wedge c = t$ … (iii)
Also $s \geq b$, $s \geq c$

$\Rightarrow$ $s \geq b \vee c \geq a \vee c$ (since $a < b$)

$= x \vee (y \wedge z) \vee y = x \vee y$ (by absorption law)

$= s$

So we get that $b \vee c = a \vee c = s$ …. (iv)

Now we can conclude that S is a sublattice of L.

Part-(ii): Now we prove that all the elements of S are distinct. We know that $a < b$.

Suppose $t = c$.

Then $a \wedge c = t = c$ (by (iii))

$\Rightarrow$ $c \leq a \Rightarrow a = a \vee c = s$ (by (iv)), a contradiction to (i).

This shows that $t < c$.

Part-(iii): Suppose $c = s$. Then $c = s = b \vee c$ (by (iv))

$\Rightarrow$ $b \leq c \Rightarrow b = b \wedge c \Rightarrow b = b \wedge c = t$ (by (iii))

$\Rightarrow$ $b = t$, a contradiction to (i).

This shows that $c < s$.

Part-(iv): Suppose that $t = a$.

Then $s = a \vee c$ (by (iv))

$= t \vee c$ (since $t = a$, our supposition here)

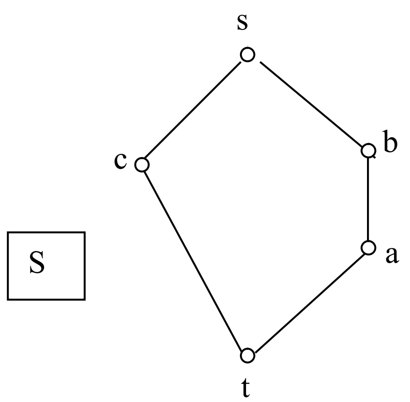
$= c$ (since $t \leq c$)

$\Rightarrow$ $s = c$, a contradiction to the fact that $c < s$.

Part-(v): Suppose $s = b$. Then $t = b \wedge c$ (by (iii)) $= s \wedge c = c$

(since $s = b$, the supposition here), a contradiction to the fact that $t < c$.

Thus all the elements of S are distinct, and the Hasse diagram of the lattice S is given.



This shows that S is isomorphic to the pentagon lattice N_5. The proof is complete.

13.3. DISTRIBUTIVE LATTICES:

13.3.1. Definition: A lattice L is said to be a **distributive** lattice if it satisfies the following laws:

(i) $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$, and

(ii) $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$, for all $x, y, z \in L$.

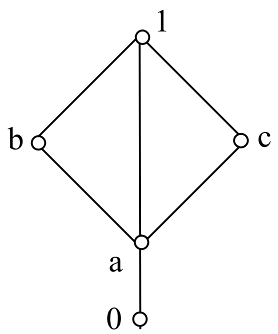
These two laws are called the distributive laws.

Theorem 13.3.4., says that the two laws (i) and (ii) given here are equivalent.

13.3.2. Examples: (i) For any set X, the lattice $(P(X), \cup, \cap)$ is a distributive lattice.

(ii) Every chain is a distributive lattice.

L



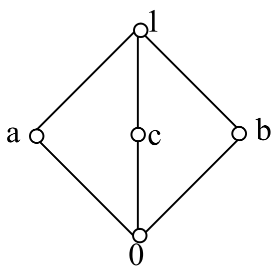
(iii) Consider the lattice L = (0, a, b, c, 1) whose Hasse diagram is given here. We can observe that this lattice is a distributive lattice .

(iv) The "diamond lattice" V_3^5 ; and the "pentagon lattice" V_4^5 are not distributive lattices.

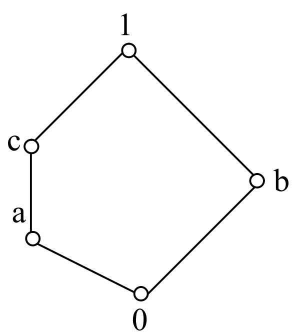
In V_3^5 , $a \vee (b \wedge c) = a \neq 1 = (a \vee b) \wedge (a \vee c)$.

In V_4^5, $a \vee (b \wedge c) = a \neq c = (a \vee b) \wedge (a \vee c)$.

These are the two smallest non-distributive lattices.



Diamond V_3^5



Pentagon V_4^5

13.3.3. Theorem: Prove that the following properties of a lattice L are equivalent:

(i) $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$ for all $x, y, z \in L$;

(ii) $(a \wedge b) \vee c = (a \vee c) \wedge (b \vee c)$ for all $a, b, c \in L$;

(iii) $(a \wedge b) \vee (b \wedge c) \vee (c \wedge a) = (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$ for all $a, b, c \in L$.

Proof: (i) $\Rightarrow$ (ii): Suppose $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$. for all $x, y, z \in L$. So

$(a \vee c) \wedge (b \vee c) = [(a \vee c) \wedge b] \vee [(a \vee c) \wedge c]$ (by (i))

$= [(a \vee c) \wedge b] \vee c$ (by commutative and absorption laws)

$= [(a \wedge b) \vee (c \wedge b)] \vee c$ (by (i))

$= (a \wedge b) \vee [(c \wedge b) \vee c]$ (by associative law)

$= (a \wedge b) \vee c$ (by absorption law)

This proves (ii).

(ii) $\Rightarrow$ (iii): Suppose (ii).

$(a \wedge b) \vee (b \wedge c) \vee (c \wedge a)$

$= (a \wedge b) \vee [(b \wedge c) \vee (c \wedge a)]$

$= \{a \vee [(b \wedge c) \vee (c \wedge a)]\} \wedge \{b \vee [(b \wedge c) \vee (c \wedge a)]\}$ (by (ii))

$= \{a \vee (b \wedge c)\} \wedge \{b \vee (c \wedge a)\}$ (by commutative, associative and absorption laws)

$= \{(a \vee b) \wedge (a \vee c)\} \wedge \{(b \vee c) \wedge (b \vee a)\}$ (by (ii))

$= (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$ (by idempotent law)

(iii) $\Rightarrow$ (i): Suppose that $a \leq c$.

Then $a \wedge b \leq c \wedge b \Rightarrow (a \wedge b) \vee (c \wedge b) = (c \wedge b)$ ….. (*)

Also $a \vee c = c$. Now

$(a \wedge c) \vee (b \wedge c) = (a \wedge c) \vee [(a \wedge b) \vee (c \wedge b)]$ (by (*))

$= (a \wedge b) \vee (b \wedge c) \vee (c \wedge a)$

$= (a \vee b) \wedge (b \vee c) \wedge (c \vee a)$ (by (iii))

$= (a \vee b) \wedge (b \vee c) \wedge c$ (since $a \leq c$)

$= (a \vee b) \wedge c$ (by absorption law)

Now we proved that $(a \vee b) \wedge c = (a \wedge c) \vee (b \wedge c)$.

This shows that (i) is true. The proof is complete.

13.3.4. Corollary: If L is a distributive lattice, then it is a modular lattice.

Proof: Assume that L is a distributive lattice.

Let x, y, z $\in$ L and $x \leq z$. Then by the Theorem 13.3.3., we have that

$(x \wedge y) \vee (y \wedge z) \vee (z \wedge x) = (x \vee y) \wedge (y \vee z) \wedge (z \vee x)$.

Since $x \leq z$, we have that $x \wedge z = x$ and $x \vee z = z$, and so

$(x \wedge y) \vee (y \wedge z) \vee x = (x \vee y) \wedge (y \vee z) \wedge z$

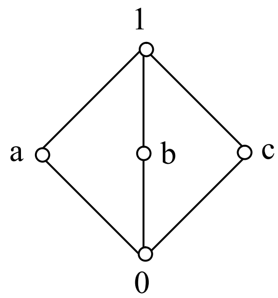
$\Rightarrow x \vee (y \vee z) = (x \vee y) \wedge z$ (by absorption laws).

This shows that L is a modular lattice.

13.3.5. Note: The converse of the Corollary 13.3.4, is not true. That is, there exist modular lattices which are not distributive.

For example, consider the diamond lattice. This lattice is a modular lattice, but not a distributive lattice.

Diamond Lattice L_1



13.3.6. Theorem: A modular lattice L is distributive $\Leftrightarrow$ none of its sub lattices is isomorphic to the Diamond Lattice L_1 (the diamond lattice is also denoted by V_3^5).

Proof: We know that V_3^5 is not distributive.

In a contrary way, suppose that L has a sublattice S which is isomorphic to V_3^5.

Then S is not distributive $\Rightarrow$ L is not distributive, a contradiction.

Hence we conclude that L contains no sublattice which is isomorphic to the Diamond lattice.

Converse: Suppose that L is modular lattice which is not distributive.

Part-(i): Since L is not destributive, by the Theorem 13.3.3., there exists x, y, z $\in$ L such that $(x \wedge y) \vee (y \wedge z) \vee (z \wedge x) < (x \vee y) \wedge (y \vee z) \wedge (z \vee x)$.

Write $s = (x \wedge y) \vee (y \wedge z) \vee (z \wedge x)$,

$t = (x \vee y) \wedge (y \vee z) \wedge (z \vee x)$, $a = s \vee (x \wedge t)$, $b = s \vee (y \wedge t)$, and $c = s \vee (z \wedge t)$.

Part-(ii): Since L is modular and $s < t$, we have that

$a = (s \vee x) \wedge t$, $b = (s \vee y) \wedge t$ and $c = (s \vee z) \wedge t$.

Now $x \wedge t = x \wedge (x \vee y) \wedge (y \vee z) \wedge (z \vee x)$

$= x \wedge (y \vee z)$ (by absorption law) … (i)

Similarly, $y \wedge t = y \wedge (z \vee x)$ and $z \wedge t = z \wedge (x \vee y)$ … (ii)

Now $x \vee s = x \vee (x \wedge y) \vee (y \wedge z) \vee (z \vee x)$

$= x \vee (y \wedge z)$ (by absorption law) … (iii)

Similarly, $y \vee s = y \vee (z \wedge x)$ and $z \vee s = z \vee (x \wedge y)$ ... (iv)

Part-(iii): Now $a \vee b = s \vee (x \wedge t) \vee s \vee (y \wedge t)$ (by the definition of a and b)

$= s \vee (x \wedge t) \vee (y \wedge t)$ (by idempotent law)

$= s \vee \{x \wedge (y \vee z)\} \vee \{y \wedge (z \vee x)\}$ (by (i) and (ii))

$= s \vee \{x \wedge (y \vee z) \vee y\} \wedge (z \vee x)\}$

(by modular law since $x \wedge (y \vee z) \leq x \leq z \vee x$)

$= s \vee [\{(x \vee y) \wedge (y \vee z)\} \wedge (z \vee x)]$ (by modular law since $y \leq y \vee z$)

$= s \vee t$ (by the definition of t)

$= t$ (since $s \leq t$, by the definition of s and t).

Similarly, we can get that $b \vee c = c \vee a = t$.

So we got that $a \vee b = b \vee c = c \vee a = t$ … (v)

Dually, we get that $a \wedge b = b \wedge c = c \wedge a = s$ … (vi)

Part-(iv): Now we prove that the elements s, a, b, c, t are all distinct.

Suppose $s = a$.

Then $a \wedge b = s$ (by (vi)) $= a$ (by our supposition here), and $c \wedge a =$ (by (vi)) $= a$ (by our supposition here)

$\Rightarrow a \leq b$ and $a \leq c$

$\Rightarrow a \vee b = b$ and $a \vee c = c$.

$\Rightarrow b = c = t$ (by (v))

$\Rightarrow t = t \wedge t = b \wedge c = s$ (by (vi)), a contradiction to the fact that $s < t$.

Therefore $s \neq a$.

Similarly, we can prove that $s \neq b$ and $s \neq c$.

Dually we get $t \neq a$, $t \neq b$, and $t \neq c$.

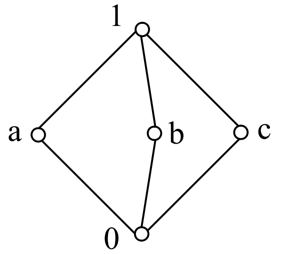
Part-(v): Suppose $a = b$.

Then $s = a \wedge b$ (by (vi)) $= a$ (by the supposition here), a contradiction to the fact that $a \neq s$.

Therefore $a \neq b$. Similarly we can get that $b \neq c$, and $a \neq c$.

Thus a, b, c are all incomparable.

This shows that the set S = {s, a, b, c, t} is a sublattice of L whose Hasse diagram is of the form which is isomorphic to V_3^5 .

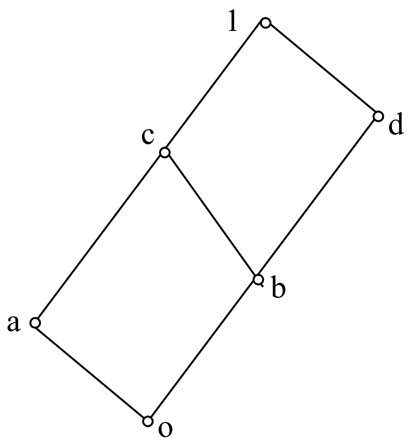


The proof is complete.

13.3.7. Theorem: A lattice $(L, \vee, \wedge)$ is distributive $\Leftrightarrow$ none of its sublattices is isomorphic to either the pentagon lattice N_5 or the Dimond lattice V_3^5 .

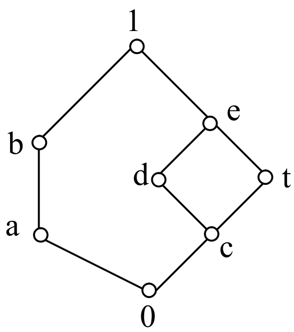
Proof: The proof follows from the Theorem 13.2.6., and Theorem 13.3.6.

13.3.8. Example: (i). Let L be a lattice whose Hasse diagram is as follows.



Then L is a distributive lattice because it has no sublattice isomorphic to either the Diamond lattice or the Pentagon lattice.

(ii). Let L be the lattice given by the following Hasse diagram.



Then L is not a modular lattice because the set
S = {o, a, b, d, 1} is a sublattice of L which is isomorphic to the Pentagon lattice.
Note that there exist some other sublattices of L which are isomorphic to the Pentagon lattice. Finding the other sublattices of L which are ismorphic to the Pentagon lattice, was left to the reader for exercise.

13.4 SOME MORE RESULTS ON DISTRIBUTIVE LATTICES:

13.4.1. Result: For a given lattice L, the following two conditions are equivalent:

(i) $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$, and

(ii) $x \wedge (y \vee z) = (x \wedge y) \wedge (x \wedge z)$ for all $x, y, z \in L$.

Proof: Part-(i): Suppose that

$$x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z) \quad \ldots \text{(i)}$$

Now $(x \wedge y) \vee (x \wedge z) = [(x \wedge y) \vee x] \wedge [(x \wedge y) \vee z]$ (by (i))

$= x \wedge [(x \wedge y) \vee z]$ (by commutative and absorption laws)

$= x \wedge [z \vee (x \wedge y)]$ (by commutative law)

$= x \wedge [(z \vee x) \wedge (z \wedge y)]$ (by (i))

$= [x \wedge (z \vee x)] \wedge [z \wedge y]$ (by associative law)

$= x \wedge (z \wedge y)$ (by commutative and absorption law)

Part-(ii): Suppose that $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z) \ldots$ (ii)

Now $(x \vee y) \wedge (x \vee z) = [(x \vee y) \wedge x] \vee [(x \vee y) \wedge z]$ (by (ii))

$= x \vee [(x \vee y) \wedge z]$ (by commutative and absorption laws)

$= x \vee [z \wedge (x \vee y)]$ (by commutative law)

$= x \vee [(z \wedge x) \vee (z \wedge y)]$ (by (ii))

$= [x \vee (z \wedge x)] \vee [z \wedge y]$ (by associative law)

$= x \vee (z \wedge y)$ (by commutative and absorption law)

Therefore $x \vee (z \wedge y) = (x \vee y) \wedge (x \vee z)$. The proof is complete.

13.4.2. Theorem: A lattice L is distributive $\Leftrightarrow$ the cancellation rule:
$x \wedge y = x \wedge z,\ x \vee y = x \vee z \Rightarrow y = z$ holds for all $x, y, z \in L$.

Proof: Suppose L is distributive.

Let $x, y, z \in L$ and $x \wedge y = x \wedge z$, $x \vee y = x \vee z$.

Then $y = (x \wedge y) \vee y$ (by absorption law)

$= (x \wedge z) \vee y$ (by given condition)

$= (x \vee y) \wedge (z \vee y)$ (by distributive law)

$= (x \vee z) \wedge (y \vee z)$ (by given condition)

$= (x \wedge y) \vee z$ (by distributive law)

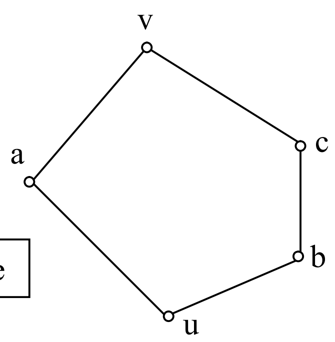
$= (x \wedge z) \vee z$ (by given condition)

$= z$ (by absorption law)

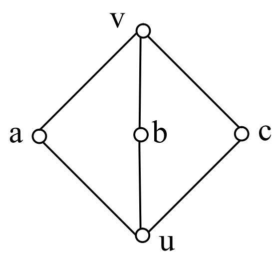
Converse: Assume the cancellation rule.

In a contrary way, we suppose that L is not distributive. Then by the Theorem 13.3.7., we have that L contains a sublattice $S = \{u, a, b, c, v\}$ which is isomorphic to either the Pentagon lattice or the Diamond lattice.

Pentagon lattice



Diamond lattice



In either case, we have that

$$a \wedge b = a \wedge c = u$$

$$a \vee b = a \vee c = v$$

and $b \neq c$, which is a contraction to our assumed cancellation law.

This shows that L is a distributive lattice.

13.4.3. Definition: A lattice L with 0 and 1 is called **complemented** if for each $x \in L$ there exists at least one element y such that $x \wedge y = 0$ and $x \vee y = 1$.

Each such y is called a complement of x. We denote the complement of x by x^1.

13.4.4. Examples: (i) Let $L = P(M)$. Then $B = M \setminus A$ is the unique complement of A.

(ii) In a bounded lattice, 1 is a complement of 0, and 0 is a complement of 1.

(iii) Every chain with more than two elements is not a complemented lattice.

(iv) The complement need not be unique. For example, in the diamond lattice, both the two elements b and c, are complements for the element a.

(v) Let L be the lattice of subspaces of the vector space $\mathbb{R}^2$. If T is a complement of a subspace S, then $S \cap T = \{0\}$ and $S + T = \mathbb{R}^2$.

Hence a complement is a complementary subspace.

13.4.5. Theorem: If L is a distributive lattice, then every element $x \in L$ has at most one complement.

Proof: Let L be a distributive lattice. Suppose $x \in L$ has two complements y_1 and y_2.
Then $x \vee y_1 = 1 = x \vee y_2$ and $x \wedge y_1 = 0 = x \wedge y_2$.
By the Theorem 13.4.2., we have that $y_1 = y_2$, a contradiction.

13.4.6. Definition: Let L be a lattice with zero. An element $a \in L$ is said to be an atom if $a \neq 0$ and if it satisfies the following condition:
$b \in L,\ 0 < b \leq a \Rightarrow b = a$.

13.4.7. Definitions: (i) An element $a \in L$ is said to be join-irreducible if it satisfies the following condition: $b, c \in L,\ a = b \vee c \Rightarrow a = b$ or $a = c$.

(ii) An element is said to be join-irreducible if it is not join-irreducible.

13.4.8. Lemma: Every atom of a lattice with zero is join-irreducible.

Proof: Let a be an atom and let $a = b \vee c,\ a \neq b$.
Then $a = \sup(b, c)$ and so $b < a$. Since a is an atom, we have that $b = 0$.
So $a = b \vee c = 0 \vee c = c$. The proof is complete.

13.4.9. Lemma: Let L be a distributive lattice and let $p \in L$ be join-irreducible with $p \leq a \vee b$. Then $p \leq a$ or $p \leq b$.

Proof: Given that $p \leq a \vee b$.

So $p = p \wedge (a \vee b)$

$= (p \wedge a) \vee (p \wedge b)$ (by distributive law)

$\Rightarrow$ $p = p \wedge a$ or $p = p \wedge b$ (since p is join-irreducible)

$\Rightarrow$ $p \leq a$ or $p \leq b$.

13.4.10. Lemma: Suppose L is a distributive lattice and $a \in L$. If a satisfies the condition: $b, c \in L$, $a \leq b \vee c \Rightarrow a \leq b$ or $a \leq c$, then a is a join-irreducible element.

Proof: Assume the condition that $a \leq b \vee c \Rightarrow a \leq b$ or $a \leq c$.

Let $b, c \in L$ such that $a = b \vee c$.

Then $a \leq b \vee c$ and so we get that $a \leq b$ or $a \leq c$.

But $a = b \vee c \Rightarrow b \leq a$ and $c \leq a$. So, we get that $a = b$ or $a = c$.

This shows that a is a join-irreducible element.

13.4.11. Theorem: Let L be a distributive lattice. Then an element $p \in L$ is join-irreducible $\Leftrightarrow$ p satisfies the following condition:

$a, b \in L$, $p \leq a \vee b \Rightarrow p \leq a$ or $p \leq b$.

Proof is the combination of Lemma 13.4.9 and Lemma 13.4.10.

13.4.12. Definitions: (i) If $x \in [a, b] = \{v \in L \ / \ a \leq v \leq b\}$ and $y \in L$ with $x \wedge y = a$ and $x \vee y = b$, then y is called a relative complement of x with respect to the interval $[a, b]$.

(ii) If all intervals $[a, b]$ in a lattice L are complemented, then L is called relatively complemented.

(iii) If L has a zero element and all $[0, b]$ are complemented, then L is called sectionally complemented.

13.5 SUMMARY:

In this lesson, we presented two important concepts: Modular Lattice, and Distributive Lattice which plays vital role in the theory of lattices. The diagrams for Diamond lattice and pentagon lattice were also included. Examples to show the difference between the two concepts Modular lattice and distributive lattice given. Few theorems on this concepts were also presented.

13.6 TECHNICAL TERMS:

Modular lattice

A lattice $(L, \vee, \wedge)$ is called a modular lattice if it satisfies the following condition:
$x \leq z \Rightarrow x \vee (y \wedge z) = (x \vee y) \wedge z$ for all $x, y, z \in L$.

Diamond lattice (Definition 13.2.3).

Pentagon lattice (Definition 13.2.4)

Distributive lattice

A lattice L is said to be a distributive lattice if it satisfies the following laws:

(i) $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$, and

(ii) $x \wedge (y \vee z) = (x \wedge y) \vee (x \wedge z)$, for all $x, y, z \in L$.

Complemented Lattice.

A lattice L with 0 and 1 is called complemented if for each $x \in L$ there exists at least one element y such that $x \wedge y = 0$ and $x \vee y = 1$.

13.7 SELF ASSESSMENT QUESTIONS:

1. Define Modular lattice and Distributive lattice. Also provide examples for each.

2. Prove that a lattice $(L, \vee, \wedge)$ is a modular lattice $\Leftrightarrow$ L contains no sublattice which is isomorphic to N_5 (the Pentagon lattice). (Theorem: 13.2.6.)

3. Prove that a modular lattice L is distributive $\Leftrightarrow$ none of its sub lattices is isomorphic to the Diamond Lattice L_1 (the diamond lattice is also denoted by V_3^5). (Theorem 13.3.6.)

4. Define complemented lattice, and give an example.

5. Prove that a modular lattice L is distributive $\Leftrightarrow$ none of its sub lattices is isomorphic to the Diamond Lattice L_1 (the diamond lattice is also denoted by V_3^5). (Theorem 13.3.6.)

6. Prove that a lattice L is distributive $\Leftrightarrow$ the cancellation rule:
$x \wedge y = x \wedge z, x \vee y = x \vee z \Rightarrow y = z$ holds for all $x, y, z \in L$.
(Theorem 13.4.2.)

13.8 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237 .

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Kuncham Syam Prasad

LESSON - 14
BOOLEAN POLYNOMIALS

OBJECTIVE:

- To know Polynomials with coefficients 0 and 1.
- To understand the Boolean polynomials.
- To identify different operations with Boolean polynomials.
- To understand equivalent expressions
- To know minterm or product term.
- To develop skills in finding dnf and cnf for a given expression.

STRUCTURE:

14.1. INTRODUCTION:

In this lesson, we study Boolean polynomials and equivalent expressions. We will make learn to express the given Boolean polynomials in terms of disjunctive normal form and conjunctive normal form.

14.2. SOME DEFINITIONS:

In this section some important useful definitions and examples were included.

14.2.1. Definition: A complemented distributive lattice is called a Boolean algebra (or a Boolean lattice).

14.2.2 Note: (i) By the Theorem 13.4.5. (of the lesson 13), we have that every element x of a complemented distributive lattice, has unique complement (which is denoted by x^1).

(ii) From (i), we can conclude that every element x of a Boolean algebra, has unique complement.

14.2.3 Notation: Henceforth, we use B to denote a Boolean algebra. So B denotes a set with the two binary operations $\wedge$ and $\vee$, with zero element 0 and a unit element 1, and the unary operation of complementation.

We write B = (B, $\wedge$, $\vee$, 0, 1) or B = (B, $\wedge$, $\vee$), in short.

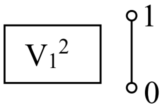
14.2.4 Examples: (i) Let M be a set and P(M) be the power set of M.

Then the system (P(M) , $\cap$, $\cup$, ϕ, M^1) is a Boolean Algebra. Here $\cap$ and $\cup$ are the set-theoretic operations: intersection and union, and the complement is the set-theoretic complement (that is, M \ A = A^1).

In this Boolean algebra, the elements ϕ and M are the "universal bounds".

If M contains exactly n elements, then P(M) contains exactly 2^n elements.

(ii) Let $\mathcal{B}$ be the lattice V_1^2, where the operations are defined by



$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
0	1	1

^{1}	
0	1
1	0

Then ($\mathcal{B}$, $\wedge$,$\vee$, 0, 1, ^{1}) is a Boolean algebra.

(iii) Consider the Boolean algebra $\mathcal{B}$ given in (ii).

Let n be a positive integer.

Consider the set $\mathcal{B}^n$, the Cartesian product of n copies of $\mathcal{B}$.

The set $\mathcal{B}^n$ is a Boolean algebra with respect to the operations given below:

$(i_1,\ldots, i_n) \wedge (j_1,\ldots, j_n) := (i_1 \wedge j_1, \ldots, i_n \wedge j_n)$,

$(i_1 ,\ldots, i_n) \vee (j_1,\ldots, j_n) := (i_1 \vee j_1, \ldots, i_n \vee j_n)$, $(i_1,\ldots, i_n)^1 := (i_1^{\,1}, \ldots, i_n^{\,1})$, and

$0 = (0, \ldots, 0)$, $1 = (1, \ldots, 1)$

(iv) Suppose $\mathcal{B}_1 , \mathcal{B}_2 , \ldots \mathcal{B}_n$ are Boolean algebras.

Consider $\mathcal{B} = \mathcal{B}_1 \times \mathcal{B}_2 \times \ldots \times \mathcal{B}_n$, the Cartesian product of the Boolean algebras $\mathcal{B}_1 ,\mathcal{B}_2 , \ldots \mathcal{B}_n$.

We define the operations $\vee$, $\wedge$ and complementation in $\mathcal{B}$ as follows:

$(i_1,\ldots, i_n) \wedge (j_1,\ldots, j_n) := (i_1 \wedge j_1, \ldots, i_n \wedge j_n)$,

$(i_1 ,\ldots, i_n) \vee (j_1,\ldots, j_n) := (i_1 \vee j_1, \ldots, i_n \vee j_n)$,

$(i_1,\ldots, i_n)^1 := (i_1^1, \ldots, i_n^1)$, and

$0 = (0, \ldots, 0), 1 = (1, \ldots, 1)$

where $i_k, j_k \in \mathcal{B}_k$ for $1 \leq k \leq n$.

It is easy to verify that $\mathcal{B} = \mathcal{B}_1 \times \mathcal{B}_2 \times_{\ldots} \times \mathcal{B}_n$ is a Boolean algebra.

So $\mathcal{B}$ is the direct product of the given Boolean algebras.

14.3. BOOLEAN POLYNOMIALS:

In this section, we study the concepts: Boolean polynomials and Polynomial functions.

14.3.1. Definition: Let $X = \{x_1, \ldots, x_n\}$ be a set of n symbols (called indeterminants or variables). The Boolean polynomials in the variables $x_1, \ldots, x_n$ are the objects which can be obtained by finitely many successive applications of the following:

(i) $x_1, x_2, \ldots, x_n$ and 0, 1 are Boolean polynomials;

(ii) If p and q are Boolean polynomials, then $p \wedge q$, $p \vee q$, and p^1 are Boolean polynomials.

14.3.2. Note: (i) Two polynomials P and Q are said to be equal if we get Q from P by using the properties of Boolean algebra.

(ii) We write $P_n =$ the set of all Boolean polynomials in n variables $x_1, \ldots, x_n$.

14.3.3. Example: (i) The expressions $0, 1, x_1, x_1 \vee 1, x_1 \wedge x_2, x_1^1, x_2,$ $x_1^1 \wedge (x_2 \vee x_1)$ are some examples of Boolean polynomials over $\{x_1, x_2\}$.

