Title	Discrete Mathematics
Author	
Prof. Abhay Saxena Dean (School of Technology, Management & Communication) and Head-Computer Science Department, Dev Sanskriti Vishwaviidyalya, Hardwar, India	
Editor	
Dr. Raj Kishor Bisht Associate Professor- Department of Mathematics Graphic Era Hill University, Dehradun, India	
ISBN: 978-93-84813-97-0	
License	
The course on Discrete Mathematics is released under Creative Commons Attribution-NonCommercial-ShareAlike - CC-BY-NC-SA license.	

Table of Contents

Block I: Unit I Set Theory

Learning Objectives
After completing this unit, the learner shall be able to:

- Explain the definition on Sets;
- Differentiate between various types of Sets;
- Explain relation between Sets;
- Perform various operations on Sets;
- Define Venn diagram;
- Calculate Cartesian product;
- Create a Power set using the given subsets;
- Explain counting principles;
- Represent Tree diagrams;
- Calculate Cardinality and Countability of Set;
- Define basic Set Identities and Proofs;
- Explain Pigeonhole Principle;

Set is the fundamental discrete structure upon which all other discrete structures are built. The notion of sets was introduced by German Mathematician George Cantor (1845 - 1918) for a better understanding of finite sequences. Simply set in mathematics means exactly what it means in ordinary language, namely, a collection of objects. A battalion of soldiers, a ream of papers, the alphabet, all are examples of sets.

1.1 Definition of Sets

A set can be defined as a collection of things that are brought together because they obey a certain rule. These 'things' may be anything like: numbers, people, shapes, cities etc. A fundamental concept of set theory is that of membership or belonging to a set. Some key points to be keep in mind:

- A set should be well-defined means its objects should follow certain rule
- Elements of a set should be distinguishable. Repetition will not make any difference.
- Order of elements in a set is immaterial i.e. we can take any sequence of elements in a set.

1.1.1 Notation

We use uppercase letters: $A, B, C, \ldots$ to denote a set and elements are denoted in lowercase letter: $x, y, z, p, q, r \ldots$

Elements of the set can be listed or definition of the variables can be given using a variable. For instance,

The list of elements in a set can be represented as,

$$X = \{1,3,5,7,9\}$$

In order to describe the elements of the set we can write it as,

$$X = \{x: x = 2n - 1, 1 \leq n \leq 5, n \; is \; an \; integer\}$$

we can use an *identifier* ($'x'$ for example) to denote a *representative element* , while a ':' symbol means 'such that' and then the rule that the identifier should obey:

$$\{x : x \; is \; an \; integer \; and \; |x| < 5\}$$

or

$$\{x : x \in Z, |x| < 5\}$$

The latest way to write a set known as *comprehension* notation - can be expressed in standard form as:

$x \mid P(x)$, where $P(x)$is a statement states that the set comprised of all elements $'x'$ for which P is true.

The symbol ∈ is used as follows:

- ∈ Stands for 'is an element of. . . . For Example: $snake \in Set \; of \; Reptiles$
- ∉ Stands for 'is not an element of . . . For example: $New \; Delhi \notin Set \; of \; African \; capital \; cities$

A set can be

i. Finite: for example, $set \; of \; Indian \; Citizens$
ii. Infinite: for example, $\{3, 9, 12, 15, 18, \ldots\}$

(*Note: the use of the ellipsis ......indicates that sequence of numbers is endless*).

Always, remember one thing that the order in which the elements in a set occur is immaterial.

1.2 Types of Sets

Universal set: The collection of all elements under consideration is said to be Universal set and it is denoted by U. For example, the universal set may be {alphabetic characters} or {all living people} etc.

Empty set: A set with no elements is called an empty or null set and generally denoted by ϕ. For example, Let a set $A = \{a, b, c\}$ is given then $\phi \subseteq A$. But $\phi \notin A$. Thus, the inclusion of an element is the membership of an element in a set.

Other examples,

Example 1.1

- If $U = Set \; of \; words \; in \; the \; English \; language$ then $Set \; of \; words \; with \; more \; than \; 50 \; letters = \phi$
- If $U = Set \; of \; whole \; numbers$ then $\{x \mid x^2 = 10\} = \phi$

1.3 Relations between Sets

There are many ways in which sets may be related to one another.

Equity

Two sets are said to be equal if they contain the same elements i.e. if every element of B is also in A, & every element of A is also in B, then we say A = B. e.g. if $A = \{1,2,3\}$ and $B = \{1,3,2\}$ $then$ $A = B$.Two sets A & B are set to be equal, iff $A \subseteq B$ and $B \subseteq A$ or symbolically,

$$A = B \Leftrightarrow (A \subseteq B \text{ } and \text{ } B \subseteq A)$$

Example 1.2

$\{1,2,4\} = \{1,2,2,4\};$

$\{1,4,2\} = \{1,2,4\};$

$\{1,3,5, \ldots\} = \{x: x \text{ } is \text{ } an \text{ } odd \text{ } positive \text{ } integer\}$

Subsets

It can easily be imagine a set within a set. The contained set is called a subset of the containing set. If the set A is a subset of B, we write:

$$A \subseteq B$$

Example 1.3

- ✓ The set of people living in Delhi is subset of the set of people living in India
- ✓ The set of organic compounds is the subset of chemical compounds.
- ✓ Set A is called a proper subset of a set $\boldsymbol{B}$if $A \subseteq B$ and $A \neq B$. Symbolically it is written as $\subset B$. It is also called proper inclusion. A proper inclusion is not reflexive but it is transitive i.e. $(A \subset B)$ and $(B \subset C) \Rightarrow (A \subset C)$

Note:

- ✓ *Every* set is a subset of the *universal set*, and the *empty set* is a subset of *every* set. i.e. for every set A, the empty set does not have anything that isn't in A . So for all sets A $(\phi \subseteq A)$
- ✓ If A is a subset of B i.e.$(A \subseteq B)$ and B is a subset of A i.e.$(B \subseteq A)$ then A and B should comprises of exactly similar elements, and hence they are equal. In further terms: If $(A \subseteq B)$ and $(B \subseteq A)$ then $(A = B)$

Disjoint

Disjoint sets can be defined as the sets with different elements with respect to each other.

For Instance:

If $A = Set \text{ } of \text{ } even \text{ } numbers \text{ } and \text{ } B = \{1, 3, 5, 11, 19\}$ then A and B are disjoint sets.

1.4 Operations on Sets

Intersection

The intersection of any two sets A and B, written as $A \cap B$, is the set consisting of all the elements which belong to both A and B i.e.

$$(A \cap \ B) = \{x \ : \ x \in A \ and \ x \in B\}$$

Example: 1.4 $(A \cap \ B) = (B \cap \ A)$

$$(A \cap \ A) = A \ and \ (A \cap \ \phi) = \phi$$

Note:

- ✓ Two sets $A \ and \ B$ are called disjoint iff $(A \ \cap \ B) = \phi$, i.e. $A \ and \ B$ have no element in common.
- ✓ A collection of set is called disjoint collection if, for every pair of sets in the collection the two sets are disjoint. The elements of a disjoint collection are said to be mutually disjoint.
  e.g. If $A = \{\{1, 2\}, \{3\}\}$, $B = \{\{1\}, \{2, 3\}\}$, $and \ C = \{\{1, 2, 3\}\}$now these sets are mutually disjoint because $(A \ \cap \ B) = \phi$, $(B \ \cap \ C) = \phi \ and \ (A \ \cap \ C) = \phi$. So the given sets are mutually disjoint.

Union

The union of $A \ and \ B$, written as $(A \cup B)$, is the set of all elements which are members of the set A or the set B or both it is written as

$$(A \cup B) = \{x \ : \ x \in A \ \ or \ \ x \in B\}$$

i.e.

$$A \cup B = B \cup A$$
$$A \cup \phi = A$$
$$A \cup A = A$$

Example 1.5 $A = \{0, 1, 2\}$

$B = \{0, -1, -2\}$

Then $A \cup B = \{-2, -1, 0, 1, 2\}$

$A \cap B = \{0\}$

Complements

Let $A \ and \ B$ be two sets. For any set A, the relative complement of B with respect to A, written as $A - B$ is the set consisting of all elements of A which are not elements of B i.e.

$$A - B = \{x: x \in A \ and \ x \notin B\}$$

Relative complements of B with respect to A can be written as $A \setminus B$

Let U be the universal sets. For any set A, the relative complement of A with respect to U, i.e. $U - A$ is called the absolute compliment of A. It is often called the compliment of A and denoted by A^C.

i.e. $U - A = A^C = \{X: X \in U \ and \ x \notin A\}$

Note: we can represent A^C as A' also.

1.5 Venn Diagrams

Introduction of the universal set permits the use of a pictorial device to study the connection between the subsets of a universal set and their intersection, union, difference and other operations. The diagrams used are called Venn Diagrams

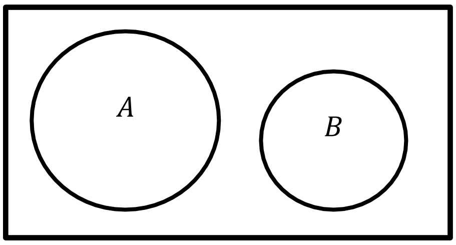
1.5.1 Definition

Venn diagram is a schematic representation of a set by a set of points. The universal set U is represented by a set of points in a rectangle and a subset say A of U is represented by the interior of circle. The operation of union and intersection can be very simply represented through Venn diagrams.

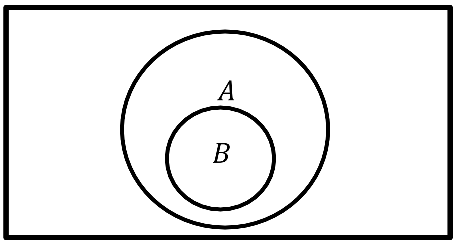
Thus “Venn diagrams are pictorial representations of sets and their inter-relations, and of them some basic results in set theory become obvious through these diagrams”.

Any closed curve enclosing an area may be supposed to represent a set.

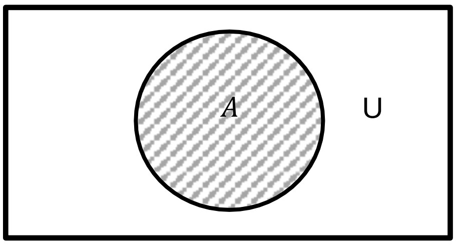
Thus, let the circles A and B represent the sets A and B respectively.



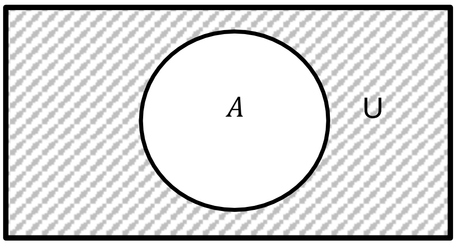
$(\text{Fig. } 1.1 \text{ a})\ A \cap B = \emptyset$



(Fig. 1.1 b) $B \subseteq A$



$(\text{Fig. } 1.1 \text{ c})\ A$



(Fig. 1.1 d) A^c

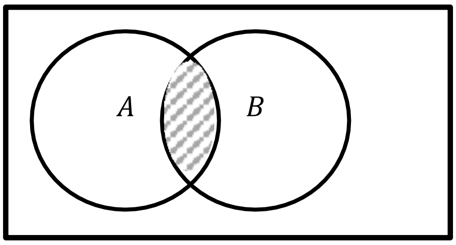
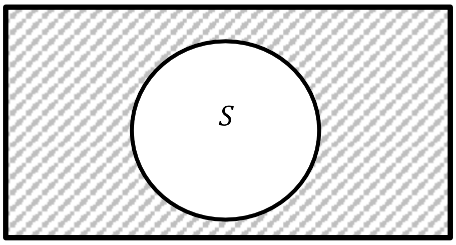


Fig. 1.1 e $A \cap B$

Thus from (Fig. 1.1 e), we see the portion common to two circles represents $A \cap B$, while $A \cup B$ is represented by the total area covered by the two circles together.

Suppose we represent the universal set U by the rectangle in the (fig. 1.1 d). The component of S with respect to U denoted by $S'\ or\ S^c$ (fig. 1.2).



(Fig. 1.2) S' *or* S^c

Venn diagrams can also help in visualizing some types of problems given below

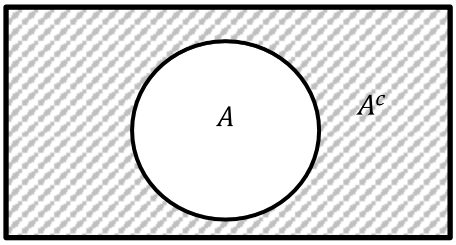
1.5.2 Complements

Suppose, if U is a universal complement or simply complement of a set A, denoted by A^c is the set of elements which belong to U but which do not belong to A:

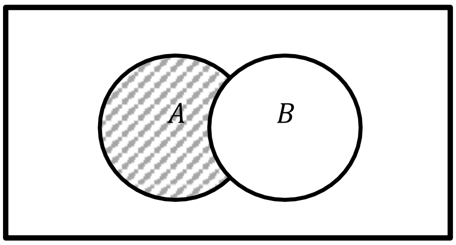
$$A^c = \{x: x \in U, x \notin A\}$$

The relative complement of a set B with respect to a set A, simply, the difference of A and B, denoted by A/B is the set of elements which belong to A but which do not belong to B.

$$A\backslash B = \{x: x \in A, x \notin B\}$$



(Fig. 1.3 a) $\boldsymbol{A^c}$ is shaded



(fig. 1.3 b) $\boldsymbol{A\backslash B}$ is shaded or $\boldsymbol{A -}$ $\boldsymbol{B}$ **or** $\boldsymbol{A{\sim}B}$

1.6 Cartesian product

Let A *and* B be two sets. The Cartesian (or direct) product of non-empty sets A *and* B is defined as the set $A \times B$ is the set of all ordered pairs such that the first member of the ordered pair is an element of A and the second member of B. The Cartesian product of A *and* B written as $A \times B$ and represented as

$$A \times B = \{(a, b): a \in A \; and \; b \in B\}$$

Example 1.6 $A = \{a, b\}$ *and* $B = \{1,2,3\}$

$$A \times B = \{(a, 1), (a, 2), (a, 3), (b, 1), (b, 2), (b, 3)\}$$

Note: if $A = \emptyset \; and \; B$ is any non-empty set then $A \times B = \emptyset = B \times A$

Remark 1: In general $A \times B \neq B \times A$

Remark 2: If any one of the sets $A \; and \; B$ is empty then $A \times B = \emptyset$

Remark 3: If $A \; and \; B$ have $m \; and \; n$ elements, then $A \times B$ has mn elements

Remark 4: If A, B, C are non-empty sets then

$$A \times (B \times C) \neq (A \times B) \times C$$

1.7 Power Sets

On several occasions testing all the combinations of elements in a set for the particular property involves numerous problems. In order to consider all such combinations in a set S, it needs to create a set which has these combinations as subsets of S

Let S be a set, and then the power set of S i.e. set of all subsets of set S can be represented as $P(S)$

Example 1.7 The power set P({3, 5, 7}) is the set of all subsets of {3, 5, 7}.

Hence, P({3, 5, 7}) = {∅, {3}, {5}, {7}, {3, 5}, {3, 7}, {5, 7}, {3, 5, 7}}.

Note that in the set of subsets, empty set and the set itself are among its members.

An empty set has only one subset i.e itself. Thus,

P (∅) = {∅}.

The set {∅} has exactly two subsets, viz., ∅ and the set {∅} itself. Therefore,

P ({∅}) = {∅, {∅}}.

1.8 Counting Principles

Initially we are introducing two basic counting principles i.e. Product Rule and Sum Rule. Later on, we will discuss the way they used to solve the counting problems

1.8.1 Product Rule

Assume that a process is broken down into two jobs and let there be $n1$ ways to conduct first job and for each first job there are $n2$ ways to conduct second job therefore in all there are $n1.n2$ ways to get the entire process done.

Example 1.8:

A company with two employees namely Sanjeev and Pankaj, leases a floor in a building having 12 offices. So in exactly how many ways does the distinct office can be assigned to both of the employees?

It can be solve as,

In the process to assign offices to two of the employees can be done by starting with Sanjeev to which an office can be assigned in 12 ways (since the total no. of offices in building is 12) later on, Pankaj can be assigned an office in 11 ways (since an office is already allotted to Sanjeev). Hence using product rule we can express this situation as,

$$12 \times 11 = \mathbf{132}$$

Ways to assign offices to these two employees.

1.8.2 Sum Rule

Let us consider if a job that can be done either in one of n1 methods or one of n2 methods thereafter none of the set of n1 methods are similar to the set with n2 methods thus, there are n1+n2 ways to conduct that job.

Example 1.9

A student can select only a project from three given lists out of which each one consists of 23, 15 and 19 projects respectively. There is no repetition of any project among these three lists. Then how many projects is there that can be selected by students?

Solution: The student can select a project either from the first list or the second list or the third list. Though no project is repeated in any of the list, using sum rule there are $23 + 15 + 19 = 57$ ways to select a project.

Assume that if a job can be done in one of two methods and within that there is one which is common to both. In such a scenario, the sum rule cannot be used to count the number of methods to conduct the job. This can be done in two or more ways. We have to see whether to do this job in many ways that will be counted repeatedly. So we have to subtract those methods which we have counted twice.

1.8.3 Subtraction Rule

If a job that can be done in $n1$ ways or $n2$ ways then the total number of ways in which that job can be accomplished will be expressed as $n1 + n2$ and subtracting the number of ways that are common in these two ways.

The rule of subtraction is also called as **principle of inclusion–exclusion**, particularly in case of counting the elements in the union of two sets. Assume that S1 and S2 are the two sets. At that time, there are |S1| ways to choice an element from S1 and |S2| ways to choice an element from S2. The total number of ways to choose an element from S1 or from S2, will be the number of ways to choose an element from the union of these two sets. It will be the sum of the number of ways to choose an element from S1 and the number of ways to choose an element from S2, minus the number of ways to choose an element that is in both S1 and S2. Mathematically it will be $|S1 \cup S2|$ways to select an element in either S1 or in S2, and $|S1 \cap S2|$ ways to choose an element common to both sets, this will give a formula

$$|S1 \cup S2| = |S1| + |S2| - |S1 \cap S2|.$$

This is the expression for the total number of unique elements in the union of two sets.

Example 1.10

How many bytes either start with a 1 bit or terminated with the two bits 00?

Solution: We can create a string of bits of length eight that will either start with 1 or terminates with two bits 00. We can create a string of bits having length eight that starts with a bit 1 in 2^7

= 128 ways. This is due to the product rule, since the first bit can be chosen in only single way and every other bits can be chosen in two ways. Similarly, we can create a string of bits with length eight terminating with the two bits 00, in 2^6 = 64 ways. This is due to the product rule, since each of the first six bits can be chosen in two ways and the last two bits can be chosen in only one way.

Some of the ways to create a string of bits with length eight starts at 1 are the same as the ways to create a string of bits with length eight that terminates with the two bits 00. There are 2^5 = 32 ways to create such a string. This uses the product rule, since the first bit can be chosen in only one way, every sixth bits can be chosen in two ways, and the last two bits can be chosen in one way. Subsequently, the number of strings of bits with length eight that starts at 1 or terminate at 00, which equals the number of ways to create a string of bits of length eight that starts at 1 or that terminates with 00, equals 128 + 64 − 32 = 160.

1.8.4 Division Rule

It is useful while solving listing problems.
"There are n/d methods to do a job if it can be done using a process that can be conducted in n ways, and for each way w, exactly d out of the n ways resemble to way w."

The division rule can be reaffirmed in terms of sets: "If a finite set S is the union of n pairwise disjoint subsets each with d elements, then $n = |S|/d$."

We can demonstrate the division rule for counting with the following example.

Example 1.11
How many dissimilar ways to seat four people about a circular table, where two seating's are assumed the same when each person has the same left and the right neighbor?

Solution: Firstly, randomly select a seat around the table and tag it seat **1**. Then tag the remaining seats in arithmetical order, arranged clockwise manner around the table.
Consider the facts,

- There are four ways to opt the person for seat **1**,
- Three ways to opt the person for seat **2**,
- Two ways to opt the person for seat **3**,
- One way to opt the person for seat **4**.

Therefore, we can say that this can be done in factorial 4 ways, i.e. $4! = 24$ ways to order the given four people for these seats.
Though, each of the four choices for seat **1** leads to the same arrangement, as it can be distinguished two arrangements only when one of the people has a different immediate left or immediate right neighbor.
Now we have four ways to select the person for seat **1**, by the division rule it will be

$$24/4 = 6$$

different seating arrangements of four people around the circular table.

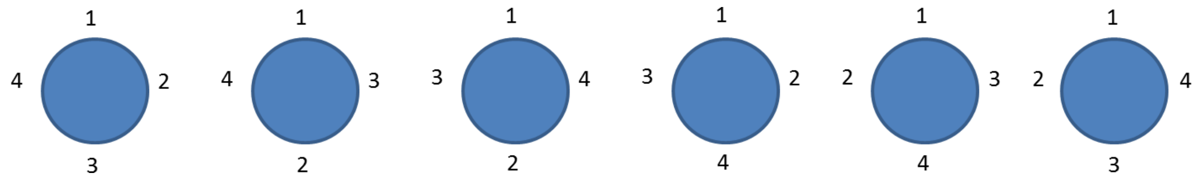


Fig. 1.4

Tree Diagrams

The problems on counting can also be solved by using trees. A tree contains a root; a number of branches originate from the root, and possible further branches originating the endpoints of other branches. To use trees in counting, we use a branch to represent each possible choice. We represent the possible outcomes by the leaves, which are the endpoints of branches not having other branches starting at them.

Example 1.12
Assume that “I Love India” T-shirts available in five variants: S, M, L, XL, and XXL. Besides this assume that each variant available in four different colors, yellow, orange, pink, and gray, except for XL, which available in orange, pink, and gray, and XXL, which have pink and gray. How many different t-shirts does a memento shop should stock to have at least one of each available variant and color of the T-shirt?

Solution: The trees diagram in the given Figure 1-5 displays all the possible variants and color pairs. It follows that the memento shop owner needs to stock 17 different T-shirts.

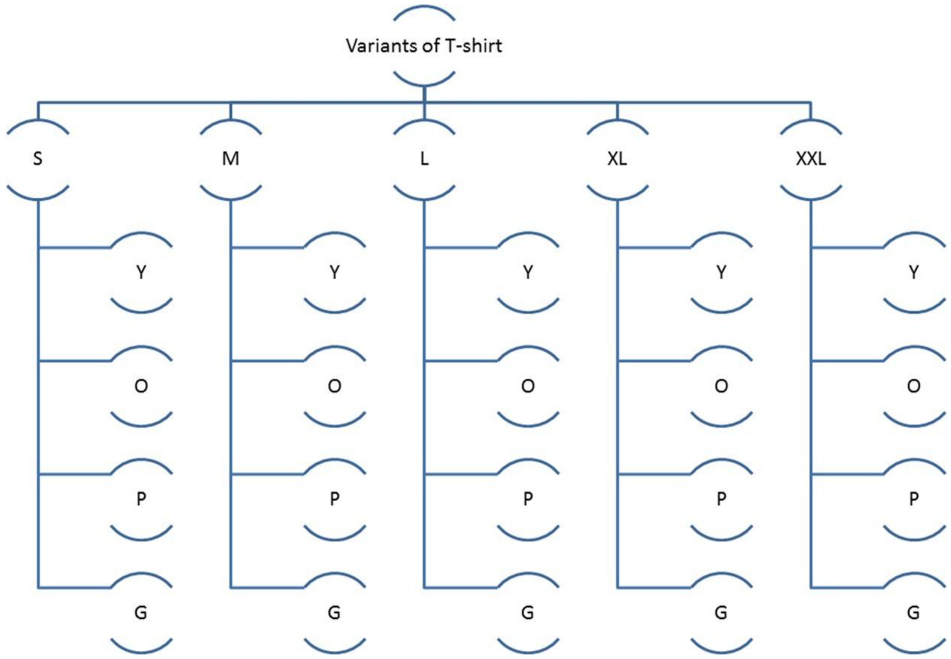


Fig. 1.5

1.9 Cardinality

Sets are broadly used in counting problems, therefore for such use we need to study about their sizes.

Let us assume that S be a set contains n distinct elements where n is any non-negative integer then the set S is said to be finite. The cardinality of set S can be represented by |S| which is in other term also called as number of elements in S.

Examples 1.13

- ✓ Let X be the set of odd positive integers less than 20. Then |X| = 10.
- ✓ Let P be the set of prime positive integers less than 10. Then |P| = 5.
- ✓ Though the null set contain no elements, it follows that |∅| = 0.

We will also be interested in sets that are not finite.

Infinite set: A set is called as infinite if it is not finite set.

Example 1.14 The set of all positive integers is infinite set.

Extending the notion of cardinality

Till now the notion of cardinality was in the scope of finite sets which is used to compare the two finite sets on the basis of their sizes. Now we are extending this notion to infinite set where the comparison can be done on the basis of the difference between their sizes with respect to each other.

These notions have vital applications to computer science. A set is said to be uncomputable in case of infeasibility of a computer program to find all its values, even with unlimited time and memory. This notion is used to explain why uncomputable sets exist.

Definition:

The sets A and B are said to have same cardinality only when their sizes are equal and have one to one correspondence between distinct elements of the set. It can be expressed as |A| = |B|.

In case of infinite sets, we need to talk of the cardinality on relatively among two sets instead of being particular to a set.

In other words, If there is a one to one correspondence between A and B, the size of A is less than or same as the size of B and we express it as |A| ≤ |B|. Here we can add that when |A| ≤ |B| and A and B have different cardinality, we can conclude that the cardinality of A is less than the cardinality of B and it will be |A| < |B|.

1.10 Countability

1.10.1 Countable Sets

Now, we will divide infinite sets into two different groups, those with the similar cardinality as the set of natural numbers and those with a dissimilar cardinality.

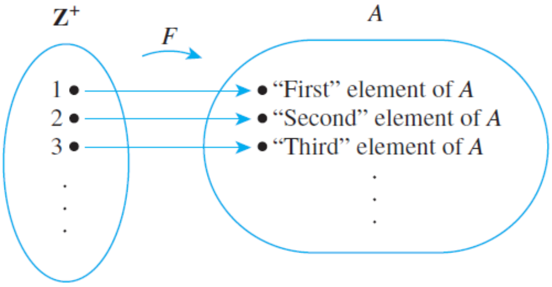


Fig. 1.6

A set is said to be countably infinite iff, it has the similar cardinality as the set of positive integers Z^+. A set is called countable iff, it is finite or countably infinite.

Example 1.15 To show a set is countable:

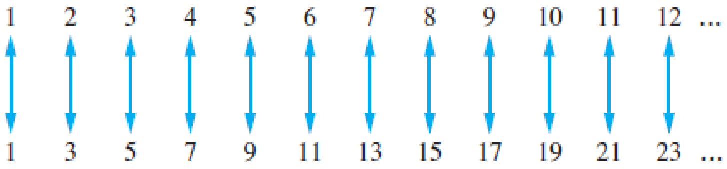


Fig. 1.7

In order to proof that the set odd positive integers is finite, we need to have one to one correspondence with the set of positive integers Z^+. It can be expressed as,

$$f(x) = 2x - 1$$

from Z^+ to the set of odd positive integers. We prove that f is a one-to-one correspondence by showing that it is both one-to-one and onto. To see that it is one-to-one, suppose that $f(x) = f(y)$.

Then $2x - 1 = 2y - 1$, so $x = y$.

To see that it is onto, suppose that n is an odd positive integer. Then n is 1 less than an even integer $2k$, where k is a natural number. Hence $n = 2k - 1 = f(k)$.

An infinite set is countable if and only if it is possible to list the elements of the set in a sequence (indexed by the positive integers). The reason for this is that a one-to-one correspondence f from the set of positive integers to a set S can be expressed in terms of a sequence

$$a_1, a_2, \dots, a_x, \dots, where\ a_1 = f(1), a_2 = f(2), \dots, a_x = f(x), \dots.$$

1.10.2 Uncountable Sets

A set that is not countable is called uncountable.

A significant proof method presented in 1879 by Georg Cantor and commonly known as the Cantor diagonalization argument. It was supposed to prove that the set of real numbers is not countable.

Let’s see that how the set of real numbers is uncountable, we assume in advance that the set of real numbers is countable and reach at a stage of contradiction. Then, the subset of all real numbers that are between 0 and 1 would also be countable (since any subset of a countable set is also countable). Under this assumption, the real numbers between 0 and 1 can be listed in some order, say, r_1, r_2, r_3, ... We can decimal represent these real numbers like

$r_1 = 0.d_{11}d_{12}d_{13}d_{14}.\dots\dots$

$r_2 = 0.d_{21}d_{22}d_{23}d_{24}.\dots\dots$

$r_3 = 0.d_{31}d_{32}d_{33}d_{34}.\dots\dots$

$r_4 = 0.d_{41}d_{42}d_{43}d_{44}.\dots\dots$

...

where d_{ij} ∈ {0, 1, 2, 3, 4, 5, 6, 7, 8, 9}. (For example, if r_1 = 0.23794102 . . . , we have d_{11} = 2, d_{12} = 3, d_{13} = 7, and so on.) Then, form a new real number with decimal expansion

r = $0.d_1d_2d_3d_4$ . . . , where the decimal digits are determined by the following rule:

$$d_i = \begin{cases} 4\ if\ d_{ii} \neq 4 \\ 5\ if\ d_{ii} \neq 4 \end{cases}$$

(As an example, suppose that r_1 = 0.23794102 . . . , r_2 = 0.44590138 . . . , r_3 = 0.09118764 . . . , r_4 = 0.80553900 . . . , and so on. Then we have r = $0.d_1d_2d_3d_4$ . . . = 0.4544 . . . , where d_1 = 4 because d_{11} ≠ 4, d_2 = 5 because d_{22} = 4, d_3 = 4 because d33 ≠ 4, d4 = 4 because d44 ≠ 4, and so on.)

Decimal expansion of each real number is unique by itself. Consequently, the real number r is not equal to any of its component because the decimal expansion of r differs from the decimal expansion of their components to the right of the decimal point.

Though there is a real number r between 0 and 1 that is not in the list, the supposition that all the real numbers between 0 and 1 could be listed should be false. Thus, all the real numbers between 0 and 1 cannot be listed, so the set of real numbers between 0 and 1 is uncountable. Any set having an uncountable subset is uncountable. Therefore, the set of real numbers is uncountable.

Results about Cardinality

- ✓ If X and Y are countable sets, then their union is also countable.
- ✓ If X and Y are sets with |X| ≤ |Y| and |Y| ≤|X|, then |X| = |Y|. In other terms, there is a one-to-one correspondence between X and Y.