(ii) Since every Boolean polynomial over $x_1, \ldots, x_n$ is also a Boolean polynomial Over $x_1, \ldots, x_n, x_{n+1}$, we have that $P_1 \subset P_2 \subset \ldots \subset P_n \subset P_{n+1} \subset \ldots$

14.3.4. Definition: Let B be a Boolean algebra, B^n the direct product of n copies of B, and p a Boolean polynomial in P_n. Then we define a function p_B as follows: p_B : $B^n \to B$; $(a_1, \ldots, a_n) \mapsto p_B (a_1, \ldots, a_n)$. This function p_B is called the Boolean polynomial function induced by p on B. Here $p_B(a_1, \ldots, a_n)$ is the element in B which is obtained from p by replacing each x_i by $a_i \in B$, $1 \le i \le n$.

14.3.5. Example: Suppose that $\mathcal{B}$ denotes the Boolean algebra {0, 1} with the usual operations. Let $n = 2$, $p = x_1 \wedge x_2$, $q = x_2 \wedge x_1$.

Then $p_{\mathcal{B}} : \mathcal{B}^2 \to \mathcal{B}$: $(0, 0) \mapsto 0$, $(0, 1) \mapsto 0$, $(1, 0) \mapsto 0$, $(1, 1) \mapsto 1$; and

$q_{\mathcal{B}}: \mathcal{B}^2 \mapsto \mathcal{B}$: $(0, 0) \mapsto 0$, $(0, 1) \mapsto 0$, $(1, 0) \mapsto 0$, $(1, 1) \mapsto 1$. Therefore $p_{\mathcal{B}} = q_{\mathcal{B}}$.

14.3.6 Note: The Example 16.2.5 shows that the two different Boolean polynomials p and q have the same Boolean polynomial function $p_{\mathcal{B}} = q_{\mathcal{B}}$.

14.3.7 Notation: Let B be a Boolean algebra. Using the notation introduced in the Definition 14.3.4, we define $P_n(B) = \{ \bar{p}_{\mathcal{B}} \ / \ p \in P_n \}$.

14.3.8 Theorem: Let B be a Boolean algebra. Then the set $P_n(B)$ is a Boolean algebra, and also it is a subalgebra of the Boolean algebra $F_n(B)$ of all functions from B_n into B.

Proof: We have to verify that $P_n(B)$ is closed with respect to $\vee$, $\wedge$, and 'the complement of functions'. Also we have to verify that $P_n(B)$ contains f_0 and f_1.

Let $a_1, \ldots, a_n \in B$.

Then $(\bar{p}_B \wedge \bar{q}_B)(a_1, \ldots, a_n) = \bar{p}_B(a_1, \ldots, a_n) \wedge \bar{q}_B(a_1, \ldots, a_n) = \overline{(p \wedge q)}_B (a_1, \ldots, a_n)$

$\Rightarrow (\bar{p}_B \wedge \bar{q}_B) = \overline{(p \wedge q)}_B$.

Now we proved that for all $\bar{p}_B, \bar{q}_B \in P_n(B)$, $\bar{p}_B \wedge \bar{q}_B = \overline{(p \wedge q)}_B \in P_n(B)$.

For $\vee$ and ^{1} we proceed similarly.

Also $\bar{0} = f_0$, $\bar{I} = f_1$ where $f_0 : B^n \to B$ is defined by $f_0(x) = 0$;

and $f_1 : B^n \to B$ defined by $f_1(x) = 1$ for all $x \in B^n$.

14.3.9 Definition: Two Boolean Polynomials $p, q \in P_n$ are equivalent (in symbols $p \sim q$) if their Boolean polynomial functions on $\mathcal{B}$ are equal.

That is, $p \sim q \Leftrightarrow \bar{p}_{\mathcal{B}} = \bar{q}_{\mathcal{B}}$.

14.3.10. Lemma: The relation $\sim$ defined in 16.2.9 is an equivalence relation on P_n.

Proof: Since $\overline{p}_{\mathcal{B}} = \overline{p}_{\mathcal{B}}$, we have that $p \sim p$ for all $p \in P_n$. Let $p, q, r \in P_n$.

Suppose $p \sim q \Rightarrow \overline{p}_{\mathcal{B}} = \overline{q}_{\mathcal{B}} \Rightarrow \overline{q}_{\mathcal{B}} = \overline{p}_{\mathcal{B}} \Rightarrow q \sim p$.

Suppose $p \sim q$ and $q \sim r \Rightarrow \overline{p}_{\mathcal{B}} = \overline{q}_{\mathcal{B}}$ and $\overline{q}_{\mathcal{B}} = \overline{r}_{\mathcal{B}} \Rightarrow \overline{p}_{\mathcal{B}} = \overline{r}_{\mathcal{B}} \Rightarrow p \sim r$.

This shows that $\sim$ is an equivalence relation.

14.3.11. Notation: (i) Consider the relation $\sim$ defined in 16.2.9.

By the above lemma14.3.10, this relation is an equivalence relation.

(ii) The equivalence class containing an element $p \in P_n$ is denoted by $[p]$.

The set of all equivalence classes is denoted by $P_n / \sim$.

So we have that $P_n / \sim \; = \{ [p] \; / \; p \in P_n\}$.

14.3.12 Theorem: (i) $P_n / \sim \; = \{ [p] \; / \; p \in P_n\}$ is a Boolean algebra with respect to the usual operations on equivalence classes $[p] \wedge [q] := [p \wedge q]$ and $[p] \vee [q] := [p \vee q]$.

(ii) $P_n/\sim \; \cong_b P_n(\mathcal{B})$.

Proof: (i) Suppose $[p_1] = [p_2]$ and $[q_1] = [q_2]$. Then $p_1 \sim p_2$ and $q_1 \sim q_2$

$\Rightarrow (p_1)_{\mathcal{B}} = (p_2)_{\mathcal{B}}$ and $(q_1)_{\mathcal{B}} = (q_2)_{\mathcal{B}}$

$\Rightarrow (p_1 \wedge q_1)_{\mathcal{B}} = (p_2 \wedge q_2)_{\mathcal{B}}$

$\Rightarrow p_1 \wedge q_1 \sim p_2 \wedge q_2 \Rightarrow [p_1 \wedge q_1] = [p_2 \wedge q_2]$.

In the same way, we get that $[p_1 \vee q_1] = [p_2 \vee q_2]$.

Therefore the operations $\wedge, \vee$ on $P_n / \sim$ are well defined.

Let $[p], [q] \in P_n / \sim$. Then $[p] \vee [q] = [p \vee q] \in P_n / \sim$.

Similarly, $[p] \wedge [q] = [p \wedge q] \in P_n / \sim$. Therefore $P_n / \sim$ is a lattice.

Now it is easy to verify that $P_n / \sim$ is a Boolean algebra.

(ii) Define a mapping $h : P_n(\mathcal{B}) \rightarrow P_n / \sim$ by $h(\overline{p}_{\mathcal{B}}) := [p]$.

Now we have that $\overline{p}_{\mathcal{B}} = \overline{q}_{\mathcal{B}} \Leftrightarrow p \sim q \Leftrightarrow [p] = [q]$.

Therefore h is well defined and one-one.

Let $[p] \in P_n / \sim$. Then $\overline{p}_{\mathcal{B}} \in P_n(\mathcal{B})$ and $h(\overline{p}_{\mathcal{B}}) := [p]$. This shows that h is onto.

Now $h(\overline{p}_{\mathcal{B}} \vee \overline{q}_{\mathcal{B}}) = h((\overline{p \vee q})_{\mathcal{B}}) = [(p \vee q)] = [p] \vee [q] = h(\overline{p}_{\mathcal{B}}) \vee h(\overline{q}_{\mathcal{B}})$.

Also $h(\overline{p}_{\mathcal{B}} \wedge \overline{q}_{\mathcal{B}}) = h((\overline{p \wedge q})_{\mathcal{B}})$

$= [(p \wedge q)] = [p] \wedge [q] = h(\overline{p}_{\mathcal{B}}) \wedge h(\overline{q}_{\mathcal{B}})$, and

$h[(\overline{p}_{\mathcal{B}})^1] = h(\overline{p^1}_{\mathcal{B}}) = [p^1] = [h(\overline{p}_{\mathcal{B}})]^1$. Therefore h is a lattice homomorphism.

Also $h(\overline{0}_{\mathcal{B}}) = 0$, $h(\overline{1}_{\mathcal{B}}) = 1$.

Since $h(\overline{p}_{\mathcal{B}}) \wedge h(\overline{p^1}_{\mathcal{B}}) = h(\overline{p}_{\mathcal{B}} \wedge \overline{p^1}_{\mathcal{B}}) = h(\overline{0}_{\mathcal{B}}) = 0$, and $h(\overline{p}_{\mathcal{B}}) \vee h(\overline{p^1}_{\mathcal{B}}) = 1$, we have that h is a Boolean homomorphism. So h is a Boolean isomorphism.

14.3.13. Note: For any two equivalent polynomials, the corresponding polynomial functions are equal (on any Boolean algebra).

14.3.14 Theorem: Let $p, q \in P_n$, $p \sim q$, and B an arbitrary Boolean algebra. Then $\overline{p}_{\mathcal{B}} = \overline{q}_{\mathcal{B}}$.

Proof: Since B is a finite Boolean algebra, we have that B is a Boolean subalgebra of $P(X) \cong_b \mathcal{B}^x$ for some set X. Now it is sufficient to prove the result for $\mathcal{B}^x$.

We know (from the definition) that

$p \sim q \Leftrightarrow \overline{p}_{\mathcal{B}} = \overline{q}_{\mathcal{B}}$

$\Leftrightarrow \overline{p}_{\mathcal{B}}(i_1, \ldots, i_n) = \overline{q}_{\mathcal{B}}(i_1, \ldots, i_n)$ for all $i_1, \ldots, i_n \in \mathcal{B}$. Let $f_1, \ldots, f_n \in \mathcal{B}^x$.

Let $x \in X$. For notational convenience we write $A = \mathcal{B}^X$.

Now we have that $(\overline{p}_A(f_1, \ldots, f_n))(x) = \overline{p}_{\mathcal{B}}(f_1(x), \ldots, f_n(x))$

$= \overline{q}_{\mathcal{B}}(f_1(x), \ldots, f_n(x)) = (\overline{q}_A(f_1, \ldots, f_n))(x)$.

Hence $\overline{p}_A = \overline{q}_A$.

14.3.15 Notation: From now onwards, we simply write $\overline{p}$ instead of $\overline{p}_B$ if the domain of $\overline{p}$ is clear. We may replace a given polynomial p by an equivalent polynomial which is in more simple or more systematic form.

14.4. NORMAL FORMS:

14.4.1 Definition: Let N be a subset of P_n. Then N is said to be a system of normal forms if it satisfies the following two conditions:

(i) $p \in P_n \Rightarrow$ there corresponds $q \in N$ such that $p \sim q$;

(ii) $q_1, q_2 \in N$, $q_1 \neq q_2 \Rightarrow q_1 \nsim q_2$.

14.4.2 Notation: We write $p + q$ for $p \vee q$, and pq for $p \wedge q$.

14.4.3 Note : (i) Consider the function $p = x_1x_2^1x_3^1$. It is clear that $\overline{p}$ takes the value 1 only at $(x_1, x_2, x_3) = (1, 0, 0)$ and is zero elsewhere.

(ii) Consider the function $q = x_1x_2^1x_3^1 + x_1x_2x_3$.

This q takes value 1 exactly at (1, 0, 0) and (1, 1, 1).

14.4.4 Note : Let f be a function from $\mathcal{B}^n$ into $\mathcal{B}$.

(i) we find out each $(b_1, \ldots, b_n)$ satisfying the condition: $f(b_1, \ldots, b_n) = 1$.

Also we write down the corresponding product term $x_1^{b_1}\ x_2^{b_2} \ldots x_n^{b_n}$, where $x^1 = x$ and $x^0 = x$.

(ii) The sum $p = \sum_{f(b_1,\ldots,b_n)=1} x_1^{b_1} x_2^{b_2} \ldots x_n^{b_n}$ induces the function $\overline{p} = f$.

Now we represented f as the sum of the product terms of the type $x_1^{c_1} \ldots x_n^{c_n}$.

(iii) We replace each product term $x_1^{b_1}\ x_2^{b_2} \ldots x_n^{b_n}$ in p by $1\ x_1^{b_1}\ x_2^{b_2} \ldots x_n^{b_n}$.

Note that in the representation for p, 1 is the coefficient of the term $x_1^{b_1}\ x_2^{b_2} \ldots x_n^{b_n}$.

(iv) We add to the representation of p the terms of the form $0\ x_1^{c_1} \ldots x_n^{c_n}$ for all the terms $x_1^{c_1} \ldots x_n^{c_n}$ that do not appear in p. (Note that in the representation for p, 0 is the coefficient of these additional terms $x_1^{c_1} \ldots x_n^{c_n}$).

(v) Now by selecting different combinations of zeroes and ones as coefficient of these terms, we get different functions $\mathcal{B}^n \rightarrow \mathcal{B}$.

14.4.5 Notation: Consider the collection N_d of all polynomials in P_n of the form $\sum_{(i_1,\ldots,i_n)} d_{i_1i_2\ldots i_n}\ x_1^{i_1} \ldots x_n^{i_n}$ where each $d_{i_1},\ldots,i_n$ is 0 or 1, and each i_j is 0 or 1.

So we can write

$$N_d = \{ \sum_{(i_1,\ldots,i_n)} d_{i_1i_2\ldots i_n}\ x_1^{i_1} \ldots x_n^{i_n} \ /\ \text{each } d_{i_1},\ldots,i_n \text{ is 0 or 1, and each } i_j \text{ is 0 or 1}\}$$

14.4.6. Theorem: N_d is a system of normal forms in P_n.

Proof: **Part**-(i) In this part we prove that if $p, q \in N_d$ and $p \sim q$, then $p = q$.

Let $p, q \in N_d$ and $p \sim q$.

We follow the notation given in the above Notation 14.4.5.

Now $p = \{ \sum_{(i_1,\ldots,i_n)} d_{i_1 i_2 \ldots i_n} x_1^{i_1} \ldots x_n^{i_n} \}$, and $q = \{ \sum_{(j_1,\ldots,j_n)} e_{j_1,\ldots j_n} x_1^{j_1} \ldots x_n^{j_n} \}$ … (i)

Let $(k_1, k_2, \ldots k_n) \in \{0, 1\}^n$.

Since $p \sim q$, we have that $\overline{p}_{\mathcal{B}} = \overline{q}_{\mathcal{B}}$

$$\Rightarrow \overline{p}_{\mathcal{B}} (k_1, k_2, \ldots k_n) = \overline{q}_{\mathcal{B}}(k_1, k_2, \ldots k_n)$$

$$\Rightarrow \{ \sum_{(i_1,\ldots,i_n)} d_{i_1 i_2 \ldots i_n} x_1^{i_1} \ldots x_n^{i_n} \}(k_1, k_2, \ldots k_n)$$

$$= \{ \sum_{(j_1,\ldots,j_n)} e_{j_1,\ldots j_n} x_1^{j_1} \ldots x_n^{j_n} \}(k_1, k_2, \ldots k_n) \quad \ldots \text{(ii)}$$

Consider the L.H.S of (ii). If $(k_1, k_2, \ldots k_n) = (i_1, i_2, \ldots i_n)$, then

$$d_{i_1 i_2 \ldots i_n} x_1^{i_1} \ldots x_n^{i_n} (k_1, k_2, \ldots k_n) = d_{i_1 i_2 \ldots i_n}.$$

If $(k_1, k_2, \ldots k_n) \neq (i_1, i_2, \ldots i_n)$, then $d_{i_1 i_2 \ldots i_n} x_1^{i_1} \ldots x_n^{i_n} (k_1, k_2, \ldots k_n) = 0$.

Now consider the R.H.S of (ii).

If $(k_1, k_2, \ldots k_n) = (j_1, j_2, \ldots j_n)$, then $(e_{j_1,\ldots j_n} x_1^{j_1} \ldots x_n^{j_n}) (k_1, k_2, \ldots k_n) = e_{j_1,\ldots j_n}$.

If $(k_1, k_2, \ldots k_n) \neq (j_1, j_2, \ldots j_n)$, then $(e_{j_1,\ldots j_n} x_1^{j_1} \ldots x_n^{j_n}) (k_1, k_2, \ldots k_n) = 0$.

This shows that $d_{k_1 k_2 \ldots k_n} = e_{k_1,\ldots k_n}$. This is true for all n-tuples $(k_1, k_2, \ldots k_n)$.

Hence $p = q$.

Part-(ii): Write $N_d(B) = \{\overline{p_B} \ / \ p \in N_d\}$.

Now $N_d(B) = \{\overline{p_B} \ / \ p \in N_d\} \subseteq P_n/\sim \ \Rightarrow \ |N_d(B)| \leq |P_n/\sim|$.

Write $B = \{0, 1\}$ the Boolean algebra consisting of two elements.

We know that $F_n(B) = \{f : B^n \to B\}$.

Define $\phi : N_d(B) \to F_n(B)$ by $\phi(\overline{p_B})(i_1, i_2, \ldots i_n) = d_{i_1 i_2 \ldots i_n}$

where $p = \sum_{(i_1,\ldots,i_n)} d_{i_1 i_2 \ldots i_n} x_1^{i_1} \ldots x_n^{i_n}$. Suppose $q = \sum_{(j_1,\ldots,j_n)} e_{j_1,\ldots j_n} x_1^{j_1} \ldots x_n^{j_n}$.

Now $\phi(\overline{p_B}) = \phi(\overline{q_B}) \Leftrightarrow d_{i_1 i_2 \ldots i_n} = e_{i_1 i_2 \ldots i_n}$ for all n-tuples $(i_1, i_2, \ldots i_n)$

$$\Leftrightarrow p \sim q \Leftrightarrow \overline{p_B} = \overline{q_B}.$$

This show that ϕ is well defined and one-one.

Part-(iii): Let $f \in F_n(B)$, then by above note $f = \overline{p}$ where $p = \sum_{f(b_1,\ldots,b_n)=1} x_1^{b_1},\ldots,x_n^{b_n}$

$$= \sum_{(i_1,\ldots,i_n)} d_{i_1 i_2 \ldots i_n}\ x_1^{i_1} \ldots\ x_n^{i_n} \quad \text{with } d_{i_1 i_2 \ldots i_n} = 1 \Leftrightarrow f(i_1, i_2, \ldots i_n) = 1.$$

Now $\phi(\overline{p_B})(i_1, i_2, \ldots i_n) = d_{i_1 i_2 \ldots i_n} = f(i_1, i_2, \ldots i_n)$. This shows that ϕ is onto.

Part-(iv): From Part-(ii) and Part-(iii), we get that ϕ is a bijection and

so $|N_d(B)| = |F_n(B)| = 2^{2^n}$.

Now $|N_d(B)| \leq |P_n/\sim| = |P_n(B)| \leq |F_n(B)|$ (by the Theorem 14.3.8) $= |N_d(B)|$.

$\Rightarrow$ $|N_d(B)| = |P_n/\sim| = 2^{2^n}$.

$\Rightarrow$ For any $q \in P_n$ there exists $\overline{p_B} \in N_d(B)$ such that $\overline{q_B} = \overline{p_B}$.

14.5. DISJUNCTIVE NORMAL FORMS:

14.5.1. Definition: $N_d = \{ \sum_{(i_1,\ldots,i_n)} d_{i_1 i_2 \ldots i_n}\ x_1^{i_1} \ldots\ x_n^{i_n}$ / each $d_{i_1},\ldots,i_n$ is 0 or 1, and each i_j is 0 or 1} is called the system of disjunctive normal forms. Each summand is called a minterm.

14.5.2. Corollary: (i) N_d has 2^{2^n} elements.

(ii) P_n splits into 2^{2^n} different equivalence classes.

Proof: (i) There are 2^n distinct product terms of the form $x_1^{i_n}, \ldots., x_n^{i_n}$. Now the general form of a sum of products can be represented as $\sum_{(i_1,\ldots,i_n)} d_{i_1 i_2 \ldots i_n}\ x_1^{i_1} \ldots\ x_n^{i_n}$. In this representation,

$d_{i_1,\ldots,i_n}$ is the coefficient of $x_1^{i_n}, \ldots., x_n^{i_n}$. It is clear that $d_{i_1,\ldots,i_n} = 0$ or $= 1$.

Hence there exists 2^{2^n} such representations. This shows that N_d has 2^{2^n} elements.

(ii) In the part-(iv) of the proof of the Theorem 14.4.6., we proved that $|P_n/\sim| = 2^{2^n}$.

This shows that there exists 2^{2^n} equivalence classes.

In other words, we can say that P_n splits into 2^{2^n} different equivalence classes.

14.5.3. Definition: A Boolean algebra B is said to be polynomially complete if $P_n(B) = F_n(B)$.

14.5.4. Corollaries: (i) $|P_n/\sim| = 2^{2^n}$ and $P_n(\mathcal{B}) = F_n(\mathcal{B})$, where $\mathcal{B} = \{0, 1\}$.

This means the Boolean algebra $\mathcal{B}$ is polynomially complete.

(ii) If $|B| = m > 2$, then $|P_n(B)| = |P_n/\sim| = 2^{2^n} < m^{m^n} = |F_n(B)|$; and so $P_n(B) \subseteq F_n(B)$. This means if $|B| > 2$, then B is not polynomially complete.

(iii) If $p = \sum_{(i_1,\ldots,i_n)} d_{i_1,\ldots,i_n} x_1^{i_1}, \ldots, x_n^{i_n}$, then $d_{i_1,\ldots,i_n} = \bar{p}(i_1, \ldots, i_n)$.

(iv) If $p \in P_n$, then $p \sim \sum_{\bar{p}(i_1,\ldots,i_n)=1} x_1^{i_1}, \ldots, x_n^{i_n}$.

Proof: (i) By the Corollary 14.5.2. (ii), we have that $|P_n/\sim| = 2^{2^n}$.

Since $P_n(\mathcal{B}) \subseteq F_n(\mathcal{B})$ and $|P_n(\mathcal{B})| = 2^{2^n} = |F_n(\mathcal{B})|$, we have that $P_n(\mathcal{B}) = F_n(\mathcal{B})$.

(ii) is clear.

(iii) Let $p = \sum_{(i_1,\ldots,i_n)} d_{i_1,\ldots,i_n} x_1^{i_1}, \ldots, x_n^{i_n}$. Let $(j_1, j_2, \ldots j_n)$ be an n-tuple of $\{0, 1\}$.

Then $\bar{p}(j_1, j_2, \ldots j_n) = \sum_{(i_1,\ldots,i_n)} d_{i_1,\ldots,i_n} x_1^{i_1}, \ldots, x_n^{i_n} (j_1, j_2, \ldots j_n)$.

Consider the R.H.S.

If $(i_1, i_2, \ldots i_n) \neq (j_1, j_2, \ldots j_n)$, then the related term is zero.

If $(i_1, i_2, \ldots i_n) = (j_1, j_2, \ldots j_n)$, then the related term is equal to $d_{j_1,\ldots,j_n}$.

Therefore $\bar{p}(j_1, j_2, \ldots j_n) = d_{j_1,\ldots,j_n}$.

(iv) From (iii), we have that $d_{i_1,\ldots,i_n} = \bar{p}(i_1, \ldots, i_n)$.

If $\bar{p}(i_1, \ldots, i_n) = 0$, then $d_{i_1,\ldots,i_n} = 0$, and so the term $d_{i_1,\ldots,i_n} x_1^{i_1}, \ldots, x_n^{i_n} = 0$.

So in the representation $p = \sum_{(i_1,\ldots,i_n)} d_{i_1,\ldots,i_n} x_1^{i_1}, \ldots, x_n^{i_n}$, we may remove the terms $d_{i_1,\ldots,i_n} x_1^{i_1}, \ldots, x_n^{i_n} = 0$ when $\bar{p}(i_1, \ldots, i_n) = 0$.

If $\bar{p}(i_1, \ldots, i_n) = 1$, then $d_{i_1,\ldots,i_n} = \bar{p}(i_1, \ldots, i_n) = 1$, and so the related term $d_{i_1,\ldots,i_n} x_1^{i_1}, \ldots, x_n^{i_n} = x_1^{i_1}, \ldots, x_n^{i_n}$.

Hence we conclude that $p \sim \sum_{\bar{p}(i_1,\ldots,i_n)=1} x_1^{i_1}, \ldots, x_n^{i_n}$.

14.5.5. Example: We consider P_2 the set of polynomials in two variables x_1, x_2. Observe that the polynomial expressions $0x_1^1x_2^1 + 0x_1^1x_2 + 0x_1x_2^1 + 0x_1x_2$, and 0 are equivalent. So these two polynomials belongs to unique equivalence class in $P_n/\sim$.

Let us call this class as Class # 1.

Observe that the two polynomials $0x_1^1x_2^1 + 0x_1^1x_2 + 0x_1x_2^1 + 1x_1x_2$, and x_1x_2 are equivalent. So these two polynomials belongs to unique equivalence class in $P_n/\sim$.

Let us call this class as Class # 2.

In this way, we can find all $|P_n/\sim| = 2^{2^2} = 16$ equivalence classes.

These equivalence classes were presented in the Table-1.

Table-1

$0x_1^1x_2^1 + 0x_1^1x_2 + 0x_1x_2^1 + 0x_1x_2$, …, 0, …	← Class # 1
$0x_1^1x_2^1 + 0x_1^1x_2 + 0x_1x_2^1 + 1x_1x_2$, …, x_1x_2,…	← Class # 2
$0x_1^1x_2^1 + 0x_1^1x_2 + 1x_1x_2^1 + 0x_1x_2$, …,$x_1x$…	← Class # 3

……………………………………………………………………

……………………………………………………………………

……………………………………………………………………

$1x_1^1 + 1x_1^1x_2 + 1x_1x_2^1 + 1x_1x_2$, …,1, …	← Class # 16

Since a term with coefficient 0 is equal to zero, we may omit the terms with coefficient 0.

If 1 is the coefficient of a term, then by omitting 1 there is no change in the truth value of the term. So if a term has the coefficient 1, then we may not write down the coefficient.

Note that the expressions $1x_1x_2$ and x_1x_2 are different, but they are equivalent polynomial expressions. So these two belong to unique equivalence class in $P_n/\sim$.

So we can take x_1x_2 has a representative of the class to which it belongs.

Such representatives from the equivalence classes 1 to 16 were presented in the Table-2.

Table-2

…, 0, …	← Class # 1
…, x_1x_2 , …	← Class # 2
…, $x_1x_2^1$, …	← Class # 3
…,$x_1x_2^1$, + x_1x_2, …	← Class # 4
…, $x_1^1x_2$, ….	← Class # 5
…, $x_1^1x_2 + x_1x_2$, …	← Class # 6
…, $x_1^1x_2 + x_1x_2^1$, …	← Class # 7
…, $x_1^1x_2 + x_1x_2^1 + x_1x_2$,…	← Class # 8

…, $x_1^1x_2^1$, …	← Class # 9
…, $x_1^1x_2^1 + x_1x_2$, …	← Class # 10
…, $x_1^1x_2^1 + x_1x_2^1$,…	← Class # 11
…,$x_1^1x_2^1 + x_1x_2^1 + x_1x_2$,…	← Class # 12
…, $x_1^1x_2^1 + x_1^1x_2$, …	← Class # 13
…, $x_1^1x_2^1 + x_1^1x_2 + x_1x_2$,…	← Class # 14
…, $x_1^1x_2^1 + x_1^1x_2 + x_1x_2^1$, …	← Class # 15
$x_1^1x_2^1 + x_1^1x_2 + x_1x_2^1 + x_1x_2$,..	← Class # 16

Observe the Table-2. The expression $x_1x_2^1 + x_1x_2$ is a representative of class-4.

Now we have that $x_1x_2^1 + x_1x_2 \sim x_1(x_2^1 + x_2) \sim x_1.1 \sim x_1$.

So instead of $x_1x_2^1 + x_1x_2$, we can take x_1 as the representative of class-4.

Note that the representative x_1 got simpler form.

In the Table-3, we present such simple representatives.

Table-3

…, 0, …	← Class # 1
…, x_1x_2, …	← Class # 2
…, $x_1x_2^1$, …	← Class # 3
…, x_1 ,…	← Class # 4
…, $x_1^1x_2$, ….	← Class # 5
…,x_2,…	← Class # 6
…,$x_1^1x_2 + x_1x_2^1$,…	← Class # 7
…, $x_1 + x_2$, …	← Class # 8
…, $x_1^1x_2^1$, …	← Class # 9
…, $x_1^1x_2^1 + x_1x_2$, …	← Class # 10
…, x_2^1, …	← Class # 11
…, $x_1 + x_2^1$, …	← Class # 12
…, x_1^1, …	← Class # 13
…, $x_1^1 + x_2$ …	← Class # 14
…, $x_1^1 + x_2^1$, …	← Class # 15
…, 1, …	← Class # 16

14.5.6. Note: (i) . Now we recollect the notation. Let $e \in \{0, 1\}$ for $1 \le i \le n$.

Define $x_i^e = x_i$ if $e = 1$ $= \overline{x_i}$ if $e = 0$

(ii). For any Boolean polynomial f in $x_1, x_2, \ldots, x_n$,

consider $\sum [f(e_1, e_2, \ldots, e_n)\ x_1^{e_1}\ x_2^{e_2} \ldots x_n^{e_n}] = m_0 \vee m_1 \vee \ldots \vee m_{(2^n - 1)}$, where

$m_i = f(d_1, d_2, \ldots, d_n)\ x_1^{d_1}\ x_2^{d_2} \ldots x_n^{d_n}$ and $d_i = 0$ or 1 for $1 \le I \le n$.

14.5.7. Example: Consider the function $f(x_1, x_2) = (x_1 \vee x_2)^1 \wedge ((x_1 \wedge x_2) \vee x_2^{\,1})$ on two variables x_1 and x_2, on the Boolean algebra $B = \{0, 1\}$.

Now $f(0, 0) = ((0 \wedge 0)^1) \wedge ((0 \wedge 0) \vee 0^1) = 0^1 \wedge (0 \vee 0^1) = 1 \wedge (0 \vee 1) = 1 \wedge 1 = 1$.

Similarly, $f(0, 1) = 0$, $f(1, 0) = 0$ and $f(1, 1) = 0$.

Here we have m_0, m_1, m_2, m_3 [because $3 = 2^n - 1$, where n is the number of the variables]. Observe that $m_0 = f(0, 0) \wedge x_1^0 \wedge x_2^0 = f(0, 0) \wedge x_1^1 \wedge x_2^{\,1} = x_1^1 \wedge x_2^{\,1}$.

$m_1 = f(0, 1) \wedge x_1^0 \wedge x_2^1 = 0 \wedge x_1^1 \wedge x_2 = 0$.

$m_2 = f(1, 0) \wedge x_1^1 \wedge x_2^0 = 0 \wedge x_1 \wedge x_2^{\,1} = 0$,

$m_3 = f(1, 1) \wedge x_1^1 \wedge x_2^1 = 0 \wedge x_1 \wedge x_2 = 0$.

Therefore $\vee[f(e_1, e_2) \wedge x_i^{e_i} \wedge x_i^{e_2}] = m_0 \vee m_1 \vee m_2 \vee m_3$

$$= x_1^{\,1}\ x_2^{\,1} \vee 0 \vee 0 \vee 0 = x_1^{\,1}\ x_2^{\,1}.$$

14.5.8. Lemma: Let f(x) be a Boolean polynomial in one variable x with coefficients from the Boolean algebra B. Then $f(x) = (f(1) \wedge x) \vee (f(0) \wedge x^1)$.

Proof: (This proof is by inductive method). We know that the elements of B and x are Boolean expressions.

Step-(i): Fix $b \in B$ and suppose $f(a) = b$ for all $a \in B$.

Then f is a constant function and $f(1) = f(0) = b$.

Now $(f(1) \wedge x) \vee (f(0) \wedge x^1) = (b \wedge x) \vee (b \cap x^1) = b \wedge (x \vee x^1) = b \wedge 1 = b = f(x)$.

Therefore in this case, the result is true.

Step-(ii): Suppose $f(a) = x$ for all $a \in B$. Then $f(1) = x$ and $f(0) = x$.

Consider $(f(1) \wedge x) \vee (f(0) \wedge x^1) = (x \wedge x) \vee (x \wedge x^1) = x \vee 0 = x = f(x)$.

Therefore in this case, the result is true.

Step-(iii): Suppose the result is true for two Boolean polynomials f(x) and g(x).

Now $f(x) \vee g(x) = [(f(1) \wedge x) \vee (f(0) \wedge x^1)] \vee [(g(1) \wedge x) \vee (g(0) \wedge x^1)]$

(by supposition)

$= (f(1) \wedge x) \vee (g(1) \wedge x) \vee (f(0) \wedge x^1) \vee (g(0) \wedge x^1)$

$= [(f(1) \vee g(1)) \wedge x] \vee [f(0) \vee g(0)) \wedge x^1] = [(f \vee g)(1) \wedge x] \vee [(f \vee g)(0) \wedge x^1]$.

This shows that the result is true for the polynomials $f \vee g$.

Next we prove for $f(x) \wedge g(x)$.

$f(x) \wedge g(x) = [(f(1) \wedge x) \vee (f(0) \wedge x^1)] \wedge [(g(1) \wedge x) \vee (g(0) \wedge x^1)]$ (by supposition)

$= \{[(f(1) \wedge x) \vee (f(0) \wedge x^1)] \wedge (g(1) \wedge x)\} \vee \{[(f(1) \wedge x) \vee (f(0) \wedge x^1)] \wedge (g(0) \wedge x^1)\}$

$= \{[(f(1) \wedge x) \wedge (g(1) \wedge x)] \vee [(f(0) \wedge x^1) \wedge (g(1) \wedge x)]\}$

$\vee \{[f(1) \wedge x) \wedge (g(0) \wedge x^1)] \vee [(f(0) \wedge x^1) \wedge (g(0) \wedge x^1\}\}$

$= \{[f(1) \wedge g(1) \wedge x] \vee [0]\} \vee \{[0] \vee [f(0) \wedge x^1 \wedge g(0)]\}$

$= \{[(f \wedge g)(1)] \wedge x\} \vee \{[(f \wedge g)(0)] \wedge x^1\}$.

Step-(iv): Now we show that if the result is true for $h(x) = x^1$.

$[h(1) \wedge x] \vee [h(0) \wedge x^1] = (0 \wedge x) \vee (1 \wedge x^1) = 0 \vee x^1 = x^1 = h(x)$.

Since all the polynomials are written by using x, x^1, b, $\wedge$, $\vee$ and ^{1}, we may conclude that the result is true for all polynomials $f(x)$ in one variable x with coefficients from the Boolean algebra. Hence the lemma is proved for all polynomials $f(x)$ in one variable x with coefficients from the Boolean algebra.

14.5.9 Lemma: Boolean polynomial $f(x_1, x_2, \ldots, x_n)$ is equal to $[f(1, x_2, \ldots, x_n) \wedge x_1] \vee [f(0, x_2, \ldots, x_n) \wedge x_1^1]$.

Proof: Write B^*= the set of all Boolean polynomials in the variables $x_2, x_3, \ldots, x_n$.

Since B^* is a Boolean algebra, we may consider a Boolean polynomial in the variables $x_1, x_2, \ldots, x_n$ as a Boolean polynomial h in the single variable x_1 with coefficients from the Boolean algebra B^*.

Now $f(x_1, x_2, \ldots, x_n) = h(x_1) = (h(1) \wedge x_1) \vee (h(0) \wedge x_1^1)$ (by the Lemma 14.5.8.)

$= [f(1, x_2, \ldots, x_n) \wedge x_1] \vee [f(0, x_2, \ldots, x_n) \wedge x_1^1]$.

Hence $f(x_1, \ldots, x_n) = [f(1, x_2, \ldots, x_n) \wedge x_1] \vee [f(0, x_2, \ldots, x_n) \wedge x_1^1]$.

The proof is complete.

14.5.10. Theorem: If $f(x_1, x_2, \ldots, x_n)$ is a Boolean polynomial, then

$f(x_1, x_2, \ldots, x_n) = \vee (f(e_1, e_2, \ldots, e_n) \wedge x_1^{e_i} \wedge \ldots \wedge x_n^{e_n})$.

Proof: The proof is by induction on the number of variables.

If n = 1, then there is only one variable x. Then we have to show $f(x) = \vee(f(e) \wedge x^e)$.

Consider $\vee(f(e) \wedge x^e) = (f(1) \wedge x^1) \vee (f(0) \wedge x^0) = (f(1) \wedge x) \vee (f(0) \wedge x^1)$.

This is true by the Lemma 14.5.8.

Assume the induction hypothesis, that is, the result is true for (n -1) variables.

Let $f(x_1, x_2, \ldots, x_n)$ be a polynomial in n variables.

$f(x_1, x_2, \ldots, x_n) = [f(1, x_2, \ldots, x_n)) \wedge x_1] \vee [f(0, x_2, \ldots, x_n)) \wedge x^1]$ (by the Lemma 14.5.9.)

$= \{[\vee(f(1, e_2, \ldots, e_n) \wedge x_2^{e_2} \wedge \ldots \wedge x_n^{e_n}] \wedge x_1\} \vee \{[\vee(f(0, e_2, \ldots, e_n) \wedge x_2^{e_2} \wedge \ldots \wedge x_n^{e_n}] \wedge x_1^1\}$

$= \{\vee[f(1, e_2, \ldots, e_n) \wedge x_1^1 \wedge x_2^{e_2} \wedge \ldots \wedge x_n^{e_n}]\} \vee \{\vee [f(0, e_2, \ldots, e_n) \wedge x_1^0 \wedge x_2^{e_2} \wedge \ldots \wedge x_n^{e_n}]\}$

$= \vee [f(e_1, e_2, \ldots, e_n) \wedge x_1^{e_1} \wedge \ldots \wedge x_n^{e_n}]$. The proof is complete..

14.5.11. Note: (i) An expression of the form $x_{i_1}^{e_1} \wedge x_{i_2}^{e_2} \wedge \ldots \wedge x_{i_k}^{e_k}$ is called a product term or minterm. The union of such product terms is called a sum of products.

(ii) A disjunctive normal form (d.n.f, in short) for a Boolean polynomial $f(x_1, x_2, \ldots, x_n)$ is a sum of products (of the form $x_{i_1}^{e_1} \wedge x_{i_2}^{e_2} \wedge \ldots \wedge x_{i_k}^{e_k}$) which represents f

14.5.12. Black Box Method: (To find d.n.f of a given Boolean polynomial).

In this discussion, the coefficients in the Boolean Polynomial f are taken from the Boolean Algebra {0, 1}.

Now the truth table of functional values of the polynomial f determines the disjunctive normal form simply by including each product term that occurs when the function takes value 1.

[If f do not take value 1, then $f(d_1, \ldots, d_n) = 0$ and so the corresponding

$m_i = f(d_1, \ldots, d_n) .\ x_1^{d_1}\ x_2^{d_2} \ldots\ x_n^{d_n} = 0\ \ x_1^{d_1}\ x_2^{d_2} \ldots\ x_n^{d_n} = 0$].

14.5.13. Example: Consider the function

$f(x_1, x_2, x_3) = [x_1 \wedge ((x_2 \vee x_3)^1)] \vee \{[(x_1 \wedge x_2) \vee x_3^1] \wedge x_1\}$.

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	f^1
0	0	0	0	1
0	0	1	0	1
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	1	0
1	1	1	1	0

The functional values of f and f^1 are mentioned in the table.

(i) From the table, it is clear that f takes value 1 only when $(d_1, d_2, d_3) = (1, 0, 0)$, $(d_1, d_2, d_3) = (1, 1, 0)$, and $(d_1, d_2, d_3) = (1, 1, 1)$.

Therefore $m_4 = f(d_1, d_2, d_3) \wedge x_1^{d_1} \wedge x_2^{d_2} \wedge x_3^{d_3} = f(1, 0, 0) \wedge x_1^1 \wedge x_2^0 \wedge x_3^0$

$= 1 \wedge x_1 \wedge x_2^1 \wedge x_3^1 = x_1 .\, x_2^1\, x_3^1$;

$m_6 = f(1, 1, 0) \wedge x_1^1 \wedge x_2^1 \wedge x_3^0 = x_1x_2\, x_3^1$;

$m_7 = f(1, 1, 1) \wedge x_1^1 \wedge x_2^1 \wedge x_3^1 = 1 \wedge x_1 \wedge x_2 \wedge x_3 = x_1x_2x_3$.

Since $m_i = 0$ for all $i \notin \{4, 6, 7\}$, we have that the disjunctive normal form is

$\vee\, m_i = x_1\, x_2^1\, x_3^1 \vee x_1x_2\, x_3^1 \vee x_1x_2x_3$.

(ii) Next we find the disjunctive normal form for $f^1(x_1, x_2, x_3)$.

Observe the table. For five values of (d_1, d_2, d_3) we have $f^1(d_1, d_2, d_3) = 1$.

By following the same steps as in (i), we get that

$f^1(x_1, x_2, x_3) = x_1^1x_2^1x_3^1 \vee x_1^1x_2^1x_3 \vee x_1^1x_2x_3^1 \vee x_1^1x_2x_3 \vee x_1x_2^1x_3$.

14.6. CONJUNCTIVE NORMAL FORMS:

In this section we discuss another normal form named as Conjunctive normal form.

14.6.1. Note: (i) By the duality principle (also refer Theorem 14.5.10.), we have that

$f(x_1, x_2, \ldots, x_n) = \wedge\, [f(e_1, \ldots, e_n) \vee (x_1^1)^{e_1} \vee \ldots \vee (x_n^1)^{e_n}]$.

This form is called the conjunctive normal form (c.n.f., in short) of the function $f(x_1, x_2, \ldots, x_n)$.

(ii) We may represent this conjunctive normal form as follows:

$$f = \prod_{(i_1,\ldots,i_n)} (d_{i_1,\ldots,i_n}) + x_1^{i_1} + \ldots + x_n^{i_n}).$$

(iii) It is clear that the conjunctive normal form of a given function f is the complement of the disjunctive normal form of $f^1(x)$.

(iv) In the above example, $f^1(x_1, x_2, x_3) = x_1^1x_2^1x_3^1 \vee x_1^1x_2^1x_3 \vee x_1^1x_2x_3^1 \vee x_1^1x_2x_3 \vee x_1x_2^1x_3$ is the disjunctive normal form of f^1. Therefore the conjunctive normal form is

$f = [\, f^1(x_1, x_2, x_3)\,]$

$= (x_1^1x_2^1x_3^1)^1 \wedge (x_1^1x_2^1x_3)^1 \vee (x_1^1x_2x_3^1)^1 \vee (x_1^1x_2x_3)^1 \vee (x_1x_2^1x_3)^1$

$= (x_1 \vee x_2 \vee x_3).(x_1 \vee x_2 \vee x_3^1).(x_1 \vee x_2^1 \vee x_3).\,(x_1 \vee x_2^1 \vee x_3^1).(x_1^1 \vee x_2 \vee x_3^1)$.

14.6.2. Example: (i) Now we wish to find the disjunctive normal form of

$p = ((x_1 + x_2)^1 x_1 + x_2^{111})^1 + x_1x_2 + x_1x_2^1$. We list the values of $\bar{p}$ in the table.

Now $\bar{p}(0, 0) = ((0 + 0)^1 0 + 0^{111})^1 + 00 + 00^1 = 0$.

Similarly we get the values for $\bar{p}(0, 1)$, $\bar{p}(1, 0)$, and $\bar{p}(1, 1)$.

Observe the table. We follow the block box method (See 14.5.12.).

b_1	b_2		$\bar{p}(b_1, b_2)$
0	0		0
0	1		1
1	0		1
1	1		1

Now we get that $p = 0x_1^1x_2^1 + 1x_1^1x_2 + 1x_1x_2^1 + 1x_1x_2 = x_1^1x_2 + x_1x_2^1 + x_1x_2$.

This is the d.n.f of p.

(ii) Observe that $p = x_1^1x_2 + x_1x_2^1 + x_1x_2 = x_1^1x_2 + x_1(x_2^1 + x_2) = x_1^1x_2 + x_1(1)$

$= x_1^1x_2 + x_1 = (x_1^1 + x_1)(x_2 + x_1) = (1)(x_2 + x_1) = x_2 + x_1$.

Note that p reduced to its simpler form $x_2 + x_1$.

14.6.3. Note: To get a simple form of the given Boolean polynomial, we may follow the following steps:

Step-(i): Find the disjunctive normal form.

Step-(ii): Reduce the d.n.f by using the laws of Boolean algebra.

14.6.4. Problem: Find a Boolean polynomial p that induces the function f given by the following table:

b_1	b_2	b_3		$f(b_1, b_2, b_3)$	
0	0	0		1	←
0	0	1		0	
0	1	0		0	
0	1	1		1	←
1	0	0		1	←
1	0	1		0	
1	1	0		0	
1	1	1		0	

Solution: We marked the lines where the value of f is 1.

Write $p = x_1^1x_2^1x_3^1 + x_1^1x_2x_3 + x_1x_2^1x_3^1$

(these product terms are related to the rows corresponding to the arrows).

Now p induces the function f.

14.6.5. Note: Consider f and p given in the Problem 14.6.4.

$p = x_1^1x_2^1x_3^1 + x_1^1x_2x_3 + x_1x_2^1x_3^1$

$\sim x_1^1x_2^1x_3^1 + x_1x_2^1x_3^1 + x_1^1x_2x_3$ (by associative and commutative laws)

$\sim (x_1^1 + x_1)\, x_2^1x_3^1 + x_1^1x_2x_3$ (by distributive law)

$\sim x_2^1x_3^1 + x_1^1x_2x_3$.

Therefore $p \sim q$ where $q = x_2^1x_3^1 + x_1^1x_2x_3$.

So q is also a solution to our problem 14.6.4.

That is, $\overline{q} = \overline{p} = f$. In other words both q and p induces the same function f.

14.6.6. Problem: Find the c.n.f for $p = x_1^1x_2 + x_1x_2^1$.

Solution: The d.n.f for p was given. One way of getting c.n.f from d.n.f is to write p as $(p^1)^1$. We expand p^1 by using the de Morgan's laws.

$p = x_1^1x_2 + x_1x_2^1 \sim (x_1^1x_2 + x_1x_2^1)^{11} \sim ((x_1^1x_2 + x_1x_2^1)^1)^1$

$\sim ((x_1 + x_2^1)\,(x_1^1 + x_2))^1 \sim (x_1x_1^1 + x_1x_2 + x_2^1x_1^1 + x_2^1x_2)^1$

$\sim (x_1x_2 + x_2^1x_1^1)^1$ (by complement laws) $\sim (x_1^1 + x_2^1)\,(x_1 + x_2)$.

Now p gets the form "product of sums form". This form is the required c.n.f.

14.7 SUMMARY:

In this lesson, we studied Boolean polynomials in n variables, and equivalent expressions. We will make learn to express the given Boolean polynomials in terms of disjunctive normal form and conjunctive normal form. Few necessary examples were included for the convenience of the reader to learn to obtain disjunctive normal form and conjunctive normal form for the given Boolean polynomials.

14.8 TECHNICAL TERMS:

System of normal forms

Let N be a subset of P_n. Then N is said to be a system of normal forms if it satisfies the following two conditions: (i) $p \in P_n \Rightarrow$ there corresponds $q \in N$ such that $p \sim q$; and (ii) $q_1, q_2 \in N$, $q_1 \neq q_2 \Rightarrow q_1 \nsim q_2$.

Product term (or minterm).

An expression of the form $x_{i_1}^{e_1} \wedge x_{i_2}^{e_2} \wedge \ldots \wedge x_{i_k}^{e_k}$ is called a product term or minterm.

Disjunctive normal form:

Boolean polynomial $f(x_1, x_2, \ldots, x_n)$ which is a sum of products (of the form $x_{i_1}^{e_1} \wedge x_{i_2}^{e_2} \wedge \ldots \wedge x_{i_k}^{e_k}$) that represents the given Boolean polynomial f is named as disjunctive normal form of f.

Conjunctive normal form:

The form $f = \prod_{(i_1,\ldots,i_n)} (d_{i_1,\ldots,i_n}) + x_1^{i_1} + \ldots + x_n^{i_n})$ is named as the conjunctive normal form of f.

14.9 SELF ASSESSMENT QUESTIONS:

1. Write the Boolean function values for $f : A^2 \rightarrow A$, where A = {0, 1} with

$f(x_1, x_2) = (x_1 \wedge \overline{x_1}) \vee x_2$.

Ans / Solution:

(x, y)	f
(0, 0)	0
(0, 1)	1
(1, 0)	0
(1, 1)	1

2. Consider the Boolean polynomial $f(x, y, z) = x \wedge (y \vee z^1)$. If B ={0, 1}, compute the truth table of the function $f : B_3 \rightarrow B$ defined by f.

[that is, If B = {0, 1}, compute the truth table of the function f: $B_3 \rightarrow B$ defined by f].

Ans:

x	y	z	$x \wedge (y \vee z^1)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

3. Rewrite (or simplify) the given Boolean polynomial to obtain the requested format.

(i). $(x \wedge y^1 \wedge z) \vee (x \wedge y \wedge z)$; two variables and one operation.

(ii). $(y \vee z) \vee x^1 \vee (w \wedge w^1)^1 \vee (y \wedge z^1)$; two variables and two operations.

Ans: (i). $x \wedge z$. (ii). $y \vee x^1$.

4 . Write the disjunctive and conjuctive normal form for

$f(x_1, x_2, x_3) = [\,[x_1 \wedge \overline{(x_2 \vee x_3)}]\,] \vee \{[(x_1 \wedge x_2) \vee \overline{x_3}] \wedge x_1\}$, by writing minterms and maxterms.

Ans / Hint / Solution: Minterms : $x_1 \wedge \overline{x_2} \wedge x_3$, $x_1 \wedge x_2 \wedge \overline{x_3}$, $x_1 \wedge x_2 \wedge x_3$

Disjunctive normal form : $(x_1 \wedge \overline{x_2} \wedge x_3) \vee (x_1 \wedge x_2 \wedge \overline{x_3}) \vee (x_1 \wedge x_2 \wedge x_3)$

Maxterms : $\overline{x_1} \vee \overline{x_2} \vee \overline{x_3}$, $\overline{x_1} \vee \overline{x_2} \vee x_3$, $\overline{x_1} \vee x_2 \vee \overline{x_3}$, $\overline{x_1} \vee x_2 \vee x_3$, $x_1 \vee \overline{x_2} \vee x_3$.

Conjunctive normal form :

$(x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee \overline{x_3}) \wedge (x_1 \vee \overline{x_2} \vee x_3) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3}) \wedge (\overline{x_1} \vee x_2 \vee \overline{x_3})$.

5. Find the c.n.f. for $p = x_1^1 x_2 + x_1 x_2^1$.

Ans / Hint / Solution: $(x_1^1 + x_2^1)(x_1 + x_2)$.

14.10 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Kedukodi Babushri Srinivas

LESSON - 15
FINITE BOOLEAN ALGEBRAS

OBJECTIVE:

- To know the Extended notion of lattice: Boolean algebra.
- To know the Important theorem namely representation theorem
- To Learn various properties of Boolean algebras

STRUCTURE:

15. 1. INTRODUCTION:

In 1854, George Boole (1815 - 1864) tried to find a mathematical model for human reasoning, and he introduced an important class of algebraic structure. In his honor this structure is called as 'Boolean algebra'. This Boolean algebra is a special type of lattice.

Boolean Algebra is an algebra of logic. One of the earliest investigators of symbolic logic was George-Boole who invented a systematic way of manipulating logic symbols which was referred as Boolean Algebra. It has become now an indispensable tool to computer scientists because of its direct applicability to switching circuit theory in physics, and the logical design of digital computers. The symbols 0 and 1 used in this unit have certian logical significance.

15.2. BOOLEAN ALGEBRAS AND PROPERTIES:

Now we recollect some important definitions and examples which are essential in the study of this Lesson.

15.2.1. Definition: A complemented distributive lattice is called a Boolean algebra (or a Boolean lattice).

15.2.2 Note: (i) By the Theorem 13.4.5. (of the lesson 13), we have that every element x of a complemented distributive lattice, has unique complement (which is denoted by x^1).

(ii) From (i), we can conclude that every element x of a Boolean algebra, has unique complement.

15.2.3 Notation: Henceforth, we use B to denote a Boolean algebra. So B denotes a set with the two binary operations $\wedge$ and $\vee$, with zero element 0 and a unit element 1, and the unary operation of complementation.

We write B = (B, $\wedge, \vee$, 0, 1) or B = (B, $\wedge, \vee$), in short.

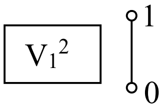
15.2.4 Examples: (i) Let M be a set and P(M) be the power set of M.

Then the system (P(M) , $\cap$, $\cup$, ϕ, M^1) is a Boolean Algebra. Here $\cap$ and $\cup$ are the set-theoretic operations: intersection and union, and the complement is the set-theoretic complement (that is, $M \setminus A = A^1$).

In this Boolean algebra, the elements ϕ and M are the "universal bounds".

If M contains exactly n elements, then P(M) contains exactly 2^n elements.

(ii) Let $\mathcal{B}$ be the lattice $V_1{}^2$, where the operations are defined by



$\wedge$	0	1
0	0	0
1	0	1

$\vee$	0	1
0	0	1
0	1	1

^{1}	
0	1
1	0

Then ($\mathcal{B}$, $\wedge$,$\vee$, 0, 1, ^{1}) is a Boolean algebra.

(iii) Consider the Boolean algebra $\mathcal{B}$ given in (ii). Let n be a positive integer.

Consider the set $\mathcal{B}^n$, the Cartesian product of n copies of $\mathcal{B}$.

The set $\mathcal{B}^n$ is a Boolean algebra with respect to the operations given below:

$(i_1,\ldots, i_n) \wedge (j_1,\ldots, j_n) := (i_1 \wedge j_1, \ldots, i_n \wedge j_n)$,

$(i_1 ,\ldots, i_n) \vee (j_1,\ldots, j_n) := (i_1 \vee j_1, \ldots, i_n \vee j_n)$, $(i_1,\ldots, i_n)^1 := (i_1{}^1, \ldots, i_n{}^1)$, and

$0 = (0, \ldots, 0)$, $1 = (1, \ldots, 1)$

(iv) Suppose $\mathcal{B}_1 , \mathcal{B}_2 , \ldots \mathcal{B}_n$ are Boolean algebras.

Consider $\mathcal{B} = \mathcal{B}_1 \times \mathcal{B}_2 \times \ldots \times \mathcal{B}_n$, the Cartesian product of the Boolean algebras

$\mathcal{B}_1 ,\mathcal{B}_2 , \ldots \mathcal{B}_n$.

We define the operations $\vee$, $\wedge$ and complementation in $\mathcal{B}$ as follows:

$(i_1,\ldots, i_n) \wedge (j_1,\ldots, j_n) := (i_1 \wedge j_1, \ldots, i_n \wedge j_n)$,

$(i_1 ,\ldots, i_n) \vee (j_1,\ldots, j_n) := (i_1 \vee j_1, \ldots, i_n \vee j_n)$,

$(i_1,\ldots, i_n)^1 := (i_1{}^1, \ldots, i_n{}^1)$, and $0 = (0, \ldots, 0), 1 = (1, \ldots, 1)$

where $i_k, j_k \in \mathcal{B}_k$ for $1 \leq k \leq n$.

It is easy to verify that $\mathcal{B} = \mathcal{B}_1 \times \mathcal{B}_2 \times_{\ldots} \times \mathcal{B}_n$ is a Boolean algebra.

So $\mathcal{B}$ is the direct product of the given Boolean algebras.

15.2.5 Theorem (De Morgan's Laws): For all x, y in a Boolean Algebra B, we have that $(x \wedge y)^1 = x^1 \vee y^1$ and $(x \vee y)^1 = x^1 \wedge y^1$.

Proof: We have $(x \wedge y) \vee (x^1 \vee y^1)$

$= (x \vee x^1 \vee y^1) \wedge (y \vee x^1 \vee y^1)$ (by distributive law)

$= (1 \vee y^1) \wedge (1 \vee x^1)$ (by complement laws)

$= 1 \wedge 1$ (by universal bound laws)

$= 1$ (by idempotent laws).

Also $(x \wedge y) \wedge (x^1 \vee y^1) = (x \wedge y \wedge x^1) \vee (x \wedge y \wedge y^1)$ (by distributive law)

$= (0 \wedge y) \vee (x \wedge 0)$ (by complement laws)

$= 0 \vee 0$ (by universal bound laws)

$= 0$ (by idempotent laws).

From the facts proved above, we can conclude that $x^1 \vee y^1$ is the complement of $x \wedge y$. The other part follows from the duality principle.

15.2.6 Corollary: In a Boolean algebra B, we have $x \leq y \Leftrightarrow x^1 \geq y^1$ for all $x, y \in B$.

Proof: We have $x \leq y \Leftrightarrow x \vee y = y$ (by the definition of $\leq$)

$\Leftrightarrow (x \vee y)^1 = y^1$ (by taking complement)

$\Leftrightarrow x^1 \wedge y^1 = y^1$ (by the Theorem 15.2.5)

$\Leftrightarrow x^1 \geq y^1$ (by the definition of $\leq$)

15.2.7 Theorem: In a Boolean algebra B, the following conditions are equivalent:

(i) $x \leq y$; (ii) $x \wedge y^1 = 0$; (iii) $x^1 \vee y = 1$; (iv) $x \wedge y = x$; and

(v) $x \vee y = y$ for all $x, y \in B$.

Proof: (i) $\Rightarrow$ (ii): $x \leq y \Rightarrow x = x \wedge y$

$\Rightarrow x \wedge y^1 = x \wedge y \wedge y^1 = x \wedge 0 = 0$.

(ii) $\Rightarrow$ (iii): $(x \wedge y^1) = 0$

$\Rightarrow (x \wedge y^1)^1 = 0^1 = 1 \Rightarrow x^1 \vee y = 1$

(iii) $\Rightarrow$ (iv): $(x^1 \vee y) = 1$

$\Rightarrow x \wedge 1 = x \wedge (x^1 \vee y)$

$\Rightarrow x = (x \wedge x^1) \vee (x \wedge y) = 0 \vee (x \wedge y) = x \wedge y.$

(iv) $\Rightarrow$ (v): $x \wedge y = x$

$\Rightarrow (x \wedge y) \vee y = x \vee y \Rightarrow y = x \vee y.$

(v) $\Rightarrow$ (i) follows from the definition of $\leq$.

15.2.8 Definition: Let B_1 and B_2 be Boolean algebras. Then a mapping $f : B_1 \to B_2$ is said to be a (Boolean) homomorphism from B_1 into B_2 if f is a (lattice) homomorphism and $f(x^1) = (f(x))^1$ for all $x \in B$.

Equivalently, we can define the Boolean homomorphism as follows:

Let B_1 and B_2 be Boolean algebras. A mapping $f : B_1 \to B_2$ is said to be a (Boolean) homomorphism from B_1 into B_2 if it satisfies the following three conditions:

(i) $f(x \wedge y) = f(x) \wedge f(y)$;

(ii) $f(x \vee y) = f(x) \vee f(y)$; and

(iii) $f(x^1) = (f(x))^1$, for all $x, y \in B_1$.

15.2.9 Definition: Let $f : B_1 \to B_2$ be a Boolean homomorphism.

(i) If f is one-one, then we say that f is a monomorphism.

(ii) If f is onto, then we say that f is an epimorphism.