1.11 Basic Set Identities and Proofs

Useful Definitions

For A, B subsets of the universal set U

$x \in A \cap B \Leftrightarrow x \in A \ \ and \ \ x \in B$

$x \in A \cup B \Leftrightarrow x \in A \ \ or \ x \in B$

$x \in A - B \Leftrightarrow x \in A \ \ and \ \ x \notin B$

$x \in A^C \Leftrightarrow x \notin A$

$x, y \in A \times B \Leftrightarrow x \in A \, and \, y \in B$

1.11.1 Basic set Identities

- ✓ Cumulative Law
  - ○ $A \cup B = B \cup A$
  - ○ $A \cap B = B \cap A$
- ✓ Associative Law
  - ○ $(A \cup B) \cup C = A \cup (B \cup C)$
  - ○ $(A \cap B) \cap C = A \cap (B \cap C)$
- ✓ Distributive Laws
  - ○ $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
  - ○ $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
- ✓ Identity Laws
  - ○ $\emptyset \cup A = A \cup \emptyset = A$
  - ○ $A \cap \emptyset = \emptyset \cap A = \emptyset$
  - ○ $A \cap U = U \cap A = A$
  - ○ $A \cup U = U \cup A = U$
- ✓ Complement Law
  - ○ $A \cup A^C = U$
  - ○ $A \cap A^C = \emptyset$
  - ○ $U^C = \emptyset$
  - ○ $\emptyset^C = U$
- ✓ Double Complement Law
  - ○ $(A^C)^C = A$
- ✓ Idempotent Law
  - ○ $A \cup A = A$
  - ○ $A \cap A = A$
- ✓ De Morgan’s Law
  - ○ $(A \cup B)^C = A^C \cap B^C$
  - ○ $(A \cap B)^C = A^C \cup B^C$
- ✓ Alternative representation for set difference
  - ○ $A - B = A \cap B^C$

1.11.2 Proofs

Distributive Law

For sets A, B, C prove that $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

Proof: Let $x \in A \cup (B \cap C) \quad \Leftrightarrow \quad x \in A \, or \, x \in (B \cap C)$

$\Leftrightarrow \qquad x \in A \quad or \quad (x \in B \ \ and \ \ x \in C)$

$\Leftrightarrow \qquad (x \in A \quad or \quad x \in B) \quad and \quad (x \in A \quad or \quad x \in C)$

$\Leftrightarrow \qquad x \in (A \cup B) \quad and \quad x \in (A \cup C)$

$\Leftrightarrow \qquad x \in \{(A \cup B) \cap (A \cup C)\}$

Thus, we can say $\quad A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$

1.12 Pigeonhole Principle

The pigeonhole principle states that if n+1 pigeons fly into n pigeonholes, then at least one pigeonhole must contain two or more pigeons. This principle is illustrated in Figure 1.8 for 5 pigeons and 4 pigeonholes. Illustration (a) shows the pigeons perched next to their holes, and (b) shows the correspondence from pigeons to pigeonholes. The pigeonhole principle is sometimes called the Dirichlet box principle because it was first stated formally by J. P. G. L. Dirichlet (1805–1859).

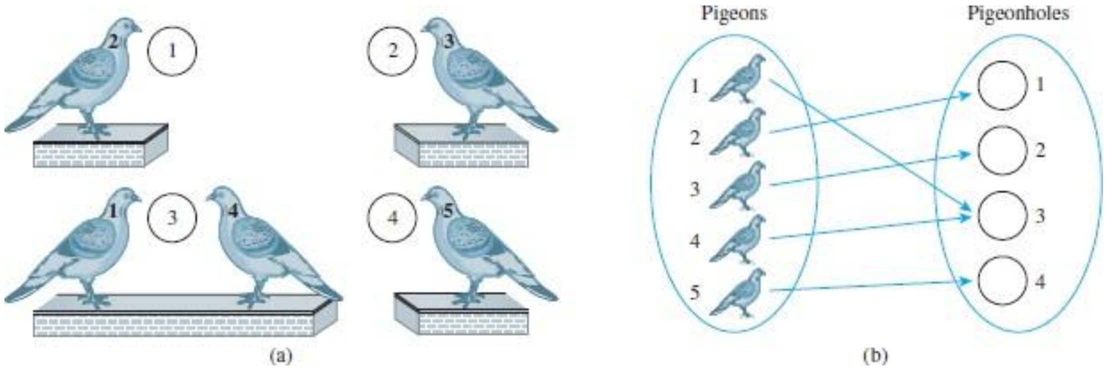


Fig. 188

Illustration (b) suggests the following mathematical way to phrase the principle.

Generalized Pigeonhole Principle

A generalization of the pigeonhole principle states that if n pigeons fly into m pigeonholes and, for some positive integer k, k < n/m, then at least one pigeonhole contains k + 1 or more pigeons. This is illustrated in Figure 1.9 for m = 4, n = 9, and k = 2. Since 2 < 9/4 = 2.25, at least one pigeonhole contains three (2 + 1) or more pigeons. (In this example, pigeonhole 3 contains three pigeons.)

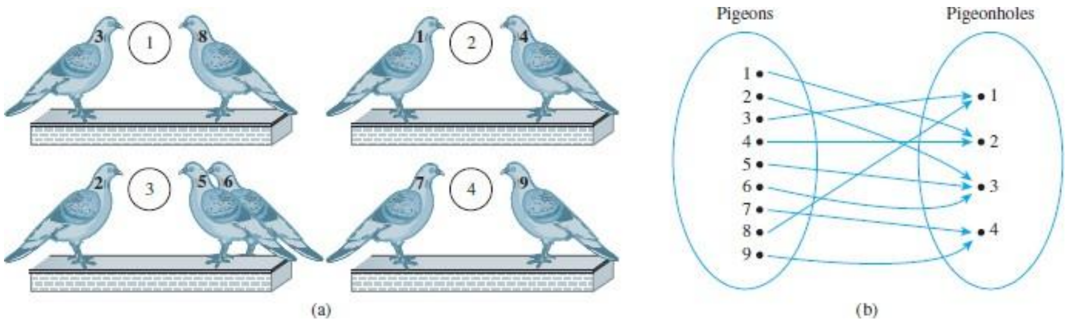


Fig. 1.9

Problems for Exercise

1. Rephrase the statements given below using set notations
   a. The element x is not a member of A
   b. The element b is a member of B.
   c. Y is a subset of Z.
   d. X is not a subset of Z.
   e. S contain all the elements of U
   f. H and G contain the same elements.

2. Write down the elements of the subsequent sets; assume $I = \{1, 2, 3, \ldots\}$.
   a. $B = \{b: b \in I, 4 < x < 15\}$
   b. $E = \{y: y \in I, y\ is\ odd, y < 20\}$
   c. $A = \{n: n \in I, 5 + n = 6\}$

3. Which of these sets are equal : $\{x, y, z\}, \{z, x, y\}, \{y, z, x\}, \{x, z, y\}, \{y, x, z\}$.

4. Distinguish between $X \subseteq Y \quad and \quad X \subset Y$

5. Draw a Venn diagram of sets X, Y, Z such that
   a. X and Y have elements in common
   b. Y and Z have elements in common
   c. X and Z are disjoint

6. State the De Morgan's Laws.

7. Prove the Distributive Law: $X \cup (Y \cap Z) = (X \cup Y) \cap (X \cup Z)$.

8. State the finite sets among the following:
   a. $S = \{seasons\ in\ the\ year\}$
   b. $T = \{states\ in\ the\ Country\}$
   c. $P = \{positive\ integers\ less\ than\ 1\}$
   d. $D = \{dogs\ living\ in\ \ the\ India\}$

9. Find the power set $P(X)$ of $X = \{6, 7, 8, 9\}$

10. Determine the size of the power set mentioned below:
    a. $\{y: y\ is\ a\ days\ of\ the\ week\}$
    b. $\{y: y\ is\ a\ letter\ in\ the\ word\ \text{“}INDIA\text{”}\}$
    c. $\{y: y\ is\ a\ positive\ divisor\ of\ 6\}$

Block I: Unit II Relation

Learning Objectives
After completing this unit, the learner shall be able to:

- Explain the definition of relation;
- Explain various types of relations;
- Define Composition of Relation;
- Represent the Pictorial Representation of Relation;
- Explain Properties of Relation;

Relationships among elements of sets occur in many perspectives. Each day we deal with relationships such as those between as business with its telephone number, an employee with his or her salary, a person with a relative, and so on. In mathematics we study relationship between a positive integer with one that it divides, a real number with one that is larger than it, and so on. Relations can solve problems such as knowing which pair of cities are linked by airline flights in a network finding, a sustainable order of different phases of a complex project, or bringing out a useful way to store information in computer databases.

The notion of a relation is elementary concept in day to day life as well as in mathematics. We already used various relations. The act of comparing objects which are related to one another is associated with a relation.

The relation among words suggests some used to examples of relation such as the relation of the father with son, mother with son, brother with sister etc. Familiar examples of relation in arithmetic are relations such as “greater than”, “less than” or “equal to”. We also know relation among the area of a circle with its radius and between the area of a square and its side.

The straightest way to express a relationship among elements of two sets is to use ordered pairs made up of two related elements. For this reason, sets of ordered pairs are said to be a binary relation.

2.1 Definition of Relation

Let A and B be sets. A binary relation from A to B is a subset of $A \times B$.

It states that some of the elements in set A are related to some of the elements in set B. In fact, if R is a binary relation from A to B and if the ordered pair (a, b) is in R, we would say that the element a is related to the element b.

Example 2.1 Consider $A = \{a, b\}$ be a set of two students and consider $b = \{CS121, CS131, CS141\}$ be the set of three courses. The Cartesian product $A \times B$ gives all the alike pairings of students and courses. On the other hand $R = \{(a, CS121), (b, CS131)\}$ represents the courses opt by the students.

A binary relation R from A to B assigns to each ordered pair (a, b) in $A \times B$ exactly one of the following statements.

1. a is related to b i.e., $a\ R\ b$ or $(a, b) \in R$
2. a is not related to b i.e., $a \not R\ b$ or $(a, b) \notin R$

The elements of a set A are termed as ***domain*** of a relation R and the elements of set B are termed as the ***range*** of R

Remark: $R \subseteq A \times B$ means R is a relation from A to B. If $B = A$ then R is said to be relation from A to A. In such a case R is called a relation in A. Therefore, any relation in A is a subset of $A \times A$. Thus, the set $A \times A$ itself describes relation in A and is said to be as a universal relation in A. While the empty set which is again a subset of $A \times A$ is called a void relation in A.

2.2 Types of Relations

2.2.1 Inverse Relation

Assume R be some relation from set A to set B. the inverse of R, symbolized by R^{-1}, is the relation from B to A with involves those ordered pairs, which, when reversed, belong to R; i.e.,

$$R^{-1} = \{(b, a): (a, b) \in R\}$$

Example 2.2 The inverse of the relation $R = \{(1, y), (1, z), (3, y)\}$from $A = \{1,2,3\}\ to\ B = \{x, y, z\}$ is as follows:

$$R^{-1} = \{(y, 1), (z, 1), (y, 3)\}$$

Obviously, if R is some relation, then $(R^{-1})^{-1} = R$. Also, the domain and range of R^{-1} are the same, respectively, to the range and domain of R. Besides, if R is a relation on A, then R^{-1} is also a relation on A.

2.2.2 Combined Relation

Since relations from A to B are known to be the subsets of $A \times B$, two relations from A to B can be combined in some way two sets can be combined. Consider the following examples.

Example 2.3 Let $A = \{1,2,3\}$ and $B = \{1,2,3,4\}$. Let R_1 and R_2 be two relations from A to B and the two relations are defined as follows:

$$R_1 = \{(1,1), (2,2), (3,3)\} \text{ and } R_2 = \{(1,1), (1,2), (1,3), (1,4)\} .$$

We can combine the two relations to obtain other relations as follows:

(i) $R_1 \cup R_2 = \{(1,1), (1,2), (1,3), (1,4), (2,2), (3,3)\}$

(ii) $R_1 \cap R_2 = \{(1,1)\}$

(iii) $R_1 - R_2 = \{(2,2), (3,3)\}$

(iv) $R_2 - R_1 = \{(1,2), (1,3), (1,4)\}$

2.3 Composition of Relation

Relation is an association between two sets and is a group of ordered pairs that contains one object from each of the given set. If the element x is from the first set and the element y is from the second set, then these elements are said to be related, if the ordered pair $(x,\ y)$ is in some relation.

Definition

Assume that we have three sets $A,\ B$ and C. A relation R defined from A to B, and a relation S defined from B to C. We can now define a new relation called as the composition of R and S, written as $S\ o\ R$. This new relation can be defined as follows:

"If a is an element in A and c is an element in C, then $a(S\ o\ R)c$, if and only if, there exists some element b in B, such that $a\ R\ b$ and $b\ S\ c$. Thus, we have a relation $S\ o\ R$ from a to c, if and only if, we can reach from a to c in two steps; i.e. from a to b related by R and from b to c related by S. In this way, relation $S\ o\ R$ can be inferred as R followed by S, since this is the order in which the two relations need to be considered, first R then S."

Example 2.4 Let us consider the three sets $A = \{1,2,3,4\}$, $B = \{a, b, c, d\}$ and $C = \{x, y, z\}$. Let *R* be a relation from *A* to *B* and *S* is a relation from *B* to *C*. The two relations are defined as follows:

$$R = \{(1,a),(2,d),(3,a),(3,b),(3,d)\} \text{ and}$$

$$S = \{(b,x),(b,z),(c,y),(d,z)\},$$

then

$$S\ o\ R = \{(2,z),(3,x),(3,z)\}$$

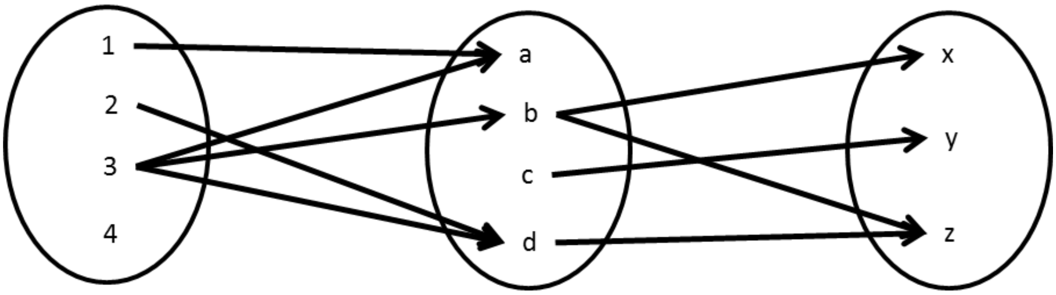


Fig: 2.1

Note: $(R\ o\ S)\ o\ T\ =\ R\ o\ (S\ o\ T)$

Remark: Let R be a relation on set A. the power $R^n, n = 1,2,3, \ldots$ are defined inductively by $R^1 =\ R$ and $R^2 =\ R\ o\ R, R^3 =\ R^2\ o\ R$ and so on

Likewise, $$R^{n+1} =\ R^n\ o\ R$$

Example 2.5: Let $R = \{(1,1),(2,2),(3,2),(4,3)\}$

Find the power $R^n, n = 2,3,4, \ldots$?

Solution:

Since $R^2 =\ R\ o\ R$,

we find that $R^2 = \{(1,1),(2,2),(3,2),(4,2)\}$.

Furthermore,

Since $R^3 = R^2 \ o \ R, R^3 = \{(1,1),(2,2),(3,2),(4,2)\}$.

Further computation shows that R^4 is the same as R^3, so $R^4 = \{(1,1),(2,2),(3,2),(4,2)\}$. it shows that $R^n = R^3 for\ n = 4,5,6,7,\ldots$.

2.4 Domain & Range

Domain: Let the relation R is defined from *A* to *B*. Then domain of R, written as (R) , is defined as

$$\boldsymbol{Dom(R)} = \{a \in A: (a,b) \in R\ for\ some\ b \in B\}$$

Range: Let the relation R is defined from *A* to *B*. Then range of R, written as (R) , is defined as

$$\boldsymbol{Ran(R)} = \{b \in B: (a,b) \in R\ for\ some\ a \in A\} .$$

Example 2.6 : Let $A = \{2,3,4\}\ and\ B = \{3,4,5\}$. List the elements of each relation R defined below and the domain and range.

$a \in A$ is related to $b \in B$, that is, a R b if and only if $a < b$

Solution:

$2 \in A$ is less than $3 \in B$, then $2R3$. Similarly, $2R4$, $2R5$, $3R4$, $3R5$, $4R5$

Thus,

$$R = \{(2,3),(2,4),(2,5),(3,4),(3,5),(4,5)\}$$

$$\boldsymbol{Dom(R)} = \{2,3,4\} \text{ and } \boldsymbol{Ran(R)} = \{3,4,5\}$$

2.5 Pictorial Representation of Relation

Initially we consider a relation S on the set R of real numbers; i.e., S is a subset of

$$R^2 = R \times R.$$

Since R^2 can be denoted by the set of points in the plane, we can picture S by featuring those points in the plane which belong to S. The pictorial representation of the relation is sometimes called the graph of the relation.

Frequently, the relation S consists of all ordered pairs of real numbers which satisfy some given equation

$$E(x,y) = 0$$

Representation of Relations on Finite Sets

Assume A and B are finite sets. The following are two of the illustrating a relation R from A to B.

(i). From a rectangular array whose rows are labeled by the elements of A and whose columns are labeled by the elements of B. Put 1 in each position of the array whenever

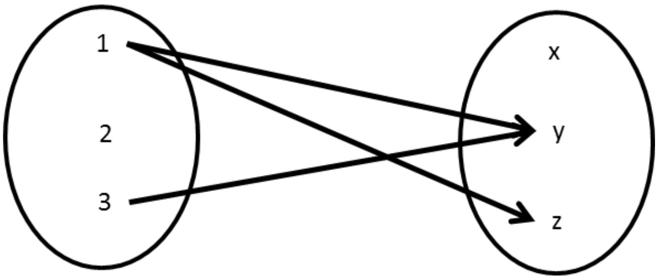
$a \in A$ is related to $b \in B$, that is, if $(a, b) \in R$, otherwise put 0. This array is said to be the matrix of the relation.

(ii). Write down the elements of A and the elements of B in two disjoint disks, and then draw an arrow from $a \in A$ to $b \in B$ whenever a is related to b. The picture will be known as the arrow diagram of the relation.

Example 2.7 Let $A = \{1,2,3\}$ and $B = \{x, y, z\}$, then the relation R from A to B defined as $R = \{(1, y), (1, z), (3, y)\}$ can be represented in the above described ways (i) and (ii) as follows:

	x	*y*	*z*
1	*0*	*1*	*1*
2	*0*	*0*	*0*
3	*0*	*1*	*0*

(i)



(ii)

Fig: 2.2

Directed Graphs of the Relations on Sets

There is alternative way of visualizing a relation R when R is a relation from a finite set to itself. First we write down the elements of the set, and then we draw an arrow from each element x to each element y whenever x is related to y. This diagram is said to be the directed graph of the relation. Figure below shows the directed graph of the following relation R on the set $A = \{1,2,3,4\}$

$$R = \{(1,2), (2,2), (2,4), (3,2), (3,4), (4,1), (4,3)\}$$

Observe that there is an arrow from 2 to itself, since 2 is related to 2 under R.

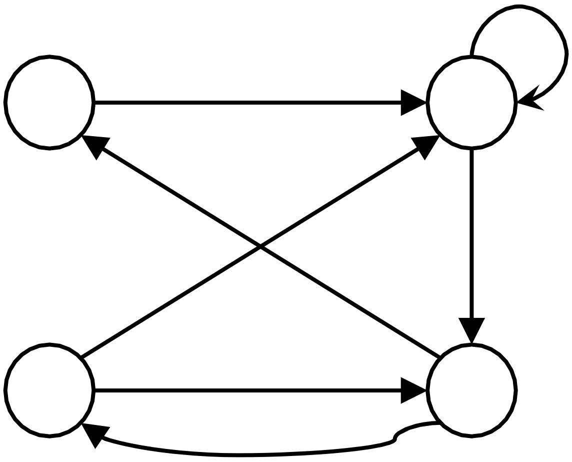


Fig: 2.3

2.6 Properties of Relation

Let R be the relations on a set A. i.e. $R \subseteq A \times A$.

2.6.1 Reflexive relation

A binary relation R in a set A is reflexive if, for each $a \in A$, aRa , that is,

$$(a, a) \in R \quad or \quad R = \{(a, a): \forall a \in A\}$$

The relation $\leq$ is reflexive in a set of real numbers since, for some x, we have $x \leq x$. The relation $<$ is not reflexive in the set of real numbers, and the relation of proper inclusion is not reflexive in the family of subsets of a universal set.

2.6.2 Symmetric Relation

A relation R in a set A is symmetric if, for each a and b in A, whenever $a\ R\ b$, then $b\ R\ a$, i.e. whenever $(a, b) \in R$ then $(b, a) \in R$.

The relations $\leq$, $<$, $\geq$, $>$ cannot be considered as symmetric relations in the set of real numbers, while the relation of equality is the sample of symmetric relation. The relation of similarity in the set of triangles in a plane can be called as both reflexive and symmetric. The relation of being a brother is not symmetric in the set of people. Though, it is symmetric in the set of all males.

2.6.3 Transitive Relation

A relation R in a set A is transitive if, for every a, b and c in A, whenever aRb and bRc, then aRc.

The relations $\leq$, $<$, $\geq$, $>$ and $=$ are transitive in the set of real numbers. The relations $\subseteq$, $\subset$ and equality are also transitive in the family of subsets of a universal set. The relation of similarity of triangles in a plane is also transitive. While the relation of being a mother is not a transitive relation at all.

2.6.4 Irreflexive relation

A relation R in a set A is irreflexive if, for each $a \in A, (a, a) \notin R$

Note that any relation which is not reflexive is not essentially irreflexive, and vice versa. The relation < in the set of real numbers is irreflexive because for no x do we have $x < x$. Likewise, the relation of proper inclusion in the set of all nonempty subsets of a universal set is irreflexive. The following is a simple example of a relation on {1,2,3} which is not reflexive and not irreflexive.

Example 2.8 $R = \{(1,1), (1,2), (3,1)\}$ is neither reflexive nor irreflexive.

2.6.5 Antisymmetric Relation

A relation R in a set A is antisymmetric if, for each a, b in A, whenever aRb and bRa, then $a = b$. Symbolically R is antisymmetric in A if whenever $(a, b) \in R$ and $(b, a) \in R$, then $a = b$.

2.7 Partial Ordering

A relation R on a set S is said to be a partial order if it is reflexive, antisymmetric and transitive. i.e.

1. *Reflexivity:* $aRa\ \ for\ all\ a \in S$

2. *Antisymmetric:* $aRa\ \ and\ bRa\ \Rightarrow\ a = b$

3. *Transitive* $aRb\ \ and\ bRc\ \Rightarrow aRc$

A set S together with a partial order R is known as a partial order set or a poset and is denoted by (S, R)

Example 2.9 The relation "greater than or equal to (≥)" on Z, the set of integers, is a partial ordering.

Reflexivity: Since $a \geq a$ for every integer $a,$ ≥ is reflexive

Antisymmetric: Since $a \geq b$ and $b \geq a\ \Rightarrow a = b,$ ≥ is antisymmetric

Transitive Since $a \geq b$ and $b \geq c\ \Rightarrow a \geq c,$ ≥ is transitive.

Hence, ≥ is partial ordering on Z, and (Z, ≥) is a poset.

Example 2.10

Let $/$ be divides relation R on a set N of positive integers. That is, $\forall\ a, b \in N, a/b \Leftrightarrow b = ka$ for some integer k. Prove that $/$ is partial relation on N.

Solution: Reflexive: We have, $a \in N$, a is a divisor of a i.e. aRa.Therefore, R is reflexive.

Antisymmetric: If a is a divisor of b then b cannot be a divisor of a unless $a = b$. Thus, $aRb\ and\ bRa$ imply $a = b$. Thus, R is antisymmetric.

Transitive: Lastly, a is a divisor of b and b is divisor of c implies a is a divisor of c. So, R is transitive.

Since R is reflexive, antisymmetric and transitive, So, R is a partial order relation.

Observe that on the set of all integers, the above relation is not partial order set as a and $(-a)$both divide each other without being equal.

Problems for Exercises

1. State the difference between ordered pair and a set assuming (a, b) as orderd pair and $\{a, b\}$ as a set.

2. Find the value of p and q if$(3p, p - 2q) = (6, -8)$.

3. Let $X = \{male, female\}$ and $Y = \{cat, dog, fish\}$. Evaluate
   a. $X \times Y$
   b. $Y \times X$

4. Assume, R as the relation from $P = \{1, 2, 3, 4\}$ to $Q = \{x, y, z\}$ defined
$$R = \{(1, y), (1, z), (3, y), (4, x), (4, z)\}$$
   a. Find out the domain and range of R
   b. Find the R^{-1} of R.

5. Each of the following is a relation on the positive integers N
   a. “p is less than q”
   b. “pq is a square root of an integer”.
   c. $p + q = 20$
   d. $p + 4q = 20$

Determine whether the relations are among: reflexive, symmetric, antisymmetric, transitive.

6. Let $P = \{1, 3, 5, 7\}$ and let R defined by “ $a\ divides\ b$ ”, written $a|b$.
$$R = \{(1,1), (1,3), (1,5), (1,7), (3,3), (3,5), (3,7), (5,5), (5,7), (5,8)\}$$
Draw the directed graph of R.

7. Let R and S are transitive relations on a set A. Show that $R \cap S$ is transitive.

8. Let $A = \{1, 2, 3\}$, $B = \{a, b, c\}$, and $C = \{x, y, z\}$. Consider the following relations R from A to B and S from B to C:
$$R = \{(1, b), (2, a), (2, c)\} \quad and \quad S = \{(a, y), (b, x), (c, y), (c, z)\}$$
Find the composition relation $R \circ S$.

9. Let R defined from $A = \{1, 3, 5, 6\}$ to $B = \{red, green, blue, white\}$ as
$$R = \{(1, red), (1, white), (3, blue), (5, white)\}$$
Draw an arrow diagram of the relation R

10. Prove $(A \times B) \cap (A \times C) = A \times (B \cap C)$.

Block I: Unit III Function

Learning Objectives
After completing this unit, the learner shall be able to:

- Define a Function;
- Classify Functions;
- Know the various types of Functions;
- Explain Composition of Functions;
- Define Recursive Function;

Function plays an important role in Mathematics, Computer Science and many applications. We are primarily concerned with discrete functions which transform a finite set into another finite set. Computer output can be considered as a function of the input. A compiler transforms a program into a set of machine language instructions (the object program). Functions can also be used for counting and establishing the cardinality of sets.

A function is something that associates each element of a set with an element of another set (which may or may not be the same as the first set). The concept of function appears quite often even in non-technical contexts.
For example,

- ✓ A social security number uniquely identifies the person
- ✓ The income tax rate varies depending on the income
- ✓ The final letter grade for a course is often determined by test and exam scores, homework and projects, and so on.

In all these cases to each member of a set (social security number, income, tuple of test and exam scores, homework and projects) some member of another set (person, tax rate, letter grade, respectively) is assigned.

It might have noticed, a function is quite like a relation. In fact, in this case a function is a special type of binary relation.

In other words, "A relation is just a correspondence between two things. But function is the validity of a relation under certain conditions. A mother is related to his son or son is related to his mother. But if A is the mother of B then vice-versa is not true. It is the example of function."

3.1 Definition of Function

Let A and B be two non-empty sets. A function f from A to B is a set of ordered pairs

$$f \subseteq A \times B$$

With the property that for each element x in A there is a unique element y in B such the $(x, y) \in f$. The statement "f is a function from A to B" is usually represented symbolically by

$$f: A \rightarrow B$$

A function can be represented pictorially as shown below

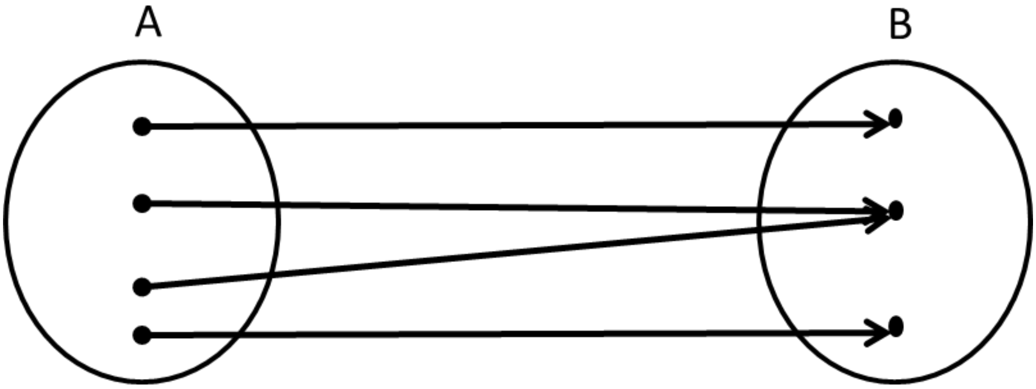


Fig 3.1: Graphical representation of a function

Note:

- ✓ There may be some elements of the set B which are not associated to any element of the set A.
- ✓ Each element of the set A must be associated to one and only one element of the set B.

If f is a function from A to B, then A is called domain of f denoted by dom f, its members are the first co-ordinates of the ordered pairs belonging to f and the set B is called the co-domain. If $(x, y) \in f$, it is customary to write $y = f(x)$, y is called the image of x ; and x is a pre-image of y. y is also called the value of f at x. The set consisting of all the images of the elements of A under the function f is called the range of f. It is denoted by $f(A)$

Thus, the range of $f = \{f(x): for\ all\ x \in A\}$

Example 3.1

Let f be the function from the set of natural numbers N to N that maps each natural number x to x^2 . Then the domain and codomain of this f are N, the image of, say *3*, under this function is *9*, and its range is the set of squares, i.e. { *0, 1, 4, 9, 16,* } .

3.2 Classification of Function

Functions can be classified mainly in two groups

a. Algebraic Functions
b. Transcendental functions

Algebraic Functions

A function which consists of a finite number of terms involving powers and roots of the independent variables x and the four fundamental operations of addition, subtraction, multiplication and division is called algebraic function. Three particular cases of the algebraic functions are:

(i) **Polynomial functions.** A function of the form $a_o x^n + a_1 x^{n-1} + \cdots + a_n$ where n is a positive integer and $a_o, a_1, \ldots a_n$ are real constants and $a_o \neq 0$ is called a polynomial of x in degree n. e.g. $f(x) = 2x^3 + 5x^2 + 7x - 3$ is a polynomial of degree 3.

(ii) **Rational functions.** A function of the form $\frac{f(x)}{g(x)}$ where $f(x)$ and $g(x)$ are polynomails in x, $g(x) \neq 0$ is called a rational function, e.g. $F(x) = \frac{x^2+2x+1}{x+2}$

(iii) **Irrational functions.** A function which involving radicals are called irrational functions. $f(x) = \sqrt[3]{x} + 5$ is an irrational function.

Transcendental Functions

A function which is not algebraic is called transcendental function.

(i) Trigonometric functions. The six functions $\sin x, \cos x, \tan x, \sec x, cosec\ x, \cot x$ where the angle x is measured in radian are called trigonometric functions

(ii) Inverse Trigonometric functions. The six functions $\sin^{-1} x$, $\cos^{-1} x$, $\tan^{-1} x$, $\sec^{-1} x$, $cosec^{-1} x$, $\cot^{-1} x$ are called inverse trigonometric functions.

(iii) Exponential functions. A function $f(x) = a^x (a > 0)$ satisfying the law $a^1 = a$ and $a^x a^y = a^{x+y}$ is called the exponential function

(iv) Logarithm functions. The inverse of the exponential function is called the logarithm function. .
So, if $y = a^x (a > 0, a = 1, x \in R, y > 0$ then $x = \log_a y$ is called Logarithm function.