(iii) If f is a bijection, then we say that f is an isomorphism.

If there is a Boolean isomorphism between B_1 and B_2, then we write $B_1 \cong_b B_2$.

15.2.10 Theorem: Let $f : B_1 \to B_2$ be a Boolean homomorphism. Then

(i) $f(0) = 0$, $f(1) = 1$;

(ii) for all $x, y \in B_1$, $x \leq y \Rightarrow f(x) \leq f(y)$; and

(iii) $f(B_1)$ is a Boolean subalgebra of B_2.

Proof: Let $a, b \in B_1$.

(i) Now $f(0) = f(a \wedge a^1)$ (by complement law)

$= f(a) \wedge f(a^1)$ (since f is a homormorphism)

$= f(a) \wedge (f(a))^1$ (since f is a homormorphism)

$= 0$ (by the complement laws).

Also $f(1) = f(a \vee a^1)$ (by the complement law)

$= f(a) \vee f(a^1)$ (since f is a homormorphism)

$= f(a) \vee ((f(a))^1$ (since f is a homormorphism)

$= 1$ (by complement law).

The proof is complete for (i).

(ii) Suppose $a \leq b$.

Then $a = a \wedge b$

$\Rightarrow f(a) = f(a \wedge b) = f(a) \wedge f(b)$ (since f is ahomormorphism)

$\Rightarrow f(a) \leq f(b)$ (by the definition of $\leq$).

(iii) To show that $f(B_1)$ is a Boolean subalgebra, it is enough to prove that $f(B_1)$ is closed under the operations $\vee$, $\wedge$, and ^{1}. Now let $f(a), f(b) \in f(B_1)$.

So $f(a) \vee f(b) = f(a \vee b) \in f(B_1)$ (since $a \vee b \in B_1$);

And $f(a) \wedge f(b) = f(a \wedge b) \in f(B_1)$ (since $a \wedge b \in B_1$);

and $[f(a)]^1 = f(a^1) \in f(B_1)$ (since $a^1 \in B_1$).

This shows that $f(B_1)$ is a Boolean subalgebra of B_2.

15.2.11 Example: (i) If $M \subset N$, then the mapping $f : P(M) \rightarrow P(N)$ defined by $f(A) = A$, is a lattice monomorphism, but not a Boolean homomorphism.

To verify this, let $A \in P(M)$.

Now $f(A^1) = f(M \setminus A) = M \setminus A \neq N \setminus A = N \setminus f(A) = (f(A)^1$.

This shows that f is not a Boolean homomorphism.

Also $f(1) = f(M) = M \neq N =$ the unit element in $P(N)$.

(ii) Suppose $M = \{1, \ldots, n\}$. Then $\{0, 1\}^n$ and $P(M)$ are Boolean algebras.

The mapping $f : \{0, 1\}^n \rightarrow P(M)$ defined by $f((i_1, \ldots, i_n)) = \{k \mid i_k = 1\}$, is a Boolean isomorphism. (verification is straight forward).

(iii) Let X be a set, and A a subset of X.

We know that the characteristic function of A is defined as:

$$\chi_A : X \rightarrow \{0, 1\}; \quad x \mapsto \begin{cases} 1 & \text{if } x \in A \\ 0 & \text{if } x \notin A \end{cases}.$$

Define $h : P(X) \rightarrow \{0, 1\}^x$ as follows: $h(A) = \chi_A$.

This mapping h is a Boolean isomorphism. So $P(X) \cong_b \{0, 1\}^x$.

15.2.12 Definitions: If $x \in [a, b] = \{v \in L \mid a \leq v \leq b\}$ and $y \in L$ with $x \wedge y = a$ and $x \vee y = b$, then y is called a relative complement of x with respect to [a, b]. If all intervals [a, b] in a lattice L are complemented, then L is called relatively complemented. If L has a zero element and all [0, b] are complemented, then L is called sectionally complemented.

15.2.13 Theorem: Let L be a lattice. Then we have the following:

(i) L is a Boolean algebra $\Rightarrow$ L is relatively complemented

(ii) L is relatively complemented $\Rightarrow$ L is sectionally complemented.

(iii) L is finite and sectionally complemented $\Rightarrow$ every non-zero element a of L is a join of finitely many atoms.

(iv) If B is a finite Boolean algebra, then every element x in B is equal to the union of atoms that are $\leq x$.

Proof: (i) Let L be a Boolean algebra and let $a \leq x \leq b$. Define $y := b \wedge (a \vee x^1)$.

Now we prove that y is a complement of x in [a, b].

We have $x \wedge y = x \wedge (b \wedge (a \vee x^1))$

$= x \wedge (a \vee x^1)$ (by modular law and $x \leq b$)

$= (x \wedge a) \vee (x \wedge x^1)$ (by distributive law)

$= x \wedge a$ (since $x \wedge x^1 = 0$)

$= x$ (since $x \leq a$) .

Also we have $x \vee y = x \vee (b \wedge (a \vee x^1))$

$= x \vee (b \wedge a) \vee (b \wedge x^1))$ (by distributive law)

$= x \vee a \vee (b \wedge x^1)$ (since $a \leq b$)

$= x \vee (b \wedge x^1)$ (since $a \leq x$)

$= (x \vee b) \wedge (x \vee x^1)$ (by distributive law)

$= b \wedge 1$ (since $x \leq b$, and by complement law)

$= b$ (by universal bound laws).

This shows that y is the complement of x in [a, b].

Thus L is relatively complemented.

(ii) Suppose L is relatively complemented. Then by definition [a, b] is complemented for all $a, b \in L$ such that $a \leq b$.

Since $0 \leq b$, we have [0, b] is complemented for all $b \in L$.

Therefore L is sectionally complemented.

(iii) Let $a \in L$. Suppose $\{ x \mid x$ is an atom and $x \leq a\} = \{p_1, \ldots, p_n\}$.

Write $b = p_1 \vee \ldots \vee p_n$.

Since each $p_i \leq a$, we have that $b \leq a$. We have to show that $b = a$.

In a contrary way, suppose $b \neq a$. Suppose c is the complement of b in $[0, a]$.

Now $c \neq 0$ (if $c = 0$, then $a = b \vee c = b \vee 0 = b$, a contradiction).

Since c is non-zero, there exists an atom p such that $p \leq c$.

Since $p \leq c \leq a$, we have that $p \in \{p_1, \ldots, p_n\}$.

Now $p = p_i \leq b$. So $p = p \wedge b \leq c \wedge b = 0$, a contradiction.

Hence $a = b$. Therefore we get that $a = p_1 \vee \ldots \vee p_n$.

(iv) Follows from the facts proved above.

15.3. REPRESENTATION THEOREM:

In this section we prove a representation theorem of Boolean algebra, namely Stone's Representation Theorem.

15.3.1 Theorem: (Representation Theorem) Let B be a finite Boolean algebra, and A denotes the set of all atoms in B. Then B is Boolean isomorphic to P(A).

That is, $(B, \wedge, \vee) \cong_b (P(A), \cap, \cup)$.

Proof: **Part**-(i): Let $v \in B$ be an element and write

$A(v) := \{a \in A \ / \ a \leq v\} := \{a \ / \ a$ is an atom and $a \leq v\}$.

Consider the mapping $h : B \rightarrow P(A)$ defined by $h(v) = A(v)$.

We show that h is a Boolean isomorphism.

Part-(ii): In this part, we show that h is a Boolean homomorphism.

Let a be an atom and for $v, w \in L$. we have $a \in A(v \wedge w)$

$\Leftrightarrow a \leq v \wedge w \Leftrightarrow a \leq v$ and $a \leq w$ (by definition of $\wedge$)

$\Leftrightarrow a \in A(v) \cap A(w)$.

This proves that $h(v \wedge w) = h(v) \cap h(w)$.

Also we have $a \in A(v \vee w) \Leftrightarrow a \leq v \vee w$

$\Leftrightarrow a \leq v$ or $a \leq w$ (by definition of $\vee$)

$\Leftrightarrow a \in A(v) \cup A(w)$.

This shows that $h(v \vee w) = h(v) \cup h(w)$.

Now $a \in A(v^1) \Leftrightarrow a \leq v^1 \Leftrightarrow a = a \wedge v^1$ (by the definition of $\leq$)

$\Leftrightarrow a \wedge v = a \wedge v^1 \wedge v \Leftrightarrow a \wedge v = a \wedge 0 = 0 \neq a$

$\Leftrightarrow a \not\leq v$ (by the definition of $\leq$) $\Leftrightarrow a \notin A(v) \Leftrightarrow a \in A \setminus A(v)$.

This shows that $h(v^1) = (h(v))^1$. Hence h is a Boolean isomorphism.

Part-(iii): Note that $h(0) = \phi$ (by the definition of h).

By the Theorem 15.2.13 (iv), we have that every $v \in B$ can be expressed as the join of finitely many atoms: $v = a_1 \vee \ldots \vee a_n$ with $a_1 \leq v$, for all i.

Now we show that h is one-one.

For this, suppose $h(v) = h(w)$. That is, $A(v) = A(w)$.

Then $a_i \in A(v) \Rightarrow a_i \in A(w) \Rightarrow a_i \leq w$. This is true for all i.

Therefore $v = a_1 \vee \ldots \vee a_n \leq w \Rightarrow v \leq w$.

In the same way, we can show that $w \leq v$.

Hence $v = w$. This shows that h is one-one.

Part-(iv): Now we show that h is onto. For this, take $C \in P(A)$.

Suppose $C = \{c_1, \ldots, c_n\}$, and write $v = c_1 \vee \ldots \vee c_n$.

Now $A(v) \supseteq C$, and so $h(v) \supseteq C$.

Conversely, $a \in h(v)$, then a is an atom with $a \leq v = c_1 \vee \ldots \vee c_n$ and so $a \leq c_i$, for some $i \in \{1, \ldots, n\}$. Since a and c_i are atoms, we have that $a = c_i \in C$.

Therefore $h(v) \subseteq C$.

Hence $h(v) = C$. This shows that h is onto. The proof is complete..

15.3.2 Theorem: (i) The cardinality of a finite Boolean algebra B is always of the form 2^n, where n is the number of atoms in B. Also $B \cong_b P(\{1, \ldots, n\})$.

(ii) If B_1, B_2 are two finite Boolean algebras such that the number of atoms in each is n, then we have that $B_1 \cong_b P(\{1, \ldots, n\}) \cong_b B_2$, and so $B_1 \cong_b B_2$.

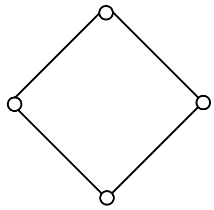
(iii) By the observation made in the Example 15.2.11 (ii), we have that for every finite Boolean algebra $B \neq \{0\}$, there is some $n \in \mathbb{N}$ with $B \cong_b \{0, 1\}^n$.

15.3.3 Note: The lattice of the divisors of 30, that is, the Boolean algebra $B = (\{1, 2, 3, 5, 6, 10, 1, 30\}$, gcd, lcm, 1, 30, complement with respect to 30), has $8 = 2^3$ elements. So it is isomorphic to the eight element Boolean algebra $P(\{a, b, c\})$ (that is, the power set of three elements) for some three distinct elements a, b, c.

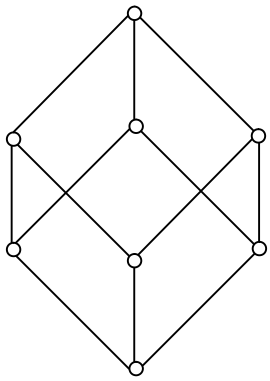
15.3.4 Note: In the following, we present all non-isomorphic Boolean algebras of order $\leq$ 8 (= 2^3).

$\mathcal{B}^0 = V_1{}^1$

$\mathcal{B}^1 = V_1{}^2$



$\mathcal{B}^2 = V_1{}^4$



$\mathcal{B}^3$

15.3.5 Remark: (i) Every finite Boolean algebra is isomorphic to the Boolean algebra P(A) where A is set of all atoms of B.

(ii) In case of infinite Boolean algebras we have the following result: "if B is an infinite Boolean algebra, then there is a set M and a Boolean monomorphism (called “Boolean embedding”) from B to P(M)". This result is known as Stone’s Representation Theorem.

15.3.6 Definition: Let B be a Boolean algebra and let X be a set. For any two mappings f and g from X into B, we define the functions

$f \wedge g$, $f \vee g$, f^1, f_0, f_1 from X into B, as follows:

$f \wedge g : X \to B$ by $(f \wedge g)(x) = f(x) \wedge g(x)$;

$f \vee g : X \to B$ by $(f \vee g)(x) = f(x) \vee g(x)$;

$f^1 : X \to B$ by $(f^1)(x) = (f(x))^1$;

$f_0 : X \to B$ by $f_0(x) = 0$; and $f_1 : X \to B$ by $f_1(x) = 1$ for all $x \in X$.

15.3.7 Result: Let B be a Boolean algebra and let X be a set. For any two mappings f and g from X into B, we defined the functions $f \wedge g$, $f \vee g$, f^1, f_0, f_1 from X into B, in the definition 15.3.6.

(i) Write B^X = the set of all mappings from X into B.

Then $(B^X, \wedge, \vee, f_0, f_1, {}^1)$ is a Boolean algebra.

(ii) If X = B^n (the Cartesian product of n copies of B), then we write

$F_n(B) := B^{B^n}$ (the set of all functions from B^n to B).

Now it is clear that $F_n(B)$ is Boolean algebra.

15.3.8 Theorem: If $S_1 = \{x_1, x_2, \ldots .x_n\}$ and $S_2 = \{y_1, y_2, \ldots y_n\}$ are any two finite sets with n elements, then the lattices $(P(S_1), \subseteq)$ and $(P(S_2), \subseteq)$ are isomorphic.

Consequently, the Hasse diagrams of these lattices may be drawn identically.

Proof: Arrange the sets as known in Fig. 1,

so that each element of S_1 is directly over the correspondingly numbered element in S_2

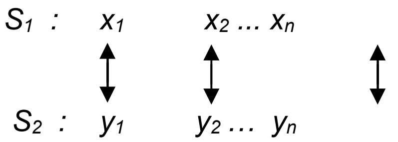
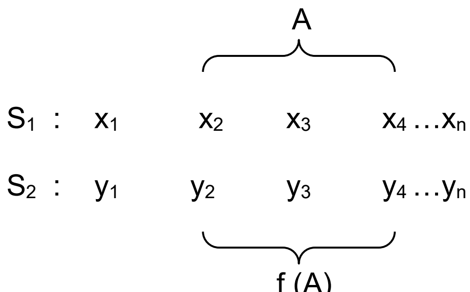


Fig. 1

Let A be a subset of S_1

Define f (A) = subset of S_2 consisting of all elements that correspond to the elements of A .



It can be easily seen that f is one-one and onto.

Also $A \subseteq B$ if and only if $f(A) \subseteq f(B)$ for all $A, B \in P(S_1)$.

Therefore the lattices $(P(S_1), \subseteq)$ and $(P(S_2), \subseteq)$ are isomorphic.

15.4 SUMMARY:

This unit provided the fundamental idea of the algebraic system namely Boolean algebra with two binary operations (join and meet) and a unary operation (complementation). Several properties of the Boolean algebras were discussed. The reader able to know the applications of Boolean algebra in various branches like computer science, electrical engineering (switching networks), and so on. Particularly, devices such as mechanical switches, diodes, magnetic dipoles, and transistors are two state devices. In case of two state devises, the Boolean logic can be applied. The important example of finite Boolean algebra is the power set of A for any finite set A. Few examples and fundamental results related to Boolean algebra were included for better understanding of the reader.

15.5 TECHNICAL TERMS:

Boolean algebra:
A complemented distributive lattice is called as Boolean algebra.

Finite Boolean Algebra:
A Boolean Algebra with finite number of elements is called a finite Boolean Algebra.

De Morgan's Laws:
For all x, y in a Boolean Algebra B, we have that $(x \wedge y)^1 = x^1 \vee y^1$ and $(x \vee y)^1 = x^1 \wedge y^1$. These two laws are called as De Morgan Laws.

Boolean homomorphism

Let B_1 and B_2 be Boolean algebras. A mapping $f : B_1 \to B_2$ is said to be a (Boolean) homomorphism from B_1 into B_2 if it satisfies the following three conditions:
(i) $f(x \wedge y) = f(x) \wedge f(y)$; (ii) $f(x \vee y) = f(x) \vee f(y)$; and
(iii) $f(x^1) = (f(x))^1$, for all $x, y \in B_1$.

Representation Theorem
Let B be a finite Boolean algebra, and A denotes the set of all atoms in B. Then B is Boolean isomorphic to P(A), the power set of A.

15.6 SELF ASSESSMENT QUESTIONS:

1. Find: Whether or not the lattice $D_{20} = \{1, 2, 4, 5, 10, 20\}$ is a Boolean algebra ?

Ans: The number of elements in the given set is six. We know that in any Boolean algebra the number of elements is of the form 2^n. Here $6 \neq 2^n$ (for any positive integer n), and hence the given set can not be a Boolean algebra.

2. Whether or not the lattice $D_{30} = \{1, 2, 3, 5, 6, 10, 15, 30\}$ a Boolean algebra ?
Ans: The given set has 2^3 elements. This set is Boolean isomorphic to the power set of a set of three elements. and hence it is a Boolean algebra.

3. Let S = {a, b, c}, T = {2, 3, 5}. Show that the Boolean lattices $(P(S), \subseteq)$ and $(P(T), \subseteq)$ are isomorphic.
Ans: Define $f : P(S) \to P(T)$ by $f(\{a\}) = \{2\}$, $f(\{b\}) = \{3\}$, $f(\{c\}) = \{5\}$, $f(\{a, b\}) = \{2,3\}$, $f(\{b, c\}) = \{3, 5\}$, $f(\{a, c\}) = \{2, 5\}$, $f(\{a, b, c\}) = \{2, 3, 5\}$, $f(\phi) = \phi$. Then f is an isomorphism.

4. Show that ({1, 2, 3, 6, 9, 18}, gcd, lcm) does not form a Boolean algebra for the set of positive divisors of 18.
Ans: (Similar answer as in above 1). The number of elements in the given set is six. We know that in any Boolean algebra the number of elements is of the form 2^n. Here $6 \neq 2^n$ (for any positive integer n), and hence the given set can not be a Boolean algebra.

5. Define the system Boolean algebra and give two examples.
Ans: (Refer: Definition 15.2.1, and Examples 15.2.4.)

6. State representation theorem and prove it.
Ans: (Refer: Theorem 15.3.1.)

7. Give all the non-isomorphic Boolean algebras of order ≤ 8 $(= 2^3)$.

Ans: (Refer Note 15.3.4)

8. If L is a lattice, then prove the following:

(i) L is a Boolean algebra $\Rightarrow$ L is relatively complemented

(ii) L is relatively complemented $\Rightarrow$ L is sectionally complemented.

(iii) L is finite and sectionally complemented $\Rightarrow$ every non-zero element a of L is a join of finitely many atoms.

(iv) If B is a finite Boolean algebra, then every element x in B is equal to the union of atoms that are $\leq x$.

Ans: (Refer: Theorem 15.2.13.)

9. State and Prove Demorgon Laws.
Ans: (Refer: Theorem 15.2.5.)

15.7 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr.Kedukodi Babushri Srinivas

LESSON-16

IDEALS, FILTERS AND SOLUTIONS OF BOOLEAN EXPRESSIONS

OBJECTIVE:

- To define the substructure ideal in Boolean algebra
- To understand Maximal ideal, filters in Boolean algebra
- To Learn to find the solutions of Boolean equations.
- To have proper understanding of the substructures of Boolean algebras.
- To develop skills in solving problems.

STRUCTURE:

16.1. INTRODUCTION:

In this Lesson, we define the substructures ideals, and filters in Boolean algebras. In the later sections we provide a procedure to find the solutions of Boolean equations.

16.2. IDEALS:

We define the substructure namely ideal and maximal ideal of a Boolean algebra.

16.2.1. Notation : Let B be a Boolean algebra and $b, c \in B$. Then we write $b + c$ instead of $b \vee c$, and bc instead of $b \wedge c$.

16.2.2. Definition: Let B be a Boolean algebra, and $I \subseteq B$. The subset I is said to be an ideal in B (in symbols, we write $I \leq B$) if I is non-empty and if

$$ib \in I \text{ and } i + j \in I \text{ for all } i, j \in I \text{ and } b \in B.$$

16.2.3 Result: Let B be a Boolean algebra and I is an ideal in B. If we take $b = i^1$, then we get that $0 = i^1 \wedge i = b \wedge i \in I$. Thus $0 \in I$ for every ideal I of a Boolean algebra B.

16.2.4 Example: (i) Let B be a Boolean algebra. The subsets
$\{0\}$ and B of the set B are ideals of the Boolean algebra B. These two ideals are called trivial ideals. All the other ideals of B, if exist, are called proper ideals of B.

(ii) Let A be a non-empty set. Write B = $\wp(A)$, the power set of A. Then B together with the operations $\wedge$ (where $\wedge$ is the set theoretic intersection), $\vee$ (where $\vee$ is the set theoretic union) is a Boolean algebra.
Write I = {X / X is a finite subset of A}.
It is clear that I is a non-empty subset of B.

Let X, Y $\in$ I, Z $\in$ B $\Rightarrow$ X, Y are finite subsets of A.

$\Rightarrow$ $X \vee Y$, $X \wedge Z$ are finite subsets of A

$\Rightarrow$ X + Y, X.Z are finite subsets of A

$\Rightarrow$ X + Y, X.Z $\in$ I.

This shows that I is an ideal of B.

16.2.5 Definition: Let B_1, B_2 be Boolean algebras and $h : B_1 \rightarrow B_2$ a Boolean homomorphism. Then the set $\{b \in B_1 \mid h(b) = 0\}$ is called the kernal of h, and the set is denoted by Ker h.

16.2.6 Note: Let $h : B_1 \rightarrow B_2$ be a Boolean homomorphism. Then Ker h is an ideal of B_1.
[Verification: Let x, y $\in$ Ker h, b $\in$ B $\Rightarrow$ h(x) = 0 and h(y) = 0

$\Rightarrow$ $h(x \vee y) = h(x) \vee h(y) = 0 \vee 0 = 0$ and $h(x \wedge b) = h(x) \wedge h(b) = 0 \wedge h(b) = 0$

$\Rightarrow$ $x \vee y$, $x \wedge b \in$ Ker h. This shows that Ker h is an ideal of B_1].

16.2.7 Theorem: Let B be a Boolean algebra and I a non-empty subset of B.
Then the following conditions are equivalent:

(i) $I \leq B$ (That is, I is an ideal of B);

(ii) If $i, j \in I$ and $b \in B$ such that $b \leq i$, then $i + j \in I$ and $b \in I$.

(iii) There exists a Boolean algebra B_1 and a Boolean homomorphism $h : B \rightarrow B_1$ such that I = Ker h.

Proof: (i) $\Rightarrow$ (ii): Let $i, j \in I$ and $b \in B$ such that $b \leq i$.

Since I is an ideal of B, we have that $i + j \in I$.

Since $b \leq i$, and I is an ideal, we have that $b = b \wedge i = bi \in I$.

(ii) $\Rightarrow$ (iii): Let I satisfies the condition (ii). Now define a relation $\sim$ on B by

$$b_1 \sim b_2 \Leftrightarrow b_1 + b_2 \in I.$$

Then $\sim$ is an equivalence relation.

The equivalence class containing $b \in B$ is denoted by [b].

Consider the set B/$\sim$ of all equivalence classes.

We define the operations +, . and "." on B/$\sim$ as follows:

$$[b_1] + [b_2] = [b_1 + b_2], \text{ and } [b_1]\,[b_2] = [b_1 b_2].$$

Now we verify that these two operations on B/~ are well defined.

For this, suppose $[b_1] = [c_1]$ and $[b_2] = [c_2]$. This implies

$b_1 \sim c_1$ and $b_2 \sim c_2 \Rightarrow b_1 + c_1 \in I$ and $b_2 + c_2 \in I$

$\Rightarrow b_1 + b_2 + c_1 + c_2 \in I. \quad \Rightarrow b_1 + b_2 \sim c_1 + c_2 \quad \Rightarrow [b_1 + b_2] = [c_1 + c_2]$.

This shows that the operation + on B/~ is well defined.

Now $[b_1] = [c_1]$ and $[b_2] = [c_2]$. This implies

$b_1 \sim c_1$ and $b_2 \sim c_2 \Rightarrow b_1 + c_1 \in I$ and $b_2 + c_2 \in I$

$\Rightarrow b_1 + b_2 + c_1 + c_2 \in I$.

$\Rightarrow b_1b_2 + c_1c_2 \in I$ (by (ii) and since $b_1b_2 + c_1c_2 \leq b_1 + b_2 + c_1 + c_2$)

$\Rightarrow [b_1b_2] = [c_1c_2]$.

This shows that the operation product on B/~ is well defined.
Now it is easy to verify that (B/~ + , .) is a Boolean algebra with zero [0] and unit [1].
Define a mapping $h : B \to B/\sim$, by $h(b) = [b]$.

Let $a, b_1, b_2 \in B$. Now $h(b_1 + b_2) = [b_1 + b_2] = [b_1] + [b_2] = h(b_1) + h(b_2)$, and

$h(b_1 . b_2) = [b_1 . b_2] = [b_1] . [b_2] = h(b_1) . h(b_2)$.

We know that $a + a^1 = 1 \Rightarrow [a] + [a^1] = [1] \Rightarrow [a]^1 = [a^1] \Rightarrow (h(a))^1 = h(a^1)$.

This shows that h is a Boolean homomorphism.

Now we show that Ker h = I.

Let $b \in \text{Ker } h \Leftrightarrow h(b) = [0] \Leftrightarrow [b] = [0] \Leftrightarrow b - 0 \in I \Leftrightarrow b \in I$.

This shows that Ker h = I.

(iii) ⇒ (i): Proof follows from the Note 16.2.6.

16.2.8 Definition: Let B be a Boolean algebra and I an ideal of B.
Now define a relation ~ on B by $b_1 \sim b_2 \Leftrightarrow b_1 + b_2 \in I$.
This is an equivalence relation.
The equivalence class containing $b \in B$ is denoted by [b].

Consider the set B/ ~ of all equivalence classes.

We define the operations +, . and "." on B/ ~ as follows:

$[b_1] + [b_2] = [b_1 + b_2]$, and $[b_1] [b_2] = [b_1b_2]$.

In the proof of the Theorem 16.2.7, we verified that these two operations on B/ ~ are well defined, and (B/~ + , .) is a Boolean algebra with zero [0] and unit [1].

This Boolean algebra (B/~, + , .) is called the Boolean factor algebra and it is denoted by B/I.

16.2.9 Example: (i) Let M be a set, and $N \subseteq M$.

We know that $\wp(N)$ and $\wp(M)$ are Boolean algebras.

Since $N \subseteq M$, we have that $\wp(N) \subseteq \wp(M)$, and so $\wp(N)$ is an ideal of $\wp(M)$.

(ii) For $[p] \in P_n/\sim$, we define $([p]) = \{ [p][q] \ / \ q \in P_n\}$.

Now $([p])$ is an ideal of $P_n/\sim$.

16.2.10 Definition: Let B be a Boolean algebra.
(i) Suppose $b \in B$. The set $\{ bc \mid c \in B\}$ is denoted by (b).
It is easy to verify that the set (b) is an ideal of B.
This ideal (b) is called a principal ideal.

(ii) Let M be an ideal of B such that $M \neq B$. The ideal M is said to be a maximal ideal of B if it satisfies the following condition:

I is an ideal of B, and $M \subseteq I \subseteq B \Rightarrow M = I$ or $I = B$.

16.2.11 Theorem: Let B be a Boolean algebra and $b \in B$.
Then $(b) = \{a \in B \ / \ a \leq b\}$.

Proof: We know that $(b) = \{ bc \ / \ c \in B \}$. Write $X = \{a \in B \ / \ a \leq b\}$.

Now we have to show that $(b) = X$.

Let $x \in (b) \Rightarrow x = bc$ for some $c \in B \Rightarrow x = b \wedge c \leq b \Rightarrow x \in X$.

Converse: Let $y \in X \Rightarrow y \leq b \Rightarrow y = b \wedge y = by \in (b)$.
This shows that $(b) = X$.

16.2.12 Theorem: Let M be an ideal of a Boolean algebra B, then the following two conditions are equivalent:

(i) M is a maximal ideal.

(ii) $b \in B \Rightarrow b \in M$ or $b^1 \in M$, but not both.

Proof: (i) $\Rightarrow$ (ii): **Part**-(i): In a contrary way, suppose that there exists $b \in B$ such that $b \notin M$ and $b^1 \notin M$. Write $J = \{x + m \ / \ m \in M, b \in B \text{ and } x \leq b\}$.

Now $M \subseteq J$ and $b \in J \setminus M$. So M is a proper subset of J.

Part-(ii): Now we verify that J is an ideal of B.

Let $x_1 + m_1, x_2 + m_2 \in J$ and $c \in B$ with $x_1 \leq b$, $x_2 \leq b$, $m_1 \in M$, $m_2 \in M$.

Now $x_1 \leq b$, $x_2 \leq b \Rightarrow x_1 + x_2 = x_1 \vee x_2 \leq b$.

Also $m_1, m_2 \in M$ and M is an ideal $\Rightarrow m_1 + m_2 \in M$.

Therefore $(x_1 + m_1) + (x_2 + m_2) = (x_1 + x_2) + (m_1 + m_2) \in J$.

Consider $(x_1 + m_1)c = x_1c + m_1c \in J$

[because $x_1c = x_1 \wedge c \leq x_1 \leq b$ and $m_1c \in M$ (since M is an ideal)].

This shows that J is an ideal containing $M \cup \{b\}$.