3.3 Types of Function

3.3.1 Into Function

Let f is a function from A to B. f is called into function if some elements of B are left, that is, some elements of B are not the image of any element of A. In other words the range is proper subset of co-domain B. So, a function is into function if its range is a proper subset of co-domain (i.e., Range ⊂ co-domain) or we may say that there is a at least one element of co-domain which does not correspond to any element of the domain.

Let us consider the following example to understand the concept.

Example 3.2 Let $A = \{1,2,3\} and\ B = \{p, q, r, s, t\}$. Let f and g are two functions from A to B given as $f = \{(1, p), (2, q), (3, r)\}\ \ and\ \ g = \{(1, p), (2, q), (3, q)\}$. f and g are into functions. For better understanding, look at the diagrammatic representation of the two functions given below in figure 3.2:

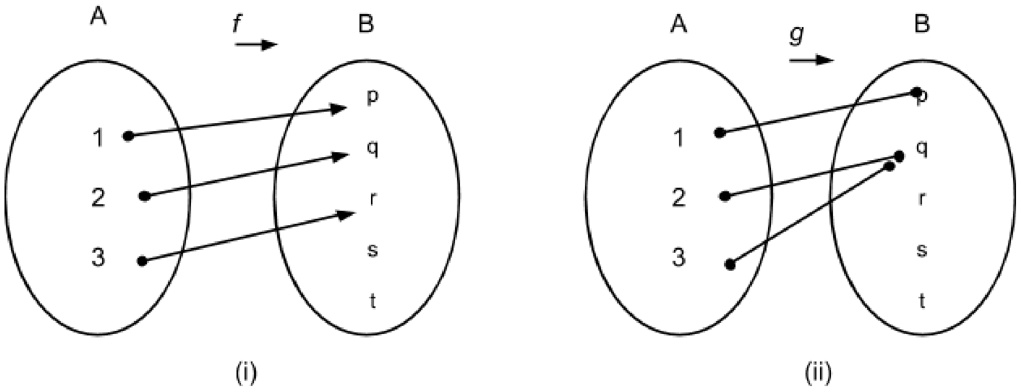


Fig 3.2: Graphical representation of functions $\boldsymbol{f}$ and $\boldsymbol{g}$ given in example 3.2

3.3.2 Onto Function

Let f is a function from A to B. f is called onto function if every element in B is the image of some element of , that is, no element is left in B which is not the image of any element of A. In other words the range of f is same as the co-domain B.

Let us consider the following examples to understand the concept.

Example 3.3 Let $A = \{2,4,5\}\ and\ B = \{a, b, c\}$. Let f be a function from A to B given as $f = \{(2, a), (4, b), (5, c)\}$, then f is an onto function.

Example 3.4 Let $A = \{1,2,3,4\}\ and\ B = \{a, b, c\}$. Let g be a function from A to B given as $g = \{(1, a), (2, b), (3, c), (4, c)\}$ then g is an onto function.

Here in the examples 3.3 and 3.4 it can be observed that all the elements of B are associated to some element of A, i.e., the range of the function is B itself (that is, Codomain). Such functions are called onto functions. So here Range = Codomain. Onto functions are called surjective functions or surjection. The graphical representation of the functions given in examples 3.3 and 3.4 are shown below in figure 3.3:

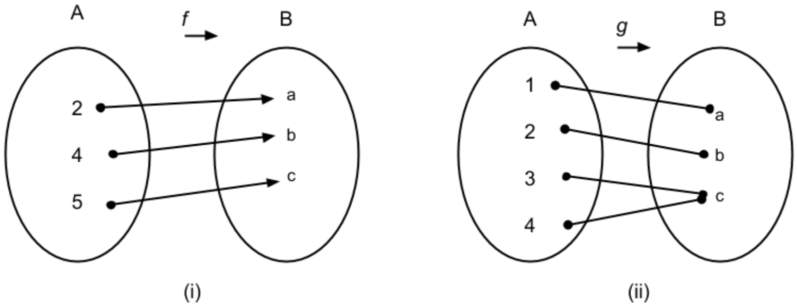


Fig 3.3: Graphical representation of functions f and g given in examples 3.3 and 3.4

Methods of finding surjective function

Let $f: A \to B$

(i) Take any element of the domain say y
(ii) Put $f(x) = y$
(iii) Solve $f(x) = y\ for\ x$

(iv) Let $x = g(y)$

(v) If $\forall\, y \in B, x = g(y) \in A,$ then f is a surjective function.

Example 3.5 Find whether the following functions are surjective or not

(i) $f\colon R \to R$ given that $f(x) = 2x^3 - 1 \;\forall\; x \in R$

(ii) $g\colon I \to I$ given that $g(x) = 3x + 2 \;\forall x \in I$

Solution.

(i) Let y be an element of R such that $f(x) = y$

$$2x^3 - 1 = y$$

$$2x^3 = y + 1$$

or $$x = (\frac{y+1}{2})^{\frac{1}{3}}$$

It is clear that there is no value of y in R for which $x \notin R$

$\therefore f$ is a surjective function

(ii) Let y be an element of I such that $g(x) = y$

$\therefore$ $$3x + 2 = y$$

or $$x = \frac{1}{3}(y - 2)$$

or

clearly, if we put $y = 1$

$$x = -\frac{1}{3}$$

which is not an element of I, the domain of g

$\therefore g$ is not a surjective function.

3.3.3 One-one Function

Let f is a function from A to B. f is called one-one function if every element in B is the image of only one element of A, that is, no element of B is the image of more than one element of A. Let us consider the following example:

Example 3.6 Let $A = \{1, 4, 7\}, \quad B = \{1,\ 16, 49\}$ and f is a function from A to B given as

$$f = \{(1, 1,), (4, 16), (7, 49)\}.$$

Here to every element of A, there exists a unique image in B, i.e. no element of B is the image of more than one element of A. Thus, f is a one-one function. One-one functions are also called injective functions. We represent graphically the function f as follows:

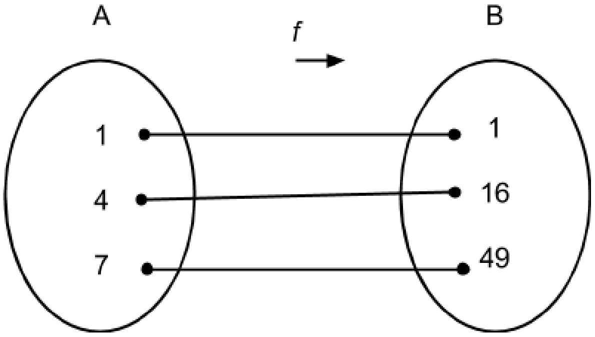


Fig 3.4: Graphical representation of function f given in example 3.6

Method of finding injective function

Let $f: A \to B$

(i) Take any two elements of the domain say a, b.
(ii) Put $f(a) = f(b)$
(iii) Solve $f(a) = f(b)$. If $f(a) = f(b)$ gives $a = b$ then the function is injective, otherwise not.

Example 3.7 Find whether the following functions are injective or not:
(i) $f: R \to R$ given that $f(x) = x^2 + 2 \; \forall \; x \in R$
(ii) $g: N \to N$ given that $g(x) = 2x + 3 \; \forall \; x \in N$

Solution.
(i) Let x and y be two elements of R such that $f(x) = f(y)$

$$f(x) = f(y)$$
$$x^2 + 2 = y^2 + 2$$

or $$x^2 = y^2$$

or $$x = \pm y$$

Hence $f(x) = f(y)$ does not give $x = y$ because for each value of x, there are two values of y

$\therefore f$ is not an injective function.

(ii) Let x and y be two elements of N such that $g(x) = g(y)$

$\Longrightarrow$ $$2x + 3 = 2y + 3$$

or $$x = y$$

Here , $g(x) = g(y)$, gives $x = y$

$\therefore g$ is an injective function.

3.3.4 Many-one function

Let f is a function from A to B. f is called many-one function if two or more than two elements of A have same image in B. In other words we can say that the function is many one when two or more elements of the domain have the same image in co-domain. Let us consider the following example:

Example 3.8 Let $A = \{p, q, r, s\}$, $B = \{P, Q, R, S\}$ and f is a function from A to B defined as

$$f = \{(p, P), (q, P), (r, R), (s, S)\}.$$

Here, elements p and q of A have the same image P in B. The function f is a many one function. Graphical representation of f is shown below in figure 3.5.

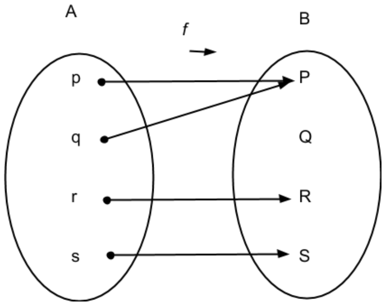


Fig 3.5: Graphical representation of function f given in example 3.8

We have defined above into, onto 1-1 and many-one functions. Combining these we get the following types of functions

3.3.5 One-one into function

A function f is called one-one into function if it is both one-one and into. It satisfies the following properties:

(i) No two elements of the domain have the same image
(ii) There is at least one element in codomain which is not the image of any elements of the domain. Fig 3.6 illustrates one-one into function

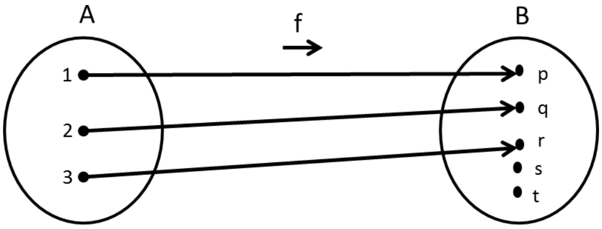


Fig: 3.6: Graphical representation of one-one into function

3.3.6 One-one onto function

A function f is called one-one onto function if it is both one-one and onto. It satisfies the following properties:

(i) No two elements of the domain have the same image.
(ii) Every element of the codomain is the image of some element of the domain.

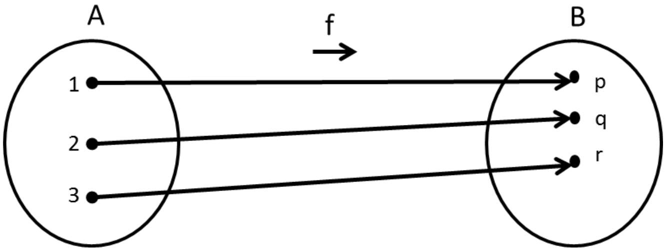


Fig: 3.7 Graphical representation of one-one onto function

One-one onto functions are also called bijective functions or bijection.

Note: For objectivity of a function, we check both the surjectivity and injectivity of a function.

Remark: A function f is one to one if and only if $f(x) \neq f(y)$ whenever $x \neq y$. This way of expressing that f is one to one is obtained by taking the contrapositive of the implication in the definition.

3.3.7 Many-one into function

A function f is called many-one into function if it is both many-one and into. It has the following properties:

(i) There are atleast 2 elements of the domain which correspond to the same element of the codomain.
(ii) There is at least one element of the codomain which is not the image of any element of the domain.

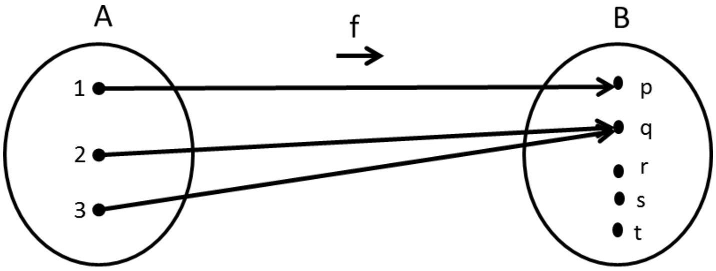


Fig: 3.8 Graphical representation of Many-one into function

3.3.8 Many-one onto function

A function f is called many-one onto function if it is both many-one and onto. It satisfies the following properties:

(i) There are atleast two elements of the domain which correspond to the same element of the codomain.

(ii) Every element of the codomain corresponds to some element of the domain.

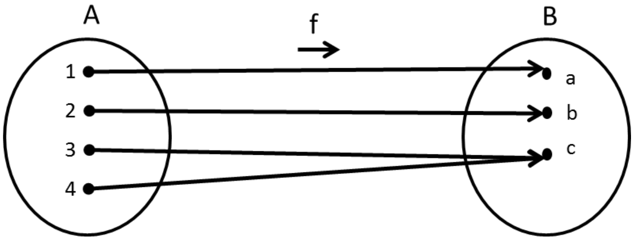


Fig: 3.9 Graphical Representation of Many-one onto function

3.3.9 Identity function

A function $f\!: A \to A$ defined by $f(x) = x$ is called the identity function. In an identity function each element of the domain corresponds to itself.

Example 3.9 The function $f\!: N \to N$ defined as $f = \{(1,1,), (2,2), (3,3), , , \dots\}$ is an identity function.

An identity function on a set A is generally denoted by I_A. It is clear from the definition that I_A is bijection on A.

3.3.10 Constant function

A function $f\!:\ A \to B$ defined by $f(x) = c$, where c is a constant, is called the identity function. In other words, a function f in which all elements of A are associated with same elements of B is called a constant function.

Example 3.10 Let $A = \{1,2,3,5\}$ and $B = \{1,2,3\}$, then the function $f\!: A \to B$ defined as $f(x) = 1$, that is, $f = \{(3,1), (2,1), (1,1), (5,1)\}$ is a constant function.

3.4 Composition of Functions

Let f: $A \rightarrow B$ and : $B \rightarrow C$. The composition of f and g, denoted by $g \circ f$, read as 'g of f results in a new function from A to C and is given by $(gof)(x) = g(f(x))$ for all x in A. Hence, the composition gof first applies f to map A into B, and it then employs g to map B to C. In other words, the range space of f becomes the domain space of g. Figure 3.10 illustrates the composition of the two function f and g.

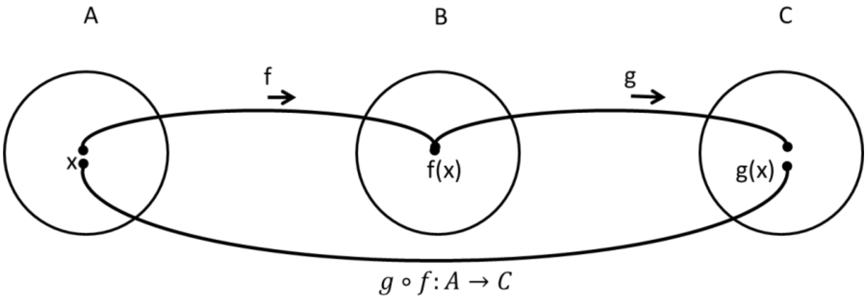


Fig 3.10: Composition of functions

Example 3.11 Let $A = \{1,2,3\}$, $B = \{a, b\}$ and $C = \{r, s\}$ and $f: A \rightarrow B$ be defined by $f(1) = a$, $f(2) = a, f(3) = b$ and $g: B \rightarrow C$ be defined by $g(a) = s, g(b) = r$.

Then $gof: A \rightarrow C$ is defined by

$$(gof)(1) = g(f(1)) = g(a) = s$$

$$(gof)(2) = g(f(2)) = g(a) = s$$

$$(gof)(3) = g(f(3)) = g(b) = r$$

3.5 Recursively defined Function

Sometimes it is difficult to define an object explicitly. However, it may be easy to define this object in terms of itself. This process is called recursion. Recursion refers to several related concepts in computer science and mathematics. One can use recursion to define sequences, functions, and sets. Let us consider the example 3.12.

Example 3.12 Let $S(n)$ be a sequence given as 1, 3, 9, 27,….. The sequence can explicitly be defined by the formula $S(n) = 3^n$ for all integers $n \geq 0$, but the sequence can also be defined recursively as follows.

(i) $S(0) = 1$
(ii) $S(n + 1) = 3\ S(n)$ for all integers $n \geq 0$

Here (ii) is the salient feature of recursion, namely, the feature of self-reference.

Problems for Exercise

1. Define a function from a set X into a set Y.

2. What is the domain, codomain, image of a function $f: X \to Y$?

3. Consider the function f from $P = \{a, b, c, d\}$ into $Q = \{x, y, z, w\}$ defined by Fig:3.11.

   a) the image of each element of P
   b) the image of f;
   c) the graph of f, i.e. write f as set of ordered pairs.

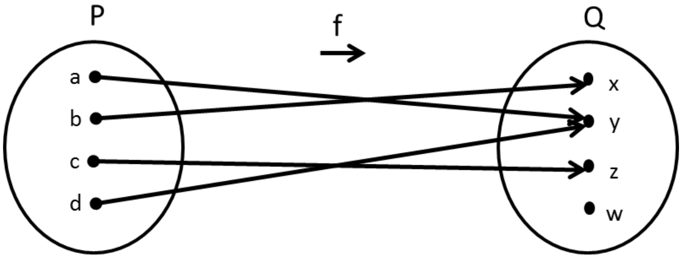


Fig 3.11

1. Let f assign to each state in India its capital city. Find:
   1.1. The domain of f, and
   1.2. $f(Uttarakhand)$, $f(Madhyapradesh)$, $f(Maharashtra)$

2. Let A be the set of polygons in the plane. Let $h: A \to N$ assign to each polygon P its number of sides. Find $h(triangle)$, $h(hexagon)$ and $h(trapezoid)$

3. Let $X = \{a, b\}$ and $Y = \{1, 2, 3\}$. Find the number n of functions:
   3.1. From X into Y and
   3.2. From Y into X.

4. Let $f: R -> R$ be the function which assigns to each real number x its square x^2. Describe different ways of defining f.

5. Determine whether the function $f(x) = x^2$ from the set of integers to the set of integers is one to one.

6. Determine whether the function $f(x) = x + 1$ is one to one.

7. Let f be the function from $\{a, b, c, d\}$ to $\{1, 2, 3\}$ defined by $f(a) = 3, f(b) = 2, f(c) = 1$ and $f(d) = 3$. Is f a onto function?

Block II: Unit I Propositional Logic

Learning Objectives
After completing this unit, the learner shall be able to:

- Define Propositional Logic;
- Determine the truth value of the given statements:
- Find the truth table of the proposition;
- Explain Tautologies and Contradictions;
- Obtain the Disjunctive Normal Form of the given statements;
- Obtain the Conjunctive Normal Form of the given statements;
- Find the Principal Disjunctive Normal Form (PDNF) of the given statements;

Reasoning revolves around reasoning. We can call it as set of rules while working with logical reasoning. The history shows great signs over the timeframe that reasoning is being a constant factor in derivation of knowledge and its representation. This is all the critical work of George Boole, a famous British mathematician which was handy taken by Gotlob Frege. Modern philosopher and Mathematician Bertand Russell with Alfred Whitehead had come out with the new definite set of logics which are common in use now.

Logic is based on truth and false of statements. However, there are many other factors which determine whether the statement is true of false. In spite of using individual statements, symbols had been used to represent arbitrary statements so that the results can be used.

The types of logic are

a. Propositional logic (logic of sentences)
b. Predicate logic (logic of objects)
c. Fuzzy logics
d. Uncertainty logics etc.

We are bit focused on Propositional logic and Predicate logics.

4.1 Propositional Logic

It is basically logic of sentences or in other words it is the logic of statements. It is the way through which one studies the joining or modifying entire propositions, statements or sentences.

One can also come up with more complicated propositions, statements or sentences as well as logical relationship or any property among them by deriving these basic statements. In this Propositional logic the basic statements can be called as indivisible units. Or in other words it cannot be further divided into statements. Hence propositional logic does not study these partial statements and hence doesn't come out with any logical properties and relations on them i.e. subject and predicate of a statement.

Among many propositional logics the truth functional propositional logic is perfect. It is based on logical operators and connectives which give rises to complex statements. Interestingly the truth value of these complex statements depend upon the truth value of simple statements and hence once can derive that the statement is true or false and not both.

This clearly shows that the sentences in this category are either true or false and also known as **propositional sentences**.

4.2 Propositions

As defined earlier a proposition is a sentence which is either true or false, but not both. We can also call like this if the proposition is true, then its truth value is true.

Example 4.1 Let us consider the following propositions along with their truth values:

1. Sky is blue True
2. Sun is yellow True

Here the truth values of both of the propositions are true.

Let us consider some propositions whose truth values are false.

3. Bus can Fly False
4. "4 + 4 = 9" False

Let us see some other examples where the sentences are neither true nor false.

5. "Open the gate"
6. "Is the tea hot?"
7. What is the temperature outside?

From these three sentences, we can come out with any result whether they are true of false, hence we can't call them propositions. . The 5th sentence is an order; the 6th sentence is a question, tea may be hot or cold. Similarly the 7th sentence asks about the temperature. Thus, no definite set of information (true/false) can we derive from these sentences, so we can say they are not propositional sentences.

4.2.1 Elements of propositions

Simple true of false statements are called as basic propositions. When these simple statements are joined or combined using the connectives, they form complex sentences. We can say that propositions and connectives are two fundamental elements of propositional logic.

Although there are many connectives, but are confining herewith five connectives, basic in nature are:

- NOT
- AND
- OR
- IF_THEN (or IMPLY)

- IF_AND_ONLY_IF

They are also denoted by the symbols: ¬, ∧, ∨, ⟶, ↔ respectively.

4.1.2 Propositional Variable

The name which is supposed to represent a proposition is commonly known as propositional variable. We can understand the same with some examples.

1. $P_{1:}$ The earth is a planet (True)
2. P_2 : Fish walks on road (False)
3. P_3 : 2+ 2 = 4 (True)

4.3 Basic Logic

The logic starts with a variable. In simple words we can define variable as a letter we use for an unknown object of any type.

Let us consider the equation $a + b = 10$. Here a and b are variables which denotes some values whose sum is 10.

Lets us take another example. We write a statement that

“Let B is superset of X”. This shows that B is a variable (where it is an unknown set) whereas we can’t call X as variable (it is a name given to set of all odd numbers).

But in case of example where $a + b = 15$ and $2a + 3b = 20$, both a and b are variables even though the values are inter dependent with each other.

It would be quite normal to say something like this: “Let x and y be two real numbers. Suppose that they satisfy the equations $x + y = 8$ and $x + 3y = 12$. Determine the values of x and y” It is then reasonable to call them variables, because initially no information is given about them. Further we have some relationships between x and y and from these relationships it is possible to deduce the exact values of x and y.

4.3.1 Logical Connectives

The words and phrases (or symbols) used to form compound propositions are called connectives. There are five basic connectives called, Negation, Conjunction, Disjunction, Implication or conditional and Equivalence or Biconditional.

Table 4.1 Connectives, symbols and symbolic form

Symbol used	**Connective word**	**Nature of statement**	**Symbolic form**
$\sim, \neg$	Not	Negation	$\sim p$
$\wedge$	And	Conjunction	$p \wedge q$
$\vee$	Or	Disjunction	$p \vee q$
$\Rightarrow, \longrightarrow$	If……then	Implication (or Conditional)	$p \longrightarrow q$
$\Leftrightarrow, \leftrightarrow$	If and only if	Equivalence (or Bi-conditional)	$p \leftrightarrow q$
$\equiv$	Equivalence	Equivalence of predicate	$p \equiv q$

Negation

If p is any proposition, the negation of p, denoted by $\sim p$ and read as not p, is a proposition which is false when p is true and true when p is false. Consider the statement

p: $Paris\ is\ in\ France$

Then the negation of p is the statement

$\sim p$: $Paris\ is\ not\ in\ France.$

Strictly speaking, negation is not a connective, since it does not join two statements and $\sim p$ is not really a compound statement. However, negation is a unary operation for the collection of statements, and $\sim p$ is a statement if p is considered a statement.

Example 4.2 The following propositions are equivalent:

p : All people are intelligent.

q : Every person is intelligent.

r : Each person is intelligent.

s : Any person is intelligent.

Example 4.3 The negation of the proposition

p : All students are intelligent

can be expressed in the following ways:

$\sim p$: Some students are not intelligent.

$\sim p$: There exists a student who is not intelligent.

$\sim p$: At least one student is not intelligent.

Example 4.4 The negation of the proposition

q : No student is intelligent .

is

$\sim q$: Some students are intelligent.

Note that "No student is intelligent" is not the negation of p; "All students are intelligent" is not the negation of q.

Conjunction

If p and q are two statements, then conjunction of p and q is the compound statement denoted by $p \wedge q$ and read as "p and q". The compound statement $p \wedge q$ is true when both p and q are true, otherwise it is false.

Example 4.5

Form the conjunction of p and q for each of the following.

a)	p: Ram is healthy	q: He has blue eyes
b)	p: It is cold	q: It is raining
c)	$p: 5x + 6 = 26$	$q: x > 3$

Solution:

a) $p \wedge q$: Ram is healthy and he has blue eyes
b) $p \wedge q$: It is cold and raining.
c) $p \wedge q$: $5x + 6 = 26$ and $x > 3$

Remarks

The symbol $\wedge$ has specific meaning which is corresponding to the connective 'and' appearing in the English language, although 'and' may also be used with some other meanings. In order to see the difference, consider the following three statements:

(i) Nilam is a girl and Arjun is a boy.
(ii) Shekhar switched on the computer and started to work
(iii) Kanchan and Sheela are friends.

In statement (i) the connective 'and' is used in the same sense as the symbol $\wedge$. In (ii) the word 'and' is used in the sense of 'and then' because the action described in "Shekhar started to work" after the action described in "shekhar switched on the computer". Finally, in (iii) the world 'and' is not at all a connective.

In logic we may combine any two sentences to form a conjunction, there is no requirement that the two sentences be related in content or subject matter. Any combinations, however absurd, are permitted, of course, we are usually not interested in sentences like ''Tanvir loves to play cricket", and 4 is divisible by 2'.

Disjunction

If p and q are two statements, the disjunction of p and q is the compound statement denoted by $p \vee q$ and read as "$p \;\; or \;\; q$". The statement $p \vee q$ is true if at least one of $p \;\; or \;\; q$ is true (The advertiser who writes 'The candidate must know English or Hindi, certainly would not reject a candidate if he knows both the languages). It is false when both $p \; and \; q$ are false.

The English word “or” can be used in two different senses – as an inclusive (“and/or”) or exclusive (“either/or”). For example consider the following statements.

1. p: He will go to Delhi or to Calcutta
2. q: There is something wrong with bulb or with the circuit.

In the compound statement (1), the disjunction of the statements p has been used in exclusive sense (p or q but not both); that is to say: one or the other possibility exists but not both. Clearly, a person can not do both.

In compound statement (2), the connective or is being used in an inclusive sense ($p\ or\ q\ or\ both$). In this case at least one of the two possibilities occurred, however both could have occurred. We shall always use ‘or’ in the inclusive sense unless it is stated.

Example 4.6 Assign a truth value to each of the following statements.

(i) $5 < 5 \bigvee 5 < 6$
(ii) $5 \times 4 = 21 \vee 9 + 7 = 17$
(iii) $6 + 4 = 10 \vee 0 > 2$

Solution:

(i) True, since one of its components $5 < 6$ is true
(ii) False, since both of its components are false.
(iii) True, since one of its components $6 + 4 = 10$ is true.

Example 4.7 If p: It is cold and q: It is raining.

Write simple verbal sentence which describes each of the following statements

a) $\sim p$ b) $p \wedge q$ c) $p \bigvee q$ d) $p \vee \sim q$

Solution:

a) $\sim p$: It is not cold
b) $p \wedge q$: It is cold and raining
c) $p \bigvee q$: It is cold or raining
d) $p \vee \sim q$: It is cold or it is not raining.

Implication (If . . . Then)
If p and q are two propositions, then ‘IF p THEN q’ is a proposition (denoted by $p \rightarrow\ q$). In $p \rightarrow\ q$, p is called hypothesis or premise and q is called conclusion or consequence.

Example 4.8:
Let p denote “It is cold” and let q denote “It rains”. Write the following statements in symbolic form

a. It rains only if it is cold.
b. A necessary condition for it to be cold is that it rain.
c. A sufficient condition for it to be cold is that it rain.

Solution:

a. $q \rightarrow p$
b. $p \rightarrow q$
c. $q \rightarrow p$

Bi-conditional (If and only if)

Let p and q are propositions. The proposition $p \leftrightarrow q$ is called bi-conditional and it is read as "p if and only if q" (or) "p iff q".

The truth value of $p \leftrightarrow q$ is true if both p and q are true or false; $p \leftrightarrow q$ is false if p and q have different truth values.

Example 4.9

Determine the truth value of each of the following statements:

a. Mumbai is in India if and only if $3 + 3 = 6$.
b. Mumbai is in India if and only if $3 + 3 = 7$.
c. Mumbai is in Australia if and only if $3 + 3 = 6$.
d. Mumbai is in Australia if and only if $3 + 3 = 7$.

Solution:

(a) and (d) are true since the sub-statements are both true in (a) and both false in (d). On the other hand, (c) and (b) are false since the sub-statements have different truth values.

Equivalence (Logical Equivalence '≡')

Two propositions p and q are said to be logically equivalent, or simply equivalent or equal, denoted by

$$p \equiv q$$

If they have identical truth values.

4.3.2 Truth Tables

The truth value of a proposition is either true (denoted by T) or false (denoted by F). A truth table is a table that shows the truth value of a compound proposition for all possible cases.

For example, consider the conjunction of any two propositions p and q . The compound statement $p \wedge q$ is true when both p and q are true, otherwise false. There are four possible cases.

1. p is true and q is true.
2. p is true and q is false.
3. p is false and q is true.
4. p is false and q is false.

There four cases are listed in the first two columns and the truth values of $\wedge q$, $p \vee q$ and $\sim p$ are shown in the table below:

Table 4.2 Truth tables of (a) $p \wedge q$, (b) $p \vee q$ and (c) $\sim p$

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

(a)

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

(b)

p	$\sim p$
T	F
F	T

(c)

The truth value of a compound depends only on the truth values of the statements being combined and on the types of connectives being used. Truth tables are especially valuable in the determination of the truth of connectives being used. Truth tables are especially valuable in the determination of the truth values of propositions constructed from simpler propositions. Note that the first columns of the table are for the variables $p,\ q,\ \dots\ \dots\ \dots$ and the number of rows depends on the number of variables. For 2 variables, 4 rows are necessary; for 3 variables, 8 rows are necessary; in general, for n variables, 2^n rows are required. The truth value at each step is determined from the previous stages by the definition of connectives. The truth value of the proposition appears in the last column.

Some Important Laws

1. Idempotent Law
   a. $p \vee p \equiv p$
   b. $p \wedge p \equiv p$
2. Associative Law
   a. $(p \vee q) \vee r \equiv p \vee (q \vee r)$
   b. $(p \wedge q) \wedge r \equiv p \wedge (r \wedge r)$
3. Commutative Law
   a. $p \vee q \equiv q \vee p$
   b. $p \wedge q \equiv q \wedge p$
4. De-Morgan's Law
   a. $\sim (p \vee q) \equiv \sim p \wedge \sim q$
   b. $\sim (p \wedge q) \equiv \sim p \vee \sim q$
5. Distributive Law
   a. $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$
   b. $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$
6. $\sim\sim p \equiv p$
7. $p \vee \sim p \equiv 1, p \wedge \sim p \equiv 0$
8. $p \vee 1 \equiv 1, p \wedge 1 \equiv p$

4.4 Tautologies and Contradictions

A compound proposition that is always true for all possible truth values of its variables or in other words, that contains only T in the last column of its truth table is called a tautology. A compound proposition that is always false for all possible values of its variables or in other words, that contains only F in the last column of its truth table is called a contradiction. Finally a proposition that is neither a tautology nor a contradiction is called a contingency.