By the definition of J, we can conclude that J is the ideal generated by $M \cup \{b\}$.

Also J contains M properly.

Part-(iii): Now $M \subseteq J \subseteq B$ and M is a maximal filter $\Rightarrow$ $M = J$ or $J = B$

$\Rightarrow$ $J = B$ (Since $M \neq J$).

Now $b^1 \in B = j$

$\Rightarrow b^1 = x + m$ for some $x \leq b$ and $m \in M$(by the definition of J)

$\Rightarrow b^1(x + m) = b^1b^1 \Rightarrow b^1 \wedge x + b^1m = b^1$

$\Rightarrow 0 + b^1m = b^1$ (since $b^1 \wedge x \leq b^1 \wedge b = 0$)

$\Rightarrow b^1m = b^1 \Rightarrow b^1 = b^1m = mb^1 \in M$ (since M is an ideal)

$\Rightarrow b^1 \in M$, a contradiction.

Therefore either $b \in M$ or $b^1 \in M$.

Part-(iv): Suppose $b, b^1 \in M$

$\Rightarrow b + b^1 \in M \Rightarrow 1 = b + b^1 \in M$

$\Rightarrow 1.c \in M$ for all $c \in B$ (since M is an ideal) $\Rightarrow c \in M$ for all $c \in B$

$\Rightarrow B = M$, a contradiction (to the fact that M is a proper filter).

(ii) $\Rightarrow$ (i): We have to show that M is a maximal ideal.

Let I be an ideal of B such that $M \subseteq I \subseteq B$. Suppose that $I \neq B$.

Now we have to show that $M = I$. In a contrary way, suppose that $M \neq I$. Then there exists $x \in I$ and $x \notin M$

$\Rightarrow$ $x \in I$ and $x^1 \in M$ (since $x \in M$ or $x^1 \in M$)

$\Rightarrow$ $x, x^1 \in I$ (since $M \subseteq I$) $\Rightarrow$ $1 = x + x^1 \in I$

$\Rightarrow$ $I = B$, a contradiction. Hence either $M = I$ or $I = B$.

The proof is complete.

16.3 FILTERS:

16.3.1 Definition: Let B be a Boolean algebra and $\phi \neq F \subseteq B$. Then F is said to be a filter (or dual ideal) if it satisfies the following two conditions:

(i) $x, y \in F \Rightarrow xy \in F$., and (ii) $a \in F,\ b \in B \Rightarrow a + b \in F$.

16.3.2 Examples: (i) Let B be a Boolean algebra. The two subsets $\{1\}$ and B are filters. These two filters are called trivial filters. The other filters, exist if any, are called as proper filters.

(ii) Let X be an infinite set. We know that $\wp(X)$ is a Boolean algebra. Write $F = \{A / A \subseteq X, \text{ and } X \setminus A \text{ is finite}\}$. Now we show that F is a filter.

Let $A, B \in F$; and $C \in B.A \in F$, and $C \in B$

$\Rightarrow X \setminus A$ is a finite set, and $C \in B$. $\Rightarrow X \setminus (A \cup C) \subseteq X \setminus A$ is a finite set

$\Rightarrow A \cup C \in F \Rightarrow A + C \in F$.

Let $A, B \in F \Rightarrow X \setminus A$ and $X \setminus B$ are finite

$\Rightarrow X \setminus (A \cap B) = (X \setminus A) \cup (X \setminus B)$ is finite. $\Rightarrow A \cap B \in F$.

This shows that F is a filter.

(iii) Let Y be a non-empty set, and $B = \wp(Y)$.

We know that B is a Boolean algebra.

Let $A \subseteq Y$. Write $F = \{C / A \subseteq C \subseteq Y\}$. Now we verify that F is a filter in B.

Let $U, V \in F$ and $Z \in B$

$\Rightarrow A \subseteq U \subseteq Y$ and $A \subseteq V \subseteq Y$

$\Rightarrow A \leq (U + Z) \leq Y$ and $A \leq (U \cap V) \subseteq Y \Rightarrow (U + Z),\ U.V \in F$.

This shows that F is a filter.

16.3.3. Theorem: Let B be a Boolean algebra and $\phi \neq I \subsetneq B$.
The following two conditions are equivalent:

(i) I is an ideal; and (ii) $F = \{x^1 / x \in I\}$ is a filter.

Proof: (i) $\Rightarrow$ (ii): Let $a, b \in F$ and $c \in B$

$\Rightarrow a^1, b^1 \in I$ and $c^1 \in B \Rightarrow a^1 + b^1 \in I$ and $a^1c^1 \in I$

$\Rightarrow (a^1 + b^1)^1 \in F$ and $(a^1c^1)^1 \in F \Rightarrow a^{11}.b^{11} \in F$ and $a^{11} + c^{11} \in F$

$\Rightarrow a.b \in F$ and $a + c \in F$. Therefore F is a filer.

(ii) $\Rightarrow$ (i): Let $x, y \in I$ and $c \in B$.

$\Rightarrow x^1, y^1 \in F$ and $c^1 \in B \Rightarrow x^1.y^1 \in F$ and $x^1 + c^1 \in F$

$\Rightarrow$ $(x^1.y^1)^1 \in I$ and $(x^1 + c^1)^1 \in I$ $\Rightarrow$ $x^{11} + y^{11} \in I$ and $x^{11} \wedge c^{11} \in I$

$\Rightarrow$ $x + y \in I$ and $xc \in I$. Therefore I is an ideal.

16.3.4 Theorem: Let B be a Boolean algebra and $F \subseteq B$. Then the following three conditions are equivalent:

(i) F is a filter in B.

(ii) There is a Boolean algebra B_1 and a Boolean homomorphism $f : B \to B_1$ such that $F = \{b \in B / f(b) = 1\}$

(iii) $a, b \in F$, $x \in B$, and $a \leq x$ $\Rightarrow$ $ab \in F$ and $x \in F$.

Proof: (i) $\Rightarrow$ (ii): Let F be a filter in B. Then by the Theorem 16.3.3, the set

$I = \{a^1 / a \in F\}$ is an ideal of B.

Hence by the Theorem 16.2.7, there exists a Boolean algebra B_1 and a Boolean homomorphism $f : B \to B_1$ such that $I = \text{Ker } f$. Now let $b \in B$.

Then $b \in F \Leftrightarrow b^1 \in I = \text{Ker } f \Leftrightarrow f(b^1) = 0 \Leftrightarrow (f(b))^1 = 0 \Leftrightarrow f(b) = 1$.

This shows that $F = \{b \in B \mid f(b) = 1\}$

(ii) $\Rightarrow$ (iii): Suppose there is a Boolean homomorphism f of B into another Boolean algebra B^1 such that$F = \{b \in B \mid f(b) = 1\}$. Let $a, b \in F$.

Then $f(a) = 1 = f(b)$

$\Rightarrow$ $f(a \wedge b) = f(a) \wedge f(b) = 1 \wedge 1 = 1$ $\Rightarrow$ $a \wedge b \in F$.

Suppose $a \in F$, $x \in B$ and $a \leq x$.

$\Rightarrow$ $f(a) = 1$, $x \in B$, $x = a \vee x$

$\Rightarrow$ $f(x) = f(a \vee x) = f(a) \vee f(x) = 1 \vee f(x) = 1$ $\Rightarrow$ $x \in F$. Hence we have (iii).

(iii) $\Rightarrow$ (i): Suppose the condition (iii). Let $a, b \in F$ and $x \in B$

$\Rightarrow$ $ab \in F$ (by (iii)), $a + x \in F$, and $a \leq a \vee x = a + x$

$\Rightarrow$ $ab \in F$, and $a + x \in F$ (by (iii)). The poof is complete.

16.3.5 Note: Let B be a Boolean algebra.
(i) Let I be a proper ideal of B and write
$\xi = \{J \mid J \text{ is an ideal of } B \text{ such that } I \subseteq J \neq B\}$.
By Zorn's Lemma, ξ contains a maximal element and this maximal element is a maximal ideal.

(ii) Let F be a proper filter of B and write

$\Omega = \{ J \mid J \text{ is a filter of } B \text{ such that } F \subseteq J \neq B\}$.

By Zorn's Lemma, Ω contains a maximal element M.

This M satisfies the following property:

K be a filer, $M \subseteq K \subseteq B \Rightarrow M = K$ or $K = B$.

16.3.6 . Definition: Let F be a proper filter in a Boolean algebra B. We say that F is a maximal filter (or ultra filter) if it satisfies the following condition:
K is a filter, $F \subseteq K \subseteq B \Rightarrow F = K$ or $K = B$.

16.3.7. Theorem: Let M be a Filter of a Boolean algebra B, then the following two conditions are equivalent:

(i) M is a maximal (ultra) filter.

(ii) $b \in B \Rightarrow b \in M$ or $b^1 \in M$, but not both.

Proof: (Similar to the proof of Theorem16.2.12.)

16.3.8. Problem: Let X be a set and write $B = \wp(X)$.
We know that $B = \wp(X)$ is a Boolean algebra. Let $x \in X$. Then

(i) $F_x = \{A \in \wp(X) \mid x \in A\}$ is a filter in $\wp(X)$.

(ii) F_x is an ultra filter.

Solution: (i) Let $U, V \in F_x$, and $C \in B$.$\Rightarrow x \in U, x \in V$ and $C \in B$

$\Rightarrow x \in U \cap V = U.V$ and $x = x + 0 \in U + C$

$\Rightarrow U. V \in F_x$ and $U + C \in F_x$. This shows that F_x is a filter.

(ii) Let J be a filter such that $F_x \subseteq J \subseteq B$.Suppose $F_x \neq J$.

Then there exists $Y \in J$ such that $Y \notin F_x \Rightarrow x \notin Y \in J$.

Now $\{x\} \in F_x \subseteq J$ and $x \notin Y \in J$

$\Rightarrow \phi = \{x\} \cap Y \in J$ (since J is a filter)

Let $Z \in B \Rightarrow \phi \subseteq Z \Rightarrow \phi \leq Z$.

Now $\phi \in J$ and $\phi \leq Z$ and J is a filter $\Rightarrow Z \in J$ (by the Theorem 16.3.4).

Now we proved that $Z \in B \Rightarrow Z \in J$. So $B \subseteq J$. Hence $J = B$.

This shows that F_x is a maximal (ultra) filter.

16.3.9. Definition: Let X be a set and write $B = \wp(X)$. We know that B is a Boolean algebra. Let $x \in X$. We know that $F_x = \{A \in \wp(X) \ / \ x \in A\}$ is an ultra filer. These filters F_x, $x \in X$ are called fixed ultra filters.

16.4 SOLUTIONS OF BOOLEAN EQUATIONS:

16.4.1 Definition: Let p and q be Boolean polynomial in P_n.

(i) Then the pair (p, q) is called an equation.

(ii) An element $(a_1, a_2, \ldots, a_n) \in B^n$ is called a solution for the equation (p, q) if $\overline{p_B}(a_1, a_2, \ldots, a_n) = \overline{q_B}(a_1, a_2, \ldots, a_n)$.

(iii) If $(a_1, a_2, \ldots, a_n)$ is a solution for the equations (p_i, q_i) for all $i \in I$, then we say that $(a_1, a_2, \ldots, a_n)$ is a common solution of all equations (p_i, q_i).
In this case, we also say that $(a_1, a_2, \ldots, a_n)$ is a solution of the system $\{(p_i, q_i) \mid i \in I\}$.

16.4.2 Notation: (i) Sometimes we write $p = q$ instead of (p, q).

(ii) Suppose $p = x_1{}^1x_2 + x_3$ and $q = x_1(x_2 + x_3)$.
If $(x_1, x_2, x_3) = (1, 0, 1)$, then $p = 1$ and $q = 1$.
Therefore $(1, 0, 1)$ is a solution for (p, q).
In this case, we say that $(1, 0, 1)$ is a solution of the equation $p = q$.

(iii) Suppose $p = x + x^1$ and $q = 0$. Then $p = q$ have no solution.

16.4.3 Theorem: The equations $p = q$ and $pq^1 + p^1q = 0$ have the same solutions for any two Boolean polynomials p and q in P_n.

Proof: Let B be a Boolean algebra and $(a_1, a_2, \ldots, a_n) \in B^n$.

Let $p, q \in P_n$. Write $a = \overline{p_B}(a_1, a_2, \ldots, a_n)$ and $b = \overline{q_B}(a_1, a_2, \ldots, a_n)$.

Now $(a_1, a_2, \ldots, a_n)$ is a solution for $p = q$.

$\Leftrightarrow \overline{p_B}(a_1, a_2, \ldots, a_n) = \overline{q_B}(a_1, a_2, \ldots, a_n) \Leftrightarrow a = b$

$\Leftrightarrow 0 = a \wedge a^1$

$= (a \vee a) \wedge (a^1 \vee a^1) = (a + a)(a^1 + a^1)$

$= (a + b)(a^1 + b^1)$ (since $a = b$)

$= aa^1 + ab^1 + ba^1 + bb^1 = 0 + ab^1 + ba^1 + 0$

$= ab^1 + ba^1 = ab^1 + a^1b$

$= \overline{p_B}(a_1, \ldots, a_n) \,.\, \overline{q_B}^1 (a_1, \ldots, a_n) + \overline{p_B}^1 (a_1, \ldots, a_n) \,.\, \overline{q_B}(a_1, \ldots, a_n)$

$= (\overline{p_B}\,\overline{q_B}^1 + \overline{p_B}^1\,\overline{q_B})(a_1, \ldots, a_n) = \overline{(pq^1 + p^1q)_B}(a_1, a_2, \ldots, a_n)$

$\Leftrightarrow$ $0\,(a_1, \ldots, a_n) = \overline{(pq^1 + p^1q)_B}(a_1, a_2, \ldots, a_n)$

$\Leftrightarrow$ $(a_1, a_2, \ldots, a_n)$ is a solution for the equation $0 = pq^1 + p^1q$.

Thus we proved that $(a_1, a_2, \ldots, a_n) \in B^n$ is a solution for $p = q$

$\Leftrightarrow$ $(a_1, a_2, \ldots, a_n)$ is a solution for $pq^1 + p^1q = 0$. The proof is complete.

16.4.4 Theorem: Let B be a Boolean algebra and$(a_1, a_2, \ldots, a_n) \in B^n$.

Let $p_i, q_i \in P_n$ for $1 \leq i \leq n$.

Then the following two conditions are equivalent:

(i) $(a_1, a_2, \ldots, a_n)$ is a solution for the system $\{(p_i, q_i) \mid 1 \leq i \leq m\}$

(ii) $(a_1, a_2, \ldots, a_n)$ is a solution for the equation

$p_1q_1^1 + p_1^1q_1 + p_2q_2^1 + p_2^1q_2 + \ldots + p_mq_m^1 + p_m^1q_m = 0$

Proof: Now $(a_1, a_2, \ldots, a_n)$ is a solution for the system$\{(p_i, q_i) \,/\, 1 \leq i \leq m\}$

$\Leftrightarrow$ $(a_1, a_2, \ldots, a_n)$ is a solution for $p_i = q_i$ for all $1 \leq i \leq m$.

$\Leftrightarrow$ $(a_1, a_2, \ldots, a_n)$ is a solution for $p_i^1q_i + p_iq_i^1 = 0$ for all $1 \leq i \leq m$.

$\Leftrightarrow$ $(a_1, a_2, \ldots, a_n)$ is a solution for $(p_1^1q_1 + p_1q_1^1) + \ldots + (p_m^1q_m + p_mq_m^1)$

$= 0 + 0 + \ldots + 0 = 0.$

The proof is complete.

16.4.5 Note: How to find a common solution for a given system of equations:

Step-(i): Suppose the given system of equation is $(p_i, q_i) \mid 1 \leq i \leq m\}$

Step-(ii): Write down the expression $(p_1^1q_1 + p_1q_1^1) + \ldots + (p_m^1q_m + p_mq_m^1)$

Step-(iii): Express polynomial (in step-(ii)) in conjunctive normal form.
Suppose the conjunctive normal form is $\prod t_i$ where t_i is a sum terms. (Note that each t_i has the form $x_1^{e_1}x_2^{e_2}\ldots x_n^{e_n}$ with each $e_i = 0$ or 1).

Step-(iv): Note that $(a_1, \ldots, a_n)$ is a solution for $\prod t_i$
$\Leftrightarrow$ it is a solution for at least one t_i.
Find the solutions for each equation $t_i = 0$.

Step-(v): All the solutions obtained in Step-(iv) form the set of all solutions of the given system of equations.

16.4.6 Problem: Solve the system of equations $(x_1x_2, x_1x_3 + x_2)$, and $(x_1 + x_2^1, x_3)\}$

Solution: **Step**-(i): In our usual notation $p_1 = x_1x_2$, $q_1 = x_1x_3 + x_2$,

$p_2 = x_1 + x_2^1$, $q_2 = x_3$.

Step-(ii): Consider the expression $[(x_1x_2)^1(x_1x_3 + x_2) + (x_1x_2)(x_1x_3 + x_2)^1]$

$$+ [(x_1 + x_2^1)^1 (x_3) + (x_1 + x_2^1)(x_3)^1].$$

Step-(iii): The conjunctive normal form for this expression (the detailed steps left to the reader as exercise) is $(x_1 + x_2 + x_3^1)(x_1^1 + x_2^1 + x_3^1) = 0$.

Step-(iv): Now $t_1 = (x_1 + x_2 + x_3^1)$ and $t_2 = (x_1^1 + x_2^1 + x_3^1)$.

Now (a_1, a_2, a_3) is a solution for $t_1 = (x_1 + x_2 + x_3^1) = 0$

$\Leftrightarrow$ $a_1 + a_2 + a_3^1 = 0$ $\Leftrightarrow$ $a_1 = a_2 = a_3^1 = 0$

$\Leftrightarrow$ $a_1 = 0$, $a_2 = 0$, $a_3 = 1$.

Thus $(0, 0, 1)$ is a solution for $t_1 = 0$.

Now (b_1, b_2, b_3) is a solution for $t_2 = (x_1^1 + x_2^1 + x_3^1)$

$\Leftrightarrow$ $b_1^1 + b_2^1 + b_3^1$ $\Leftrightarrow$ $b_1^1 = b_2^1 = b_3^1 = 0$

$\Leftrightarrow$ $b_1 = b_2 = b_3 = 1$. Thus $(1, 1, 1)$ is a solution for t_2.

Step-(v): Conclusion: $\{(0, 0, 1), (1, ,1 ,1)\}$ is the set of all solutions of all solutions for the given system of equations.

16.4.7 Theorem: Let $E = \{p_j = 0 \mid j \in J\}$ be a system of equations over a Boolean algebra B with $p_j \in P_n$ for all $j \in J$.

Write $I = \{b_1 \overline{p_1} + \ldots + b_m \overline{p_m} \mid b_j \in B \text{ for } 1 \leq j \leq n\}$

(i) Then I is an ideal in $P_n(B)$.

(ii) $(a_1, \ldots, a_n) \in B^n$ is a common solution of $\{p_j = 0 \ / \ j \in J\}$

$\Leftrightarrow$ $\bar{i}(a_1, \ldots, a_n) = 0$ for all $i \in I$.

Proof: (i) Let $x, y \in I$, $\bar{q} \in P_n(B)$.

$\Rightarrow$ $x = b_1 \overline{p_1} + \ldots + b_m \overline{p_m}$ and $y = c_1 \overline{r_1} + \ldots + c_k \overline{r_k}$.

$\Rightarrow$ $x + y = b_1 \overline{p_1} + \ldots + b_m \overline{p_m} + c_1 \overline{r_1} + \ldots + c_k \overline{r_k} \in I$.

Note that $p_j = 0 \Rightarrow p_j.q = p_j \wedge q = 0 \wedge q = 0 \Rightarrow p_j.q = 0$.

Now $x \overline{q} = (b_1 \overline{p_1} + \ldots + b_m \overline{p_m}). \overline{q} = b_1 \overline{p_1 q} + \ldots + b_m \overline{p_m q}) \in I$.

This shows that I is an ideal.

(ii) Now $(a_1, \ldots, a_n)$ is a common solution of $\{p_j = 0 \ / \ j \in J\}$

$\Leftrightarrow$ $p_j (a_1, \ldots, a_n) = 0$ for all $j \in J$.

$\Leftrightarrow$ $b_j p_j (a_1, \ldots, a_n) = 0$ for all $j \in J$ and $b_j \in B$

$\Leftrightarrow$ $(b_1 p_1 + \ldots + b_m p_m)(a_1, \ldots, a_n) = 0$ for all $j \in J$ and $b_j \in B$ for all m.

$\Leftrightarrow \bar{i} (a_1, \ldots, a_n) = 0$ for all $i \in I$. The proof is complete.

16.5 SUMMARY:

In this lesson, we have introduced ideals and filters which are important substructures of a Boolean algebra. We have given a method to find a common solution for a given system of equations.

16.6 TECHNICAL TERMS:

1. Ideal:
A non-empty I of a Boolean algebra B is said to be an ideal of B and if $ib \in I$ and $i + j \in I$ for all $i, j \in I$ and $b \in B$.

2. Principal ideal:
Let B be a Boolean algebra. Suppose $b \in B$. The set $\{ bc \mid c \in B\}$ is denoted by (b). It is easy to verify that the set (b) is an ideal of B. This ideal (b) is called a principal ideal.

3. Filter:
Let B be a Boolean algebra, and $\phi \neq F \subseteq B$. Then F is said to be a filter (or dual ideal) of B if (i) $x, y \in F \Rightarrow xy \in F$, and (ii) $a \in F, b \in B \Rightarrow a + b \in F$.

4. Kernal of a Boolean homomorphism:
Let B_1, B_2 be Boolean algebras and $h : B_1 \rightarrow B_2$ a Boolean homomorphism. Then the set $\{b \in B_1 \mid h(b) = 0\}$ is called the kernal of h, and the set is denoted by Ker h.

5. maximal filter (or ultra filter):
Let F be a proper filter in a Boolean algebra B. We say that F is a maximal filter (or ultra filter) if it satisfies the following condition:
K is a filter, $F \subseteq K \subseteq B \Rightarrow F = K$ or $K = B$.

16.7 SELF ASSESSMENT QUESTIONS:

1. If B is a Boolean algebra and I a non-empty subset of B, then prove that the following conditions are equivalent:
   (i) $I \leq B$ (That is, I is an ideal of B);
   (ii) If $i, j \in I$ and $b \in B$ such that $b \leq i$, then $i + j \in I$ and $b \in I$.
   (iii) There exists a Boolean algebra B_1 and a Boolean homomorphism $h : B \rightarrow B_1$ such that $I = \text{Ker } h$.

Ans: (Theorem: 16.2.7).

2. If B is a Boolean algebra and $\phi \neq I \subsetneq B$, then prove that the following two conditions are equivalent: (i) I is an ideal; and (ii) $F = \{x^1 / x \in I\}$ is a filter.

Ans: (Theorem 16.3.3.)

3. Define ideal and filter in a Boolean algebra and give examples each.

Ans: (Refer: Definition 16.2.2., Example 16.2.4.(ii), Definition 16.3.1., Example 16.3.2.)

4. Prove that the equations $p = q$ and $pq^1 + p^1q = 0$ have the same solutions for any two Boolean polynomials p and q in P_n.

Ans: (Theorem 16.4.3.)

5. Solve the system of equations $(x_1x_2, x_1x_3 + x_2)$, and $(x_1 + x_2^1, x_3)\}$

Ans: (Refer the solution of Problem 16.4.6.)

16.8 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr.Kedukodi Babushri Srinivas

LESSON-17

MINIMUM FORMS OF BOOLEAN POLYNOMIALS, KARNAUGH DIAGRAMS

OBJECTIVES:

- To know more on Boolean polynomials.
- To find minimal forms of Boolean polynomials.
- To understand the Quine-McCluskey algorithm to find minimal forms
- To apply Quine-McCluskey algorithm.
- To understand the Karnaugh-diagrams.
- To get ability to represent Boolean polynomials in terms of K-diagrams

STRUCTURE:

17.1. INTRODUCTION:

We know that by using the axioms of a Boolean algebra, we can simplify a given Boolean polynomial. The process of simplification is called the optimization or minimization of Boolean polynomials. This optimization is useful in future studies such as the simplification of switching circuits (we study this concept of switching circuit in the next coming lessons).

Boolean algebra is used as a tool for expressing problems of circuit design. In the previous lessons, we have seen some of them viz., Hasse diagrams, truth tables and logical diagrams. In this lesson, another widely-used way is discussed. This type of representation helps us to simplify the functions. We discussed a new structure, called Karnaugh diagram / map. This is an area which is subdivided into 2^n cells, one for each possible input combination for a Boolean function of n variables. Half the number of cells is associated with an input value of 1 for one of the variables and the other half the number of cells, with the input value 0 for the same variable. More precisely, the Karnaugh map corresponding to Boolean expressions in n variables is an area which is subdivided into 2^n cells (small squares) each of which corresponds to one of the fundamental products (or minterms) in n variables.

17.2 MINIMAL FORMS OF BOOLEAN POLYNOMIALS:

17.2.1 Notation: (i) Any Boolean variable x (either complemented or not), 0 and 1 are called as literals.

(ii) The symbol d_f denotes the total number of literals in a sum-of-products representation of a Boolean polynomial f.

(iii) The symbol e_f denotes the number of summands in f.

(iv) We say that f is simpler than a sum-of-product expression g if $e_f < e_g$, or $e_f = e_g$ and $d_f < d_g$.

(v) We say that f is minimal if there is no simpler sum-of-product expression which is equivalent to f.
(In other words, f is minimal if it has sum-of-product expression with the smallest possible number of literals).

(vi) In this section, a Boolean polynomial is also called as expression.

17.2.2 Definition: We say that an expression p implies an expression q if the condition:

$\bar{p}_{\mathcal{B}}(b_1, \ldots b_n) = 1 \Rightarrow \bar{q}_{\mathcal{B}}(b_1, \ldots b_n) = 1$ is true for all $b_1, \ldots b_n \in \mathcal{B}$.

In this case, we say that p is an implicant of q.

17.2.3 Note: (i) A product expression (or a product) is an expression in which + does not occur.

(ii) A prime implicant for an expression p is a product expression α which implies p, but which does not imply p if one or more factors in α are deleted.

(iii) If the set of factors of a product term p is a subset of the set of factors of a product term q, then we say that p is a subproduct of q.

17.2.4 Examples: (i) x_1x_3 is a subproduct of $x_1x_2x_3$.

(ii) x_1x_3 is a subproduct of $x_1x_2^1x_3$.

(iii) Consider the expression $p = x_1x_2x_3 + x_1x_2^1x_3 + x_1^1x_2^1x_3^1$.

Observe that $\overline{x_1x_3}\,(1, i_2, 1) = 1$, and $\bar{p}\,(1, i_2, 1) = 1$.

For other arguments, the value of $\overline{x_1x_3}$ is 0. The subproducts of x_1x_3 are x_1, and x_3.

Neither x_1 nor x_3 imply p (since $\overline{x_1^1}\,(1, 1, 0) = 1$, and $\bar{p}\,(1, 1, 0) = 0$).

So x_1x_3 is an implicant of p , and no subproduct of x_1x_3 is an implicant of p.

Therefore x_1x_3 is a prime implicant of p.

17.2.5 Theorem: A polynomial $p \in P_n$ is equivalent to the sum of all its prime implicants.

Proof: Let $\{ p_a \ / \ a \in A\}$ be the set of all prime implicants of p, and q be sum of all prime implicants p_a of p.

Part-(i): Suppose $\bar{q}(b_1, \ldots b_n) = 1$ for some $(b_1, \ldots b_n) \in \mathcal{B}^n$.

If $p_a(b_1, \ldots b_n) = 0$ for all prime implicants p_a, then the sum $\bar{q}(b_1, \ldots b_n) = 0$, a contradiction. Therefore $p_a(b_1, \ldots b_n) = 1$ for some prime implicant p_a.

Since p_a is a prime implicant, p_a implies p, and so $\bar{p}(b_1, \ldots b_n) = 1$.

Now we proved that $\bar{q}(b_1, \ldots b_n) = 1 \Rightarrow \bar{p}(b_1, \ldots b_n) = 1$.

Part-(ii): Suppose that $\bar{p}(b_1, \ldots b_n) = 1$.

We know that p can be expressed in disjunctive normal form.

Suppose that $p = s_1 + s_2 + \ldots$ is the disjunctive normal form where each s_i is a product term.

Since $\bar{p}(b_1, \ldots b_n) = 1$, there exists i such that $s_i(b_1, \ldots b_n) = 1$.

If s_i is a prime implicant, then (since $s_i(b_1, \ldots b_n) = 1$) the the sum of prime implicants equal to 1, and so $\bar{q}(b_1, \ldots b_n) = 1$.

Now s_i is a product term of the form $x_1^{b_1} \ldots x_n^{b_n}$.

If s_i is not a prime implicant, then there exists a subterm t of s_i which implies p, and $t(b_1, \ldots b_n) = 1$.

Note that we got t by removing some terms in $s_i = x_1^{b_1} \ldots x_n^{b_n}$.

If t is a prime implicant, then $\bar{q}(b_1, \ldots b_n) = 1$.

If t is not a prime implicant, then (after some steps), we get a subproduct term r of s_i such that r implies p, and there is no subproduct term of r that implies p.

Then r is an implicant of p, and $r(b_1, \ldots b_n) = 1$.

Since r is a prime implicant, and $r(b_1, \ldots b_n) = 1$, we have that the sum of prime implicants is equal to 1. That is, $\bar{q}(b_1, \ldots b_n) = 1$.

Now we proved that $\bar{q}(b_1, \ldots b_n) = 1 \Rightarrow \bar{p}(b_1, \ldots b_n) = 1$.