Example 4.10 Propositions like

a) The professor is either a woman or a man.
b) People either like watching TVs or they don't.

are always true and are called tautologies.

Example 4.11 Propositions like

a) x is prime and x is an even integer greater than 8
b) All men are good and all men are bad

are always false and are called contradictions.

Example 4.12 Prove that the following propositions are tautology

a) $p \lor \sim p$ b) $\sim(p \land q) \lor q$ c) $p \rightarrow (p \lor q)$

Solution:

(a) The truth table of the given proposition is shown below. Since the truth value is TRUE for all possible values of the propositional variables which can be seen in the last column the table, the given proposition is tautology

Table 4.3 Truth table of $p \lor \sim p$

p	$\sim p$	$p \lor \sim p$
T	F	T
F	T	T

(b) We construct the truth table for the expression in question. It can be seen that for any possible assignment of p and q, the expression $\sim(p \land q) \lor q$ is true, which establishes that it is a tautology

Table 4.4 Truth table of $\sim(p \land q) \lor q$.

p	q	$p \land q$	$\sim(p \land q)$	$\sim(p \land q) \lor q$
T	T	T	F	T
T	F	F	T	T
F	T	F	T	T
F	F	F	T	T

(c) We construct the truth table of the given expression. It can be seen from the last column of the truth table that the expression is true for all possible assignments of p and q. Hence the proposition is tautology.

Table 4.5 Truth table of $p \rightarrow (p \vee q)$

p	q	$p \wedge q$	$p \vee q$	$p \rightarrow (p \vee q)$
T	T	T	T	T
T	F	F	T	T
F	T	F	T	T
F	F	F	F	T

4.5 Normal Forms

Let’s talk about normal forms. So far we have learned about the propositions, tautologies, contradictions and putting them in truth table. Surprisingly it is not always possible to construct the truth table when there are too many variables.

So the better method is to transform the expressions p and q to some form of expressions i.e. p' and q' in such a way that they can denotes something like $p \equiv q$.

The standard forms are popularly known as **normal forms** or in some places it can also be called as **canonical forms**.

Types of normal forms:

1. **Disjunctive normal forms**
2. **Conjunctive normal forms.**

For simplifying the things, we often use the words product for the logical connective conjunction and sum for logical connective disjunction.

We can take some other common terms like Elementary Product or Elementary Sum.

4.5.1 Elementary Product

It can be defined as a product of variable and their negation in a formula. For example, let us take two atomic variables p and q then p, $\sim p$, $\sim p \wedge q$, and $\sim p \wedge q \wedge \sim q$ are the examples of elementary product.

We are already familiar that any variable p, $p \wedge \sim p$ is a contradiction. Hence, if $p \wedge \sim p$ appears in the elementary product, then the product is definitely false.

This clearly shows that the statement a necessary condition for an elementary product to be identically false is that it should contains at least one pair of factors in which one is the negation of the other.

4.5.2 Elementary Sum

An elementary sum can be defined as the sum of variable and its negation. We can take example of p and q be any two variables. Then p, $\sim p$, $\sim p \vee q$, and $\sim p \vee q \vee \sim q$ can be called as some examples of elementary sum.

For an example we know for variable p, $p \vee \sim p$ is tautology. Hence, if $p \vee \sim p$ appears in the elementary sum, then the sum is identically true.

Hence, we can derive from this that for a statement the necessary condition for an elementary sum to be identically true is that it must contain at least one pair of factor in which one is the negation of the other.

4.5.3 Disjunctive Normal Form

We can define an expressive to be Disjunctive normal form or popularly known as DNF, when it is to be the sum of elementary products.

For example, $p \vee (q \wedge r)$ and $p \vee (\sim q \wedge r)$ are in disjunctive normal form while $p \wedge (q \wedge r)$ is not in disjunctive normal form.

Procedure to obtain a disjunctive normal form of a given logical expression
In order to attain a DNF through algebraic expressions, it can be worked in three steps.

a. We should remove all → and ↔ by an equivalent expression containing the connectives $\vee \wedge, and \sim$ only
b. Elimination of ~ before sums and products by using De Mogans law or by the double negation
c. Implying the distributive law until we obtained a sum of elementary product

Let's work out the above with a suitable example.

Example 4.13 Obtain the DNF of the following:

(a) $p \wedge (p \Rightarrow q)$ (b) $p \wedge (\sim p \Rightarrow (q \vee (q \Rightarrow \sim r)))$

Solution:

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

(b) $p \wedge (\sim p \Rightarrow (q \vee (q \Rightarrow \sim r))) \equiv p \vee (\sim p \Rightarrow (q \vee (\sim q \vee \sim r)))$
$\equiv p \vee (p \vee (q \vee (\sim q \vee \sim r)))$
$\equiv p \vee p \vee q \vee \sim q \vee \sim r$
$\equiv p \vee q \vee \sim q \vee \sim r$

4.5.4 Conjunctive Normal Form (CNF)

A logical expression if it consists of a product of elementary sum is called as conjunctive normal form (CNF). Let us understand the same with a suitable example.

Example 4.14 Find the conjunctive normal form of the followings:

(a) $p \wedge (p \Rightarrow q)$ (b) $[q \vee (p \wedge r)] \wedge \sim[(p \vee r) \wedge q]$

Solution:

(a) $p \wedge (p \Rightarrow q) \equiv p \wedge (\sim p \vee q)$ it is the CNF of the statement

(b) $[q \vee (p \wedge r)] \wedge \sim[(p \vee r) \wedge q] \equiv [q \vee (p \wedge r)] \wedge [\sim(p \vee r) \vee \sim q]$

$$\equiv [q \vee (p \wedge r)] \wedge [(\sim p \wedge \sim r) \vee \sim q]$$
$$\equiv (q \vee p) \wedge (q \vee r) \wedge (\sim p \vee \sim q) \wedge (\sim r \vee \sim q)$$

After understanding DNF and CNF we are going to learn about PDNF i.e. Principal disjunctive Normal form.

4.5.5 Principal Disjunctive Normal Form (PDNF)

Let p and q be two statement variables. If we create all possible formulae that consist of conjunction of p or $\sim p$ with q or $\sim q$ excluding the forms where a variable and its negation both appear and any form equivalent to previously obtained form, we are going to get the following forms:

$p \wedge q$, $\sim p \wedge q$, $p \wedge \sim q$, and $\sim p \wedge \sim q$

We can call these forms as *minterms* for the two variables p and q.

Interestingly we do find that all minterms are different. If there are n variables in a statement formula, then there will be 2^n minterms.

Let's find the minterms of the three variables p, q and r, they are

$p \wedge q \wedge \sim r$, $p \wedge q \wedge r$, $p \wedge \sim q \wedge r$, $\sim p \wedge q \wedge \sim r$, $\sim p \wedge q \wedge r$, $p \wedge \sim q \wedge \sim r$, $\sim p \wedge \sim q \wedge r$,and $\sim p \wedge \sim q \wedge \sim r$.

An equivalent formula consisting of disjunctions of min-terms alone for a given formula is commonly called as principal disjunctive normal form (PDNF). Let's do it with an example:

Example 4.15 Write the PDNF of $p \vee (p \wedge q)$

Solution:

$$p \vee (p \wedge q) \equiv (p \wedge T) \vee (p \wedge q) \quad (since \quad p \equiv (p \wedge T))$$
$$\equiv (p \wedge (q \vee \sim q)) \vee (p \wedge q) \quad (since \quad p \vee \sim p \equiv T)$$
$$\equiv ((p \wedge q) \vee (p \wedge \sim q)) \vee (p \wedge q) \quad (using\ distributive\ law)$$
$$\equiv (p \wedge q) \vee (p \wedge \sim q)$$

Example 4.16 Find the PDNF of the following statements:

1. $p \vee (p \to q)$
2. $p \vee (p \wedge q)$
3. $p \to q$

Solution: Let's try to find out the PDNF through truth table:

Finding of Principal Disjunctive Normal Form using Truth Table

Firstly, for every truth value T of the given formula in the truth table, write the minterm corresponding to the truth values of the variables. We all know that the minterm consists of the variable itself if its truth value is true and negation of the variable if its truth value is false.

The disjunction of these minterms is the PDNF of the given formula. The PDNF of $p \vee (p \wedge q)$ can be find as below:

Table 4.6 Truth table of $p \vee (p \wedge q)$and corresponding minterms

p	q	$p \wedge q$	$p \vee (p \wedge q)$	$Minterm$
T	T	T	T	$p \wedge q$
T	F	F	T	$p \wedge {\sim}q$
F	T	F	F	
F	F	F	F	

The truth table, clearly shows that only two truth values are true for the given formula.
Hence, the PDNF is $(p \wedge q) \vee (p \wedge {\sim}q)$.

4.5.6 Principal Conjunctive Normal Form (PCNF)

Let p and q be two statement variables. If we try to create all possible formulae that consist of disjunctions of p or ${\sim}p$ with q or ${\sim}q$ excluding the forms where a variable and its negation both appear in any form equivalent to previously obtained form, we will have the following forms:

$p \vee q$, ${\sim}p \vee q$, $p \vee {\sim}q$, and ${\sim}p \vee {\sim}q$

We can call these forms as *maxterms* for the two variables p and q. In PDNF we do have minterms but in PCNF we will have *maxterms.*

We can understand with an example for three variables p, q and r, the maxterms are: $p \vee q \vee {\sim}r$, $p \vee q \vee r$, $p \vee {\sim}q \vee r$, ${\sim}p \vee q \vee {\sim}r$, ${\sim}p \vee q \vee r$, $p \vee {\sim}q \vee {\sim}r$, ${\sim}p \vee {\sim}q \vee r$,and ${\sim}p \vee {\sim}q \vee {\sim}r$.

For a given formula, an equivalent formula consisting of conjunction of max-terms alone is popularly called as Principal conjunctive normal form (PCNF). We can understand it better with the help of following example:

Example 4.17 Write the PCNF of $p \wedge (p \vee q)$.

Solution:

$$
\begin{aligned}
& p \wedge (p \vee q) \equiv (p \vee F) \wedge (p \vee q) \quad (since \quad p \vee F \equiv p) \\
& \equiv (p \vee (q \wedge {\sim}q)) \wedge (p \vee q) \quad (since \quad q \vee {\sim}q \equiv F) \\
& \equiv ((p \vee q) \wedge (p \vee {\sim}q)) \wedge (p \vee q) \quad (using\ distributive\ law) \\
& \equiv ((p \vee q) \wedge (p \vee q)) \wedge (p \vee {\sim}q) \quad (using\ associative\ law) \\
& \equiv (p \vee q) \wedge (p \vee {\sim}q) \qquad (since \quad p \wedge p \equiv p)
\end{aligned}
$$

We will try to find the Principal Conjunctive Normal Form Using Truth Table
The PCNF of a given formula using the truth table can be find out as follows.

For any given formula in the truth table,for every truth value F we write the max-term corresponding to the truth values. As maxterm consists of the variable itself it its truth value false and the negation of the variable if its truth value is true.

The conjunction of these maxterms will be the PCNF of the given formula.

PCNF of $p \wedge (p \vee q)$ can be obtained as given in the table below:

Table 4.7 Truth table of $p \wedge (p \vee q)$ and corresponding maxterms

p	q	$p \vee q$	$p \wedge (p \vee q)$	$Maxterm$
T	T	T	T	
T	F	T	T	
F	T	T	F	$p \vee \sim q$
F	F	F	F	$p \vee q$

Hence, the PCNF is $(p \vee q) \wedge (p \vee \sim q)$.

In last we can say that PDNF and PCNF techniques are very useful in understanding the logics effectively.

Problems for Exercise

1. Let p be “He is tall” and let q be “He is handsome”. Write each of the following statements in symbolic form using p and q (Assume that “He is short” means “He is not tall”, i.e., $\sim p$).
    a. He is tall and handsome.
    b. It is false that he is short or handsome.
    c. He is tall but not handsome.
    d. He is neither tall nor handsome.
2. Determine the truth value of each of the following statements:
    a. 1+1=5 or 2+2=4
    b. 2+5=9 or 1+7=8
3. Find the truth table of the proposition $\sim (p \wedge \sim q)$.
4. Find the truth tables of the following:
    a. $p \wedge (q \vee r)$
    b. $(p \wedge q) \vee (p \wedge r)$
5. Prove the associative law : $(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$
6. Find a principle disjunctive normal form $\sim (p \vee q) \leftrightarrow (p \wedge q)$
7. Obtain a principal disjunctive normal form$(\sim p \vee \sim q) \rightarrow (\sim p \wedge r)$.
8. Obtain a principle conjunctive normal form of $(q \vee (p \wedge q)) \wedge \sim (p \vee r) \wedge q)$
9. Obtain a PDNF for $p \leftrightarrow q$
10. Obtain PDNF of $p \vee (\sim p \wedge \sim q \wedge r)$.

Block II: Unit II Inference

Learning Objectives

After completing this unit, the learner shall be able to:

- Define an Inference;
- State the rule of the inference used in the given arguments;
- Using modus ponens or modus tollens to make an argument for given statement;
- Check whether the given argument is valid or not;
- Translate the given statement into symbolic form;

An inference can be defined as a process or a method of making logical conclusions on the basis of premises known or expected to be true. The derived conclusion is considered to be natural.

The propositions that are assumed to be true are called hypotheses or premises. The proposition derived by the using the rules of inference is called conclusion. The process of deriving conclusions based on the assumption of premises is called a valid argument.

It can be explained as the illogical, but rational stuff via perceiving facts patterns, context for understanding. In other words, inference does not used to derive conclusions but unlocks the new avenues for inquiry. Inferences can be categorized in two types as:

1. Inductive Inference
2. Deductive inference.

Let us look at the example of inductive inference:

1. All Cricketers are fit.
2. Sachin is a Cricketer.
3. Therefore, Sachin is fit.

Here we can check truthfulness of the premises and conclusion, besides logic & inference are interrelated: does the truth of the assumption follow the premises?

To determine form of the inference is significant because on the basis of it, validity of an inference can be determined. However, the word “valid” itself depicts the form of inference rather than the truthiness of premises or the conclusion. It is possible that an inference can said to be valid even if some portion is false and may get invalid in spite of some portion of it is true. Thus, a valid form along with true premises will always have a true conclusion.

For instance,

1. All fruits come from trees.
2. Apple is a type of fruit.
3. Therefore, Apple comes from a tree.

The truthiness of conclusion is closely related to premises, too.

Now we bring an invalid form of inference.

1. All M are F
2. B is a F
3. Thus, B is a M

Now let’s see how it is invalid,

1. All mangoes are fruit. (Correct)
2. All Bananas are fruit. (Correct)
3. Therefore, Bananas are mangoes. (Wrong)

A false conclusion may also come out of a valid argument with false premises :

1. All Stars are Bright.
2. LED is Bright.
3. Thus, LED is a Star.

If in case a false conclusion is made through making a valid argument from a given premises still then an inference remains valid and follows the form of a correct inference.

A true conclusion from premises can be derived by using a valid argument.

1. All tall people are actors (although wrong)
2. Amitabh Bachhan is tall (right, valid)
3. Thus, Amitabh Bachhan is an actor (Right)

Let us look at the example of deductive inference:

Usually we read such news in newspaper that “A cricket team from a village in Uttarakhand surprises by winning game by game. The team even overthrows the Mumbai team”

Inference: The Village in Uttarakhand is not a village anymore.

Inference is made on the ground of some known facts: The village was remote and historically had never distinguished itself; the resources need for cricket was short due to the lack of cricket clubs and a coach for proper training. Large cities might field good teams due to the greater availability of high quality players and resources; and teams that can practice longer under the guidance of coach can reasonably be expected to be better.

5.1 Rule of Inference

A rule of inference can be state as a form of logic where a function takes premises and assesses their syntax to return a conclusion. For instance, the rule of inference said to be *modus ponens* considers two premises, one in the form “If p then q”, and another in the form “p”, that returns the conclusion “q”. The rule is valid as far as semantics of classical logic are concerned, in the sense that conclusion will be true provided premises are true.

Usually, a rule of inference retains truth is a kind of semantic property. It retains a common place in many-valued logic. But a rule of inference is strictly syntactic with no requirement to retain any semantic property. Typically, only recurring rules are significant; those provides a means to verify whether the given formulation is a conclusion of a given set of formulation according to the rule.

Well known rules of inference in propositional logic comprise modus ponens, modus tollens, and contraposition. Rules of inferences are used by first order predicate logic while treating logical quantifiers.

Standard form of rules of inference

In formal logic and other concerned domains, rules of inference are generally given in the subsequent standard form:

Premise 1 → Premise 2 → Premise n → Conclusion

It means that on every occasion during some logical induction the given premises may achieved, the particular conclusion can be considered as well.

The actual language of expression to explain both of the premises & conclusions is based on the perspective of the logical induction. For example, one may use logical formula as

$$A \to B$$

In prepositional logic, it is said to be *modus ponens* rule. Rules of inference are usually expressed as a model using syntactical variables. In the rule mentioned above, the syntactical variables A and B can assume any element of the universe to produce a countless set of inference rules.

Derivations can be expressed as proof method made up of a set of rules linked to one another to form a proof. Any derivation ends with only one conclusion that is the statement said to be derived. If in case, the premises are gone unfulfilled then the derivation is said to be a proof of theoretical statement "if the premises exists, then the conclusion also exist".

5.2 Modus ponens

Under propositional logic, modus ponens is a rule of inference that can be expressed as as "A implies B; A is declared as true, so therefore B should be true."

Modus Ponens are among the most frequently used notions in logic and should not be interpreted as a law in logic but it's a tool for the derivation of proofs that comprises the rule for definition and substitution as well. Though it is allowed to ignore a conditional statement from the logical derivation or an argument thus it is occasionally called as rule for detachment.

The conviction in the inference is the acceptance that if the previous statements are true, then the final conclusion will also true. In other words,

If A implies B and A is true, then B is true.

Example 5.1 Let us consider the following argument

"If it is mango, It must be a fruit"

"It is mango."

Thus, “It must be a fruit”

Formally*, Modus ponens* can be specified as:

$$\frac{A \to B, A}{\therefore B}$$

It means that whenever there is an implication as $A \to B$ and A comes in a line of proof then in place of A, B can be placed in the later line. B will be their only hint that is taken further in more complex derivation.

It has a close relationship with another logical form of argument, *modus tollens*. Both of them have similar but invalid forms that will be discussed later.

5.2.1 Formal Notation
Logical notaion may be used to depict modus ponens rule

$$A \to B, \qquad A \vdash B$$

Where, $\vdash$ is a logical sign means that B is a syntactic result of $A \to B$ and A in few logical methods.

$$((A \to B) \land A) \to B$$

where A, and B are propositions stated in some formal system.

5.3 Modus tollens

Under propositional logic, modus tollens is inference rule and valid argument form. It is an implementation of the common fact that in case of a statement which is true, then its contrapositive is also true.

Modus tollens is also well known as law of contraposition, confirms the inference from A implies B and the contradictory of B, to the contradictory of A.

Formally, modus tollens rule can be expressed as

$$\frac{A \to B, \neg B}{\therefore \neg A}$$

where $A \to B$ stands for the statement “ A implies B ” (and $\neg A \to \neg B$ is called the “contrapositive”). $\neg B$ stands for “it is not the occasion that B” (or in brief “not B”). Then, whenever “$A \to B$” and “$\neg B$” each appear by themselves as a line of a proof, then “$\neg A$” can validly be positioned on a later line.

Modus tollens has close relationship with *modus ponens*. There are two alike, but invalid, forms of argument: asserting the consequent and contradicting the predecessor.

5.3.1 Formal notation

In logical notations, modus tollens can be represented as:

$$A \rightarrow B, \ \ \neg B \vdash \neg A$$

Where, $\vdash$ is a logical sign in the sense that $\neg A$ is a syntactic result of $A \rightarrow B$ and $\neg B$ in some logical system

$$((A \rightarrow B) \wedge \neg B) \rightarrow \neg A$$

where A and B are propositions expressed in some formal system

Example 5.2

"If the smoke sensor detected fire, the fire alarm will ring"

"The fire alarm won't ring"

Thus, "no smoke was detected by the smoke sensor"

Assuming that the premises are equally true (the fire alarm will ring if it detects smoke, and does indeed won't ring), it follows that no smoke has been detected. This is a valid argument since it is not possible for the inference to be false if the premises are true. (It is feasible that there may have been the smoke that smoke sensor did not detected, but that does not nullify the argument; the first premise is "if the smoke sensor detects smoke." It means the more important thing is that whether the sensor detects the smoke irrespective of whether smoke is present there or not.

5.3.2 Relation to modus ponens

In propositional logic, **material implication** is a valid rule of replacement that allows for a conditional statement to be replaced by a disjunction if and only if the antecedent is negated. The rule states that p implies q is logically equivalent to $\sim p$ or q and can replace each other in logical proofs.

Each use of modus tollens can be adapted to a use of modus ponens and one use of substitution to the premise which is a material implication. For example:

If A, then B, (premise – material implication)

If not B, then not A. (Derived by Substitution)

Not B. (premise)

Thus, not A (derived by modus ponens)

Likewise, each use of modus ponens can be converted to use of modus tollens and substitution.

5.4 Validity

Logically, an argument is said to be valid if it is in such a form so that almost it become impossible for an inference to be false having premises true. It is not always essential that a valid argument needs to be true but if the assertion is true then it would assure that the conclusion based on that assertion will be true. It means that an argument schema is valid iff every argument of that logical structure is valid.

5.4.1 Validity of an argument

Though the logical consequence of the premises is the inference derived from it. So an argument's validity is based on the validity of the premises. Hence the truth value of an argument is the criteria on the basis of which it will be called as valid while on the other hand the negation of this condition is a contradiction.

Example 5.3 Let us consider the following argument:

All birds have wings.

parrot is a bird.

Then, parrot has wings.

Here truth values of the premises and conclusion are not responsible to make this argument valid but conclusion needs a logical context and that gives these two premises. Otherwise the argument would still remain valid where the truth value of premises and conclusion both are false.

The argument given in example 2.4 is of the same logical form as above in example 2.3. Difference is that the truth value of premises and conclusion is false but the argument would be equally valid.

Example 5.4 Let us consider the following argument:

All vegetables are green.

carrot is a vegetable.

Therefore, Carrot is green.

It doesn't matter how the universe is, but it is not always possible that these argument would give true premises concurrently with false conclusion.

Example 5.5 Let us consider the following argument:

All human are immortal.

Sachin is a man.

Therefore, Sachin is mortal.

Here, the conclusion is not drawn from the premises so it contradicts the derived logic and hence the argument is invalid while in general the conclusion can be considered as true.

From a standard point of view, whether an argument is valid is a matter of the argument's logical form. Numerous techniques are engaged by logicians to represent an argument's logical form. A simple example, applied to two of the above illustrations, is the following:

Let the letters 'H', 'M', and 'S' stand, individually, for the set of men, the set of mortals, and Sachin. Using these symbols, an argument may be summarized as:

All H are M.

S is a H.

Thus, S is a M.

Likewise, another argument becomes:

All H are not M.

S is a H.

Thus, S is a M.

An argument can formally called as valid if the conclusion is drawn on the basis of the premises then no matter whether the premises are true or not. On the other hand, if in an argument a conclusion is not drawn on the basis of the premises then no matter the premises could be true but the argument will be called as invalid.

5.4.2 Validity of Statements

A statement can be called valid, i.e. logical truth, if it is true in all interpretations.

5.5 Predicate Logic

In mathematical logic, predicate logic is common term used to represent the formal logical systems where it is comprises of the syntactical variables

There are two frequently used quantifiers are the

1. Existential ∃ ("there exists") quantifiers
2. Universal ∀ ("for all") quantifiers.

The variables may possibly be any elements in the universe under consideration, or possibly relations or functions over that universe of discourse. For instance, an existential quantifier over a function symbol would be inferred as modifier "there is a function".

The foundations of predicate logic were given unconventionally by Gottlob Frege and Charles Sanders Peirce.

Let us consider the following sentences:

1. Mohit is a student.
2. Shridhar is a student.
3. Seema is a student.

If we write the propositions for these three sentences, we will require three propositions. In the same way if we have a list of hundred students, then it is not suitable to write hundred propositions because the part ‘is a student’ of the sentence is repeated in all these sentences. Hence, it is better to assign a variable (say x) in place of the name of the student and keep the remaining part as it is, and define a set X of students from where x can take its values.

The sentence can be written as ‘x is a student’ in which the part ‘is a student’ is called *predicate*, and the set X is called the universe of discourse for x. The complete sentence is called predicate on x. A predicate on x is denoted by the symbols $P,\ Q,\ R$ and so on, with x in braces, that is, $P(x), Q(x), R(x)$, and so on, respectively.

For example,

$$P(x)\colon x\ is\ a\ student$$

$$Q(x)\colon x\ is\ an\ animal$$

A predicate can be defined without defining its universe of disclosure. In this case, the variable can take any value from the universal set. A predicate can also be defined over more than one variable. For example, consider the predicate on two variables.

$$P(x,y)\colon x\ is\ greater\ than\ y$$

If we replace x by 6 and y by 3, then it becomes a proposition ‘$6\ \ is\ \ greater\ \ than\ \ 3$’ whose truth value is $true$.

5.6 Quantification

Let us first assume the following sentence:

Rajesh is brilliant and Mohit is brilliant and Akansha is brilliant.

If we form a set A of three students, then the sentence can be transcribed as follows:

All the students of the set A are brilliant.

For writing a representational form of the sentence, we need a predicate on a variable x like $P(x)$: $x\ \ is\ \ brilliant$, and the domain of x (called universe of discourse) defined as the set A, and a symbol for the phrase ‘for all’. The symbol is called quantifier. Thus, quantifier is a symbol that quantifies the variables. At the time when we use quantifier before a predicate, the predicate becomes a proposition.

In logic, an idea that states the quantity of subjects in the domain of discourse assigned with a symbol and satisfies an open formula is called ***quantification*.**

Two fundamental kinds of quantification in predicate logic are:

1. Universal quantification

2. Existential quantification

5.6.1 Universal Quantification

It is used when a statement is true for all values given in the universe of discourse. It is denoted by the symbol ∀. The universal quantification of $P(x)$ is the statement

$P(x)$ for all values x in the universe of discourse and is denoted by $\forall x P(x)$. We read $\forall x P(x)$ as 'for all $xP(x)$' or 'for every $xP(x)$'.

Note that $\forall x P(x)$ is true when $P(x)$ is true for every x and is false when there is any x for which $P(x)$ is not true.

Example 5.6 Let $P(x)$: $x\ is\ even\ number$ and universe of discourse for x is the set $\{1,2,3,4\}$. Find the truth value of $\forall x P(x)$.

Solution: As every number in the set is not an even number, the statement $\forall x P(x)$ is false.

Example 5.7 Let $P(x)$: $x \neq 5$ and universe of discourse for x is the set $\{1,2,3,4\}$. Find the truth value of $\forall x P(x)$.

Solution: As for every number x in the set $x \neq 5$, the statement $\forall x P(x)$ is true.

5.6.2 Existential Quantification

The existential quantifier is used on the occasion of a statement is true for some values given in the universe of discourse. It is represented by the symbol ∃. The existential quantification of $P(x)$ is the statement

There exists some x in the universe of discourse such that $P(x)$ and it is symbolized by the symbol $\exists x P(x)$.

Note that $\exists x P(x)$ is true when $P(x)$ is true for at least one value of x in the universe of discourse and is false when $P(x)$ is false for every x in the universe of discourse.

Example 5.8 Let $P(x)$: $x\ is\ even\ number$ and universe of discourse for x is the set $\{1,2,3,4\}$. Find the truth value of $\exists x P(x)$.

Solution: As some numbers in the set are even numbers, the statement $\exists x P(x)$ is true.

Example 5.9 Let $P(x)$: $x > 5$ and universe of discourse for x is the set $\{1,2,3,4\}$. Find the truth value of $\exists x P(x)$.

Solution: As none of the number in the set is greater than 5, the statement $\exists x P(x)$ is false.

Problems for Exercise:

1. State the rule of the inference used in the following arguments
   a. If it rains, then the schools are closed; it rains. Therefore, the schools are closed.
   b. If it rains, then the schools are closed; the schools are not closed. Therefore it does not rain.
2. Using modus ponens or modus tollens, make an argument for each one in the following.
   a. If this student is honest, she will not try to cheat when she takes a test.
   This student tried to cheat on a test.
   Therefore, ________________________ by modus ________
   b. If it is raining today, I will take my umbrella.
   It is raining today.
   Therefore, _________________________by modus ________
   c. $(a \lor b) \to c$
   b
   Therefore, __________________________by modus ________
   d. I always bring my lunch on Friday.
   I will buy my lunch today.
   Therefore, __________________________by modus ________
3. Supply the missing statement or reason in the following
   a. $p \to \sim q$; p ∴ _______
   b. $\sim p \to q$; $\sim p$ ∴_______
   c. $(\sim p \lor q) \to \sim(q \land r)$; $\sim p \lor q$ ∴ _________
   d. $(\sim p \land q) \to (q \land \sim r)$; $\sim p \land q$ ∴ _________
   e. $(\sim p \lor q) \to \sim(q \land r)$; $q \land r$ ∴ _________
   f. $(\sim p \land q) \to (q \land \sim r)$; $\sim (q \land \sim r)$ ∴ _________
4. Check whether the argument is valid or not
   a. If I plant a tree, then I will get dirt under my nails. I didn’t get dirt under my nails. Therefore, I didn’t plant a tree.
   b. If I don’t change my oil regularly, my engine will die. My engine died. Thus, I didn’t change my oil regularly
   c. If I don’t tie my shoes, then I trip. I didn’t tie my shoes. Hence, I tripped.
   d. All racers live dangerously. Arnav is a racer. Therefore, Arnav lives dangerously
5. Translate the following into symbolic form:
   a. Everybody loves him
   b. Somebody cried out for help and called the police.
   c. Nobody can ignore him.
6. State whether the following are true or false, where $x, y\ and\ z$ range over the integers.
   a. $\forall x, \exists\, y.\ (2x - y = 0)$
   b. $\forall x, x < 10 \Rightarrow \forall y, (y < x \Rightarrow y < 9)$
   c. $\exists\, y. \exists\, z. y + z = 100$
7. Formalize the following (over the real numbers):
   a. Negative numbers don’t have square roots
   b. Every positive number has exactly two square roots

Block II: Unit III Notion of Proof

Learning Objectives
After completing this unit, the learner shall be able to:

- Verify the equality of two sets using a truth table;
- Verify the equality of two sets using Venn Diagram;
- Prove with contradiction method;
- Verify whether or not a given proposition formula is a tautology;
- Disprove the given statements by finding a counterexample;

6.1 Notion of Proof

Here all the discussion is about diverse methods of proof. Proving a theorem or a mathematical statement is fundamentally proving the validity of an argument. So far, equivalences and implications in propositional logic were in our discussion. We shall use some of these equivalences and implications to describe various methods of proof. Before defining the different methods of proof, we shall discuss some terms used to represent the statements.

Theorem:

A theorem is a statement, fact, or consequence that can be shown to be true.

Proposition:

It is deliberated as less significant theorem.

Occasionally, to prove the theorem, we first prove some parts of the theorem individually, and then those results are used to prove the theorem.

Lemma:

A lemma is deliberated as a less significant theorem that is used to prove other theorems.

Corollary:

A corollary is a theorem that can be proved straight from a theorem that has been proved.

Generously, a theorem is a valid argument comprises of some premises and an inference, or more precisely, it may be inferred as the universal quantification of a conditional statement. In few cases, a theorem may be a logical statement as well.