Part-(iii): From Part-(i) and Part-(ii), we conclude that

$\bar{q}(b_1, \ldots b_n) = 1 \Leftrightarrow \bar{p}(b_1, \ldots b_n) = 1$.

Now we have that $\bar{q}(b_1, \ldots b_n) = 0 \Leftrightarrow \bar{p}(b_1, \ldots b_n) = 0$.

This shows that p is the sum of its prime implicants.

17.2.6 Definition: A sum of prime implicants of p is said to be irredudant if it is equivalent to p, but does not remain equivalent if any one of its summands is omitted.

17.2.7 Note: (i) A minimal sum-of-product expression is irredundant.

(ii) To get a minimal expression for a given polynomial p, first we get the set of irredundant expressions for p, and then we select that irredundant expression with the least number of literals.

(iii) Prime implicants are obtained by starting with the disjunctive normal form d for the Boolean polynomial p and then by applying the rule $yz + yz^1 \sim y$, (from left to right) wherever necessary.

In particular, we use $\alpha\beta\gamma + \alpha\beta^1\gamma \sim \alpha\gamma$, where α, β, and γ are some product expressions.

The set all sub expressions of the d.n.f. of p which cannot be simplified further by this procedure, is the set .of prime implicants.
The sum of these prime implicants obtained is equivalent to p (we may say that the sum of these prime implicants is equal to p).

17.2.8 Example: Let p be the Boolean polynomial.
We use ϖ, x, y, z instead of x_1, x_2, x_3, x_4.

Step-(i): Suppose the disjunctive normal form d for p is given by

$d = \varpi xyz^1 + \varpi xy^1z^1 + \varpi x^1yz + \varpi x^1yz^1 + \varpi^1x^1yz + \varpi^1x^1yz^1 + \varpi^1x^1y^1z.$

Step-(ii): Observe the following:

$\varpi xyz^1 + \varpi xy^1z^1 = \varpi xz^1,$

$\varpi x^1yz + \varpi x^1yz^1 = \varpi x^1y,$

$\varpi xyz^1 + \varpi x^1yz^1 = \varpi yz^1,$

$\varpi^1x^1yz + \varpi^1x^1yz^1 = \varpi^1x^1y,$

$\varpi^1x^1yz + \varpi^1x^1y^1z = \varpi^1x^1z,$

$\varpi x^1yz + \varpi^1x^1yz = x^1yz,$

$\varpi x^1yz^1 + \varpi^1x^1yz^1 = x^1yz^1.$

Now we use the above mentioned equations in the process of minimizing the polynomial expression. In general, this procedure is repeated again and again.
Whenever a product term is used (in this process), we place a tick mark.
At any step, the product terms that cannot be ticked, are prime implicants.

Step-(iii): In this example, the second round of simplifications yields:

$\varpi x^1 y + \varpi^1 x^1 y = x^1 y$,

$x^1 yz + x^1 yz^1 = x^1 y$.

These four expressions $\varpi x^1 y$, $\varpi^1 x^1 y$, $x^1 yz$, and $x^1 yz^1$ are ticked.

Finally we get that

$p \sim \varpi xz^1 + \varpi yz^1 + \varpi x^1 z + x^1 y$, which is a sum of prime implicants of p.

McCluskey improved this method, and the improved method is called as Quine-McCluskey algorithm.

17.3 QUINE-MCCLUSKEY ALGORITHM:

17.3.1 Algorithm:

Step-1: Consider the d.n.f. of the given Boolean polynomial.
We represent all the product terms of the d.n.f. in terms of zero-one-sequences (In other words, the product terms are represented by binary n-tuples).
In particular, x_1^1 and x_1 are denoted by 0 and 1, respectively.

(For example, the product term $\varpi^1 x^1 y^1 z$ is denoted by 0001).

Missing variables are indicated by a dash.

(For example, the product term $\varpi^1 x^1 z$ is denoted by 00-1).

Step-2: The product expressions, regarded as binary n-tuples, are partitioned into classes according to the numbers of ones in the expression. So we write the n-tuples according to increasing numbers of ones. In our example, the order is given below:

Table-1

$\varpi^1 x^1 y^1 z$	0 0 0 1
$\varpi^1 x^1 yz^1$	0 0 1 0
$\varpi^1 x^1 yz$	0 0 1 1
$\varpi x^1 yz^1$	1 0 1 0
$\varpi xy^1 z^1$	1 1 0 0
$\varpi x^1 yz$	1 0 1 1
ϖxyz^1	1 1 1 0

Step-3: If two of these expressions differ in exactly one position, then they are of the form $p = i_1 i_2 \ldots i_r \ldots i_n$ and $q = i_1 i_2 \ldots i^1_r \ldots i_n$, where all i_k are from {0, 1, -} and the i_r is in {0,1}.

Now instead of $p + q$ we write $i_1 i_2 \ldots i_{r-1} - i_{r+1} \ldots i_n$.

Also we place a tick mark at both p and q.
Now we consider our example. From the Table-1, we get the Table-2.
In getting table-2, we use all the terms in table-1.

So all the terms of table-1 are to get a tick mark, and so no one of these product terms is a prime implicant.

Table - 2

0	0	-	1	
0	0	1	-	✓
-	0	1	0	✓
-	0	1	1	✓
1	0	1	-	✓
1	-	1	0	
1	1	-	0	

Step-4: The unticked terms in table-2, are prime implicants.

At present, from table-2, the prime implicants that are obtatined are 00-1 ($\varpi^1 x^1 z$), 1-10 (ϖyz^1), and 11-0 (ϖxz^1). The expressions with ticks are not prime implicants and so we have to go for further reduction.

The further reduction gives us only one term -01- ($x^1 y$).
Thus we got all the prime implicants, namely

Table-3

0	0	-	1	$\varpi^1 x^1 z$
1	-	1	0	ϖyz^1
1	1	-	0	ϖxz^1
-	0	1	-	$x^1 y$

Note that the sum of all prime implicants of a given polynomial may not be in the minimal form.

Step-5: We know that the sum of all prime implicants of p is equivalent to p.
Observe the Table-4. This table-4 is called as prime implicants table.
The binary n-tuples related to the product terms of the d.n.f. are used for column headings. The prime implicants are used for row headdings.
A product term u is said to cover another product term v if u is a subproduct of v.
A cross mark (that is, $\times$) is placed at the junction of the i^{th} row and j^{th} column if the prime implicant in the i^{th} row covers the product term of the j^{th} column.
Now we select a minimal subset S of the set of prime implicants so that each product term of the d.n.f. is covered by at least one of the prime implicant in S.
A prime implicant is called a main term(or essential) if it covers a product expression (of the d.n.f.) which is not covered by any other prime implicant.
The sum of the main terms is called the core.
[Consider the example and observe table-4.

Table-4: **Prime implicants table**

	0001	0010	0011	1010	1100	1011	1110
00 - 1	×		×				
1 - 10				×			×
11 - 0					×		×
- 01 -		×	×	×		×	

any of the prime implicant.

Note that 00-1, 11-0, -01- are main terms.

So the core = (00-1) + (11-0) + (-01-)

$= \varpi^1x^1z + \varpi xz^1 + x^1y$.]

Step-6: If the set of all prime implicants in the core cover all the product terms in the d.n.f., then the core is the (unique) minimal form of d.
If the set of all prime implicants in the core do not cover all the product terms in the d.n.f., then we go further.
[Consider our example. From the above step-5, we know that the

core = $\varpi^1x^1z + \varpi xz^1 + x^1y$.

In this example, the set of all prime implicants in the core cover all the product terms in the d.n.f. Hence the minimal expression is $\varpi^1x^1z + \varpi xz^1 + x^1y$.]

Step-7: Suppose the product terms of d.n.f. which are not covered by the prime implicants of the core are $q_1, \ldots, q_k$, and the prime implicants not in the core are $p_1, \ldots, p_m$. We form the next table (similar to prime implicants table) with q_j as the column headings, and p_i as row headings.
The mark $\times$ is placed in the entry (i, j) (that is, at the junction of the i^{th} row and j^{th} column) to indicate the fact that p_i covers q_j. We then find a minimal sub-collection of $p_1, \ldots, p_m$ which covers all of $q_1, \ldots, q_k$ and add them to the core.

17.3.2 Problem: Determine the minimal form of p, which is given in its disjunctive normal form

$$p = v^1w^1x^1y^1z^1 + v^1w^1x^1yz^1 + v^1w^1xy^1z^1 + v^1w^1xyz^1$$
$$+ v^1wx^1y^1z + v^1wx^1yz^1 + v^1wxy^1z + v^1wxyz^1$$
$$+ v^1wxyz + vw^1x^1y^1z^1 + vw^1x^1y^1z + vw^1xy^1z$$
$$+ vwx^1yz^1 + vwxy^1z^1 + vwxyz^1 + vwxyz$$

Solution: We follow the Quine-McClusky algorithm. We are not going for detailed steps, as the steps in detailed presented in the algorithm. We can write p in the binary form, as follows:

p = 00000 + 00010 + 00100 + 00110
+ 01001 + 01010 + 01101 + 01110
+ 01111 + 10000 + 10001 + 10101
+ 11010 + 11100 + 11110 + 11111.

First we write these binary 5-tuples in the order and then we form the Table-1.
We got the Table-2 from Table-1. All the product terms in Table-1 are ticked.
So we do not get a prime implicant from Table-1. Now observe Table-2.
We got Table-3 from Table-2. All the product terms in Table-2 are not ticked.

The terms which are not ticked are denoted by J, I, H, G, F, E.
For example, J represents the prime implicant "-0000".

Table-1

							Row numbers
0 ones	0	0	0	0	0	✓	(1)
1 one	0	0	0	1	0	✓	(2)
	0	0	1	0	0	✓	(3)
	1	0	0	0	0	✓	(4)
2 ones	0	0	1	1	0	✓	(5)
	0	1	0	0	1	✓	(6)
	0	1	0	1	0	✓	(7)
	1	0	0	0	1	✓	(8)
3 ones	0	1	1	0	1	✓	(9)
	0	1	1	1	0	✓	(10)
	1	0	1	0	1	✓	(11)
	1	1	0	1	0	✓	(12)
	1	1	1	0	0	✓	(13)
4 ones	0	1	1	1	1	✓	(14)
	1	1	1	1	0	✓	(15)
5 ones	1	1	1	1	1	✓	(16)

Observe the Table-3.
No two terms in table-3 can be used to get a term to the next table. So each term of table-3 is a prime implicant. The prime implicants obtained from table-3 are denoted by D, C, B, A.
Final list of prime implicantsis A, B, C, D, E, F, G, H, I, J.

Table - 2

(1)(2)	0	0	0	-	0	✓
(1)(3)	0	0	-	0	0	✓
(1)(4)	-	0	0	0	0	J
(2)(5)	0	0	-	1	0	✓
(2)(7)	0	-	0	1	0	✓
(3)(5)	0	0	1	-	0	✓
(4)(8)	1	0	0	0	-	I
(5)(10)	0	-	1	1	0	✓
(6)(9)	0	1	-	0	1	H
(7)(10)	0	1	-	1	0	✓
(7)(12)	-	1	0	1	0	✓
(8)(11)	1	0	-	0	1	G

(9)(14)	0	1	1	-	1	F
(10)(14)	0	1	1	1	-	✓
(10)(15)	-	1	1	1	0	✓
(12)(15)	1	1	-	1	0	✓
(13)(15)	1	1	1	-	0	E
(14)(16)	-	1	1	1	1	✓
(15)(16)	1	1	1	1	-	✓

Table - 3

(1)(2), (3)(5)	0	0	-	-	0	D
(2)(5), (7)(10)	0	-	-	1	0	C
(7)(10), (12)(15)	-	1	-	1	0	B
(10)(15), (14)(16)	-	1	1	1	-	A

Step-4: In the prime implicants table (that is, in Table-4), for convenience, we give the product terms of d.n.f. in + binary 5-tuples in column form.

Table - 4		(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)
		0	0	0	1	0	0	0	1	0
		0	0	0	0	0	1	1	0	1
		0	0	1	0	1	0	0	0	1
		0	1	0	0	1	0	1	0	0
		0	0	0	0	0	1	0	1	1
- 1 1 1 -	A									
- 1 - 1 0	B							×		
0 - - 1 0	C		×			×		×		
0 0 - - 0	D	×	×	×		×				
1 1 1 – 0	E									
0 1 1 – 1	F									×
1 0 - 0 1	G								×	
0 1 - 0 1	H						×			×
1 0 0 0 -	I				×				×	
- 0 0 0 0	J	×			×					

(10)	(11)	(12)	(13)	(14)	(15)	(16)
0	1	1	1	0	1	1
1	0	1	1	1	1	1
1	1	0	1	1	1	1
1	0	1	0	1	1	1

0	1	0	0	1	0	1
×				×	×	×
×		×			×	
×						
			×		×	
				×		
	×					

Observe the prime implicants table. The product terms D, H, G, B, E, and A are the main terms. So we have that
Core = the sum of the main terms = D + H + G + B + E + A
The product term of column (4) is the only product term that is not covered by the main terms in the core. This term 10000 is denoted by q_1.
The prime implicants C, F, I, and J are not in the core. Now we form Table-5.

Table - 5		(4)
0 - - 1 0	C	
0 1 1 - 1	F	
1 0 0 0 -	I	×
- 0 0 0 0	J	×

This means that the minimal form is

(i) D + H + G + B + E + A + J (if we use I);
(ii) D + H + G + B + E + A + J (if we use J).

Note that the minimal form is not unique.
In our usual notation, the minimal form of p (given in (i)) is given by

$P = v^1w^1z^1 + v^1wy^1z + vw^1y^1z + wyz^1 + vwxz^1 + wxy + vw^1x^1y^1.$

17.4. KARNAUGH DIAGRAMS:

Now we study a method of representing a given Boolean polynomials (in n variables) in the form of a diagram called as Karnaugh diagrams (that contains 2^n cells where n is the number of variables).

17.4.1 Example: Consider the Boolean polynomial $p = x_1x_2$. The following (Table-1) is the truth table for p. As there are only two variable x_1 and x_2 there exists four possibilities for x_1x_2 : 00, 01, 10, 11. Observe the following table-1.

Table - 1

Row	b_1	b_2	Minterm	$\overline{p}(b_1, b_2) = b_1 b_2$
(1)	1	1	x_1x_2	1
(2)	1	0	$x_1x^1_2$	0
(3)	0	1	$x^1_1x_2$	0
(4)	0	0	$x^1_1x^1_2$	0

In the fourth column, there is only one 1 and this 1 is related to $(b_1, b_2) = (1, 1)$. This 1 is related to the unique minterm x_1x_2. The other values of p are equal to 0. For two input variables b_1, b_2, the Karnaugh diagram (table-2) has b_1 and b^1_1 as column headings and b_2 and b^1_2 as row headings.

	b_1	b^1_1
b_2	(1)	(3)
b_2^1	(2)	(4)

Table - 2

Each box at the junction of a row and a column represents a minterm. A shaded box represents the value 1, and an unshaded box represents the value 0.
For the given function $\overline{p} = x_1x_2$, (refer table-3) the shaded box represents the value 1. The other boxes which do not have shade represents the value 0.

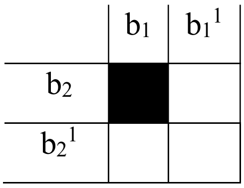


Table - 3

17.4.2 Example: Karnaugh diagram for three input variables b_1, b_2, b_3 will be in form given in table-4.

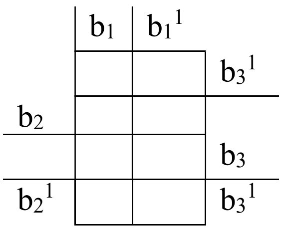


Table - 4

17.4.3 Example: Karnaugh diagram for four input variables is of the form (called the standard square (SQ)) given in table-5.

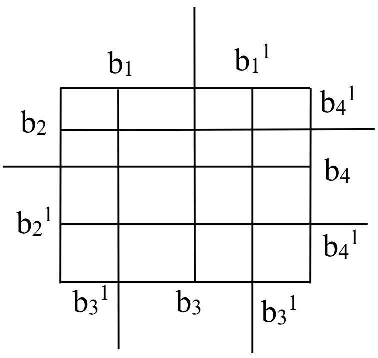
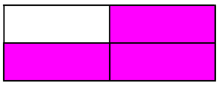


Table - 5

17.4.4 Example: Now we present the Karnaugh diagrams of some polynomials in two variables: x_1 and x_2 :

$x_1^1 + x_2^1$:



$x_1x_2 + x_1^1x_2^1$:

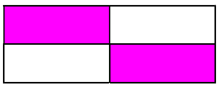
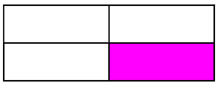
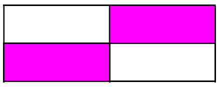


Table - 6

$x_1^1x_2^1$:



$x_1^1x_2 + x_1x_2^1$:



17.4.5 Example: The standard square (that is, the Karnaugh diagram for four variables) enables us to construct Karnaugh diagrams for five or more input variables. This was illustrated by the following tables.

(i) We follow the diagram in Table-7 to represent a Boolean polynomial with five variables.

b_5	b^1_5
SQ	SQ

Table - 7

(ii) We follow the diagram in Table-8 to represent a Boolean polynomial with six variables.

	b_5	b_5^1
b_6	SQ	SQ
b_6^1	SQ	SQ

Table - 8

17.4.6 Note: We use the Karnaugh diagrams to simplify the Boolean polynomials. Consider the collection of the box portions with shade. We try to collect as many shaded box portions of the diagram as possible to form a bigger block. This big box represents a "simple" polynomial. (We may use a part of the diagram more than once, because the polynomials corresponding to blocks are connected by +).

17.4.7 Problem : Simplify the polynomial $p = (x_1 + x_2)(x_1 + x_3) + x_1x_2x_3$ by using its Karnaugh diagram.

Solution: The given polynomial is $p = (x_1 + x_2)(x_1 + x_3) + x_1x_2x_3$.
The Karnaugh diagram for this polynomial is given in table-9.

Observe table-9. Here the boxes 1,3,4,5,7 are the shaded boxes.
These shaded boxes represent 1 (the value of the function).

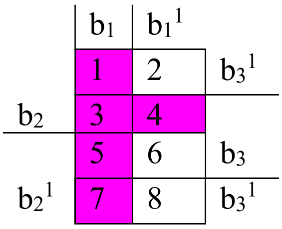


Table - 9

The shaded boxes (1), (3), (5), (7) forms a big shaded rectangular region.
It is clear that this rectangular region represents the variable x_1. Therefore this x_1 is a prime implicant. Now consider the shaded rectangular region formed by the shaded boxes (3), (4) . This shaded rectangular region represents the term x_2x_3 . This is also a prime implicants. We can understand from table-9 there are only two prime implicants for the given polynomials. We know that every polynomial is equal (in other words, equivalent) to sum of its prime implicants. Thus we conclude that $p \sim x_1 + x_2x_3$.

17.5. MINIMIZATION OF BOOLEAN EXPRESSIONS USING K-Maps:

The process of minimization of circuits is important in circuit design. The aim of minimization is to reduce the number of gates to a minimum. Minimization of an expression is the selection of the simplest representative expression of an equivalence class to serve as our circuit. K-maps are used in the minimization process for functions of six or fewer variables.

Two minterms or fundamental products (cells in a K-map) are said to be adjacent if they have the same variables and if they differ in exactly one literal which must be a complemented variable in one product and uncomplemented in the other.
For example,

1. xyz^1 and xy^1z^1 are adjacent (here, note that in the terms xyz^1 and xy^1z^1 , the difference is: one term contains y and other term contains y^1, the difference is in only one variable, the other parts are same).

2. x^1yzw and x^1yz^1w are adjacent (here, note that in the terms x^1yzw and x^1yz^1w , the difference is: one term contains z and other term contains z^1, the difference is in only one variable, the other parts are same).

3. x^1yzw and xyz^1w are not adjacent as they differ in two literals (here, note that in the terms x^1yzw and xyz^1w , the difference is in two variables. The difference is: one term contains x^1, z and other term contains x, z^1. Hence the given terms are not adjacent).

17.5.1 Theorem: Sum of two adjacent products P_1 and P_2 is a fundamental product with one less literal.
Proof: Two adjacent products P_1 and P_2 are represented as

$$P_1 = a_1a_2 \ldots a_{r-1}a_ra_{r+1} \ldots a_k$$

$$\text{and} \quad P_2 = a_1 a_2 \ldots a_{r-1} a_r^1 a_{r+1} \ldots a_k$$

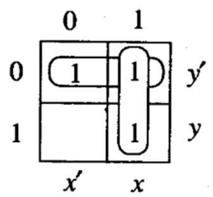
$$\text{Then } P_1 \oplus P_2 = a_1 a_2 \ldots a_{n-1} a_r a_{r+1} \ldots a_k (a_r \oplus a_r^1) = a_1 a_2 \ldots a_{r-1} a_{r+1} \ldots a_k$$

17.5.2 Example: For three variables, $xyz^1 \oplus xy^1z^1 = xz^1 (y \oplus y^1) = zz^1$.

The above result and the absorption operation $xyz + xyz^1 = xy$ help us in grouping the terms. Minimization involves grouping of adjacent cells with l's in them into a largest possible block of such cells. Simplified expression must contain minimum number of such blocks.

17.5.3 Note: In case of two variables, a block will be either a pair of adjacent squares or an individual square.

17.5.4 Example: Minimize the expression $f = xy \oplus xy^1 \oplus x^1y^1$
Solution: The K-map for the given expression is shown in the following Figure.



Therefore, f contains two blocks corresponding to x and other to y^1.
Hence $f = x \oplus y^1$.

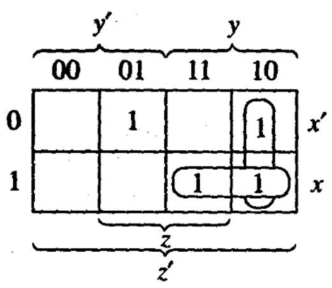
17.5.5 Note: In the case of 3 variables, a basic rectangle contains either a square, or two adjacent squares, or four squares which form a one-by-four or a two-by- two rectangle.
A maximal basic rectangle is a block.

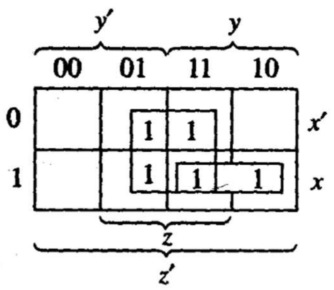
17.5.6 Example: Minimize the following expressions:

(a) $f_1 = xyz \oplus xyz^1 \oplus x^1yz^1 \oplus x^1y^1z$

(b) $f_2 = xyz \oplus xyz^1 \oplus xy^1z \oplus x^1yz \oplus x^1y^1z$

Solution K-maps for the given expressions are given in Figures.





Their minimized expressions are

(a). $x^1y^1z \oplus yz^1 \oplus xy$, (b). $z \oplus xy$.

17.6 SUMMARY:

In this lesson, we have learnt simplification (or minimization) of Boolean expressions. This optimization is useful in the simplification of switching circuits which will be studied in the next coming lessons. More importantly, we have discussed Quine-McCluskey Algorithm for minimization of Boolean expressions.

In the later parts of this lesson, we discussed the process of reducing the number of terms in a Boolean expression Karnaugh diagram / map. The method described was introduced by Maurice Karnaugh in 1953. This method is usually applied only when the function involve six variables or less. It has enormous applications in electronics and communications engineering and information technology.

17.7 TECHNICAL TERMS:

1. Implicant

We say that an expression p implies an expression q if the condition: $\overline{p}_{\mathcal{B}}(b_1, \ldots b_n) = 1 \Rightarrow \overline{q}_{\mathcal{B}}(b_1, \ldots b_n) = 1$ is true for all $b_1, \ldots b_n \in \mathcal{B}$. In this case, we say that p is an implicant of q.

2. Prime implicant

A prime implicant for an expression p is a product expression α which implies p, but which does not imply p if one or more factors in α are deleted.

3. Quine-McCluskey Algorithm (Algorithm 17.3.1.)

4. Standard square (SQ)) in terms of Karnaugh diagram for four input variables.

Karnaugh diagram for four input variables is given in Table-5 here (it is called as the standard square (SQ)).

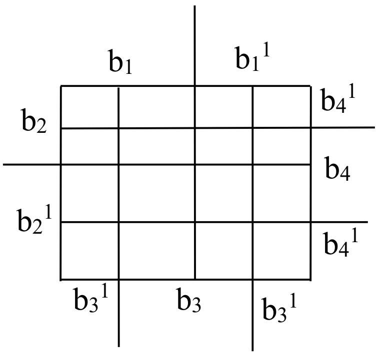


Table - 5

5. Karnaugh map:

Corresponding to Boolean expressions in n variables, we use a square to represent the Boolean expression. The area of the square is subdivided into 2^n cells (small squares) each of which corresponds to one of the fundamental products or minterms in n variables. Such diagrams used are called as Karnaugh diagrams / maps.

6. Adjacent:

Two minterms or fundamental products (cells in a K-map) are said to be adjacent if they have the same variables and if they differ in exactly one literal which must be a complemented variable in one product and uncomplemented in the other.

17.8 SELF ASSESSMENT QUESTIONS:

1. Prove that a polynomial $p \in P_n$ is equivalent to the sum of all its prime implicants.
Ans: (Theorem 17.2.5.)

2. Determine the minimal form of p, which is given in its disjunctive normal form
$p = v^1w^1x^1y^1z^1 + v^1w^1x^1yz^1 + v^1w^1xy^1z^1 + v^1w^1xyz^1$
$+ v^1wx^1y^1z + v^1wx^1yz^1 + v^1wxy^1z + v^1wxyz^1$
$+ v^1wxyz + vw^1x^1y^1z^1 + vw^1x^1y^1z + vw^1xy^1z$
$+ vwx^1yz^1 + vwxy^1z^1 + vwxyz^1 + vwxyz$
Ans: (Problem 17.3.2.)

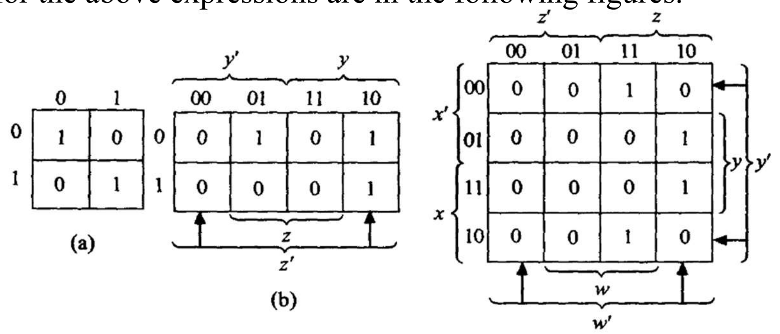
3. Find all prime implicants of $xy^1z + x^1yz^1 + xyz^1 + xyz$ and form the corresponding prime implicants table.
Ans: (Use the procedure given in 17.3.1.(refer steps 1 to 4)).

4. Simplify the polynomial $p = (x_1 + x_2)(x_1 + x_3) + x_1x_2x_3$ by using its Karnaugh diagram.
Ans: (refer: Problem : 17.4.7.).

5. What do you mean by Karnaugh diagram/map. Give an example.
Ans: (Refer: matter before Example 17.4.1, and this example also).

6. Find the K-map for the following expressions:
(a) $(x * y) \oplus (x^1 * y^1)$
(b) $(x^1 * y^1 * z \oplus x^1 * y * z^1 \oplus x * y * z^1)$
(c) $(x^1 * y^1 * z * w) \oplus (x^1 * y * z * w^1) \oplus (x * y^1 * z * w) \oplus (x * y * z * w^1)$

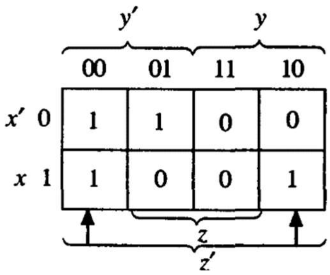
Ans: K- maps for the above expressions are in the following figures.



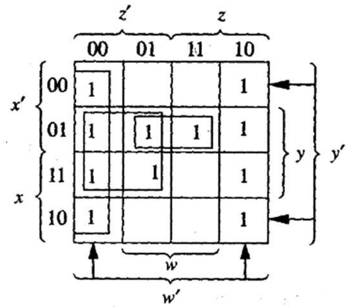
7. For the Boolean expression represented by the following truth table, give K-map representation. Also write the expression.

x	y	z	f(x, y, z)
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Ans: The Boolean expression represented by the given table is $x^1y^1z^1 \oplus x^1y^1z \oplus xy^1z^1 \oplus xyz^1$. The following figure represents the K-map for the expression $x^1y^1z^1 \oplus x^1y^1z \oplus xy^1z^1 \oplus xyz^1$.



8. Minimize the expression: $w^1 \oplus y * (x^1 \oplus z^1)$ and provide K-map.
Ans: Minimized expression is : $w^1 \oplus yz^1 \oplus wx^1y$.



The K-map is shown in the Figure.

17.9 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Harikrishnan Panackal

LESSON-18

SWITCHING CIRCUITS AND GATING NETWORKS

OBJECTIVE:

- ❖ To know Switching circuits.
- ❖ To use the concepts of Boolean expressions.
- ❖ To understand the Gating networks.
- ❖ To identify different types of Gates.
- ❖ To have proper understanding of Swiching circuits and Gating networks.
- ❖ To develop skills in finding Gating networks for the given expressions.

STRUCTURE:

18.1. INTRODUCTION:

The most important application of Boolean algebra lies in the realm of electrical engineering. The devices such as mechanical switches, diodes, magnetic dipoles, and transistors are two state devices. The two states may be realized as current or no current, magnetized or not magnetized, high potential or low potential, and closed or open. Boolean algebra can be applied to any two state device. In this lesson, we study the switching circuits and gating networks.