6.2 Proof by implication

As we know $p \vdash q$ means there is a proof of q by put on inference rules to p, while $p \rightarrow q$ states that q holds each time p does. These are not the similar things: provability ($\vdash$) is separate to the theory (it's a declaration about whether a proof exists or not) while implication (→) is inside (it's a logical linkage for making compound propositions). But most of the time they mean almost equivalent.

For instance, suppose that $p \to q$ can be proved without any assumptions:

$$\vdash p \to q$$

Since we can always take no notice of extra premises, we get

$$p \vdash p \to q$$

And thus

$$p \vdash q, p \to q$$

which gives

$$p \vdash q$$

by put on modus ponens to the right-hand side.

So we can go from $\vdash p \to q$ to$p \vdash q$.

This shows that provability is in a sense weaker as compared to implication: it holds (supposing modus ponens) whenever implication does. But we frequently don't use this fact much, since $p \to q$ is a much more suitable statement than $p \vdash q$.

6.3 Converse

In logic, the converse of an implicational statement is the consequence of reversing its two parts. For the implication $p \to q$, the converse is $q \to p$

Consider S be a statement of the form p implies q ($p \to q$). Then the converse of S is the statement q implies p ($q \to p$). Usually, the verity of S says nothing about the verity of its converse, unless the predecessor p and the resultant q are logically equivalent.

For instance, assume the true statement "If I am a human, so I am mortal." The converse of that statement is "If I am mortal, so I am a human," which is not essentially true.

On the other side, the converse of a statement with mutually inclusive terms resides true, given the truth of the original proposition. Therefore, the statement "If I am a bachelor, then I am a single man" is logically equivalent to "If I am a single man, then I am a bachelor."

A truth table depicts that S and the converse of S are not logically equal unless both terms imply each other:

p	q	$p \to q$	$q \to p$
T	T	T	T
T	F	F	T
F	T	T	F
F	F	T	T

Going from a statement to its converse is the misconception of asserting the resultant. Though, if the statement S and its converse are equal (i.e. if p is true if and only if q is also true), then asserting the resultant will be valid.

6.4 Inverse

In logic, an inverse is a type of conditional sentence which is an instantaneous inference made from a different conditional sentence. Every conditional sentence has an inverse: the contrapositive of the converse. The inverse of $p \to q$ is therefore $\neg p \to \neg q$.

For instance, replacing propositions in natural language by logical variables, the inverse of the conditional proposition, “If it’s raining, then game will be abandoned” is “If it’s not raining, then game will not be abandoned”.

Since a double negation has no logical consequence, the inverse of the inverse is logically same to the original condition.

The inverse and the converse of a conditional are logically same to each other, just as the conditional and its contrapositive are logically same to each other. But the inverse of a conditional is not inferable from the conditional.

For instance, “If it’s not raining, then game will not abandoned” cannot be inferred from “if it’s raining, then game will abandoned”. It could easily be the case that game will abandoned no matter what the weather is.

6.5 Contrapositive

Logically, contraposition rule states that a condition statement is reasonably same to its contrapositive. The contrapositive of the statement has its predecessor and resultant inverted and flipped: the contrapositive of $p \to q$ is thus $\neg q \to \neg p$.

For instance, the proposition “All Mangoes are fruits” can be reaffirmed as the condition “if somewhat is Mango, then it is fruit”. At this moment, the law states that statement is indistinguishable to the contrapositive ”If somewhat is not fruit, then it is not mango”

In order to verify an implication, a proof can be specified by its contrapositive that statement.

Example 6.1

Prove that if $p + q \geq 93$, then $p \geq 47$ or $q \geq 47$, p and q being positive integers.

Solution

There looks to be no way to verify the given fact directly. Instead, one can prove by taking the contrapositive: not ” $p \geq 47$ or $q \geq 47$” implies not “$p + q \geq 93$”. By De Morgan’s law, the negation of ” $p \geq 47$ or $q \geq 47$” is “not $p \geq 47$ and $q \geq 47$” i.e., “$p \leq 46$ and $q \leq 46$” So the contrapositive proposition is if “$p \leq 46$ and $q \leq 46$” then $p + q \leq 93$. This follows immediately from property of inequalities: $a \leq c$ and $b \leq d$ imply that $a + b \leq c + d$ for all real numbers a, b, c, d

6.6 Negation

Logically, negation, also known as logical counterpart, is an act that takes a proposition p to another proposition "$not\ p$", written as $\neg p$, which is inferred instinctively as being true when p is false and vice versa.

Proof of negation is a kind of inference rule which clarifies how to prove a negation:

To prove $\neg p$, assume p and derive illogicality.

Example 6.2

Prove that $\sqrt{2}$ is not rational.

Solution:

Suppose $\sqrt{2}$ is rational and it is equivalent to a division a/b with a and b comparatively prime. Then we would get $a^2 = 2b^2$, hereafter a^2 is even and so is a. Write $a = 2c$ and plug it back in to get $2c^2 = b^2$, from which we determine that b is even as well. This is a contradiction ever since a and b were assumed to be comparatively prime.

6.7 Contradiction

It consists of a logical inconsistency between two or more propositions. It occurs when the propositions, taken together, yield two inferences which form the logic, typically opposite inversions of one another.

Proof by contradiction is based on the fact that a statement is either true or false but not both at the same time. We get at a contradiction when we arrive at a situation where we say that a statement is both true and false at the same time. This shows that our initial assumptions are inconsistent.

To prove that a statement p is true, we assume that $\neg p$ is true, and taking $\neg p$ as premise, we draw a contradiction F as the conclusion. $\neg p \Rightarrow F$ proves that $\neg p \rightarrow F$ is true; thus, $\neg p$ must be false, that is, p must be true. We can summarize the steps as follows:

1. Assume that p is true.
2. Using this assumption show a contradiction.

Example 6.3

Prove the statement 'if $3m + 1$ is even, then m is odd' utilizing the method of proof by contradiction.

Solution

Here $p: 3m + 1\ is\ even$ and $q: m\ is\ odd$.

Let us consider that p is true and $\sim q$ is true.

Assume, $m\ is\ even$ and $3m + 1\ is\ even$.

Consider, $m = 2a$ for some integer, then

$$3m + 1 = 3.2a + 1 = 6a + 1$$

Since $6a = 2(3a)$

This infers that $6a\ is\ an\ even\ number$

$\Rightarrow 6a + 1\ is\ an\ odd\ number$

$\Rightarrow 3m + 1\ is\ an\ odd\ number$

This is a contradiction to the statement that $3m + 1\ is\ even$. Hence $m\ is\ not\ even$, i.e., $m\ is\ odd$. This proves the statement 'if $3m + 1\ is\ even, then\ m\ is\ odd$'.

6.8 Direct Proof

In mathematics and logic, a direct proof is a way of confirming the truth or falsehood of a given statement by a straightforward combination of well-known facts, usually axioms, existing theorems, without making any further assumptions.

In direct proof, we can restate the theorem or statement in the form of conditional statement $p \rightarrow q$. We start with the statement that p is true and then use the rules of inferences with given axioms or already proved theorems and definitions to show that q is also true.

Example 6.4

Show that the square of an even number gives an even number.

Solution

First, we will reorganize the sentence. We have to verify 'if $k\ is\ an\ even\ number$, then $k^2\ is\ also\ an\ even\ number$'.

Here, p: $k\ is\ an\ even\ number$.

And q: $k^2 is\ an\ even\ number$.

Let us assume that $k\ is\ an\ even\ number$. Then we can write $k = 2n$, where $n \in Z$

$$k^2 = (2n)^2 = 4n^2 = 2(2n^2)$$

This implies k^2 is an even number.

Example 6.5

Show that the sum of two odd integers is an even number.

Solution

Let m and n be odd integers.

Here p: $m\ is\ odd\ and\ n\ is\ odd$.

q: $m + n \ is \ even$.

Let us assume p is true, i.e., m and n are odd integers.

As m and n are odd integers, we can state that $m = 2a + 1$ and $n = 2b + 1$ for some integers a and b

Now $m + n = 2a + 1 + 2b + 1$

$= 2(a + b) + 2$

$= 2(a + b + 1)$

Hence, $m + n \ is \ even \ number$.

6.9 Proof by using truth table

In the method of proving by using truth tables, the validity of propositional formulas is determined with respect to Boolean interpretations. Specifically, for discerning whether a propositional formula for Boolean interpretations is a tautology.

Initially, the truth table of the various logical connectives has to create.

One line for each Boolean interpretation has to be written of the set of variables that we are opting. If there are n variables that we are opting then the amount of lines will be 2^n.

There are as a result, two lines in the truth table for the only non-trivial unary connective

p	$\sim p$
T	F
F	T

. . and four lines in the truth table for the binary connectives

p	q	$p \wedge q$	$\sim(p \wedge q)$	$\sim(p \wedge q) \vee q$
T	T	T	F	T
T	F	F	T	T
F	T	F	T	T
F	F	F	T	T

6.9.1 Proof of tautology

Truth tables can be used to verify whether or not a given proposition formula is a tautology for Boolean interpretations.

Let p be a proposition formula we wish to validate.

Afterward, determine its truth table.

In the column under the main connective of p, it can found the truth value of p for every Boolean interpretation.

- If this comprises nothing but T, then p is a tautology
- If this comprises nothing but F, then p is a contradiction
- If this comprises T for some Boolean interpretation and F for others, then p is a liable statement.

Example 6.6

Whenever $p \Rightarrow q$ and $q \Rightarrow r$ are established as true, then $p \Rightarrow r$ is accepted as true

Solution

p	q	r	$p \Rightarrow q$	$q \Rightarrow r$	$p \Rightarrow r$
T	T	T	T	T	T
T	T	F	T	F	F
T	F	T	F	T	T
T	F	F	F	T	F
F	T	T	T	T	T
F	T	F	T	F	T
F	F	T	T	T	T
F	F	F	T	T	T

Both premises are true as seen in the first, fifth, seventh, and eight rows of the truth table. Subsequently in each case the conclusion is also true, the argument is valid.

This rule is valid rule of inference due the implication

$(p \Rightarrow q) \wedge (q \Rightarrow r) \Rightarrow (p \Rightarrow r)$ is a tautology

There are numerous arguments in mathematics comprises chain of if-then statements due to the fact that some statement implies a second and the second implies a third, anyone can determine that first statement implies the third one.

6.10 Proof by counter example

In the field of logic and especially in its applications to mathematics, a counterexample is an exception to a proposed general rule or law. For instance, consider the proposition “all fruits are sweet”. Because this statement claims that a definite property (sweetness) holds for all fruits, even a single example of acerbic fruit will prove it false. Thus, any acerbic fruit is a counterexample to “all fruit are sweet”. More specifically, a counterexample is a particular instance of the falsity of a universal quantification (a “for all” statement).

Proof technique

Let X be the statement:

$\forall m \in K : Q(m)$ (For all the elements m of a given set K, the property Q holds.)

Such a statement need not necessarily be true.

Let Y be the statement:

$\exists n \in K: \neg Q(n)$ (There exists at least one element n of the set K such that the property Q does not hold.)

It follows immediately by De Morgan's laws that if Y is true, then X must be false.

Such a statement Y is stated to as a counterexample to X.

The method to prove or disprove a statement in the form of X by verifying either truth or the falsehood of a statement in the form of Y is said to be proof by counterexample.

Example 6.7

Statement: It is not necessary that all linear functions in one variable should perpendicular to one another.

Proof:

To establish that this is true, we need to find a pair of linear functions in one variable that are not perpendicular. We recommend the following counterexample:

$$f_1(k) = 3k + 4 \qquad f_2(k) = 2k - 1$$

To observe, these two linear functions are not perpendicular, we notice that the slope of the first function is 3. Thus, the slope of a perpendicular line must be $-\frac{1}{3}$. However, the slope of $f_2\ is\ 2$, not $-\frac{1}{3}$. So, it is not necessary.

Note: While proving by counterexample only stating the counterexample is not sufficient but explaining why it is a counterexample is also must.

Problems for Exercise:

1. Prove that the following sets are equal. Verify it with a truth table or a Venn Diagram. You may assume that *A*, *B*, and *C* are nonempty sets. Also assume that *U* is the universe.
    a. $A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$
    b. $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
    c. $A^c - B = (A \cup B)^c$
    d. $A - B^c = A \cap B$
    e. $A - B) \cup (A \cap C) = A - (B - C)$
2. Prove that if A and B are finite sets then $|A \cup B| \leq |A| + |B|$ and that equality holds when $A \cap B = \emptyset$.
3. Prove the following by contrapositive:
    a. An integer n is even if and only if $n + 1$ is odd.
    b. If n and m have the same parity then $n + m$ is even
4. Prove with contradiction method that $\sqrt{2}$ is irrational.
5. Show that this identity holds for all positive integers:
$$1 + 2 + 3 + \cdots + n = \frac{n(n+1)}{2}$$
6. Show that $n! > 2^n for\ n \geq 4$
7. Prove that every positive integer n:
$$2 + 4 + 6 + \cdots + 2n = n(n+1)$$
8. Disprove the following statements by finding a counterexample for each of them:
    a. ∀ real numbers a and b, if $b^2 > a^2$, then $b > a$
    b. ∀ real numbers $x, y\ and\ z$, if $x > y$, then $xz > yz$
9. Prove that there is a prime number between 45 and 54.
10. Prove that:
Let m and n be integer. Then, there is no integer k such that
$(3m + 2)(3n + 2) = 3k + 2$

Block III: Unit I Combinatorics

Learning Objectives
After completing this unit, the learner shall be able to:

- Explain the features of combinatorics;
- Define Mathematical Induction;
- Apply the principle of mathematical induction to prove statements;
- Apply recursion to define sequences, functions, sets;
- Explain recursive definition of the factorial function;
- Apply Permutations and Combinations techniques to solve mathematical problems;

In many discrete problems, we are challenged with problem of counting. *Combinatorics* is the branch of mathematics deals with the study of problems concerning the counting and discrete structures those are finite or countable.

Features of combinatorics

- ✓ The configurations of a specified class and size get counted.
- ✓ Helps in taking decision when definite criteria are encountered and construction and exploration of objects meeting those criteria.
- ✓ Evaluating ‘largest’, ‘smallest’, or ‘optimal’ objects.
- ✓ Solution to the combinatorial problems can be obtained by applying algebraic techniques.

Combinatorics has many applications in mathematical optimization, computer science etc. Graph theory is one of the oldest and most frequently accessible part of combinatorics having numerous associations with other areas. Estimations in analysis of algorithm and finding formulas is significant application of combinatorics in the field of computer science.

Those who study combinatorics are called as combinatorialist or a combinatorist.

7.1 Mathematical Induction

Mathematical induction is a technique by virtue of which one can prove mathematical statements involving positive integers. The word induction means the method of inferring a general statement from the validity of particular cases.

Before describing the method of mathematical induction, let us try to understand its power. To do this, let us consider the statement:

$$1+2+3+\cdots\ldots\ldots\ldots\ldots\ldots\ldots\ldots+n=\frac{n(n+1)}{2}$$

It is easy to check that this statement is true for $n=1$, for $n=2$ and $n=3$ etc.

From the above, one cannot conclude that the statement is true for all positive n as one can never be sure that the statement does not fail for some untried value of n. But it is also impossible to substitute infinite number of possible values of n. Mathematical induction reduces the proof to a finite number of steps and guarantee that there is no positive n for which the statement fails to be determined.

A formal statement of principle of mathematical induction can be stated as follows.

Let $S(n)$ be a statement that involves positive integer $n = 1,2,3,$................ Then $S(n)$ is true for all positive integer n provided that

1. $S(1)$ is true
2. $S(k+1)$ is true whenever $S(k)$ is true.

So, there are three steps of proof using the principle of mathematical induction.

Step 1(Inductive base): Verify that $S(k)$ is true.

Step 2(Inductive hypothesis): Assume that $S(k)$ is true for an arbitrary value of k.

Step 3(Inductive Step): Verify that $S(k+1)$ is true on the basis of the inductive hypothesis.

Note (Change of inductive base): The principle of mathematical induction defined above begins at $n = 1$ and proves the statement for $n = k + 1$ assuming that the statement is true for $n = k(k \geq n_0)$

Example 7.1 Show that

$$1^2 + 2^2 + 3^2 + \cdots \ldots\ldots\ldots\ldots\ldots\ldots + n^2 = \frac{n(n+1)(2n+1)}{6}, n \geq 1$$

by mathematical induction.

Solution: Let $S(n)$ be the given statement

1. Inductive base: For $n = 1$ we have

$$1^2 = \frac{1(1+1)(2+1)}{6} = 1$$

So, $S(1)$ is true.

2. Inductive hypothesis: Assume that $S(k)$ is true i.e,

$$1^2 + 2^2 + 3^2 + \cdots \ldots\ldots\ldots\ldots\ldots + k^2 = \frac{k(k+1)(2k+1)}{6}$$

3. Inductive Step: We wish to show the truth of $S(k+1)$i.e.,

$$1^2 + 2^2 + 3^2 + \cdots \ldots\ldots\ldots\ldots\ldots + (k+1)^2 = (k+1)(k+2)(2k+3)/6$$

which has been obtained by substituting $k+1$ for n is $S(n)$

Now, $1^2 + 2^2 + 3^2 + \cdots \ldots\ldots\ldots\ldots\ldots\ldots. + (k+1)^2 = (1^2 + 2^2 + 3^2 + \cdots \ldots\ldots + k^2) + (k+1)^2$

$$= \frac{k(k+1)(2k+1)}{6} + (k+1)$$

$$= (k+1)\left[\frac{2k^2 + 7k + 6}{6}\right]$$

$$= \frac{(k+1)(k+2)(2k+3)}{6}$$

which is $S(k+1)$. That is $S(k+1)$ is true whenever $S(k)$ is true.

by the principle of mathematical induction $S(n)$ is true for all positive integer n

In some cases, the principle of mathematical induction is insufficient to prove certain statements. Thus, we have another form of mathematical induction known as strong form of mathematical induction.

Principle of strong mathematical induction

Let $S(n)$ be a statement defined on positive integers $n \in N$ such that it has the following properties:

1. $S(m)$ is true for some $m \in N$.
2. Whenever $S(m), S(m+1), S(m+2), \ldots\ldots\ldots, S(k)$ are true, $S(k+1)$ is true, where $k \geq m$.

Then, $S(n)$ is true for all natural numbers $n \geq m$

The principle of strong mathematical induction has more assumptions than simple mathematical induction principle. Although the strong form of principle of mathematical induction appears to be different from the weak form, the two forms are actually equivalent because each can be obtained from the other. So, we can use either form of the mathematical induction.

7.2 Recursive Mathematical Definition

Sometimes it is difficult to define an object explicitly. However, it may be easy to define object in terms of itself. This repetition of method in a self-similar way is known as recursion. Recursion refers to several related concepts in computer science and mathematics. One can use recursion to define sequences, functions, sets.

A problem can be clearly expressed through recursion and hence, it is a powerful tool for solving problem. A sequence or functions can be described by a recursive procedure and a recursive definition provides an easy way to find the successive terms of the sequence or values of the function.

Example 7.2 Let us consider the sequence$1,3,9,27, \ldots \ldots \ldots$ The sequence can be defined explicitly by the formula $S(n) = 3^n$ for all integers $n \geq 0$, but the sequence can also be defined recursively as follows.

(i) $S(0) = 1$

(ii) $S(n + 1) = 3\, S(n)$ for all integer $n \geq 0$

Here (ii) is the salient feature of recursion, namely the feature of self-reference.

Example 7.3 Write the recursive definition of the following sequences:

$$2,4,8,16, \ldots \ldots \ldots \ldots$$

Solution: Let the n^{th} term of the sequence be denoted by $S(n)$. The first term of the sequence is 2, and each successive term can be obtained by multiplying the preceding terms by 2. Thus, the sequence $S(n)$ can be defined as follows:

$$S(0) = 1$$

$$S(n + 1) = 2\ \ S(n), \qquad n \geq 2$$

7.2.1 Recursively Defined Functions

A function whose domain is the set of non-negative integers can be defined recursively using the recursive definition. The basis step defines the function for some primitive values and the recursive step provides a way to calculate the value of the function for the other integers.

Example 7.4 Write a recursive definition of the function $f(x) = 2^x$ defined from the set of natural numbers (including 0) to the set of natural numbers.

Solution: Since $f(0) = 1$ and $f(x + 1) = 2^{x+1} = 2.2^x = 2.\, f(x)$ the function can be defined recursively as follows:

$$f(x) = \{\begin{matrix} 1 & for\ x = 0 \\ x.f(x-1) & for\ x \geq 1 \end{matrix}$$

7.2.2 Recursively Defined Sets

A set is said to be recursively defined if the elements of the set can be defined using the recursive definition. The basis step defines the primitive elements of the set and the recursive definition generates the other elements of the set.

Example 7.5 Write the recursive definition for the elements of the following set:

$$A = \{1, 4, 7, 10, \ldots \ldots\}$$

Solution The first term is 1, and each successive term can be obtained from the previous term by adding 3. Thus, the elements of the set A can be defined recursively as follows:

(i) $1 \in A$

(ii) If $x \in A, then\ x + 3 \in A$

7.3 Basics of Counting

Let us consider the representation of characters in a computer. One bit (0 or 1) can represent two characters, and two bits (00, 01, 10, 11) can represent four characters. Using the product rule, it can be calculated that n bits are used to represent 2^n characters. This calculation will help us determine the number of bits required to represent n characters and amount of memory needed to represent them.

Combinatorics is concerned with arrangements and selection of objects. It plays an important role in various problems in discrete mathematics such as the generation of different codes and passwords from a set of given symbols, generation of different groups from a set of given objects, complexity of algorithms, and calculation of probabilities of events.

Here, we shall discuss two basic counting principles – the product rule and the sum rule – and their role in solving different counting problems.

7.3.1 Sum Rule

Let us consider two events E_1 and E_2 that cannot occur simultaneously. Suppose the event E_1 can occur in n_1 ways and event E_2 can occur in $n_2$ways. Then the event E_1 or E_2can occur in $n_1 + n_2$ ways.

In a generalized way, let us consider n events E_1, E_2, ... , E_n such that no two events can occur simultaneously. If the events E_1, E_2, ... , E_n can occur in n_1, n_2, ... , n_n ways, respectively, then one of events can occur in $n_1 + n_2 + \cdots \ldots \ldots + n_n$ ways.

Example 7.6 In how many ways can we select a student's representative from 4 boys and 3 girls?

Solution: A boy can be selected in 4 ways and a girl can be selected in 3 ways. Since the representative may be a boy or a girl, the total number of ways to select a representative is $4 + 3 = 7$.

7.3.2 Product Rule

Let us consider two events E_1 and E_2. Suppose the event E_1 can occur in n_1 ways and for each of these n_1 ways the event E_2 can occur in $n_2$ways. Then the event E_1 and E_2 can occur in n_1n_2 ways.

In a generalized way, let us consider n events E_1, E_2, ... , E_n. If the events E_1, E_2, ... , E_n can occur in n_1, n_2, ... , n_n ways, respectively, then all of events can occur in n_1n_2 ... ... n_n ways.

Example 7.7 A building has 7 floors and each floor has 10 rooms. How many ways are there to get a room for rent?

Solution: A floor can be chosen in 7 ways. As every floor contains 10 rooms, the total number of ways to pick a room is $7 \times 10 = 70$.

7.3.3 Counting ways of forming numbers from a set of digits

For given set of digits, each time we have to find the numbers of certain digits, proper care must be taken before counting the numbers. There are two cases in forming a number of certain digits – repetition is allowed and repetition is not allowed.

Example 7.8 How many different three-digit numbers can be formed by using the digits $1, 2, 3, 4, 5\ and\ 6$ when (a) repetition is not allowed and (b) repetition is allowed?

Solution:

(a) Let a three-digit number be represented by three places - - - . Since repetition is not allowed, the first place can be occupied in 6 different ways, the second place in 5 different ways, and the third place in 4 distinct ways. Therefore, using the product rule, the total number of means to fill the three places is $6\ .\ 5\ .\ 4 = 120$.

(b) When repetition is allowed, the following method can be used to count the numbers. There will be 6 ways of filling each of the first, second, and third places. Thus, using the product rule, the total number of ways to fill the three places is $6\ .\ 6\ .\ 6 = 216$.

7.3.4 Inclusion – Exclusion Principle

Suppose two tasks A and B can occur in $n_1 and\ n_2$ ways, where some of the $n_1 and\ n_2$ ways may be the same. In this situation, we cannot apply the sum rule, because the same number of ways will be counted twice. In such situations, we apply the inclusion – exclusion principle. According to this principle, if A and B are two set, then the number of elements in the set $A \cup B$ is given by

$$n(A \cup B) = n(A) + n(B) - n(A \cap B)$$

This principle holds for any number of sets. For three sets, it can be stated as follows:

$$n(A \cup B \cup C) = n(A) + n(B) + n(C) - n(A \cap B) - n(B \cap C) - n(A \cap C) + n(A \cap B \cap C)$$

Example 7.9 There are 70 students in a class. The class teacher decided to organize two competitions – singing and dancing. Every student has to participate in at least one competition. The students who participated in the dance competition also get a chance to perform during the annual function. The students who participate in both the activities get 10 additional points in general proficiency, 50 students participate in the singing competition and only 30 students do not get additional points in their general proficiency. Find the number of students who get the chance to perform during the annual function, but do not get additional points.

Solution: Let S denote the set of students who participate in the singing competition and D denote the set of students who participate in the dancing competition.

Given that $|S \cup D| = 70\ and\ \ |S| = 50$

$$|D| = Number\ of\ students\ who\ get\ a\ chance\ to\ perform\ during\ the\ annual\ function$$

$$|S \cap D| = Number\ of\ students\ who\ get\ 10\ additional\ points\ in\ general\ proficiency$$

Given that $|(S \cap D)'| = 30,\ |S \cap D| = 70 - 30 = 40$.

We know that $|S \cup D| = |S| + |D| - |S \cap D|$. Hence,

$$|D| = 70 - 50 + 40 = 60$$

Thus, the number of students who get the chance to perform during the annual function but do not get additional points is

$$|D| - |S \cap D| = 60 - 40 = 20$$

7.3.5 Permutations and combinations

Consider the problem of counting the arrangements of certain elements from a given set of elements. In this problem, the order of elements is important because each order will give a new arrangement. If we are asked to count the number of ways to select certain elements from a given set of elements, then the order of the elements is not important. Two different orders of elements will give the same selection. These counting problems are of special interest. Here we shall discuss the methods to find the answers for these counting problems.

Permutation

Any arrangement of n objects in a given order is called a permutation of the objects (taken all at a time). The arrangement of any $r \leq n$ of these objects is called an $r - permutation$. The number of r – permutation of a set with n distinct elements is denoted by $P(n, r)$ or ${}^{n}P_{r}$.

Example 7.10 Consider a set of letters$\{a, b, c\}$.

(a) $abc, acb, bac, bca, cab, cba$ are permutation of three objects taken all at a time.

(b) ab, ba, ac, ca, bc, cb are permutations of any two of the three objects.

Theorem 7.1 The number of r-permutations of a set with n distinct elements is $P(n, r) = n(n-1)(n-2) \ldots \ldots (n-r+1)$.

Proof: The first element can be selected in n different ways. Now, $n - 1$ elements are left in the set, and thus, there are $n - 1$ ways to choose the second element. Similarly, the third element can be selected in $n - 2$ ways. Continuing like this, the r^{th} element can be selected in $n - r + 1$ ways. Thus, using the product rule, the total number of ways for r-permutations is given by $P(n, r) = n(n-1)(n-2) \ldots \ldots \ldots (n-r+1)$.

Theorem 7.2 Prove that $P(n, r) = \frac{n!}{(n-r)!}$.

Proof: $P(n,r) = n(n-1)(n-2)\ldots\ldots\ldots(n-r+1)$

$$= \frac{n(n-1)(n-2)\ldots\ldots\ldots(n-r+1)(n-r)!}{(n-r)!} = \frac{n!}{(n-r)!}$$

Example 7.11 From a set of 5 books, in how many ways can 4 books be arranged in a bookshelf?

Solution: Here $n = 5$ and we have to find 4-premutations from a set of 5 elements. Thus, the total numbers of ways to arrange 4 books from a set of 5 books is $P(5,4) = \frac{5!}{1!} = 120$

Example 7.12 In how many ways can 5 students arrange themselves in a row?

Solution: The permutation of n objects taken all at a time is given by $n!$. Thus, the total number of ways is $5! = 5\,.\,4\,.\,3\,.\,2\,.\,1 = 120$

Combination

Let us consider a set of n objects. An r-combination from the set of n objects is any selection of r objects, where the order of the object does not matter. Consider a set of letter$\{a,\ b,\ c\}$. Then $abc,\ acb,\ bac,\ bca,\ cab,\ cba$ represent different permutations, but they all represent the same combination.

Number of r-combinations from set of n elements

The number of r-combinations from a set of n objects is denoted by $C(n,r)\ or\ C^n_r\ or\ \binom{n}{r}$.

Theorem7.3 The number of r-combinations from a set of n objects equals

$$C(n,r) = \frac{n!}{r!\,(n-r)!}$$

Proof: $C(n,r)$ represents the number of r-combinations from the set of n objects. Then each of the r-combinations contains r objects, which can rearrange themselves in $r!$ Ways. Thus $C(n,r).r! = P(n,r)$

$$C(n,r).r! = \frac{n!}{(n-1)!}$$

$$C(n,r) = \frac{n!}{r!\,(n-1)!}$$

The following are some important results:

1. $C(n,0) = \frac{n!}{0!(n-0)!} = 1$
2. $C(n,n) = \frac{n!}{n!(n-n)!} = 1$
3. $C(n,r) = \frac{n!}{r!(n-r)!} = \frac{n!}{[n-(n-r)]!(n-r)!} = \frac{n!}{(n-r)![n-(n-r)]!} = C(n,n-r)$

Example 7.13 Find the number of diagonals of a polygon having n sides

Solution: To count the number of diagonals, first we shall have to count the number of ways to join two points in a polynomial of n sides, and this equals $C(n,2) = \frac{n(n-1)}{2}$. Since there will be n sides in a polygon, the total number of diagonals $\frac{n(n-1)}{2} - n = \frac{n(n-3)}{2}$.

Problems for Exercise:

1. Let P be the proposition that the sum of the first n odd numbers in n^2; that is,
$$P(n): 1 + 3 + 5 + \cdots + (2n - 1) = n^2$$
(The nth odd number is $2n - 1$, and the next odd number is $2n + 1$.). Prove P is true for every positive integer $n \in N$.
2. Prove the following proposition:
$$P(n): 1 + 4 + 7 + \cdots + (3n - 2) = \frac{n(3n - 1)}{2}$$
3. Prove the following proposition:
$$P(n): \frac{1}{1(3)} + \frac{1}{3(5)} + \frac{1}{5(7)} + \cdots + \frac{1}{(2n - 1)(2n + 1)} = \frac{n}{2n + 1}$$
4. Assume a is a nonzero real number and n is a nonnegative integer. Give a recursive definition of a^n.
5. Give a recursive definition of $\sum_{k=0}^{n} a_k$.
6. Let Fibonacci numbers, $f_0, f_1, f_2, \dots,$ are defined by the equations $f_0 = 0; f_1 = 1; f_n = f_{n-1} + f_{(n-2)}$. Find the Fibonacci number f_4.
7. There are four bus lines between A and B; and three bus lines between B and C. Find the number of ways a person can travel:
   a. By bus from A to C by way of B;
   b. Roundtrip by bus from A to C by way of B.
8. Give a recursive definition of the factorial function.
9. Find $2!, 3!, and\ 4!$
10. Compute:
$$a)\ \binom{16}{3},\quad and\quad b)\ \binom{12}{4}$$

Block III: Unit II Recurrence Relation

Learning Objectives
After completing this unit, the learner shall be able to:

- Define recurrence relation;
- Model counting and other such problems using a recurrence relation;
- Find the solution of recurrence relations;
- Find the order and degree of the given recurrence relation;
- Solve linear homogeneous recurrence relation with constant coefficients;
- Solve linear non-homogeneous recurrence relation with constant coefficient;

Suppose we ask to any known fellow the age of his oldest daughter. He could tell directly that she is 19 years old. Or he could tell that she is 6 years older than his second daughter. If we ask for the age of the second daughter, instead of telling us that she is 13 years old, he might tell us that she if 5 years older than his third daughter. In turn, he could tell us that his third daughter is 2 years older than his only son. When he tells us that his only son is 6 years old, we would have no difficulty in figuring out that his third daughter is 8 years old, his second daughter is 13 years old, and his oldest daughter is 19 years old.

Several observations can be drawn from above example:

(a) Using our prior knowledge can be a concise way to give information.
(b) We do need to do some work to make use of the knowledge we already have.
(c) We might try to refer to some prior knowledge in successive steps

Such a chain of reference can only be terminated when we reach a point where we know explicitly what to do without referring to other prior knowledge.

The amount of bacteria in a colony gets doubled in every single hour. If initially a colony have five bacteria, how many will be in next n hours? To resolve this problem, assume a_n be the amount of bacteria at the end of n hours. Since the amount of bacteria doubles each hour, the relationship $a_n = 2a_{n-1}$ holds whenever n is a positive integer. This type of counting problems is known as recurrence relation. We will discuss a variety of counting problems that can be modeled using solve the recurrence relations. Recursive definitions are used to solve the recurrence relations.

8.1 Definition

A recurrence relation is an equation that uses a recursive definition. We already know that recursive definitions can be used to define functions, sets and sequences. In the recursive definition, the recursive step is basically a relationship or a formula through which we calculate the next term with the help of the exiting terms. This is a recurrence relation. More explicitly, a recurrence relation is an equation that relates a_r with one or more preceding terms in the sequence, namely $a_{r-1}, a_{r-2}, \ldots\ldots\ldots, a_0$ for all integers r with $r \geq r_0$, where r_0 is a non-negative integer used to define the initial condition.

Note: $Recurrence\ relation\ + initial\ conditions = Ways\ of\ representation\ of\ a\ sequence$

$$Recurrence\ relation = Difference\ equation$$

Recursive definition of the sequence is the recurrence relation with initial condition.

8.2 Modeling with Recurrence relation

Many counting and other such problems can be modeled through recurrence relations, and therefore, these relations play an important role in solving such problems. First, we will learn to model a recurrence relation with the help of a few examples and then finding the solution of recurrence relations.

Example 8.1(Compound Interest)

Suppose that a person deposits Rs.10,000 in a saving account at a bank yielding 11% per year with compounded annually. How much be the amount after 30 years.

Solution: To solve the problem, consider P_n that denote the amount later n years. Since the amount in the account next n years equals the amount in the account next $n - 1$ years plus interest for the n^{th} year, we see that the sequence $\{P_n\}$ satisfies the recurrence relation.

$$P_n = P_{n-1} + 0.11P_{n-1} = (1.11)P_{n-1}$$

The initial condition is $P_0 = 10{,}000$

Now, we can use an interactive approach to find a formula for P_n

$$P_1 = (1.11)P_0$$

$$P_2 = (1.11), P_1 = (1.11)^2P_0$$

When the initial condition $P_0 = 10{,}000$ is put then $P_n = (1.11)^n. 10{,}000$

$$P_{30} = (1.11)^{30}. 10{,}000$$

$$= Rs. 2{,}28{,}922$$

Example 8.2 (Chess Tournament)

In a chess tournament, there are r players and each player plays with every other player. Let a_r be the total number of games in the tournament. Find the recurrence relation for a_r.

Solution: Let a_{r-1} denote the number of games in a class tournament of $r - 1$ players. Now, the r^{th} player will play with each of the $r - 1$ players. Thus, the total number of games in a tournament of r players shall be given by

$$a_r = a_{r-1} + (r - 1)$$

Since there will be no game if there is only 1 player, $a_1 = 0$

Therefore, the required recurrence relation is,

$$a_r = a_{r-1} + (r - 1)\ for\ r \geq 2$$

With the initial condition $a_1 = 0$

8.3 Order and Degree of Recurrence Relations

Let a_r be a numeric function. A recurrence relation is an expression of the form

$$a_r = F(a_{r-1}, a_{r-2}, \dots \dots \dots, a_{r-k}, \mathrm{r})$$

where,

F is a function of some of the variables $a_{r-1}, a_{r-2}, \dots \dots, a_{r-k}, r$

(for our purpose, we shall consider F as a polynomial that depends on finitely many variables $a_{r-1},\ a_{r-2},\ \dots \dots,\ a_{r-k}\ and\ r$) . This relationship is used to find the r^{th} term with the help of one or more previous terms.

The order of the recurrence relation $a_r = F(a_{r-1}, a_{r-2}, \dots \dots \dots, a_{r-k}, \mathrm{r})$ is k , where a_r depends on some of the previous k terms and k is the smallest such integer. If the recurrence relation is $a_r = F(a_{r-1}, a_{r-2}, \dots \dots \dots, a_0, \mathrm{r})$, where a_r depends on all of its previous terms, the order is not defined. The degree of the recurrence relation is the degree of F considering Fas polynomial in its variables excluding r. A recurrence relation is called linear if its degree is 1.

In other words, the order of a recurrence relation can be calculated as the difference between the largest and the smallest subscripts (terms) of a appearing in the recurrence relation. Similarly, just like calculating the degree of a polynomial of finitely many variables. For example, the degree of the polynomials $f(x) = x^2 + 2x + 3$ and $f(x, y) = x + x^2y + 4$ are 2 and 3, respectively. We can calculate the degree of a recurrence relation by assuming the terms $a_r, a_{r-1}, a_{r-2}, \dots \dots$ as variables.

In general,

$$Order\ of\ a\ recurrence\ relation = \frac{Largest\ argument - Smallest\ argument}{Unit\ of\ increment}$$

Example 8.3 Find the order and degree of the following recurrence relations:

(a) $a_r = 2a_{r-1} - a_{r-2}$

(b) $a_r = a_{r-1} + r$

(c) $a_r = ra_{r-1} + a_{(r-2)} + r^2$

(d) $a_r = ra_{r-1} + a^2_{r-2}$

(e) $a_r = \sqrt{a_{r-1}} + a_{r-2}$

(f) $a_r = a_{r-1}a_{r-2} + r$

Solution:

(a) Order = 2, degree = 1

(b) Order = 1, degree = 1

(c) Order = 2, degree = 1

(d) Order = 2, degree = 2

(e) Order = 2, degree not defined

(f) Order = 2, degree = 2

A recurrence relation is said to be homogenous if it does not contain a term that depends only on r. A recurrence relation that is not homogeneous is called non-homogeneous. For example, the recurrence relation $a_r = a_{r-1} + a_{r-2}$ is homogenous, whereas the recurrence relation $a_r = a_{r-1} + r$ is non-homogeneous.

8.4 Linear Homogenous Recurrence Relations

One important class of recurrence relations can be explicitly solved in a systematic way.

A linear recurrence relation with constant coefficients is a recurrence relation of the form

$$a_r = c_1 a_{r-1} + c_2 a_{r-2} + \cdots \ldots \ldots \ldots \ldots + c_k a_{r-k} + f(r)$$

Where $c_1, c_2, \ldots \ldots \ldots, c_k$ are real numbers and $c_k \neq 0$.

A linear recurrence relation with constant coefficients is called homogeneous if $f(r) = 0$; otherwise, it is called non-homogenous. The solution of a recurrence relation is obtained in two parts – homogeneous solution and particular solution.

8.4.1 Solving Linear Homogeneous Recurrence Relation with Constant Coefficients Method 1

Suppose the recurrence relation is given

$$a_n = c_1 a_{n-1} + c_2 a_{n-2}$$

c_1 and c_2 are real number constants.

Solution or complete solution of a recurrence relation is an expression for a_n which satisfies the given recurrence relation.

The general solution of a recurrence relation is that in which the number of arbitrary constants is equal to the order of the recurrence relation.

The particular Integral (P.I.) or particular solution is that solution which is obtain from the general solution by giving the particular value to the constant.

Thus, the complete solution of the recurrence relation is

$$a_n = complementary\ function\ (C.F.) + particular\ integral\ (P.I.)$$

for linear homogeneous recurrence relation we will have a condition:

$$P.I. = 0$$

Thus, in the case the complete solution will be defined as:

$$a_n = C.F.$$

Suppose, $a_n = r^n$ is the genera; solution

Then we will make the Auxiliary equation (A.E.) first

$$r^n = c_1 r^{n-1} + c_2 r^{n-2}$$

Implies $r^{n-2}[r^2 - c_1 r - c_2] = 0$

$$r^2 - c_1r - c_2 = 0 \;\; (A.E.)$$

Above equation is the auxiliary equation and n has two distinct roots $r1$ and $r2$.

Hence, the sequence $\{a_n\}$ is a solution to the recurrence relation if and only if .

$$a_n = l_1r_1^n + l_2r_2^n \qquad for\ n = 0, 1, 2, \dots \dots \quad where\ l_1\ \ and\ \ l_2\ are\ constants.$$

Let us take an example to explain the above method

Example 8.4 Solve the recurrence relation.

$$a_{n+3} - 2a_{n+2} - 5a_{n+1} + 6a_n = 0$$

Solution: Put $a_n = r^n$ is the general solution

Then $\qquad (r^3 - 2r^2 - 5r + 6)r^n = 0$

$A.E.is \qquad r^3 - 2r^2 - 5r + 6 = 0$

$$(r - 1)(r + 2)(r - 3) = 0$$

$$r = 1, -2, 3$$

Thus, complete solution is $a_n = C.F.$

$$a_n = l_1(1)^n + l_2(-2)^n + l_3(3)^n$$

Method 2

Suppose, the recurrence relation

$$a_n = c_1a_{n-1} + c_2a_{n-2}$$

c_1 and c_2 be real numbers $c_2 \neq 0$

$$A.E. = r^2 - c_1r - c_2 = 0$$

Let $r_1 and\ r_2$be the roots now

$$r_1 = r_2 = r_0, \qquad then$$

The solution of the recurrence relation is $a_n = (l_1 + l_2n)r_1^n \qquad for\ n = 0, 1, 2, \dots \dots$ where $l_1 and\ l_2$ are constants.

Example 8.5 Solve the recurrence relation.

$$a_{n+2} - 2a_{n+1} + a_n = 0$$

Solution: Put $a_n = r^n$ is the general solution

$$(r^2 - 2r + 1)r^n = 0$$

$A.E.$ is $$r^2 - 2r + 1 = 0$$

$$(r-1)^2 = 0, \qquad r = 1, 1$$

Complete solution is $a_n = (l_1 + l_2 n)(1)^n$

$$= l_1 + l_2 n$$

Method 3

Suppose the recurrence relation is given

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots + c_k a_{n-k}$$

Then the auxiliary equation $A.E.$ is

$$(r^k - c_1 r^{k-1} - c_2 r^{k-2} - \cdots - c_k) = 0$$

Let the roots are $r_1, r_1, r_1, r_1 \ldots \ldots$

Then the general solution n of the recurrence relation is

$$a_n = (l_1 + l_2 n + l_3 n^2 \ldots + l_k n^{k-1}) r_1^n$$

Example 8.6 Find the solution of the recurrence relation $a_n = -3a_{n-1} - 3a_{n-2} - a_{n-3}$ with initial condition

$$a_0 = 1, \qquad a_1 = -2 \;\; and \;\; a_2 = -1$$

Solution

The $A.E.$ is given as

$$(r^3 + 3r^2 + 3r + 1) = 0$$

$$(r+1)^3 = 0$$

$$r = -1$$

So, the complete solution is

$$a_n = (l_1 + l_2 n + l_3 n^2)(-1)^n$$

For $n = 0$ $\qquad 1 = l_1 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (i)$

For $n = 1$, $\qquad -2 = -(l_1 + l_2 + l_3) \ldots\ldots\ldots\ldots\ldots\ldots (ii)$

For $n = 2$, $\qquad -1 = l_1 + 2l_2 + 4l_3 \ldots\ldots\ldots\ldots\ldots\ldots (iii)$

By Eq. (i) and (ii)

$$(1 + l_2 + l_3) = 2$$

$$l_2 + l_3 = 1 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (A)$$

By Eq. (i) and (iii)

$$1 + 2(l_2 + 2l_3) = -1$$

$$l_2 + 2l_3 = -1$$

$$l_2 + l_3 = 1$$

After solving the above equations we get

$$l_3 = -2$$

$$l_2 = 3$$

Hence the final complete solution is

$$a_n = (1 + 2n^2 - 3n)(-1)^n$$

$$a_n = (1 - 3n + 2n^2)(-1)^n$$

Method 4

If the roots of auxiliary equation are imaginary

Suppose the recurrence relation is given

$$a_n = c_1a_{n-1} + c_2a_{n-2}$$

c_1 and c_2 are real number constants.

According to method 1, the auxiliary equation $(A.E.)$comes out as:

$$r^2 - c_1r - c_2 = 0$$

Suppose the roots are $r_1 = a + ib$ and $r_2 = a - ib$, then the complete solution comes out as

$$a_n = l_1(a + ib)^n + l_2(a - ib)^n$$

Now put $a = r\ \cos\alpha$ and $b = r\ \sin\alpha$ then

$$a_n = r^n[l_1(\cos\alpha + i\sin\alpha) + l_2(\cos\alpha - i\sin\alpha)]$$

$$a_n = r^n[(l_1 + l_2)\cos\alpha + i\ (l_1 - l_2)\sin\alpha]$$

Say $\qquad L_1 = l_1 + l_2$ and $L_2 = (l_1 - l_2)i$

Thus, $a_n = r^n(L_1\cos\alpha + L_2\sin\alpha)$ is the complete solution where l_1 and l_2 are arbitrary constants and

$$r = \sqrt{(a^2 + b^2)} \qquad and \qquad \alpha = \tan^{-1}\left(\frac{b}{a}\right)$$

8.4.2 Solving Linear Non-homogeneous Recurrence Relation with Constant Coefficient

So far we have discussed to solve linear homogeneous recurrence relations with constant coefficients. There was no use of particular Integral (P.I,) in that case because in that case always P.I. was equal to zero.

Now we will solve linear but not homogeneous recurrence relations like :

$$a_n = 3a_{n-1} + 2n$$

is the example of a linear non homogeneous recurrence relation with constant coefficients i.e., the recurrence relation of the type

$$a_n = c_1 a_{n-1} + c_2 a_{n-2} \dots\dots\dots\dots, c_k a_{n-k} + F(n)$$

In such cases the A.E. will behave like the previous case in homogeneous equation. But the particular integral (P.I.) will be the main constraint to decide the complete solution of the equation.

Thus,

$$a_n = C.F. + P.I.$$

Now we will discuss how to find the P.I.

(i) When $F(n) = p^n$, where p is a constant.

P.I. $= \frac{p^n}{A.E.}$ and put $r = p$ provided $A.E. \neq 0$

If $A.E. = 0 \ at \ r = p$ then $F(n)$ can be written in one of the following forms given in next cases

(ii) When $(r-p)a_n = p^n$

then, $P.I. = \frac{p^n}{r-p} = np^{n-1}$

(iii) When $(r-p)^2 a_n = p^n$

then, $P.I. = \frac{p^n}{(r-p)^2} = [\frac{n(n-1)}{2}] * p^{n-2}$

(iv) When $(r-p)^3 a_n = p^n$

Then, $P.I. = \frac{p^n}{(r-p)^3} = [\frac{n(n-1)(n-2)}{3}] * p^{n-3}$

and so on.

Example 8.7 Solve $a_{n+2} - 4a_{(n+1)} + 3a_n = 5^n$

Solution Here $F(n) = 5^n$

$A.E.$ is

$$(r^2 - 4r + 3) = 0$$

$$r = 1, 3$$

$$C.F. = l_1(1)^n + l_2(3)^n$$

$$P.I. = \frac{1}{r^2 - 4r + 3} * 5^n$$

$$= \frac{5^n}{5^2 - 4.5 + 3} = \frac{5^n}{8}$$

Thus, the complete solution is

$$a_n = C.F. + P.I.$$

$$a_n = l_1 + l_2 3^n + \frac{5^n}{8}$$

Problems for Exercise

1. Find a general formula for the Fibonacci sequence

$$\left\{\begin{array}{c} f_n = f_{n-1} + f_{n-2} \\ f_0 = 0 \\ f_1 = 1 \end{array}\right.$$

2. Find the solution for the recurrence relation

$$\left\{\begin{array}{c} x_n = 6x_{n-1} - 9x_{n-2} \\ x_0 = 2 \\ x_1 = 3 \end{array}\right.$$

3. Find the solution for the recurrence relation

$$\left\{\begin{array}{c} x_n = 2x_{n-1} - 5x_{n-2}, \quad n > 2 \\ x_0 = 1 \\ x_1 = 5 \end{array}\right.$$

4. Two persons Rajan and Vijay gamble dollars on the toss of a fair coin. Rajan has Rs. 700 and Vijay has Rs. 300. In each play either Rajan wins Rs.10 from Vijay or loss Rs.10 to Vijay. The game is played without stop until one wins all the money of the other or goes forever. Find the probabilities of the following three possibilities:
   a. Rajan wins all the money of Vijay.
   b. Rajan loss all his money to Vijay.
   c. The game continues forever.
5. Find an explicit formula for the sequence given by the recurrence relation

$$\left\{\begin{array}{c} x_n = 15x_{n-2} - 10x\_(n - 3 - 60x_{n-4} + 72x_{n-5} \\ x_0 = 1, \ \ x_1 = 6, \ \ x_2 = 9, \ \ x_3 = -110, \ \ x_4 = -45 \end{array}\right.$$

6. Consider the non-homogeneous equation

$$\left\{\begin{array}{c} x_n = 3x_{n-1} + 10x_{n-2} + 7(5)^n \\ x_0 = 4 \\ x_1 = 3 \end{array}\right.$$

7. Consider the non-homogeneous equation

$$\left\{\begin{array}{c} x_n = 10x_{n-1} - 25x_{n-2} + 8(5)^n \\ x_0 = 6 \\ x_1 = 10 \end{array}\right.$$

8. Find an explicit formula for each of the sequences defined by the recurrence relation with initial conditions.
   a. $x_n = 5x_{n-1} + 3, x_1 = 3$
   b. $x_n = 3x_{n-1} + 5n, x_1 = 5$
   c. $x_n = 2x_{n-1} + 15x_{n-2}, x_1 = 2, x_2 = 4$
9. Find an explicit formula for each of the sequences defined by the non-homogeneous recurrence relations with initial conditions.
   a. $x_n = 2x_{n-1} + 15x_{n-2} + 2^n, x_1 = 2, x_2 = 4$
   b. $x_n = 4x_{n-1} + 5x_{n-2} + 3, x_1 = 3, x_2 = 5$
   c. $x_n = 3x_{n-1} - 2x_{n-2} + 2^n, x_0 = 2, x_1 = 4$
10. Show that if s_n and t_n are solutions for the non-homogeneous linear recurrence relation

$$x_n = ax_{n-1} + bx_{n-2} + f(n), n > 2,$$

Then $x_n = s_n - t_n$ is a solution for the homogeneous linear recurrence relation

$$x_n = ax_{n-1} + bx_{n-2}, \qquad n > 2$$

Block III: Unit III Generating Function

Learning Objectives
After completing this unit, the learner shall be able to:

- Define Generating functions;
- Find the generating function for a given sequence;
- Explain the properties of generating function;
- Determine the numeric function corresponding to the given generating functions;
- Find the solution of linear recurrence relations using generating functions;
- Find the solution of combinatorial problem using generating function;

Generating functions are used to represent sequence efficiently by coding the terms of a sequence as coefficient of powers of a variable in a formal series. Generating functions can be used to solve many types of counting problems, such as the number of ways to select or distribute objects of different kinds, subject to a variety of constraints, and the number of ways to make change for a dollar using coins of different denominations. Generating functions can be used to solve recurrence relations by translating a recurrence relation for the terms of a sequence into an equation involving a generating function. From this we can solve the original recurrence relation. Generating functions can also be used to prove combinatorial identities by taking advantage of relatively simple relationships between functions that can be translated into identities involving the terms of sequences.

The notion of alternative representation is of great use in computer science. Binary numbers are an alternative representation of decimal numbers. Instead of adding, subtracting, multiplying and dividing decimal numbers directly, we represent them as binary numbers, use a computer to carry out all arithmetic operations on the binary numbers (which a computer can do easily), and then obtain the results of our computation by converting the results in binary numbers into decimal numbers. Similarly, an alternative representation of a real number using logarithm is very useful in many problems. Thus, a suitably chosen alternative representation leads to efficiency and case in some operations.

9.1 Closed form expression

Generating functions are frequently articulated in closed form (instead of a series), by some expression including operations defined for proper power series. These expressions in terms of the indeterminate x may involve arithmetic operations, differentiation with respect to x and substitution into other generating functions; since these operations are also defined for functions, the result expressed like a function of x. Certainly, the closed form expression can usually be interpreted as a function that can be estimated at (adequately small) concrete values of x. However such interpretation is not required to be promising, because proper power series are not required to give a convergent series when a nonzero numeric value is replaced for x.

Generating functions are not functions in the proper sense of a domain to a codomain mapping; the name is just traditional, and they are occasionally more suitably called generating series.

Generating functions are important tools in discrete mathematics and their use is by no means confined to solve linear recurrence relations (as a closed form formula). The functions can be used to solve many types of counting problems.

9.1.1 Definition of generating function

The generating function for the sequence $a_0, a_1, \dots \dots \dots \dots a_k, \dots \dots \dots$ of real numbers is infinite series.

$$G(x) = a_0 + a_1x + a_2x^2 + \cdots \dots \dots \dots + a_kx^k + \cdots \dots \dots = \sum_{k=0}^{\infty} a_kx^k$$

Example 9.1

The generating functions for the sequences $\{a_k\}$ where $a_k = 2, a_k = 3^k \quad and \quad a_k = (k+1)$ are

$$\sum_{k=0}^{\infty} 2x^k, \qquad \sum_{k=0}^{\infty} 3^kx^k, \ and \ \sum_{k=0}^{\infty}(k+1)x^k$$

respectively.

It is often possible to find a formula (a closed form expression) for $G(x)$ which can be manipulated algebraically to provide useful combinatorial information.

9.1.2 Some special Generating Functions

1. The function given by

$$G(x) = \frac{1}{1-x}$$

the generating function of the sequence $1, 1, 1, 1,$...........since

$$G(x) = (1-x)^{-1} = 1 + x + x^2 + \cdots \dots \dots \dots \qquad |x| < 1$$

Here $\quad a_0 = 1, \ a_1 = 1, \ a_2 = 1, \dots \dots \dots$

2. The function given by

$$G(x) = \frac{1}{(1-x)^2} = \sum_{k=0}^{\infty}(k + 1)x^k$$

is the generating function of the sequence $1, 2, 3, 4, \dots \dots \dots$

$$G(x) = (1-x)^{-2} = 1 + 2x + 3x^2 + \cdots \dots \dots \dots (k+1)x^k + \cdots \dots \dots$$

3. The function given by

$$G(x) = \frac{x}{(1-x)^2} = \sum_{k=0}^{\infty}(k\ +\ 1)x^{k+1}$$

is the generating function of the sequence $0, 1, 2, 3,$ .... Since

$$G(x) = x(1-x)^{-2} = x(1 + 2x + 3x^2 + \cdots \text{..............})$$

$$= 0 + 1x + 2x^2 + 3x^3 + \cdots \ldots\ldots\ldots + kx^k + \cdots \ldots\ldots$$

4. The function given by

$$G(x) = \frac{1}{1-ax}, \qquad |ax| < 1$$

is the generating function of the sequence $1, a, a^2, a^3, \ldots\ldots$

We can derive a closed form expression for $G(x)$ by involving the formula for the sum of a geometric series. We can also apply indirect method as follows.

Example 9.2 Find the generating function for the sequence $1, a, a^2, \ldots\ldots\ldots$ where a is a fixed constant.

Solution Let $G(x) = 1 + ax + a^2x^2 + a^2x^3 + \cdots \ldots\ldots$

So, $G(x) - 1 = ax + a^2x^2 + a^2x^3 + \cdots \ldots\ldots$

or, $\frac{G(x)-1}{ax} = 1 + ax + a^2x^2 + \cdots \ldots\ldots$

or , $\frac{G(x)-1}{ax} = G(x) \Rightarrow G(x) = \frac{1}{1-ax}$

the required generating function is $\frac{1}{1-ax}$.

The results are summarized in the following table.

Table 9.1 Sequences and Generating functions

S. No	General term of sequence a_k	Generating Function $G(x)$
1	1	$\frac{1}{1-x}$
2	$k+1$	$\frac{1}{(1-x)^2}$
3	k	$\frac{x}{(1-x)^2}$
4	$k(k+1)$	$\frac{2x}{(1-x)^3}$
5	$(k+1)(k+2)$	$\frac{2}{(1-x)^3}$
6	a^k	$\frac{1}{1-ax}$

9.2 Properties of Generating Function

Now we are going to discuss some of the properties of generating function. These properties will be the helpful in finding the generating function of a complex numeric function.

Theorem 9.1 let $\{a_r\}, \{b_r\}, and\ \{c_r\}$ be three sequence, and $A(x)$, $B(x)$ and $C(x)$ be the corresponding generating function. Show the following:

(a) The generating function of the sequence $\{\alpha a_r\}$ is $\alpha A(x)$.

(b) The generating functions of the sequence $\{a_r + b_r\}$ is $A(x) + B(x)$.

(c) The generating function of the sequence $\{\alpha^r a_r\}$ is $A(\alpha x)$.

(d) If $c_r = a_r * b_r$, then $C(x) = A(x).B(x)$

Proof:

(a) If $C_r = \alpha a_r$, then $C(x) = \alpha a_0 + \alpha a_1 x + \alpha a_2 x^2 + \cdots \ldots \ldots + \alpha a_r x^r + \cdots \ldots.$

$$C(x) = \alpha(a_0 + a_1 x + a_2 x^2 + \cdots \ldots \ldots + a_r x^r + \cdots \ldots \ldots) C(x) = \alpha A(x)$$

(b) If $c_r = a_r + b_r$ then

$$C(x) = a_0 + b_0 + (a_1 + b_1)x + (a_2 + b_2)x^2 + \cdots \ldots \ldots + (a_r + b_r)x^r + \cdots \ldots \ldots C(x)$$
$$= (a_0 + a_1 x + a_2 x^2 + \cdots \ldots \ldots + a_r x^r + \cdots \ldots) + (b_0 + b_1 x + b_2 x^2 + \cdots \ldots \ldots$$
$$+ b_r x^r + \cdots \ldots \ldots) C(x) = A(x) + B(x)$$

(c) If $c_r = \alpha^r a_r$, then $C(x) = \alpha^0 a_0 + \alpha^1 a_1 x + \alpha^2 a_2 x^2 + \cdots \ldots \ldots + \alpha^n a_r x^r + \cdots \ldots \ldots$

$$C(x) = \alpha^0 a_0 + a_1(\alpha x) + a_2(\alpha x)^2 + \cdots \ldots \ldots + a_r(\alpha x)^r + \cdots \ldots \ldots C(x) = A(\alpha x)$$

(d) $c_r = a_r * b_r = \sum_{i=0}^{r} a_i b_{r-i} \qquad for\ r \geq 0$

thus $\quad c_0 = a_0 b_0,\ c_1 = a_0 b_1 + a_1 b_0,\ c_2 = a_0 b_2 + a_1 b_1 + a_2 b_0, \ldots$

We have $\quad A(x) = a_0 + a_1 x + a_2 x^2 + \cdots \ldots$ and $B(x) = b_0 + b_1 x + b_2 x^2 + \cdots \ldots \ldots$

$$A(x).B(x) = a_0 b_0 + (a_0 b_1 + a_1 b_0)x + (a_0 b_2 + a_1 b_1 + a_2 b_0)x^2 + \cdots \ldots$$
$$= c_0 + c_1 x + c_2 x^2 + \cdots \ldots = C(x)$$

Example 9.3

Determine the numeric function corresponding to each of the following generating functions:

(a) $G(x) = \frac{2}{1-4x^2}$

(b) $G(x) = \frac{1}{x^2+3x+2}$

(c) $G(x) = \frac{1}{x^2-2x-3}$

(d) $G(x) = \frac{1}{1-x^4}$

Solution:

(a) $G(x) = \frac{2}{1-4x^2}$

$= \frac{2}{(1-2x)(1+2x)}$

$= \frac{1}{1-2x} + \frac{1}{1+2x}$

$= \{1 + 2x + (2x)^2 + (2x)^3 + \cdots\} + \{1 - 2x + (2x)^2 - (2x)^3 + \cdots \ldots\}$

$= 2\{1 + (2x)^2 + (2x)^4 + \cdots\}$

$= 2 + 2^3x^2 + 2^5x^4 + \cdots \ldots$

thus, the corresponding numeric function is $a_r = \begin{cases} 2 & r = 0 \\ 0 & if\ r\ is\ odd \\ 2^{r+1} & if\ r\ is\ even \end{cases}$

(b) $G(x) = \frac{1}{x^2+3x+2}$

$= \frac{1}{(x+1)(x+2)}$

$= \frac{1}{x+1} - \frac{1}{x+2} = \frac{1}{x+1} - \frac{1}{2} . \frac{1}{1+\frac{x}{2}}$

$= (1 - x + x^2 - x^3 + \cdots \ldots \ldots) - \frac{1}{2} . (1 - \frac{x}{2} + \frac{x^2}{4} - \frac{x^3}{8} + \cdots \ldots)$

$= \frac{1}{2} - (1 - \frac{1}{2^2})\,x + (1 - \frac{1}{2^3})\,x^2 - (1 - \frac{1}{2^4})\,x^3 + \cdots \ldots \ldots$

Thus, the corresponding numeric function is $a_r = (-1)^r\,(1 - \frac{1}{2^{r+1}}) \quad r \geq 0$

(c) $G(x) = \frac{1}{x^2-2x-3}$

$= \frac{1}{(x+1)(x-3)}$

$= \frac{1}{4}[\frac{1}{x-3} - \frac{1}{x+1}]$

$= \frac{1}{4}[-\frac{1}{3}(1 + \frac{x}{3} + \frac{x^2}{9} + \cdots \ldots) - (1 - x + x^2 - x^3 + \cdots \ldots)]$

$= \frac{1}{4}[\frac{1}{3}(1 + \frac{x}{3} + \frac{x^2}{9} + \cdots \ldots) + (1 - x + x^2 - x^3 + \cdots \ldots \ldots)]$

$= -\frac{1}{4}[(\frac{1}{3} + 1) + (\frac{1}{3^2} - 1)\,x + (\frac{1}{3^3} + 1)\,x^2 + \cdots \ldots \ldots + (\frac{1}{3^{r+1}} + (-1)^r)\,x^r + \cdots \ldots]$

Thus, the corresponding numeric function $a_r = -\frac{1}{4}(\frac{1}{3^{r+1}} + (-1)^r)$

(d) $G(x) = \frac{1}{1-x^4}$

$= 1 + (x^4) + (x^4)^2 + (x^4)^3 + \cdots \ldots \ldots . + (x^4)^r + \cdots \ldots \ldots$

$= 1 + x^4 + x^8 + x^{12} + \cdots \ldots \ldots + x^{4r} + \cdots \ldots \ldots$

Thus, the corresponding numeric function is $a_r = \{ \begin{matrix} 1 & if\ \ r = 4k,\ k = 0, 1, 2, 3 \ldots . \\ 0 & otherwise \end{matrix}$

9.3 Solution of Linear Recurrence Relations using Generating Functions

We can find the solution to a recurrence relation with initial conditions by finding an explicit formula for the associated generating function.