18.2 PRELIMINARY NOTATIONS:

18.2.1 Note: Observe the Figures-1, 2, and 3.

(i) In these figures, the symbols x and y are electromagnets. These x and y determine whether the corresponding switch is open or closed.

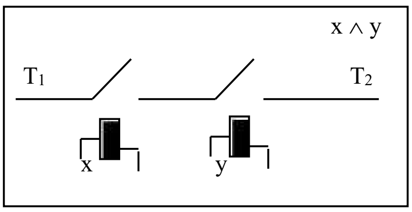


Figure-1

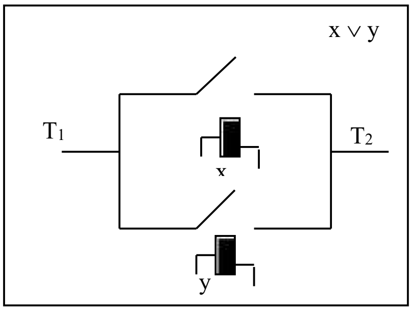


Figure-2

(ii) In the figures-1 and 2, the switches are normally held open by a spring. When the current flows through the electromagnet, the switch is pulled closed.

(iii) In figure-3, the switch is normally closed by a spring and when current flows through the electromagnet x, the switch is forced open.

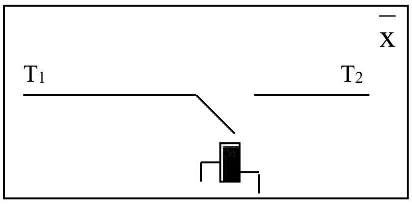


Figure-3

(iv) The flow of current through the main circuits (that is, the circuit connecting T_1 and T_2) depends on whether the electromagnets x and y are “on” or “off”.

(v) "On" is denoted by “1” and "off" by “0”.

x	y	$x \wedge y$	$x \vee y$	$\bar{x}$
0	0	0	0	1
0	1	0	1	1
1	0	0	1	0
1	1	1	1	0

Table-1

Current flow through the main circuit is denoted by 1 while no current is denote by 0.

(vi) Now the dependency was shows in the table-1.

18.2.2 Note: (i) If two terminal switching circuits f_1 and f_2 depend on the switches x_1, x_2, …, x_n, then $f_1 \vee f_2$ will denote the switching circuit determined by f_1 and f_2 in parallel (see the Figure-4).

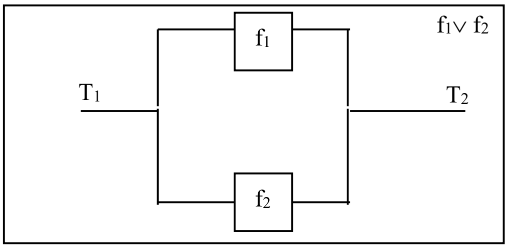


Figure-4

(ii) $f_1 \wedge f_2$ will denote the switching circuit determined by f_1 and f_2 in series (see the Figure-5).

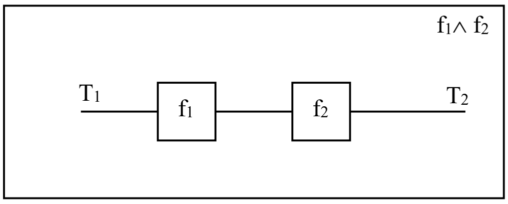


Figure-5

(iii) $\overline{f_1}$ the inverse (or the complement) of f_1 will denote the switching circuit (as in the Figure-3) that takes the value 1 when f_1 takes value 0; and takes the value 0 when f_1 takes the value 1.

18.2.3 Problem: Draw switching circuits which represent the following Boolean expressions: (i) $x_1 \wedge (x_2 \vee x_3)$, and (ii) $(x_1 \wedge x_2) \vee (x_1 \wedge x_3)$.

Solution: (i) Figure-6 represents the Boolean expression $x_1 \wedge (x_2 \vee x_3)$.

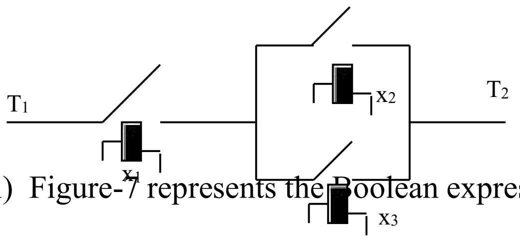


Figure-6

(ii) Figure-7 represents the Boolean expression $(x_1 \wedge x_2) \vee (x_1 \wedge x_3)$.

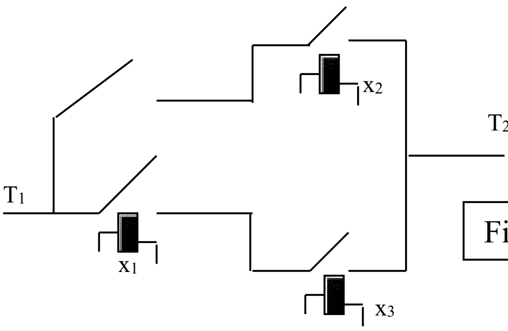


Figure-7

18.3. SWITCHING CIRCUITS:

The main object of the algebra of switching circuits is to describe electrical or electronic switching circuits.

18.3.1 Note: (i) We also use the symbols given in Figure-8, for the switches.

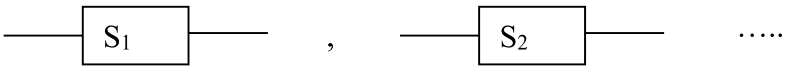
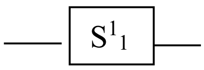
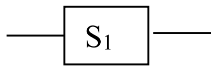


Figure-8

(ii) The symbol 

is for the complement of the switch 

In other words, S_1 and $S_1{}^1$ constitute two switches which are linked, in two separate places in a circuit so that S_1 is open $\Leftrightarrow$ $S_1{}^1$ is closed.

18.3.2 Definitions: (i). Each symbol $x_1, \ldots, x_n$ is called a switch.

(ii). Every $p \in P_n$ is called a switching circuit.

(iii). x^1_i is called the complementation switch of x_i.

(iv). $x_i x_j$ is called the series connection of x_i and x_j.

(v). $x_i + x_j$ is called the parallel connection of x_i and x_j.

(vi). For $p \in P_n$ the corresponding polynomial function $\overline{p} \in P_n(B)$ is called the switching function of p.

(vii). $\overline{P}(a_1, \ldots, a_n)$ is called the value of the switching circuit p at $(a_1, \ldots, a_n) \in B^n$. Here the elements a_i of B are called input variables.

18.3.3 Note: It is possible to model electrical circuits by using Boolean polynomials.

(i) Consider the Figure-9. This switching circuit represents the Boolean polynomial $p = x_1(x_2(x_3 + x_4) + x_3(x_5 + x_6))$.

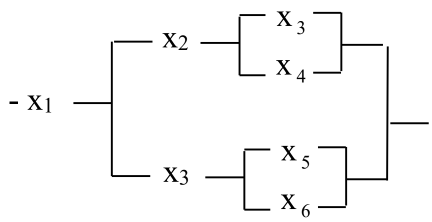


Figure-9

(ii) Consider the Figure-10. This switching circuit represents the Boolean polynomial

$q = x_1(x_2^1 (x_6 + x_3 (x_4 + x_5^1)) + x_7(x_3 + x_6)x_8^1)$.

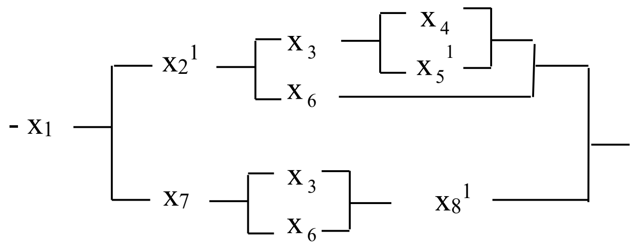


Figure-10

18.3.4 Note: Consider the figures-6 and 7 (given in the solution of the Problem 18.2.3).

Figure-6 represents the polynomial $p = x_1 \wedge (x_2 \vee x_3)$.

Figure-7 represents the polynomial $q = (x_1 \wedge x_2) \vee (x_1 \wedge x_3)$.

It is clear that $p \sim q$. That is one of the polynomials p or q can be obtained from the other by using the laws of Boolean algebra.
So there exist two different electric circuits p and q which operate "identically" if their values are equal for all possible combinations of the input variables $a_1, a_2, \ldots, a_n$.
This means, there exists two distinct electric circuits whose corresponding polynomials are p and q (p and q are different polynomials) such that $\overline{p}_B = \overline{q}_B$. (that is, $p \sim q$)

18.3.5 Note: Algorithm to find a simplified electrical circuit:

Suppose an electrical circuit (say, circuit-1) is given.

Step-(i): Find the polynomial p which represents the electrical circuit-1.

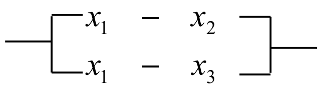
Step-(ii): By using Quine-McCluskeyalgorithm, simplify the polynomial p. Suppose a simple form of p is q. Now q got more simple form than p, and $p \sim q$.

Step-(iii): Write down the electrical circuit (say, circuit-2) which represents q.

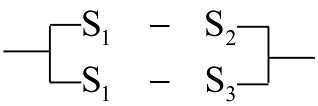
In this way we can get a simple electrical circuit (circuit-2) which operates identically to the given electrical circuit (circuit-1).

18.4 GATING NETWORKS:

18.4.1 Note: (i) We may represent the polynomial (or the circuit) $x_1x_2 + x_1x_3$ as follows:



(ii) The electrical realization for the polynomial $x_1x_2 + x_1x_3$ is given by the following figure.



18.4.2 Note: In the above Note 18.4.1., we presented a method of representing a given polynomial in the form of a switching circuit. Now we provide a new representation.

This new representation consist of some boxes, which converts input variables into values.

(i) Consider the following diagram-1. Here $a_1, \ldots, a_n$ are input variables. The polynomial $p \in P_n$ converts the given set of inputs into the value $\bar{p}(a_1, \ldots, a_n)$.

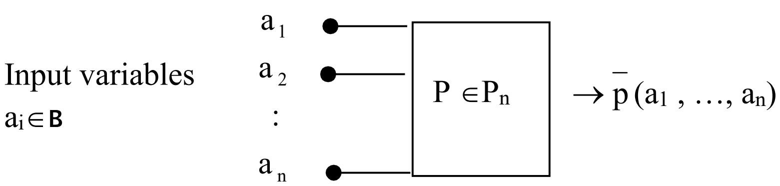


Diagram-1

(ii). For example, consider the polynomial $p = a_1a_2 + a_1a_3$. Here a_1, a_2, a_3 are input variables.

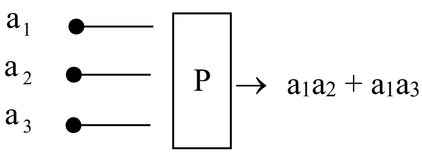


Diagram-2

polynomial p converts the variables into the value $\bar{p}(a_1, a_2, a_3) = (0 \text{ or } 1)$.

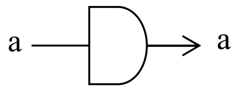
18.4.3 Note: (i). Some switches or switching circuits may be represented by some new type of diagrams which are called as gates.

By using these gates, we can represent any switching circuit as a combination of the gates. This is a symbolic representation.

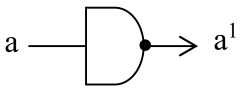
(ii). From (i), we can conclude that a gate (or a combination of gates) is a polynomial p.

(iii). A symbolic representation (that is, a combination of gates) which represents a polynomial, is called a gating network.

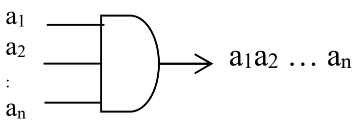
18.4.4. Notation: Different gates that we use are given below:

(i) 

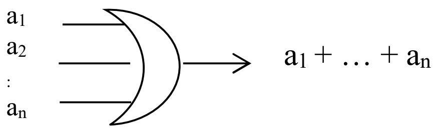
identity-gate (symbolizes x);

(ii) 

NOT-gate (or **inverter**) (symbolizes x^1);

(iii) 

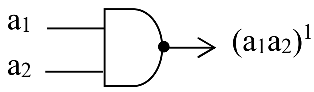
AND-gate (symbolizes $x_1x_2 \dots x_n$);

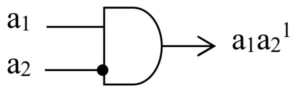
(iv) 

OR-gate (symbolizes $x_1 + \dots + x_n$)

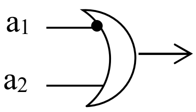
18.4.5. Notation: We also use a small black disk (either before or after) one of the other gates to indicate an inverter.

18.4.6 Example:

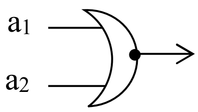
(i) 

(ii) 

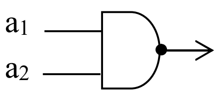
18.4.7 Definitions:

(i) 

$a_1^1 + a_2$ **Subjunction-gate**

(ii) 

$(a_1 + a_2)^1$ **NOR-gate** or **Pierce-operation**

(iii) 

$(a_1a_2)^1$ **NAND-gate** or **Sheffer-operation**

18.4.8 Problem: Write down the gating network for the polynomial $p = (x_1{}^1 x_2)^1 + x_3$.

Solution: The required gating network is given by the Figure-11.

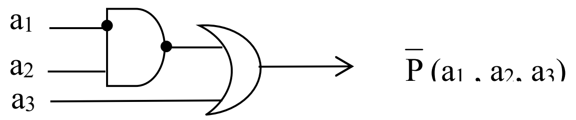


Figure-11

18.4.9 Problem: (i) Find the polynomial p which corresponds to the gating network given in the Figure-12.

(ii) Find a simplified gating network which operates in the same way as the gating

network given in Figure-12.

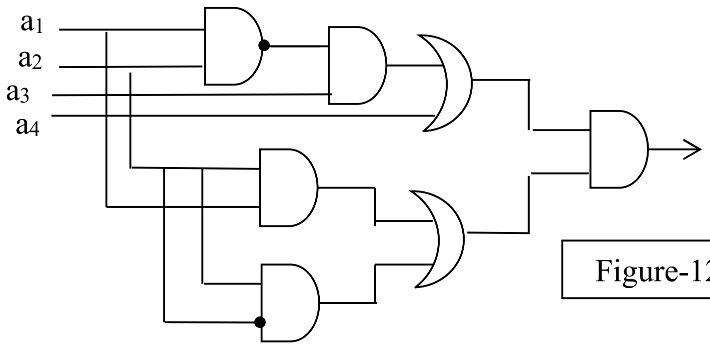


Figure-12

Solution: (i) The polynomial that represents the given gating network is
$p = ((x_1x_2)^1 x_3 + x_4)(x_1x_2 + x_3{}^1 x_4)$.

(ii) By using the Quine-McCluskey algorithm we get a
simplified form $q = x_1x_2x_4 + x_3{}^1x_4$ of p.

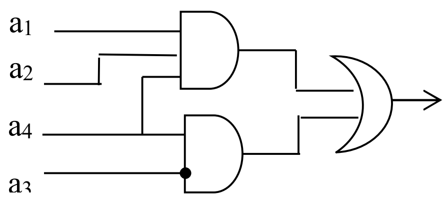


Figure-13

Now, the gating network which represents q is given by the Figure-13.

18.4.10 Note: From the problem 18.4.9, we conclude the following: (i) $p \neq q$, and $p \sim q$

(ii) The gating network for q contains very less number of gates than that of the gating network for p so q (the gating network for q) is much cheaper than q (the gating network for p).

18.4.11 Note: In the following table we present all $2^{2^2} = 16$ Boolean polynomial functions on $B = \{0, 1\}$.

The following table shows the functional values of these polynomial functions.

b_1	b_2	$\overline{p_1}$	$\overline{p_2}$	$\overline{p_3}$	$\overline{p_4}$	$\overline{p_5}$	$\overline{p_6}$	$\overline{p_7}$	$\overline{p_8}$
0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1

$\overline{p_9}$	$\overline{p_{10}}$	$\overline{p_{11}}$	$\overline{p_{12}}$	$\overline{p_{13}}$	$\overline{p_{14}}$	$\overline{p_{15}}$	$\overline{p_{16}}$
1	1	1	1	1	1	1	1
0	0	0	0	1	1	1	1
0	0	1	1	0	0	1	1
0	1	0	1	0	1	0	1

Some polynomials given in this table are important in the algebra of switching circuits. They are given below:

$\overline{p_2}$ … **AND-function**

$\overline{p_3}$ … **inhibit-function**

$\overline{p_7}$ … **antivalence-function**

$\overline{p_8}$ … **OR-function**

$\overline{p_9}$ … **NOR-function**

$\overline{p_{10}}$ … **equivalence-function**

$\overline{p_{14}}$ … **implication-function**

$\overline{p_{15}}$ … **NAND-function**

18.5 SUMMARY:

In this lesson, we have studied an important application of Boolean algebra lies in the realm of electrical engineering. We have illustrated several examples of switching circuits which will be used in the next lesson to understand several applications to use of devices such as mechanical switches, diodes, magnetic dipoles, and transistors are two state devices, etc. We also provide another representation, called gating network of Boolean expressions. For better understanding of the reader, we included examples.

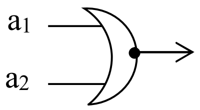
18.6 TECHNICAL TERMS:

1. Complementation switch.
x^1_i is called the complementation switch of x_i.

2. Series connection
$x_i x_j$ is called the series connection of x_i and x_j.

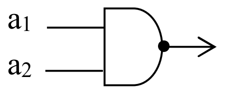
3. Parallel connection
$x_i + x_j$ is called the parallel connection of x_i and x_j.

4. NOR gate



$(a_1 + a_2)^1$ **NOR-gate** or **Pierce-operation**

5. NAND-gate



$(a_1 a_2)^1$ **NAND-gate** or **Sheffer-operation**

18.7 SELF ASSESSMENT QUESTIONS:

1. Draw the switching circuit that represent the given Boolean expression: $x_1 \wedge (x_2 \vee x_3)$.
Ans: (refer: Problem 18.2.3.(i))

2. Draw the switching circuit that represent the given Boolean expression:
$(x_1 \wedge x_2) \vee (x_1 \wedge x_3)$.
Ans: (refer: Problem 18.2.3.(ii))

3. Write down the equivalent Boolean polynomial for the given switching circuit.

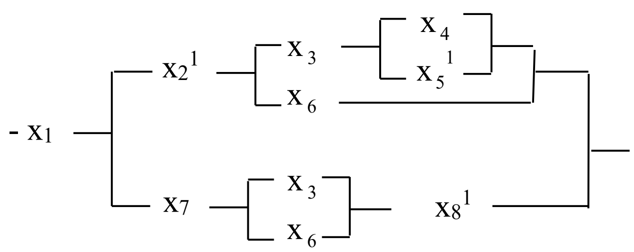


Figure-10

Ans: The required Boolean polynomial is
$q = x_1(x_2^1 (x_6 + x_3 (x_4 + x_5^1)) + x_7(x_3 + x_6)x_8^1)$.

4. Write an algorithm to find a simplified electrical circuit for a given Boolean expression.
Ans: (refer: Note. 18.3.5.)

5. Write down the gating network for the polynomial $p = (x_1{}^1 x_2)^1 + x_3$.

Ans: The required gating network is given by the Figure-11.

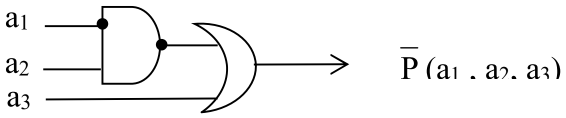


Figure-11

6. (i). Find the polynomial p which corresponds to the gating network given in the Figure-12.

(ii). Find a simplified gating network which operates in the same way as the gating network given in Figure-12.

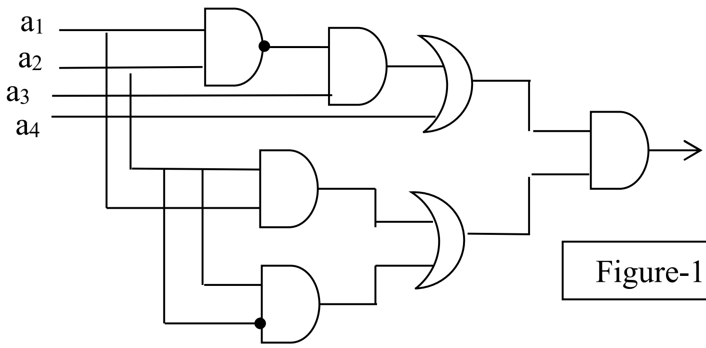


Figure-12

Ans: (i). The polynomial that represents the given gating network is
$p = ((x_1x_2)^1x_3 + x_4)\ (x_1x_2 + x_3{}^1x_4)$.

(ii). By using the Quine-McCluskey algorithm we get a simplified form $q = x_1x_2x_4 + x_3{}^1x_4$ of p.

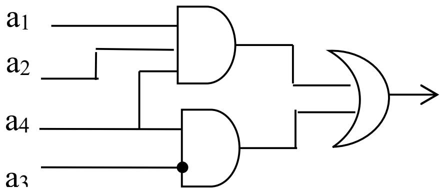


Figure-13

Now, the gating network which represents q is given by the Figure-13.

18.8 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Harikrishnan Panackal

LESSON-19
SOME APPLICATIONS

OBJECTIVE :

- To know Half Adder.
- To Understand the Applications of switching circuits
- To know the Full Adder.
- To have proper understanding of Applications.
- To develop skills to construct gatting networks.

STRUCTURE:

19.1. INTRODUCTION:

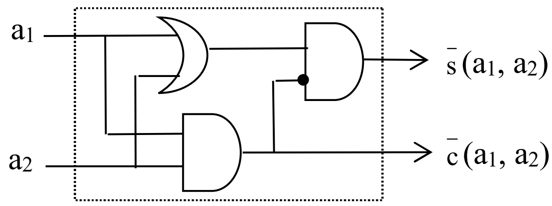
In this lesson, we study various practical applications of switching circuits or networks. Among different approaches of expressing Boolean expressions, gating network and switching circuits have several applications in science, engineering and technology;.

19.2 HALF-ADDER AND FULL-ADDER:

In this section, we study two important gatting networks namely Half Adder, and Full Adder.

19.2.1 Note: Consider the polynomials $s = (x_1x_2)^1.(x_1 + x_2)$, and $c = x_1x_2$.

(i) Now we write a gating network (refer Gating Network-1) whose input is x_1, x_2 and the output is the values of the expressions s and c at the given values $x_1 = a_1$ and $x_2 = a_2$.



Gating Network-1

(ii) Observe Gating Network-1. If a_1, a_2 is the input to the Gating Network-1, then the output is the values of $\bar{s}(a_1, a_2)$ and $\bar{c}(a_1, a_2)$ for all $a_1, a_2 \in \{0, 1\}$.

(iii) The Gating Network-1 is called half-adder.

(iv) The gating network related to half-adder is denoted by **HA**

19.2.2 Problem: Let $a_1, a_2 \in \{0, 1\}$.

(i) Find out two polynomials p and c which represents the units digit and the 2's digit (respectively) (we may call this 2's digit as carry) of the binary sum $a_1 + a_2$.

(ii) Write down the gating network to get polynomials p and c from the given input a_1, a_2.

(iii) Find a simpler gating network to get p and c from the given input a_1, a_2. (If possible, use the half-adder).

Solution: (i) Suppose we add two single digit binary numbers a_1 and a_2. Suppose p is the units digit of $a_1 + a_2$, and c is the 2's digit of $a_1 + a_2$.

For example, if $a_1 = a_2 = 1$, then the sum is 10, and so $\bar{p}(a_1, a_2) = 0$, $\bar{c}(a_1, a_2) = 1$.

The functional values for $\bar{p}(a_1, a_2)$ and $\bar{c}(a_1, a_2)$ is given by the table-1.

Observe table-1. Now by using table-1 and block box method, we get the following disjunctive normal forms for p and c.

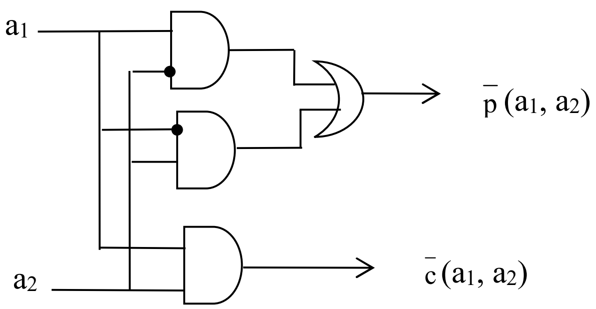
$p = x_1x_2^1 + x_1^1x_2$

$c = x_1x_2$.

a_1	a_2	$\bar{p}(a_1, a_2)$	$\bar{c}(a_1, a_2)$
1	1	0	1
1	0	1	0
0	1	1	0
0	0	0	0

Table-1

(ii) Now we draw the gating network whose input is a_1, a_2 and output is $\bar{p}(a_1, a_2)$ and $\bar{c}(a_1, a_2)$. This was given in the following gating network-2.



Gating Network-2

(iii) To find a simpler network for the gating network-2, we have to modify the expression for p by using the laws of Boolean algebra.
The expression for c is already in simplest form.

Now $p = x_1x_2^1 + (x_1^1x_2) = (x_1^1 + x_2)^1 + (x_1 + x_2^1)^1$ (by Demorgan's laws)

$= ((x_1^1 + x_2)(x_1^1 + x_2^1))^1$ (by Demorgan's laws)

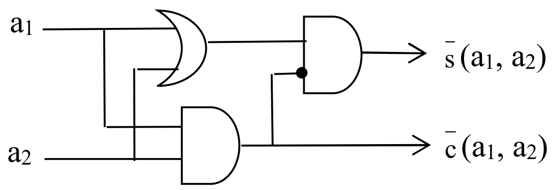
$= (x_1^1x_1 + x_2x_1 + x_1^1x_2^1 + x_2x_2^1)^1$ (by distributive law)

$= (x_1x_2 + x_1^1x_2^1)^1$ (by complement laws) $= (x_1x_2)^1 (x_1^1x_2^1)^1$ (by Demorgan's laws)

$= (x_1x_2)^1.(x_1 + x_2)$ (by Demorgan's laws)

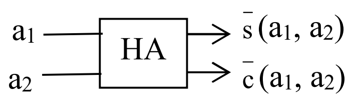
Write $s = (x_1x_2)^1.(x_1 + x_2)$. Now $p \sim s$.

The gating network for s and c is given in gating network-3.



Gating Network-3

It is clear that gating network-3 is a simpler network than the gating network-2.
By using the half-adder, we can represent the gating network-3 as follows:



19.2.3 Note: A gating network called as Full-adder can add three one-digit binary numbers. Let a_1, a_2, a_3 denote the three numbers to be added.
Suppose s denotes the units digit and c denotes the 2's digit of the sum $a_1 + a_2 + a_3$.

(i). The functional values are given in table-2.
The disjunctive normal form of the polynomials s and c are given below:

$s = x_1x_2x_3 + x_1x_2^1x_3^1 + x_1^1x_2x_3^1 + x_1^1x_2^1x_3$.

$c = x_1x_2x_3 + x_1x_2x_3^1 + x_1x_2^1x_3 + x_1^1x_2x_3$.

a_1	a_2	a_3	$\bar{s}(a_1, a_2, a_3)$	$\bar{c}(a_1, a_2, a_3)$
1	1	1	1	1
1	1	0	0	1
1	0	1	0	1
1	0	0	1	0
0	1	1	0	1
0	1	0	1	0
0	0	1	1	0
0	0	0	0	0

Table-2

(ii). Suppose s_1 denotes the units digit and c_1 denotes the 2's digit of $a_2 + a_3$. Then the functional values are given in Table-3.

a_2	a_3	$\bar{s}_1(a_2, a_3)$	$\bar{c}_1(a_2, a_3)$
1	1	0	1
1	0	1	0
0	1	1	0
0	0	0	0
1	1	0	1
1	0	1	0
0	1	1	0
0	0	0	0

Table-3

(iii) Suppose c_2 is the 2's digit of the sum $a_1 + \bar{s}_1(a_2, a_3)$. Then the functional values are given in Table-4.

a_1	$\bar{s}_1(a_2, a_3)$	$\bar{s}(a_1, a_2, a_3)$	$\bar{c}_2(a_1, \bar{s}_1(a_2, a_3))$
1	0	1	0
1	1	0	1
1	1	0	1
1	0	1	0
0	0	0	0
0	1	1	0
0	1	1	0
0	0	0	0

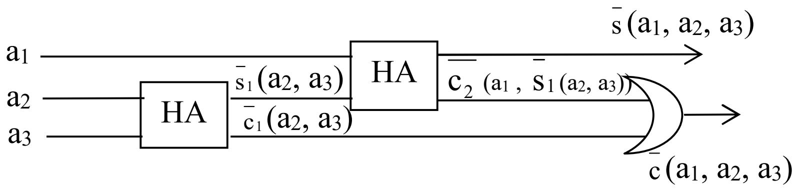
Table-4

(iv) The relation between c_1, c_2 and c presented in Table-5.

$\bar{c}_1(a_2, a_3)$	$\overline{c_2}(a_1, \bar{s}_1(a_2, a_3))$	$\bar{c}_2(a_1, a_2, a_3))$
1	0	1
0	1	1
0	1	1
0	0	0
1	0	1
0	0	0
0	0	0
0	0	0

Table-5

(v) Consider the gating network-4. Here we used two half-adders.