Example 9.4

Use generating functions to solve the recurrence relation.

(i) $a_n = 3a_{n-1} + 2 \qquad a_0 = 1$

(ii) $a_n - 9a_{n-1} + 20a_{n-2} \qquad a_0 = -3, a_1 = -10$

(iii) $a_{n+2} - 2a_{n-1} + a_n = 2^n \qquad a_0 = 2,\ a_1 = 1$

Solution

(i) Let

$$G(x) = \sum_{n=0}^{\infty} a_n x^n \quad \text{where } G(x) \text{is the generating function for the sequence } \{a_n\}.$$

Multiplying each term in the given recurrence relation by x^n and summing from 1 to ∞, we get

$$\sum_{n=1}^{\infty} a_n x^n = 3 \sum_{n=1}^{\infty} a_{n-1} x^n + 2 \sum_{n=1}^{\infty} x^n$$

$$G(x) - a_0 = 3xG(x) + 2\frac{x}{1-x}$$

$$(Since\ \ G(x) = a_0 + \sum_{n=1}^{\infty} a_n x^n, 3 \sum_{n=1}^{\infty} a_{n-1} x^n = 3xG(x), 2 \sum_{n=1}^{\infty} x^n$$

$$= 2\frac{x}{1-x})\ (1-3x)G(x) = a_0 + 2\frac{x}{1-x}$$

$$= \frac{1+x}{1-x}\ \ (Since\ \ a_0 = 1)$$

$$G(x) = \frac{1+x}{(1-x)(1-3x)} = \frac{2}{1-3x} - \frac{1}{1-x}$$

$G(x) = (1+x)/((1-x)(1-3x))\ \ =\ 2/(1-3x) - 1/(1-x"\ ")$

$$\therefore \qquad \sum_{n=0}^{\infty} a_n x^n = 2\sum_{n=0}^{\infty} 3^n x^n - \sum_{n=0}^{\infty} x^n$$

Hence $a_n = 2\,.\,3^n - 1$ which is the required solution.

(ii) Let

$$G(x) = \sum_{n=0}^{\infty} a_n x^n \ where\ G(x) is\ the\ generating\ function\ for\ the\ sequence\ \{a_n\}$$

Multiplying each term in the given recurrence relation by x^n and summing from

$2\ to\ \infty$, we get,

$$\sum_{n=2}^{\infty} a_n x^n - 9\sum_{n=2}^{\infty} a_{n-1} x^n + 20\ \sum_{n=2}^{\infty} a_{n-2} x^n = 0$$

or $\quad [G(x) - a_0 - a_1 x] - 9x[G(x) - a_0] + 20x^2 G(x) = 0$

or $\quad G(x)[1 - 9x + 20x^2] = a_0 + a_1 x - 9a_0 x$

or $\quad G(x) = \frac{a_0 \pm a_1 x - 9a_0 x}{1-9x+20x^2} = \frac{-3-10x+27x}{1-9x+20x^2}$

$\because a_0 = -3 \ \ and \ \ a_1 = -10$

$$= \frac{-3+17x}{(1-5x)(1-4x)}$$

or $\quad G(x) = \frac{2}{1-5x} - \frac{5}{1-4x}$ (by partial fraction)

$$\therefore \ \sum_{n=0}^{\infty} a_n x^n = 2\sum_{n=0}^{\infty} 5^n x^n - 5\sum_{n=0}^{\infty} 4^n x^n$$

Hence $\qquad a_n = 2.5^n - 5.4^n \; which\ is\ required\ solution.$

(iii) Let

$$G(x) = \sum_{n=0}^{\infty} a_n x^n \quad where\ G(x)\ is\ the\ generating\ function\ for\ the\ sequence\ \{a_n\}$$

Multiplying each term in the given recurrence relation by x^n and summing from 0 to ∞, we get,

$$\sum_{n=0}^{\infty} a_{n+2}x^n - 2\sum_{n=0}^{\infty} a_{n+1}x^n + \sum_{n=0}^{\infty} a_n x^n = \sum_{n=0}^{\infty} 2^n x^n$$

or $\qquad \frac{G(x)-a_0-a_1x}{x^2} - 2\left(\frac{G(x)-a_0}{x}\right) + G(x) = \frac{1}{1-2x}$

or $\qquad \frac{G(x)-2-x}{x^2} - 2\left(\frac{G(x)-2}{x}\right) + G(x) = \frac{1}{1-2x}$

or $\qquad (x^2 - 2x + 1)G(x) = 2 + 3x + \frac{x^2}{1-2x}$

or $\qquad G(x) = \frac{2}{(1-x)^2} + \frac{3x}{(1-x)^2} + \frac{x^2}{(1-2x)(1-x)^2}$

By partial fraction $\qquad \frac{x^2}{(1-2x)(1-x)^2} = \frac{1}{1-2x} - \frac{1}{(1-x)^2}$

$$\therefore \qquad G(x) = \frac{1}{(1-x)^2} + \frac{3x}{(1-x)^2} + \frac{1}{1-2x}$$

$$\therefore \qquad \sum a_n x^n = \sum (n+1)x^n + 3\sum n x^n + \sum 2^n x^n$$

Hence $\qquad a_n = (n+1) + 3n + 2^n = 1 + 4n + 2^n.$

9.4 Solution of combinatorial problem using Generating Function

Generating functions are useful in solving combinatorial problems. We know that combination of n differrent objects taken r at a time is given by $C(n,\ r)$. We also know that for a fixed positive integer n

$$C(n,0) + C(n,1) + C(n,2)x^2 + \cdots \dots\dots + C(n,n) = (1+x)^n$$

This shows that $(1 + x)^n$ is the generating function of the numeric function $a_r = C(n, r)$ and is the product of the function $(1+x)$ up to n times. The function $(1+x)$ is the sum of two terms x^0 and x^1, representing the selection of an object zero times and one time, respectively. Since each object can be selected zero times or one time, $(1 + x)$ is the factor for each and the generating function $(1 + x)^n$ is the product of all n factors. If we can select an object at most two times (zero, one, or two times), the factor corresponding to the object will be $(1 + x + x^2)$. In this way, we can find factors for different objects and, finally, the generating function of the required numeric function.

Example 9.5

(i) Find the generating function of a_r, the number of ways to select r balls from a pile of 3 green, 3 white, and 3 blue balls.

Solution

The generating function will be a multiplication of 3 factors corresponding to the 3 colours green, white, and blue. Since there are 3 balls of each colour, one can select 0, 1, 2, or 3 balls from each colour. Thus, each factor is $1 + x + x^2 + x^3$. Hence, the required generating function will be

$$G(x) = (1 + x + x^2 + x^3)^3$$

(ii) Find the generating function of a_r, the number of ways to select r objects from n objects with unlimited number of repetitions. Also find a_r.

Solution

each object can be selected 0, 1, 2, 3, … or infinite times. Thus, each factor will be $1 + x + x^2 + \cdots$ .... Since there are n objects, the required generating function is

$$G(x) = (1 + x + x^2 + \cdots \dots)^n$$

$$G(x) = (\frac{1}{1-x})^n = (1-x)^{-n}$$

Now $a_r = Coefficient\ of\ x^r in\ the\ expansion\ of\ (1-x)^{-n}$

$$a_r = \frac{(-1)^r((-n)(-n-1)\dots\dots\dots(-n-r+1))}{r!}$$

$$a_r = \frac{(n)(n+1)\dots\dots\dots(n+r-1)}{r!}$$

$$a_r = \frac{(n-1)!\,(n)(n+1)\ \dots\dots\dots(n+r-1)}{r!\,(n-1)!}$$

$$a_r = \frac{(n+r-1)!}{r!\,(n-1)!}$$

$$a_r = C(n+r-1, r)$$

Problems for Exercise

1. Find the generating functions corresponding to the following sequences:

   a. $(1, 2, 2^2, 2^3, \dots, 2^r, \dots)$

   b. $(1, \frac{2}{3}, \frac{3}{9}, \frac{4}{27}, \dots, \frac{r+1}{3^r}, \dots)$

2. Find the generating function of the numeric function defined as

$$a_r = \{ \begin{matrix} 3^r & for\ r\ is\ even \\ -3^r & for\ r\ is\ odd \end{matrix}$$

3. Find the generating function of the following numeric functions:

   a. $a_r = 5 . 2^r$

   b. $a_r = 3^r + 5^r$

4. Find the generating function of the numeric function that satisfies the following recurrence relation:

$$a_r = a_{r-1} + a_{r-2} \ for\ r \geq 2$$

$$a_0 = 0, \qquad a_1 = 1$$

5. Find the generating function of the numeric function that satisfies the recurrence relation $a_r = 2a_{r-1} + 1 \ for\ r \geq 1$, with the initial condition $a_0 = 1$. Hence find the solution of the recurrence relation.

6. Using the generating function, find the number of ways of selecting 6 objects from 3 types of objects if repetitions of up to 4 objects of each type are allowed.

7. How many solution of the equation $n_1 + n_2 + n_3 = 10$ $(n_i \geq 2)$ are possible?

8. Using the generating function , evaluate the sum $1^2 + 2^2 + 3^2 + \cdots + r^2$.

9. Find the generating function of a_r, the number of ways to select r roses from a bunch of 4 pink, 4 white, and 4 yellow roses.

10. Determine the numeric function corresponding to generating function:

$$G(x) = \frac{1}{x^2 - 2x - 3}$$

Block IV: Unit I Algebraic Structure

Learning Objectives

After completing this unit, the learner shall be able to:

- Explain Binary Composition & its properties;
- Explain the Definition of an algebraic structure;
- Define associative operation;
- Determine whether or not the given operation in Z are associative;

Algebras generally deal with discrete objects and are, therefore, is a natural part of discrete mathematics. Algebra as studied today consists largely of the investigation various types of structure which are abstractions from situations occurring widely in regular mathematical practice. Some of the most important of them are groups, rings and fields. Abstract algebra has many applications in computer science. For example, semi-groups have application to formal languages and automata theory, and groups have important application in in coding theory. Coding theory has developed techniques for introducing redundant information in transmitted data that help in detecting and sometimes in correcting errors. Some of these techniques make use of group theory. A particular important is in the use of finite machines in compiler design for the recognition of syntactically correct language structures. Another important notion in the study of algebra is that of a ring. Usually a group is equipped with one binary operation but a ring is consists of two binary operations connected by more than one relation. Coding theory represents a beautiful example of the applicability of abstract algebra. A variety of algebraic concepts can be used to describe codes and their properties.

10.1 Introduction

Let us consider a set $X = \{1, 2, 3, 4\}$. If we add any two numbers of the set X then it may be an element of the set X $(e.g.\ 1 + 2 = 3)$, or may not be an element of the set X $(e.g.\ 2 + 3 = 5)$. If it is possible to define an operation between two elements to produce an element of the set, then it builds a structure on the set X with respect to the defined operation and has a significant meaning. Different types of structures can be built by adding additional properties. These structures are quite important, in sense that the elements of the set are related to each other through some properties. An arbitrary set with one or more binary operations defined on it is generally referred to as an algebraic structure, which is useful to study the algebraic properties of the members of the set. We can relate many apparently unrelated concepts in terms of algebraic properties through the study of algebraic structures.

10.2 Binary Composition & its properties

Let X be any non-empty set. A binary operation $*$ on X is a rule to combine a pair of elements x and y of X in some way to form another element. Usually, we denote it by $x * y$. The word binary signifies that two elements are involved. Any given binary operation has certain algebraic properties, discussed below.

10.2.1 Closure Law

Let X be a non-empty set. Then the set X is called closed under $*$ if it satisfies the following property:

$$x \in X, y \in X \Longrightarrow x * y \in X$$

Example 10.1

(i) A non-empty set such as a set of natural numbers is closed under the binary operations such as addition and multiplication but are not closed under subtraction and division. This is because the difference of two natural numbers need not be a natural number, for example, $1 - 3 = -2$; similarly, the division of two natural numbers may not be a natural number, for example, $\frac{5}{2} = 2.5$

(ii) Let us consider the set $X = \{-2, -1, 0, 1, 2\}$. The set X is not closed under the binary operation addition, as $2 + 2 = 4$, which is not an element of the set X. It can be observed that X is also not closed under multiplication.

From these examples, it can be observed that not every finite set is closed under addition, multiplication, subtraction and so on.

10.2.2 Associative Law

The binary $*$ operation is said to be associative on the set X, if for all $x, y, z \in X$

$$x * (y * z) = (x * y) * z$$

If an operation is associative, then the term $x * y * z$ needs no parenthesis, and the terms $x * (y * z) = (x * y) * z$, and $x * y * z$ are equal.

Example 10.2

Addition and multiplication are associative on the set of integers.

10.2.3 Existence of Identity element

If there exists an element $e \in X$ such that for all $x \in X$

$$x * e = e * x = x$$

Then the element $e \in X$ is said to be the identity element of X.

Example 10.3

Let us consider the set of real numbers. With respect to the binary operation addition in the set of real numbers, 0 is the identity element, since for every real number $a,$

$$a + 0 = 0 + a = a.$$

With respect to the binary operation multiplication, 1 is the identity element, since for every real number a,

$$a\,.\,1 = 1\,.\,a$$

10.2.4 Existence of Inverse Element

If for each element $x \in X$ there exists an element $y \in X$ such that

$$x * y = y * x = e$$

Then the element $y \in X$ is called the inverse of $x \in X$.

Example 10.4

Let us consider the set of real numbers. With respect to the binary operation addition, for every real number a, there exist a real number $-a$ such that $a + (-a) = 0 = (-a) + a$. With respect to the binary operation multiplication. for every real number a, there exists a real number $\frac{1}{a}$ such that $a\,.\frac{1}{a} = 1 = \frac{1}{a}\,.\,a$

10.2.5 Commutative Law

The binary operation $*$ is said to satisfy the commutative law if $\forall\ x,\ y \in X$

$$x * y = y * x$$

Example 10.5

(i) In case of a set of real numbers, addition and multiplication are commutative, since for two real numbers a and b, $a + b = b + a$ and $ab = ba$.

(ii) But in case of matrices, If we took a set of square matrices of order n, then the multiplication between two matrices is not commutative.

let $A = \begin{bmatrix} 1 & 2 \\ 0 & 0 \end{bmatrix}$ and $B = \begin{bmatrix} 2 & 1 \\ 0 & 1 \end{bmatrix}$

Then $AB = \begin{bmatrix} 2 & 3 \\ 0 & 0 \end{bmatrix}$ and $BA = \begin{bmatrix} 2 & 4 \\ 0 & 0 \end{bmatrix}$

This shows that $AB \neq BA$

10.3 Definition of an algebraic structure

An algebraic structure is defined as a non-empty set with one or more than one binary operations. For example,$(N, +)$, $(Z, +)$, $(R, +, .)$ are algebraic structures. Noticeably addition and multiplication are both binary operations on the set R of real numbers. Thus, $(R, +, .)$ is an algebraic structure equipped with two operations.

Let X be a set and $*$ be a binary operation defined on the set X. The binary operation on the given set represents a structure between the elements of the set; thus, the algebraic system is also known as the algebraic structure and is denoted by $(X,*)$.

Problems for Exercise

11. Let $A = \{0,1\}$. Is A closed under:
    a. Multiplication
    b. Addition
12. Let $B = \{1,2\}$. Is B closed under:
    a. Multiplication
    b. Addition
13. Let $C = \{1, 3, 5, \dots \dots\} = \{n: n\ is\ odd\}$. Is C closed under:
    a. Multiplication
    b. Addition
14. Let $D = \{2, 4, 6, \dots \dots\} = \{n: n\ is\ even\}$. Is Dclosed under:
    a. Addition
    b. Multiplication
15. Let $F = \{2, 4, 8, \ldots\ldots\} = \{x: x = 2^n, n \in N\}$. Is F closed under:
    a. Multiplication
    b. Addition
16. Define associative operation
17. Consider the set $Z = \{\dots \dots \dots, -1, 0, 1, 2, \dots \dots\}$ of integers. Determine whether or not the following operation in Z are associative :
    a. Addition
    b. Subtraction
    c. Multiplication
18. Determine whether or not the following operations on the integers Z are associative:
    a. Division
    b. Exponentiation
19. Suppose an operation $*$ on a set S is not associative. How many ways can the product $a * b * c * d$ of the four elements be formed?

Block IV: Unit II Group I

Learning Objectives
After completing this unit, the learner shall be able to:

- Explain the definition and properties of Group;
- Define Semigroup;
- Define Monoid;
- Check whether the given set is a monoid;
- Check whether a given set forms an abelian group;

Group theory is one of the most important fundamental concepts of modern algebra. Groups come across unsurprisingly in various mathematical situations. They have found extensive uses in discipline like physics and biology predominantly in the study of crystal structure, configuration molecules and structure of human genes.

The structure of a group is one of the core mathematical structures. Hence, the study of various algebraic structures finds its origin here. In this unit, we shall define groups and study some of their basic properties.

11.1 Overview

Consider G as a nonempty set where $G \times G = \{(a,b)\colon\ a \in G,\ \ b \in G\}$.

If $f\colon\ G \times G \to G$, then f is called as binary operation on G. Hence, a binary operation on G is a function that allot an element of G to every ordered pairs of elements of G.

The symbols such as $+,\ .\ ,\ 0,\ *$ etc. are used to represent binary operations on a set. Therefore $+$ will be a binary operation on G if and only if

$$a+b \in G \quad for\ all\ \ a,b \in G\ \ and\ \ a+b\ \ is\ unique.$$

Likewise $*$ will be a binary operation on G if an only if

$$a*b \in G \quad for\ all\ \ a,b \in G\ and\ \ a*b\ \ is\ unique.$$

It is known as the closure property of the binary operation and the set G is said to be close under binary operation. A binary operation on a set G is occasionally called the composition in G. For the finite set, a binary operation on the set can be demarcated by means of a table, called the $\boldsymbol{composite\ table}$.

11.1.1 Definition and properties of Group

Consider $(G,*)$ as an algebraic structure, where $*$ is a binary operation, then the structure $(G,*)$ is called a group under * operation if the following conditions (properties) are satisfied

1. **Closure law:**
   The operation is closed under * i.e., $a * b \in G \;\; for\, all \;\; a, b \;\in G$.

2. **Associative law:**
   The binary operation is associative under * i.e.,
   $$a * (b * c) = (a * b) * c \quad for\, all \;\; a, b, c \;\in G$$

3. **Identity element**
   The existence of an identity element i.e., $e \in G$, such that $e * a = a * e = a\, for\, all\, a \in G$

4. **Inverse element**
   For every a in G, there exists an element a' (the inverse of a) in G such that
   $$a * a' = a' * a = e$$

Example 11.1

(i) A group $(Q, +)$ is a an algebraic system with the identity element 0. The inverse of $x \in Q \;\; is - x$.

(ii) A group $(R, +)$ is an algebraic system with the identity element 0. The inverse of $x \in R \;\; is - x$.

(iii) A group $(Q - \{0\}, .)$ is an algebraic system with the identity element 1. The inverse of $x \in Q - \{0\} \;\; is \;\frac{1}{x}$.

(iv) A group $(R - \{0\}, \; .)$ is an algebraic system with the identity element 1. The inverse of $x \in R - \{0\}\; is \;\frac{1}{x}$.

Theorem 11.1 (Elementary properties)

Assume $(G, \; *)$ as a group. Then the following supposed to be true:

(a) There exists one and only one identity.

(b) Each element has a unique inverse.

(c) $(a^{-1})^{-1} = a$ for each $a \in G$, where a^{-1} stands for the inverse of a.

(d) $(a * b)^{-1} = b^{-1} * a^{-1} \quad for\, all \;\; a, b \in G$.

(e) $a * b = a * c \Longrightarrow b = c \quad for\, all \;\; a, b, c \;\in G$ (left cancellation law).

(f) $b * a = c * a \Longrightarrow b = c \quad for\, all \;\; a, b, c \;\in G$ (right cancellation law).

Proof:

(a) Suppose e and e' are two elements of G that act as identity elements. Then as $e \in G$ and e' is the identity,

$$e * e' = e' * e = e$$

and as $e' \in G$ and e is the identity.

$$e' * e = e * e' = e'$$

From the two equations, $e = e'$

(b) Let $a \in G$ be any element and let a_1 and a_2 be two inverse elements of G then

$$a * a_1 = a_1 * a = e = a * a_2 = a_2 * a$$

Now $a_1 = a_1 * e = a_1 * (a * a_2) = (a_1 * a) * a_2 = e * a_2 = a_2$. Thus , the inverse of each element is unique.

(c) Since a^{-1} is the inverse of a,

$$a * a^{-1} = a^{-1} * a = e$$

which also implies that a is the inverse of a^{-1}. Thus, $(a^{-1})^{-1} = a$

(d) To prove $(a * b)^{-1} = b^{-1} * a^{-1}$, we shall show the following:

$$(a * b) * (b^{-1} * a^{-1}) = (b^{-1} * a^{-1}) * (a * b) = e$$

$(a * b) * (b^{-1} * a^{-1})$

$$= a * (b * b^{-1}) * a^{-1} \quad (using\ associative\ law)$$

$$= a * e * a^{-1} \quad (since\ b * b^{-1} = e)$$

$$= (a * e) * a^{-1} \quad (using\ associative\ law)$$

$$= a * a^{-1} = e \quad (since\ a * a^{-1} = e)$$

$(b^{-1} * a^{-1}) * (a * b)$

$$= b^{-1} * (a^{-1} * a) * b \quad (using\ associative\ law)$$

$$= b^{-1} * e * b \quad (Since\ a * a^{-1} = e)$$

$$= (b^{-1} * e) * b \quad (using\ associative\ law)$$

$$= b^{-1} * b = e \quad (since\ b * b^{-1} = e)$$

Hence, $(a * b)^{-1} = b^{-1} * a^{-1}$

(e) Let $a * b = a * c$

We know that $b = e * b$

$$= (a^{-1} * a) * b \quad (since\ a * a^{-1} = e) = a^{-1} * (a * b) \quad (using\ associative\ law)$$

$$= a^{-1} * (a * c) \quad (given\ a * b = a * c) = (a^{-1} * a) * c = e * c = c$$

Similarly, the right cancellation law can be proved.

11.2 Semi-group

Consider $(S, *)$ as an algebraic system in which S is a non-empty set and $*$ is a binary operation on set S. Thus S is closed under the operation $*$. Such a system consisting of a non-empty set S and a binary operation in S is called a $Groupoid$

An algebraic system $(S,*)$ is said to be a semi-group after following conditions been satisfied:

1. * the binary operation is a closed operation i.e., $a * b \in S \quad for\ all\ \ a, b \ \in S$. (closure law)
2. The binary operation $*$ is an associative operation i.e., $a * (b * c) = (a * b) * c \qquad for\ all\ a, b, c \in S$. (associative law).

Example 11.2

A set containing positive integers equipped with the operation addition is a semi-group.

11.3 Monoid

An algebraic system $(S,*)$ is said to be a monoid after following conditions been satisfied:

1. The binary operation $*$ is a closed operation. (closure law)
2. The binary operation $*$ is an associative operation (associative law).
3. There exists an identity element, i.e., for some $e \in S, \quad e * a = a * e = a \ \ for\ all\ a \in S$.

Therefore, a monoid is a semi-group $(S, *)$ that has an identity element.

Example 11.3

Check whether the set containing positive integers N is a monoid with under the binary operation $*$ demarcated as $a * b = LCM(a, b),\ \forall\ \ a, b \ \in N$.

Solution:

Closure property: The least common multiple (LCM) of two positive integers is a positive integer; thus, the set N is closed with respect to the binary operation LCM. For example, $LCM(3,\ 4) = 12$ and $LCM\ (4,\ 10) = 20$.

Associative law: From number theory, we know that

$$LCM(LCM(a, b), c) = LCM[a, LCM(b, c)]$$

This implies that $(a * b) * c = a * (b * c)$

Hence, the binary operation is associative.

Identity Element: We know that $LCM(1, a) = a = LCM(a, 1)$ for every positive integer a; thus, 1 is the identity element of the set of positive integers with respect to the binary operation.

Hence, the set of positive integers is a monoid with respect to the binary operation $*$.

11.4 Abelian Group

The term '*abelian*' is named after N.H. Abel, a Norwegian mathematician. Let G be a group under a binary operation $*$ is said to be $commutative$ or $abelian$ if it satisfies the commutative law i.e.

$$a * b = b * a \quad for\ all \quad a, b \in G$$

Note:

(i) A group equipped with addition as a binary operation is known as $\boldsymbol{additive\ group}$ and in case of multiplication as binary operation it known as $\boldsymbol{multiplicative\ group}$ **.**

(ii) The number of elements in a group G is said to be the order of that group. It is denoted by $o(G)$ or $|G|$. G is said to be finite if there are finite elements in it, on the other hand if there are infinite elements in G then it is said to infinite group.

Example 11.4

(i) Show that the set of integers Z forms an abelian group with respect to the addition of integers.

Solution:

$Closure\ property$: Since the sum of two integers is also an integer, the set Z is closed with respect to addition; that is, $\forall\ a, b\ \in Z,\ a + b\ \in Z$.

$Associative\ Law$: Since the sum of integers is associative, the set of integers Z satisfies the associative law; that is , $\forall\ a, b \in Z, \quad a + (b + c) = (a + b) + c$.

$Existence\ of\ Identity$: The integer $0 \in Z$ is the identity element as for all

$$a \in Z, \qquad a + 0 = a = 0 + a$$

$Existence\ of\ Inverse$: For every $a \in Z$, there exists $-a \in Z$ such that $a + (-a) = 0 = (-a) + a$. Thus the inverse of each element exists in Z with respect to addition.

$Commutative\ Law$: Since for all $a, b \in Z,\ \ a + b = b + a$, the integers satisfy the commutative law with respect to addition.

The set Z of integers satisfies all the properties of an abelian group. Thus, it forms an abelian group with respect to addition.

(ii) Show that the set of all positive rational numbers Q^+ forms an abelian group under the composition defined by $a * b = \frac{ab}{2}$.

Solution:

Closure property For every $a, b \ \in Q^+, \frac{ab}{2}$ is also in Q^+; therefore, Q^+ is closed with respect to the operation $*$.

Associative Law: let $a, b, c \in Q^+$. Then

$$(a * b) * c = \frac{ab}{2} * c = \frac{ab}{2} . \frac{c}{2} = \frac{a}{2} . \frac{bc}{2} = a * (b * c).$$

Hence, the operation $*$ is associative.

Existence of identity: If e be the identity element of Q^+,

$$a * e = a = e * a \implies \frac{ae}{2} = a \implies ae = 2a \implies a(e - 2) = 0$$

Since $a \neq 0,\ e = 2$. Thus, 2 is the identity element.

Existence of inverse: Let $a \in Q^+$. If b is the inverse of a, then $a * b = e = b * a$. Then $\frac{ab}{2} = 2 \implies b = \frac{4}{a}$. Thus , $4/a$ is the inverse of a.

Hence all of the postulates of group are satisfied. Further $* b = b * a$, the set Q^+ forms an abelian group under the given composition.

Problems for Exercise

1. Define an identity element for an operation $*$ on a set S.
2. Suppose e is a left identity and f is a right identity for an operation. Show that $e = f$.
3. Suppose an operation $*$ on a set S has an identity element e. Define the inverse of an element a in S.
4. Define
   a. Semigroup
   b. Monoid
5. Let S be a semigroup with an identity element e, and suppose b and b' are inverses of an element a in S. Show that $b = b'$, that is, that inverses are unique if they exist. We note that this result need not be true if the operation is non-associative.
6. Define the left and right cancellation laws for an operation $*$ on a set S.
7. Define commutative operation
8. Consider the set N of positive integers, and let $*$ be the operation of least common multiple (LCM) on N.
   a. Find $4 * 6,\ 3 * 5,\ 9 * 18$ and $1 * 6$
   b. Is $(N,*)$ a semi-group? Is it commutative
   c. Find the identity element of $*$.
9. Show that $(ab)^{-1} = b^{-1}a^{-1}$
10. Show that identity element in a group G is unique.
11. Prove that the set of all unimodulus complex number forms a group with respect to complex multiplication. Will it be abelian?

Block IV: Unit III Group II

Learning Objectives
After completing this unit, the learner shall be able to:

- Explain the definition and properties of cosets;
- Define the order of the group;
- Prove Lagrange's Theorem;
- Define cyclic group;
- Define permutation group;
- Define Ring;
- Define Field;

So far, we have discussed about an algebraic structure and its properties along with the group theory though it is most important fundamental concept in algebra.

Key terms that we used till now such as binary operations, composition of group, semi-group, monoid, abelian group must be kept in mind to understand the upcoming concepts.

In this unit, we are going to discuss about subgroup, permutation groups. cyclic groups, ring and fields.

Note: So far, we have been denoting a group composition by *. This notation can be replaced by the given group composition. Now, for simplification, we shall use ab in place of $a * b$ whenever required. There should be no misinterpretation that the composition is only multiplication.

12.1 Subgroup

Let $(G, *)$ be a group and $H$is a subset of G. $(H, *)$ is said to be a subgroup of G if $(H, *)$ is also a group by itself.

Now every set is a subset of itself. Thus, if G said to be a group, then the group G is a subgroup of itself G. Also let e be the identity element of the group G. Then the subset of G having only identity element is also a subgroup of G. So, the two subgroups $(G, *)$ and $(\{e\}, *)$ of the group $(G, *)$ are termed as $\boldsymbol{improper}$ or $\boldsymbol{trivial}$ subgroups, while others are termed $\boldsymbol{proper}$ or $\boldsymbol{nontrivial}$ subgroups.

Example 12.1

(i) The multiplicative group $\{1, -1, i, -i\}$ has a subgroup $\{1, -1, i, -i\}$

(ii) The group of even integers under the operation addition is a subgroup of the group of all integers under addition only.

(iii) The set Q^{+} of all non-zero positive rational numbers is a subgroup of the multiplicative group Q^{*} of all non-zero rational numbers

In order to check whether a given subset H of a group G is a subgroup of G or not, we need to go through all the axioms of the group. Following is the theorems given below to simplify this to some extent.

Theorem 12.1

A non-empty subset H of a group G is a subgroup of G if the following conditions are satisfied:

(a) $a,\ b \in H \Longrightarrow ab \in H$

(b) $a \in H \Longrightarrow a^{-1} \in H$

Proof: Let H be a subgroup of G. Then (a) and (b) hold immediately. Conversely, let the conditions (a) and (b) hold in H.