Gating Network-4

Now we understand that a_2 and a_3 are inputs of a half-adder with outputs $\bar{s}_1(a_2, a_3)$ and $\bar{c}_1(a_2, a_3)$.

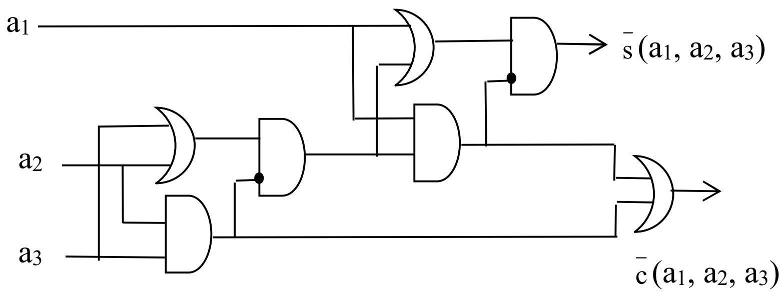
The output $\bar{s}_1(a_2, a_3)$ together with a_1 forms inputs of a second half-adder, whose outputs are $\bar{s}(a_1, a_2, a_3)$ and $\bar{c}_2(a_1, \bar{s}_1(a_2, a_3))$.

Hence $\bar{s}(a_1, a_2, a_3)$ is the final sum $a_1 + a_2 + a_3$.

Finally, the sum of $\bar{c}_1(a_2, a_3)$ and $\bar{c}_2(a_1, \bar{s}_1(a_2, a_3))$ gives $\bar{c}(a_1, a_2, a_3)$.

This gating network is called the full-adder.

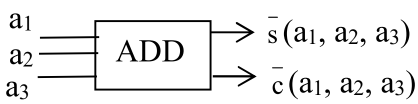
So a full-adder is composed of two half-adders and one OR-gate as shown in the gating network-4.



Gating Network-5

Observe that gating networks 4 and 5 are same.

A symbolic representation of the full adder is given by



19.2.4 Problem: Find a polynomial p satisfying the following conditions:

$p = x_2x_3$ if $x_1 = 0$

$p = x_2+x_3$ if $x_1 = 1$.

Solution: Suppose $a_1, a_2, a_3 \in \{0, 1\}$. The functional values are given in table-6.

a_1	a_2	a_3	$\bar{p}(a_1, a_2, a_3)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Table-6

By table-6 and block box method, we get the disjunctive normal form for p.

The form is $p = x_1^1x_2x_3 + x_1x_2^1x_3 + x_1x_2x_3^1 + x_1x_2x_3$.

19.2.5 Example: (Chakrabarti's Cell): Suppose $k = 2$ and $n = 5$. We want to find a polynomial p with the following conditions:

$\bar{p}(a_1, a_2, a_3, a_4, a_5) = \text{NOR}(a_3, a_4, a_5)$ if $(a_1, a_2) = (0, 0)$;

$\bar{p}(a_1, a_2, a_3, a_4, a_5) = \text{OR}(a_3, a_4, a_5)$ if $(a_1, a_2) = (0, 1)$;

$\bar{p}(a_1, a_2, a_3, a_4, a_5) = \text{NAND}(a_3, a_4, a_5)$ if $(a_1, a_2) = (1, 0)$;

$\bar{p}(a_1, a_2, a_3, a_4, a_5) = \text{AND}(a_3, a_4, a_5)$ if $(a_1, a_2) = (1, 1)$.

The polynomial p satisfying these conditions is given by

$p = x_2^1x_3^1x_4^1x_5^1 + x_1x_2^1x_3^1 + x_1x_2^1x_4^1 + x_1x_2^1x_5^1 + x_1^1x_2x_3 + x_1^1x_2x_4 + x_1^1x_2x_5 + x_2x_3x_4x_5$.

19.3 SOME APPLICATIONS:

In this section, we study some applications of switching circuits related to central lighting system in a big room; paper movements and the related control mechanism in fast printers of computers, and in the machines for paper production; and Elivator Services, etc .,

19.3.1 Some Applications:

Suppose there is a big room with central lighting system. To operate this central lighting system there are switches at three different places near by the respective entrance doors. These three switches operate alternatively. That is, each of these three switches can "switch on" or "switch off" the lighting system.

(i) We wish to determine the related switching circuit p, and its symbolic polynomial representation.
Each switch got two states: "on" or "off". We denote these three switches by x_1, x_2, x_3 and the two possible states of the switches x_i by $a_i \in \{0, 1\}$.

The light situation (whether on or off) in the room is given by the value

$\bar{p}(a_1, a_2, a_3) = 0$ (or 1) if the lights are off (or on), respectively.

We suppose that if all the three switches x_1, x_2, x_3 are in the state "on" (that is, the value of all the variables x_1, x_2, x_3 is equal to 1) then the central lighting system is on (that is, the value of $p = 1$).

So we write this situation as: $\bar{p}(1, 1, 1) = 1$.

If we operate any one of the three switches, then the lights go off, that is we have $\bar{p}(a_1, a_2, a_3) = 0$ for all (a_1, a_2, a_3) which differ in one place from $(1, 1, 1)$.

Similarly we have that $\bar{p}(a_1, a_2, a_3) = 0$ for all (a_1, a_2, a_3) which differ in three places from $(1, 1, 1)$. In other words, $\bar{p}(a_1, a_2, a_3) = 0$ if $(a_1, a_2, a_3) = (0, 0, 0)$.

(ii) Suppose the lights are in the state on. Then if we operate any two switches, the lights still stay on.

That is, we have that $\bar{p}(a_1, a_2, a_3) = 1$ for all those (a_1, a_2, a_3) which differ in two places from $(1, 1, 1)$.

(iii) If the polynomial p satisfies the said set of conditions then we get the table-7 which provides the function values.

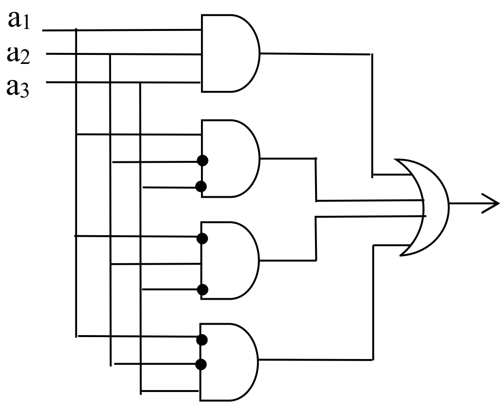
a_1	a_2	a_3	minterms	$\bar{p}(a_1, a_2, a_3)$
1	1	1	$x_1x_2x_3$	1
1	1	0	$x_1x_2x_3^1$	0
1	0	1	$x_1x_2^1x_3$	0
1	0	0	$x_1x_2^1x_3^1$	1
0	1	1	$x_1^1x_2x_3$	0
0	1	0	$x_1^1x_2x_3^1$	1
0	0	1	$x_1^1x_2^1x_3$	1
0	0	0	$x_1^1x_2^1x_3^1$	0

Table-7

From this table-7, we can get the following disjunctive normal form for p:

$p = x_1x_2x_3 + x_1x_2^1x_3^1 + x_1^1x_2x_3^1 + x_1^1x_2^1x_3$.

(iv) The gating network for p is given by the gating network-6.



Gating Network-6

Note that the above polynomial expression for p is already in minimal form.

(v) Observe that

$$p = x_1x_2x_3 + x_1x_2^1x_3^1 + x_1^1x_2x_3^1 + x_1^1x_2^1x_3.$$

$$\sim \; x_1(x_2x_3 + x_2^1x_3^1) + x_1^1(x_2x_3^1 + x_2^1x_3) = q$$

(vi) The switching circuit diagram for q was given in the Figure-1.

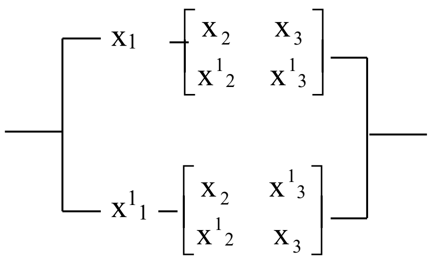


Figure-1

19.3.2 Example: In the fast printers of computers, and in the machines for paper production, a careful control of the paper movements is essential. We present a diagram (please see the diagram-2) which shows a schematic model of the method of paper movements and the related control mechanism.

The motor operates a pair of cylinders (1), which helps the movement of paper (2).
Due to this paper strip (2) the light from lamp (3) can not be fall on the photo cell (4).
If the paper strip breaks, then the light from lamp (3) be fall on the photo cell (4).
Since the photo cell (4) receives the light and passes on an impulse which switch off the motor.

(i) The light in lamp (3) can vary its brightness or it can fail. So an another photo cell (5) supervises the brightness of the lamp (3).
The work of lamp is satisfactory if its brightness is above a fixed given value a.
If the brightness is below a, and above a minimum value b, then the situation is indicated by the warning lamp (6). In this case, the paper movement mechanism still operates as it is.
If the brightness of the lamp is below b, then the photo cell (4) cannot work satisfactorily, and so the motor is switched off.

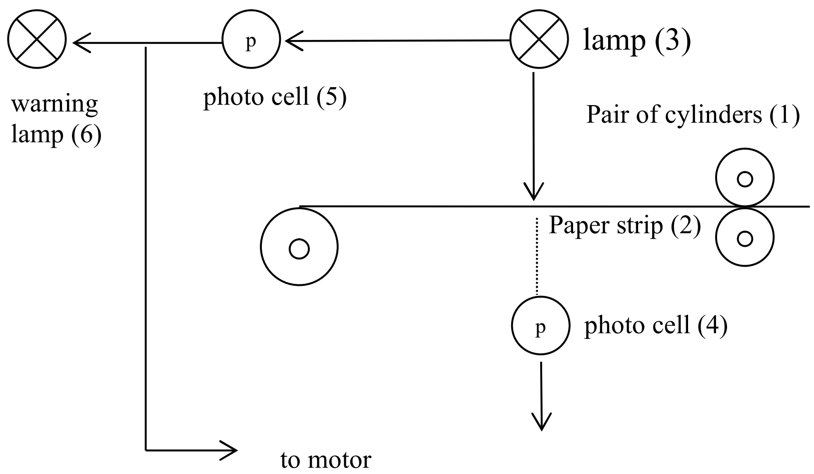


Diagram-2

(ii) Now we represents this situation in mathematical symbols as follows:

$a_1 = 1$ if the brightness of (3) $>$ a;

$a_1 = 0$ if the brightness of (3) $\leq$ a;

$a_2 = 1$ if the brightness of (3) $>$ b;

$a_2 = 0$ if the brightness of (3) $\leq$ b;

$a_3 = 1$ if the paper strip is broken;

$a_3 = 0$ if the paper strip is unbroken.

Note that $b < a$.

(iii) Suppose $\bar{p}_1(a_1, a_2, a_3)$ is the Boolean function related to the state of the motor, and $\bar{p}_2(a_1, a_2, a_3)$ is the Boolean function related to the state of warning lamp.

Now we define

$\bar{p}_1(a_1, a_2, a_3) = 1 \Leftrightarrow$ motor operates;

$\bar{p}_1(a_1, a_2, a_3) = 0 \Leftrightarrow$ motor is switched off;

$\bar{p}_2(a_1, a_2, a_3) = 1 \Leftrightarrow$ warning lamp (6) operates;

$\bar{p}_2(a_1, a_2, a_3) = 0 \Leftrightarrow$ warning lamp (6) does not operate.

(iv) The values of the functions $\bar{p}_1(a_1, a_2, a_3)$, and $\bar{p}_2(a_1, a_2, a_3)$ were presented in the table-8.

a_1	a_2	a_3	$\overline{p}_1(a_1, a_2, a_3)$	$\overline{p}_2(a_1, a_2, a_3)$
1	1	1	0	0
1	1	0	1	0
~~1~~	~~0~~	~~1~~		
~~1~~	~~0~~	~~0~~		
0	1	1	0	1
0	1	0	1	1
0	0	1	0	0
0	0	0	0	0

Table-8

Observe that the case $a_1 = 1, a_2 = 0$ cannot occur.

That is, the case $a_1 = 1, a_2 = 0$ is an impossible case.

At this situation, we may assign arbitrary values for $\overline{p_1}$ and $\overline{p_2}$ in the table-8 (Don't-care combinations).

Now in the impossible cases, we assign 0 for $\overline{p_1}$ and $\overline{p_2}$ (as don't care combinations)

(v) From table-8 we get the following disjunctive normal forms for p_1 and p_2.

$p_1 = x_1x_2x_3^1 + x_1^1x_2x_3^1 \sim x_2x_3^1$.

$p_2 = x_1^1x_2x_3 + x_1^1x_2x_3^1 \sim x_1^1x_2$.

(vi) From the above point (v), we conclude that the state of motor (p_1) is not depend on a_1.
Also it is clear that the state of warning lamp (p_2) is not depend on a_2.

(vii) The gating network for the functions $\overline{p_1}$ and $\overline{p_2}$ given in the diagram-3.

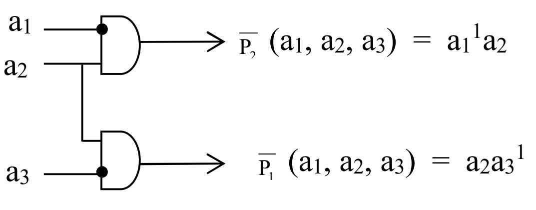


Diagram-3

19.3.3 Example: A motor is supplied by three generators. The operation of each generator is monitored by a corresponding switching element which closes a circuit as soon as a generator fails. We demand the following conditions from the electrical monitoring system:

(i) A warning lamp lights up if one or two generators fail.
(ii) An acoustic alarm is initiated if two or all three generators fail.

We determine a symbolic representation as a mathematical model of this problem. Let $a_1 = 0$ denote that generator i is operating, $i \in \{1, 2, 3\}$; $a_1 = 1$ denotes that generator i does not operate. The table of function values has two parts $\bar{p}_1(a_1, a_2, a_3)$ and $\bar{p}_2(a_1, a_2, a_3)$, defined by:

$\bar{p}_1(a_1, a_2, a_3) = 1$: acoustic alarm sounds;

$\bar{p}_1(a_1, a_2, a_3) = 0$: acoustic alarm does not sounds;

$\bar{p}_2(a_1, a_2, a_3) = 1$: warning lamp lights up;

$\bar{p}_2(a_1, a_2, a_3) = 0$: warning lamp is not lit up

Then we obtain the following table for the function values:

a_1	a_2	a_3	$\bar{p}_1(a_1, a_2, a_3)$	$\bar{p}_2(a_1, a_2, a_3)$
1	1	1	1	0
1	1	0	1	1
1	0	1	1	1
1	0	0	0	1
0	1	1	1	1
0	1	0	0	1
0	0	1	1	0
0	0	0	0	0

For p_1 we choose the disjunctive normal form, namely

$p_1 = x_1x_2x_3 + x_1 x_2x^1{}_3 + x_1x^1{}_2x_3 + x^1{}_1 x_2x_3$.

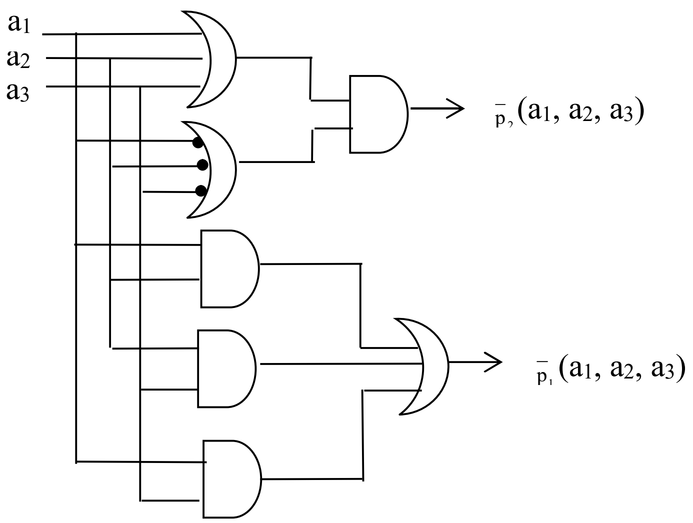
This can be simplified by using rules of a Boolean algebra:

$p \sim x_1x_2 + x_2x_3 + x_1x_3$.

For p_2 we choose the conjunctive normal form, which is preferable if there are many 1's as function values:

$p_2 = (x_1 + x_2 + x_3)(x^1{}_1x^1{}_2x^1{}_3)$.

The symbolic representation is



One of the applications of Boolean algebras in the simplification of electromechanical or electronic switching circuits. In order to economize, it is often useful to construct switching circuits in such a way that the costs for their technical realization are as small as possible, example that a minimal number of gates is used. Unfortunately, it is often difficult to decide from the diagram of a switching circuit whether its technical implementation is simple. Also, the simplest and most economical switching circuit may not necessarily be a series-parallel connection, in which case switching algebra is not of much help. Some methods of simplification other than the Quine-McCluskey algorithms are discussed in Dornhoff and Hohn (1978) and also in Hohn (1970).

19.3.4. Remark: A switching circuit p can be simplified by our methods, as follows:

(i) It can be simplified according to the laws of a Boolean algebra (example, by applying the distributive, idempotent, absorption, and de Morgan laws).

(ii) Sometimes calculating the dual $d(p)$ of p and simplifying the dual yields a simple expression.

(iii) We can also determine the minimal form of p by using the method of Quine and McCluskey. Recall that this algorithm can only be started if p is in disjunctive normal form.

(iv) Use Karnaugh diagrams.

19.3.5 Example: We give an example for the first two methods mentioned in 19.3.4.

(i) $p = (x_1^1 + x_2 + x_3 + x_4)(x_1^1 + x_2 + x_3 + x_4^1)(x_1^1 + x_2^1 + x_3 + x_4^1).$

$\sim (x_1^1 + x_2 + x_3)(x_1^1 + x_3 + x_4^1) \sim x_1^1 + x_3 + x_3x_4^1$

Here, we have used the fact $(\alpha + \beta)(\alpha + \beta^1) = \alpha$ twice.

(ii) $p = ((x_1 + x_2)(x_1 + x_3)) + (x_1x_2x_3)$

$$\sim \underbrace{((x_1 + x_2) + (x_1x_2x_3))}_{=:p_1} \underbrace{((x_1 + x_3) + (x_1x_2x_3))}_{=:p_2}.$$

Let d denote "dual of".

We have $d(p_1) = (x_1x_2)(x_1 + x_2 + x_3) \sim x_1x_2$. Therefore $d(d(p_1)) \sim x_1 + x_2$.

Also, $(x_1x_3)(x_1 + x_2 + x_3) \sim x_1x_3$. Thus $d(d(p_2)) \sim x_1 + x_3$. Altogether we have

$p \sim p_1p_2 \sim (x_1x_3)(x_1 + x_2 + x_3) \sim x_1 + x_2x_3.$

We consider two more examples of applications (due to Dokter and Steinhauer (1972)).

19.3.6 Example: An elevator services three floors. On each floor there is a call-button C to call the elevator. It is assumed that at the moment of call the cabin is stationary at one of the three floors.

Using these six input variables we want to determine a control which moves the motor M in the right direction for the current situation.

One, two, or three call-buttons may be pressed simultaneously; so there are eight possible combinations of calls, the cabin being at one of the three floors. Thus we have to consider $8 \cdot 3 = 24$ combinations of the total $2^6 = 64$ input variables.

We use the following notations: $a_1 := c_i$ (for $i = 1, 2, 3$) for the call-signals. $c_i = 0$ (or 1) indicates that no call (or a call) comes from floor $i \cdot a_4 := f_1$, $a_5 := f_2$, $a_6 := f_3$ are position signals; $f_i = 1$ means the elevator cabin is on floor $i \cdot \overline{p_1}(a_1, ..., a_6) =: M\uparrow$, $\overline{p_1}(a_1, ..., a_6) =: M\uparrow$ indicate the direction of movement to be given to the motor; then the signal $M\uparrow = 1$ means movement of the motor upward, etc. The output signals (function values) of the motor does not operate.

If a call comes from the floor where the cabin is at present, again the motor does not operate. Otherwise, the motor follows the direction of the call.

The only exception is the case when the cabin is at the second floor and there are two simultaneous calls from the third and first floor. We agree that the cabin goes down first.
Figure 8.4 shows the table of function values.

From this table we derive the switching circuits p_1 for $M\uparrow$ and p_2 for $M\downarrow$ in disjunctive normal form.
Here x_i are replaced by C_i for $i = 1, 2, 3$ and by F_{i-3} for $i = 4, 5, 6$.

$$P_1 \sim C^1_1C_2C_3F_1F^1_2F^1_3 + C^1_1C_2C^1_3F_1F^1_2F^1_3 + C^1_1C^1_2C_3F_1F^1_2F^1_3 + C^1_1C^1_2C_3F^1_1F_2F^1_3.$$

The first and third minterms are complementary with respect to C_2 and can be combined. This gives:

$$P_1 \sim C^1_1C_2C^1_3F_1F^1_2F^1_3 + C^1_1C_3F_1F^1_2F^1_3 + C^1_1C^1_2C_3F^1_1F_2F^1_3 .$$

For $M\uparrow$ we obtain

$$P_2 \sim C_1C^1_2C_3F^1_1F_2F^1_3 + C_1C^1_2C^1_3F^1_1F_2F^1_3 + C_1C_2C^1_3F^1_1F^1_2F_3$$
$$+ C_1C^1_2C^1_3F^1_1F^1_2F_3 + C^1_1C_2C^1_3F^1_1F^1_2F_3.$$

Call			Floor			Direction of motor	
c_1	c_2	c_3	f_1	f_2	f_3	M ↑	M ↓
1	1	1	1	0	0	0	0
1	1	0	1	0	0	0	0
1	0	1	1	0	0	0	0
1	0	0	1	0	0	0	0
0	1	1	1	0	0	1	0
0	1	0	1	0	0	1	0
0	0	1	1	0	0	1	0
0	0	0	1	0	0	0	0
1	1	1	0	1	0	0	0
1	1	0	0	1	0	0	0
1	0	1	0	1	0	0	1
1	0	0	0	1	0	0	1
0	1	1	0	1	0	0	0
0	1	0	0	1	0	0	0
0	0	1	0	1	0	1	0
0	0	0	0	1	0	0	0
1	1	1	0	0	1	0	0
1	1	0	0	0	1	0	1
1	0	1	0	0	1	0	0
1	0	0	0	0	1	0	1
0	1	1	0	0	1	0	0
0	1	0	0	0	1	0	1
0	0	1	0	0	1	0	0
0	0	0	0	0	1	0	0

The first two minterms are complementary with respect to C_3, the third and fourth minterm are complementary with respect to C_2. Simplification gives

$P_2 \sim C_1C^1_2F^1_1F_2F^1_3 + C_1C^1_3F^1_1F^1_2F_3 + C^1_1C_2C^1_3F^1_1F^1_2F_3$.

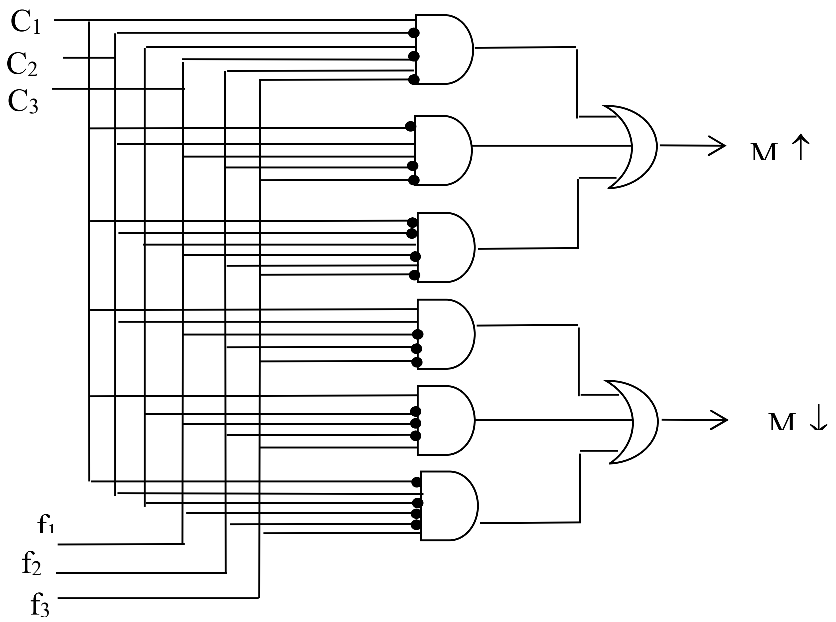


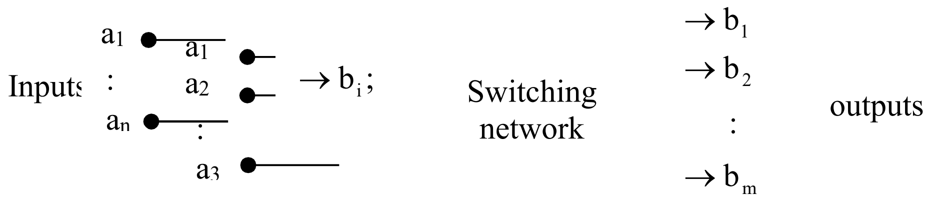
Figure 2

The two switching circuits enable us to design the symbolic representation of Figure 2 (we have six NOT-gates, AND-gates, and two OR-gates).

Observe that in above three Examples we had not only a switching circuit, but a switching network, which differ from a circuit by having multiple outputs:

Optimizing such a network reduces to the minimization of all circuits

We have precisely done that in these examples.

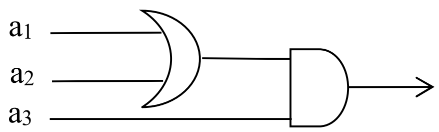


As another example of applications of this type we consider the addition of binary numbers with half-adders and adders. Decimals can be represented in terms of quadruples of binary numbers; such a quadruple is called a tetrad.

Each digit of a decimal gets assigned a tetrad; thus we use then different tetrads corresponding to

0, 1, 2, …, 9. Using four binary positions we can form $2^4 = 16$ tetrads. Since we need only ten tetrads, which are called pseudotetrads.

A binary coded decimal then uses the following association between 0, 1, …, 9 and tetrads:



$\overline{p}_1(a_1, a_2, a_3) = 1$ denotes the pseudotetrads. We have to evaluate $\overline{p}_1(a_1, a_2, a_3)$ to find out if the result of a computing operation is a psedotetrad.

	a_3	a_2	a_1	a_0	$\overline{p}_0(a_1, a_2, a_3)$
	1	1	1	1	1
	1	1	1	0	1
	1	1	0	1	1
	1	1	0	0	1
	1	0	1	1	1
	1	0	1	0	0
9	1	0	0	1	0
8	1	0	0	0	0
7	0	1	1	1	0
6	0	1	1	0	0
5	0	1	0	1	0
4	0	1	0	0	0
3	0	0	1	1	0
2	0	0	1	0	0
1	0	0	0	1	0
0	0	0	0	0	0

We represent p in disjunctive normal form:

$p = x_3x_2x_1x_0 + x_3x_2x_1x^1_0 + x_3x_2x^1_1x^1_0 + x_3x^1_2x_1x_0 + x_3x^1_2x_1x^1_0.$

The pairs of minterms 1 and 2, 3 and 6 are complementary with respect to x_0 and can be simplified:

$p \sim x_3x_2x_1 + x_3x_2x^1_1 + x_3x^1_2x_1$

$\sim x_3x_2x_1 + x_3x_2x_1 + x_3x_2x^1_1 + x_3x^1_2x_1$

$\sim (x_3x_2x_1 + x_3x_2x^1_1) + (x_3x_2x_1 + x_3x^1_2x_1)$

$\sim x_3x_2 + x_3x_1 \sim x_3(x_2 + x_1).$

This result indicates that determining if a tetrad with four positions a_0, a1, a_2, a_3 is a pseudotetrad is independent of a_0. If we use the a_i as inputs, then Figure 8.6 indicates the occurrence of a pseudotetrad.

19.4 SUMMARY:

In this lesson, we studied two important gatting networks namely: Half adder, and Full adder, also presented their diagrams. we have described some applications through examples using Boolean expressions and gating networks. These are having useful applications in electrical engineering and network engineering. In particular, in the last section, we study some applications of switching circuits related to central lighting system in a big room; paper movements and the related control mechanism in fast printers of computers, and in the machines for paper production; and Elivator Services, etc .,

19.5 TECHNICAL TERMS:

1. Half-adder.
(refer: Example. 19.2.1.)

2. Full-adder.
A gating network called as Full-adder can add three one-digit binary numbers.
(refer: Note 19.2.3.)

3. Chakrabarti's Cell:
(refer: Example 19.2.5.)

19.6 SELF ASSESSMENT QUESTIONS:

1. Draw the diagram of Half Adder.
Ans: (refer: Example. 19.2.1.)

2. What the Full-adder can do. Draw the diagram of Full adder.
Ans: A gating network called as Full-adder can add three one-digit binary numbers.
(refer: Note 19.2.3.)

3. What do you mean by Chakrabarti's Cell.
Ans: (refer: Example 19.2.5.)

19.7 SUGGESTED READINGS:

1. Bhavanari Satyanarayana and Kuncham Syam Prasad, Discrete Mathematics & Graph Theory, Prentice Hall India Ltd., New Delhi, 2014 (second edition) ISBN-978-81-203-4948-3.

2. James L. Fisher, Application Oriented Algebra (second edition) UTM, Springer, 1977.

3. Bhavanari Satyanarayana, T.V.P. Kumar and SK Mohiddin Shaw, Mathematical Foundations of Computer Science, CRC Press, London, 2019, e-ISBN-9780367367237.

4. R. Lidl and G. Pilz, Applied Abstract Algebra, second edition, UTM Springer, 1998.

Prof. Dr. Harikrishnan Panackal