The closure property is satisfied due to (a)

Now, $a, b, c \in H \Longrightarrow a, b, c \in G \Longrightarrow a(bc) = (ab)c$. Hence, the associative law holds in H.

Since it is given that

$a \in H \Longrightarrow a^{-1} \in H, a \in H, a^{-1} \in H \Longrightarrow aa^{-1} \in H \Longrightarrow e \in H.$ (since $aa^{-1} \in H \Longrightarrow aa^{-1} \in G\ and\ \ aa^{-1} = e$)

Thus, an identity element exists in H.

Condition (b) shows that the inverse of each element exists in H. Since H satisfies all the conditions of group, it forms a group by itself, and hence, it is a subgroup of G.

Theorem 12.2

A non-empty subset H of a group G is a subgroup of G if $a, b \in H \Longrightarrow ab^{-1} \in H$

Proof:

Let H be a subgroup of G. Then $a, b \in H \Longrightarrow ab^{-1} \in H$

Conversely, let the condition hold in H. Then

$$a, a \in H \Longrightarrow aa^{-1} \in H \Longrightarrow e \in H \quad (since\ aa^{-1} \in H \Longrightarrow aa^{-1} \in G\ \ and\ \ aa^{-1} = e)$$

Thus, an identity element exists in H.

For any $a \in H$,

$$e, a \in H \Longrightarrow ea^{-1} \in H \Longrightarrow a^{-1} \in H \quad (since \;\; ea^{-1} \in H \Longrightarrow ea^{-1} \in G \;\; and \; ea^{-1} = a^{-1})$$

Therefore, the inverse of each element exists in H.

For any $a, b \in H$,

$$a, b^{-1} \in H \Longrightarrow a(b^{-1})^{-1} \in H \Longrightarrow ab \in H \; (b^{-1} \in H \Longrightarrow b^{-1} \in G \; and \;\; the \; inverse \;\; of \; b^{-1} \; is \; b)$$

Hence, H is closed with respect to the given composition.

Moreover, since the elements of H are the elements of G, the associative law holds in H.

Thus, H satisfies all the conditions of a group; hence, it forms a group by itself and it is a subgroup of G.

12.1.1 Cosets

Let $a \in G$ and H be a subgroup of a group G. Then the non-empty set $\{ah : h \in H\}$ is called the left coset generated by a and H and is symbolized as aH.

Also, the set $Ha = \{ha \; : h \in H\}$ is said to be right coset and is symbolized by Ha. The element a is called a characteristic of aH and Ha.

It is obvious that both aH and Ha are subsets of G. If e be the identity element of G, then $e \in H$ and $He = H = eH$. Thus, H itself is a left as well as right coset.

Usually $aH = Ha$, but in case of abelian group, each left coset coincides with the corresponding right coset.

If the group is under the operation addition, then the right coset of H in G generated by a is demarcated as

$$H + a = \{h + a : h \in H\}$$

Also, the left coset $a + H = \{a + h : h \in H\}$.

Example 12.2 Consider the following group G of integers under the operation addition:

$$G = \{\dots, -3, -2, -1, 0, 1, 2, 3, \dots\}$$

Consider H be a subgroup of G that holds only the elements which are multiples of 3.

$$H = \{\dots, -9, -6, -3, 0, 3, 6, 9, \dots\}$$

The group G is abelian and each right coset will be equal to its corresponding left coset. Let us form the right cosets of H in G. As $0, 1, 2 \in G$,

$$H + 0 = \{\dots, -9, -6, -3, 0, 3, 6, 9, \dots\} = H$$

$$H + 1 = \{\dots, -8, -5, -2, 1, 4, 7, 10, \dots\}$$

$$H + 2 = \{\dots, -7, -4, -1, 2, 5, 8, 11, \dots\}$$

$$H + 3 = \{\dots, -9, -6, -3, 0, 3, 6, 9, \dots\} = H$$

It is noticeable that $G = H \cup (H + 1) \cup (H + 2)$. Thus, any two right cosets are either equal or disjoint. The set of all disjoint right cosets produces a partition on the set G.

Properties of Cosets

Let H be a subgroup of G, and $a, b \ \in G$, Then,

1. $a \in aH$
2. $aH = H \iff a \in H$
3. $aH = bH \ or \ aH \cap bH = \emptyset$
4. $aH = bH \iff \ a^{-1}b \in H$

Similar results hold for right cosets.

12.1.2 Index of a subgroup in a group

Assume that H is a subgroup of a group G, then the amount of distinct right (left) cosets of H in G is called the index of H in G and is represented by $[G : H]$ or by $i_G(H)$.

12.1.3 Centralizer and Normalizer

Let G be a group. The center $Z(G)$ of the group G is defined as follows:

$$Z(G) = \{z \in G : zx = xz \ \ \forall \ x \in G\}$$

It can be proved that the center of a group G is a subgroup of G.

Assume H as a subgroup of a group G. Then the centralizer $C(H)$ and normalizer $N(H)$ of H in G are demarcated as follows:

$$C(H) = \{x \in G : xh = hx \ \forall h \in H\}$$

$$N(H) = \{x \in G : xH = Hx\}$$

$$= \{x \in G: xHx^{-1} = H\}$$

It can be easily shown that $C(H)$ and $N(H)$ are both subgroups of G and $C(H) \subseteq N(H)$.

A subgroup H of a group G is said to be a $\boldsymbol{normal\ subgroup}$ of G if $aH = Ha\ \forall\ \ a \in G$.

Example 12.3 Consider the group $(Z,\ +)$. Let $H = \{3n : n \in Z\}$ prove that H is a subgroup of Z.

Solution

It is a subgroup of Z since

(i) H is non-empty.

(ii) Let $x, y \in H$. Then there exist $p, q \in Z$ such that $x = 3p, y = 3q$

Now $xy^{-1} = 3p - 3q = 3(p - q)\ where\ p - q \in Z$

So $xy^{-1} \in H$

Therefore H is a subgroup of Z.

12.1.4 Order of a group

The order of the group can be defined as the number of elements in a group.

The order of a group G is symbolized by $o(G)$. A group of finite order is said to be a finite group. By using the notion of cosets we prove a theorem due to Langrange which expresses how the order of a finite group is related to the order of its subgroup.

12.1.5 Lagrange's Theorem

Theorem 3.3 The order of each subgroup of a finite group is a divisor of the order of the group.

Proof:

Let G be a finite group of order n and H be a subgroup of G such that $o(H) = m$ and $H = \{h_1, h_2, \dots \dots, h_m\}$. For $a \in G$, the right coset Ha is defined as $Ha = \{h_1a,\ h_2a, \dots \dots, h_ma\}$ and $o(Ha) = m$.

Each right coset of H in G will have m discrete elements. Besides this, any two right cosets are either the same or disjoint. Suppose there are k disjoint right cosets of H in G, then the union of these right cosets is equal to the set G. If the k disjoint right cosets are Ha_1, $Ha_2, \dots \dots, Ha_k$, then

$$G = Ha_1 \cup Ha_2 \cup \ldots \ldots \cup Ha_k$$

$$\Longrightarrow o(G) = o(Ha_1) + o(Ha_2) + \cdots \ldots \ldots \ldots + o(Ha_k)$$

$$\Longrightarrow n = mk$$

$$\Longrightarrow \frac{n}{m} = k$$

Since $k \in Z$, m is a divisor of n. Therefore, the order of every subgroup of a finite group is a divisor of the order of the group.

Since the k is number of disjoint right cosets of H in G, thus different right (left) cosets of H in $G = \frac{n}{m} = \frac{|G|}{|H|}$.

Theorem 12.4

If H and K be two subgroups of a group G, then HK is a subgroup of G if and only if $HK = KH$.

Proof:

Let HK be a subgroup of G and $x \in HK$

$$x \in HK \Longrightarrow x^{-1} \in HK \quad (since \;\; HK \; is \; a \; group)$$

$$\Longrightarrow x^{-1} = hk \;\; for \; some \; h \in H \;\; and \; k \in K$$

$$\Longrightarrow x = (hk)^{-1} = k^{-1}h^{-1}$$

$$\Longrightarrow x \in KH \qquad (since \;\; k^{-1}h^{-1} \in KH)$$

Thus, $HK \subseteq KH$

Similarly, it can be shown that $KH \subseteq HK$ and hence $HK = KH$

Conversely, let $HK = KH$

To prove that HK is a subgroup, we shall show that if $a, b \in HK$, then $ab^{-1} \in HK$.

$a, b \; \in HK \Longrightarrow a = h_1k_1, \;\; b = h_2k_2$ for some $h_1, h_2 \in H$ and $k_1, k_2 \in K$

Then

$$ab^{-1} = h_1k_1(h_2k_2)^{-1}$$

$$= h_1k_1(k_2^{-1} \; h_2^{-1})$$

$$= h_1(k_1k_2^{-1})h_2^{-1}$$

Since $k_1\ k_2^{-1} \in K$ and $h_2^{-1} \in H, (k_1k_2^{-1})h_2^{-1} \in KH$. Given that $HK = KH$; thus, $(k_1k_2^{-1})h_2^{-1} \in HK$. Hence, $k_1k_2^{-1} = h$ and $h_2^{-1} = k$ for some $h \in H$ and $k \in K$. Then

$$ab^{-1} = h_1(hk)$$

$$= (h_1h)k$$

Thus, $ab^{-1} \in HK \quad [since \quad (h_1h)k \in HK]$

Hence, HK is a subgroup.

12.3 Cyclic Group

A group G is called a cyclic group if, for some $a \in G$, every element of G is of the form a^n, where n is some integer i.e., $G = \{a^n : n \in Z\}$. The element a is then called a generator of G.

If G is a cyclic group generated by a, it is denoted by $G =< a >$. The element of G are in the form

$$\dots\dots\dots\dots\dots a^{-2}, a^{-1}, a^0, 0, a, a^2, a^3 \dots\dots\dots\dots$$

There may be more than one generator of a cyclic group. Every cyclic group has at least two generators, generator and inverse of it.

Example 12.4 The set of integers with respect to $+$ i.e. $(Z, +)$ is a cyclic group, a generator being 1.

Solution:

We have $1^0 = 1,\ 1^1 = 1,\ 1^2 = 1 + 1 = 2,\ 1^3 = 1 + 1 + 1 = 3\ \ and\ \ so\ \ on$

Similarly, $1^{-1} = inverse\ of\ 1 = -1$

$1^{-2} = (1^2)^{-1} = -2, 1^3 = (1^3)^{-1} = (3)^{-1} = -3$ and so on

Thus, each element of G can be expressed as some integral power of 1

Similarly, we can show that -1 is also a generator.

Example 12.5 The multiplicative group $\{1, w, w^2\}$ is a cyclic group.

Solution: We have $w^0 = 1$, $w^1 = w$, $w^2 = w^2$, $w^3 = 1$ and $(w^2)^0 = 1$, $(w^2)^1 = w^2$, $(w^2)^2 = w^4 = w$

Thus each element of the group can be expressed as some integral powers of w and w^2. Hence the group is a cyclic group with generators w and w^2

Some important properties of Cyclic Groups

(i) Every cyclic group is an abelian group

(ii) If a is a generator of a cyclic group G, then a^{-1} is also a generator of G.

(iii) If a cyclic group G is generated by an element a of order n, then a^m is a generator of G if and only if the greatest common divisor of m and n is 1 i.e., if and only if m and n are relatives primes.

Infinite Cyclic Group

If H is a cyclic group generated by a subject to all the powers of a are distinct, then

$H =< a >$ is an infinite cyclic group.

12.4 Permutation Group

Let A be a finite set. Then a function $f : A \to A$ is said to be a permutation of A if

(i) f is one-one

(ii) f is onto

i.e. A bijection from A to itself is called a permutation of A.

The number of distinct elements in the finite set A is called the degree of permutation.

Consider a set $A = \{a_1, a_2, \dots \dots, a_n\}$ and let $f : A \to A$ be a bijection function. Then every element of A has a unique image in A, no two distinct elements of A have the same image, and every element of A has a unique pre-image, under f. Thus, the range of f is of the form

$$Ran(f) = \{f(a_1), f(a_2) \dots\dots\dots\dots\dots f(a_n)\}$$

In the notation of the relations the function f is given by

$$f = \{(a_1, f(a_1)), (a_2, f(a_2)), \dots\dots\dots\dots, (a_n, f(a_n))\}$$

This is written in two line notation as

$$f = \begin{pmatrix} a_1 & a_2 & & a_n \\ f(a_1) & f(a_2) & \ldots\ldots\ldots\ldots\ldots & f(a_n) \end{pmatrix}$$

Since A is a finite set, its elements can be ordered as the first, the second, ….. , the n^{th}. Therefore, it is convenient to take A to be a set of the form $\{1,\ 2,\ 3,\ \ldots\ \ldots\ ,\ n\}$ for some positive integer n instead of $\{a_1, a_2, a_3, \ldots\ldots\ldots, a_n\}$.

In general, a permutation f on the set $\{1,\ 2,\ 3,\qquad ,n\}$ can be written as

$$f = \begin{pmatrix} 1 & 2 & \ldots\ldots\ldots\ldots & n \\ f(1) & f(2) & & f(n) \end{pmatrix}$$

Obviously, the order of the column in the symbol is immaterial so long as the corresponding elements above and below in that column remain unchanged.

12.4.1 Equality of two permutations

Let f and g be two permutations on a set X. Then $f = g$ if and only if $f(x) = g(x)$ for all x in X.

Example 12.6 Let f and g be given by

$f = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \end{pmatrix}$ $\qquad\qquad$ $g = \begin{pmatrix} 3 & 2 & 1 & 4 \\ 4 & 3 & 2 & 1 \end{pmatrix}$

Evidently $f(1) = 2 = g(1)$, $\qquad f(2) = 3 = g(2)$

$f(3) = 4 = g(3)$, $\qquad f(4) = 1 = g(4)$

Thus $f(x) = g(x)$ for all $x \in \{1,\ 2,\ 3,\ 4\}$ which implies $f = g$

12.4.2 Identity Permutation

If each element of a permutations be replaced by itself. Then it is called the identity permutations and is denoted by symbol I. For example,

$I = \begin{pmatrix} a & b & c \\ a & b & c \end{pmatrix}$ is an identity permutation.

12.4.3 Product of Permutations (or composition of Permutation)

The product of two permutations f and g of same degree is denoted by $f\ o\ g\ \ or\ \ fg$, meaning first perform g and then perform f.

$$f = \begin{pmatrix} a_1 & a_2 & a_3 & & a_n \\ b_1 & b_2 & b_3 & \cdots\cdots & b_n \end{pmatrix},$$

$$g = \begin{pmatrix} b_1 & b_2 & b_3 & \dots\dots & b_n \\ c_1 & c_2 & c_3 & & c_n \end{pmatrix},$$

Then $fg = \begin{pmatrix} a_1 & a_2 & a_3 & \dots\dots & a_n \\ c_1 & c_2 & b_3 & & c_n \end{pmatrix}$

For, f replaces a_1 by b_1 and then g replaces b_1 by c_1 so that fg replaces a_1 by c_1. Similarly, fg replaces a_2 by c_2, a_3 by $c_3, \dots\dots a_n$ by c_n.

Clearly fg is also a permutation on S.

It should be observed that the permutation g has been written in such a manner that the second row of f coincides with the first row of g. This is most essential in order to find fg.

If we want to write gf, then f should be written in such a manner that the second row of g must coincide with the first row of f.

12.4.4 Inverse Permutation

Since a permutation is one-one onto map and hence it is inversible, i.e., every permutation f on a set

$$P = \{a_1, a_2 \dots\dots\dots, a_n\}$$

has a unique inverse permutation denoted by f^{-1}.

Thus if $\quad f = \begin{pmatrix} a_1 & a_2 & \dots\dots\dots & a_n \\ b_1 & b_2 & & b_n \end{pmatrix}$

Then $\quad f^{-1} = \begin{pmatrix} b_1 & b_2 & \dots\dots\dots & b_n \\ a_1 & a_2 & & a_n \end{pmatrix}$

Total Number of Permutations

Let X be a set consisting of n distinct elements. Then the elements of X can be permuted in $n!$ distinct ways. If S_n be the set consisting of all permutation of degree n, then the set S_n will have $n!$ distinct permutations of degree n . This set S_n is called the symmetric set of permutations of degree n.

Example 12.7 If $A = \{1, 2, 3\}$, then $S_3 = \{p_0, p_1, p_2, p_3, p_4, p_5\}$ where

$$p_0 = I_A = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 2 & 3 \end{pmatrix}, \; p_1 = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \end{pmatrix} \; p_2 = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \end{pmatrix} \; p_3 = \begin{pmatrix} 1 & 2 & 3 \\ 1 & 3 & 2 \end{pmatrix} \; p_4 = \begin{pmatrix} 1 & 2 & 3 \\ 3 & 2 & 1 \end{pmatrix} \; p_5 = \begin{pmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{pmatrix}$$

The multiplication table for the composition of permutation in S_3 is as given below:

Multiplication table for S_3

o	p_0	p_1	p_2	p_3	p_4	p_5
p_0	p_0	p_1	p_2	p_3	p_4	p_5
p_1	p_1	p_2	p_0	p_5	p_3	p_4
p_2	p_2	p_0	p_1	p_4	p_5	p_3
p_3	p_3	p_4	p_5	p_0	p_1	p_2
p_4	p_4	p_5	p_3	p_2	p_0	p_1
p_5	p_5	p_3	p_4	p_1	p_2	p_0

The table shows that

(i) The multiplication of any two permutations of S_3 gives a permutation of S_3. So, S_3 is closed with respect to multiplication.

(ii) Associativity law holds for $(p_1\, p_3)\, p_4 = p_5\, p_4 = p_0$ and $p_1\,(p_3\ \ p_4) = p_1\, p_1 = p_0$

(iii) Identity element exists, p_0 when composed with any permutation gives that permutation.

(iv) Every permutation has its own inverse.

Hence S_3 is group. It is a non-commutative group since $p_1\, p_2 \neq p_2\, p_1,\ \ p_3\, p_2 \neq p_2\, p_3$

Let A be a set of degree n. Let P_n be the set of all permutations of degree n on A. Then $(P_n,*)$ is a group, called a permutation group and the operation $*$ is the composition (multiplication) of permutations.

12.4.5 Cyclic Permutations

A permutation which replaces n objects cyclically is called a cyclic permutation of degree n. Let us consider the permutation.

$$P = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \end{pmatrix}$$

This assignment of values could be presented schematically as follows.

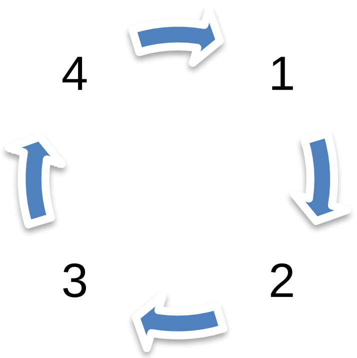


Fig 12.1 Circular permutation

Such diagrams are cumbersome, we leave out the arrows and simply write $S = (1\ \ 2\ \ 3\ \ 4)$. We read the new symbols in cyclical order from left to right as follows: 1 is replaced by 2, 2 is replaced by 3, 3 is replaced by 4, and 4 is replaced by 1

Thus, the meaning of the symbol is to replace each number which follows and the last number by the first.

Note that $(1\,2\,3\,4) = (2\,3\,4\,1) = (3\,4\,1\,2) = (4\,1\,2\,3)$. Thus a circular permutation may be denoted by more than one rowed symbols.

The number of elements permuted by a cycle is said to be its length and the disjoint cycles are those which have no common elements.

12.5 RING

A non-empty set R, together with two binary compositions $+$ and . , is said to form a ring if the following axioms are satisfied:

1. The set R is closed with respect to the binary composition $+$ and . , that is $\forall\ x, y \in R, x + y \in R\ \ and\ xy \in R$.
2. Addition is associative; that is, $x + (y + z) = (x + y) + z$ for all $x, y, z\ \in R$
3. Addition is commutative; that is, $x + y = y + x\quad \forall\ x, y \in R$
4. There exists an element denoted by 0 in R such that $0 + x = x\quad \forall\ x \in R$
5. For each element $x \in R$, there exists an element $-x \in R$ such that $x + (-x) = (-x) + x = 0$
6. Multiplication is associative; that is, $x(yz) = (xy)z\qquad \forall x, y, z \in R$
7. Multiplication is distributive with respect to addition; that is, for all $x, y, z\ \in R$, $x(y + z) = xy + xz\ (left\ distributive\ law)$ and $(y + z)x = yx + zx\ (right\ distributive\ law)$

It can be observed that R is an abelian group with respect to addition. We can have any other binary compositions in place of addition and multiplication. Since these compositions are natural and their properties are easy to understand, we have taken these compositions to define a ring; otherwise, any two symbols can be used to denote the two compositions.

12.5.1 Commutative Ring

A ring R is called a commutative ring if multiplication is also commutative, that is, $\forall\, x, y \in R$, $xy = yx$.

12.5.2 Ring with Unity

If in a ring R there exists an element $e \in R$ such that $\forall x \in R,\ xe = ex = x$, then R is called a ring with unity. Generally, we denote the unity by 1. The element 1 is called the multiplicative identity.

Example 12.8

(i) The set of integers forms a ring with respect to usual addition and multiplication. This is also a commutative ring with unity.

(ii) The set of all even integers is a commutative ring without unity with respect to usual addition and multiplication.

Theorem 12.5 If R is a ring, then the following results hold for all $x, y, z \in R$:

(a) $x\,.\,0 = 0\,.\,x = 0$
(b) $x(-y) = (-x)y = -xy$
(c) $(-x)(-y) = xy$
(d) $x(y - z) = xy - xz$

Proof:

(a) $x\,.\,0 = x\,.\,(0 + 0)$
$\Rightarrow x.\,0 = x\,.\,0 + x\,.\,0$
$\Rightarrow x\,.\,0 + 0 = x\,.\,0 + x\,.\,0$
$\Rightarrow 0 = x\,.\,0$ (using left cancellation law, as $< R, + >$ is a group)

(b) $x\,.\,0 = 0$
$\Rightarrow x(-y + y) = 0$
$\Rightarrow x(-y) + xy = 0$
$\Rightarrow x(-y) = -(xy)$
Similarly, $(-xy) = -xy$

(c) $(-x)(-y) = -[x(-y)] = -(-xy) = xy$

(d) $x(y - z) = x[y + (-z)]$
$= xy + x(-z)$
$= xy - xz$

12.5.3 Zero Divisor of Ring

Let R be a ring and 0 be the additive identity of the ring. We have already proved that for any element $x \in R,\ x0 = 0 = 0x$. However, in some of the rings, it may be possible that $xy = 0$

when neither $x = 0$ nor $y = 0$. This phenomenon leads to the definition of zero divisors. A non-zero element $x \in R$ is called a zero divisor if there exists an element $y \in R(y \neq 0)$ such that $xy = 0$ or $yx = 0$.

Example 12.9 Let M be a ring of all 2×2 matrices, with their elements as integers and addition and multiplication being the two ring operation. Then M is a ring with zero divisors, as for $A = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ and $B = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$, we have $AB = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$. Here, the null matrix is the zero element of the ring.

12.5.4 Subrings

Let S be a non-empty subset of a ring R. S is called a subring of R if S itself forms a ring under the binary compositions of R.

Example 12.10 Let us consider the ring $(R, +, .)$ of real numbers. The ring of integers $(Z, +, .)$ is a subring of $(R, +, .)$.

12.5.5 Ring Homomorphism

Let $(R, +, .)$ and $(R_1, *, o)$ be two rings. A mapping $f: R \rightarrow R_1$ is called a homomorphism if for all $x, y \in R$

$f(x + y) = f(x) * f(y)$

And $f(x.y) = f(x)\ o\ f(y)$

Here, the binary compositions in the second ring are denoted by $*, o$ in order to avoid any confusion is defining the ring homomorphism. If we take the usual notations $+, .$ in both of the rings, then the mapping $f: R \rightarrow R_1$ is called a homomorphism if for all

$x, y \in R$

$f(x + y) = f(x) + f(y)$ and

$f(x.y) = f(x).f(y)$

An isomorphism in rings is the one-one onto homomorphism.

12.5.6 Integral Domain

A commutative ring R is called an integral domain if R has no zero divisors; that is, if $ab = 0$ in R, then either $a = 0$ or $b = 0$.

Example 12.11 The ring of the integers $< Z, +, . >$ is an integral domain.

Theorem 12.6 A commutative ring R is an integral domain if for all $a, b, c \in R(a \neq 0)$ $\quad ab = ac \Longrightarrow b = c$

Proof: Let R be an integral domain. Then for $(a \neq 0)$

$ab = ac \Longrightarrow ab - ac = 0$

$\Longrightarrow a(b - c) = 0$

$\Longrightarrow a = 0$ or $(b - c) = 0$

$\Longrightarrow b - c = 0$ since $a \neq 0$

$\Longrightarrow b = c$

Conversely, let for all $a, b, c \in R(a \neq 0),\ ab = ac \Longrightarrow b = c$

Then $ab = 0 \Longrightarrow ab = a0$

$\Longrightarrow b = 0$

Thus, R is without zero divisors, and hence, R is an integral domain.

12.5.7 Division Ring or Skew Field

A ring R with unity is called a division ring or skew field if the non-zero elements of R form a group with respect to multiplication. In other words, a ring R is called a division ring or a skew field if it satisfies the following two conditions:

1. There exists unity.
2. Each non-zero element possesses a multiplicative inverse.

Since a division ring forms groups with respect to two binary operations, it must contain two identity elements 0 and 1 (with respect to addition and multiplication), and thus, a division ring has at least two elements.

12.6 Field

A commutative division ring is called a field. In other words, a ring R is called a field if it satisfies the following conditions:

1. R is commutative.
2. There exists unity.
3. Each non-zero element possesses a multiplicative inverse.

Example 12.12 The ring of rational number $(Q, +, .)$ is a field. The ring of real numbers also forms a field under usual addition and multiplication.

Theorem 12.7 A field is an integral domain.

Proof: Let $(R, +, \ .)$ be a field. Then R is a commutative ring. Let $xy = 0$ in R. We have to show either $x = 0$ or $y = 0$. Let us assume that $x \neq 0$. Then x^{-1} exists, as R is a ring. Thus, $xy = 0 \Longrightarrow x^{-1}0 = y \Longrightarrow y = 0,$ which shows that R is an integral domain.

Theorem 12.8 A field is non-zero finite integral domain.

Proof: First, we shall show that there exists unity.

Let $0 \neq x \in R$ be any element.

Then $xx_1, xx_2, \dots \dots, xx_n$ are the elements of R. If $xx_i = xx_j$ for some $i \neq j$, then by cancellation, we get $x_i = x_j$, which is not true.

Hence, $xx_1, xx_2, \dots \dots, xx_n$ are distinct elements of R placed in some order. One of these elements will be equal to x. Thus, $x = xx_i$ for some i. Let $a \in R$ be any element. Then

$$xa = (xx_i)a$$

$$\Longrightarrow xa = x(x_i a)$$

$$\Longrightarrow a = x_i a$$

Since commutative law holds in R,

$$a = x_i a = a\_x_i$$

Thus, x_i is the unity of R and we shall denote it by1. Hence, for $1 \in R,\ 1 = xx_j$ for some j, which shows that x_j is the multiplicative inverse of x. Any non-zero element of R has a multiplicative inverse, and therefore, R is a field.

Problem for Exercise:

1. Suppose H is a subset of a group G. Show that H is a subgroup of G if H has the following three properties:
    a. The identity element e belongs to H
    b. H is closed under operation of G, i.e. if $a, b \in H$ then $ab \in H$
    c. H is closed under inverses, i.e., if $a \in H$ then $a^{-1} \in H$
2. Consider the group Z of integers under addition. Let H be the subset of Z consisiting of all multiples of a positive integer m; that is, $H = \{\ldots, -3m, -2m, -m, 0, m, 2m, 3m, \ldots\}$. Show that H is a subgroup of $\boldsymbol{Z}$.
3. Let G be any group and let a be any element of G. Define the cyclic group generated by a, denoted by $gp(a)$.
4. Let a be any element in a group G. Describe the cyclic group $gp(a)$ when $gp(a)$ is finite, and define the order of a.
5. Let H be a subgroup of a group G. Define a right (left) coset of H.
6. Let H be a subgroup of a group G. Define the index of H in G, denoted by $[G: H]$
7. Consider the group Z of integers under addition and the subgroup $H = \{\ldots\ldots, -10, -5, 0, 5, 10, \ldots\ldots\}$ consisting of the multiples of 5. Find
    a. The coset of H in Z
    b. The index of H in Z
8. Consider the ring Z of integers.
    a. Is Z commutative?
    b. Does Z have a unity element?
    c. What are the units in Z
9. Prove that $\{x | x.x = e\}$ will be a subgroup of an abelian group $(G, .)$
10. Show that $a.0 = 0.a = 0$ in a ring R.

Block V: Unit I Graph Theory

Learning Objectives
After completing this unit, the learner shall be able to:

- Know the basic terminology of graph theory;
- Define various Types of Graphs;
- Define Euler Graph;
- Explain Hamiltonian path and Circuit;
- Define the set of vertices, the set of edges and the degree of each vertex in a given Graph;
- Determine whether a given multigraphs is a Graph;
- Define Graph Coloring & Chromatic number;

Many situations that occur in Computer Science, Physical Science, Communication Science, Economics and many other areas can be analysed by using techniques found in a relatively new areas of mathematics called graph theory.

Let us consider a set of different places in a city. To show the connectivity of these places, we often use a pictorial representation in which the places are denoted by dots and a line or a curve joins two dots if there is a route between those two places. This representation of places (vertices) and routes (edges) is called a graph and can be treated as an abstract mathematical system.

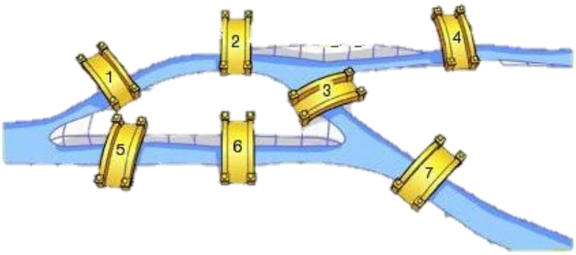


Fig. 13.1

The graphs can be used to represent almost any problem involving discrete arrangements of objects, where concern is not with the internal properties of these objects but with relationship among them.

The foundations of graph theory were originated by the $K\ddot{o}nigsberg\ bridge$ problem. $K\ddot{o}nigsberg$(Kaliningrad, a part of Russia) was a Prussian city situated on the sides of the

Pregel River. The figure 13.1 shows the river banks and the two islands formed by the splitting of the river. The river banks and islands were connected to each other through seven bridges.

The $Königsberg$ bridge problem was to find whether it was possible to walk through the town in such a way as to traverse every bridge exactly once. In 1736, Leonhard Euler, a Swiss mathematician, came out with the solution in terms of graph theory. He presented the problem in a simple way by representing the landmasses by dots and bridges by lines that connected those landmasses. Euler proved that it was not possible to walk through the town by traversing through the seven bridges, crossing each bridge exactly once. He also explained why it was not possible. He introduced the concept of degree of a node, the number of edges touching a node, and proposed that any given graph can be traversed with each edge traversed exactly once if and only if it had zero or exactly two nodes of odd degree.

Graph theory deals with study of graphs and their various properties. It has wide applications, as many real-life problems can be modeled through graphs. Communication network, data organization, link structures of web pages, job assignment, electrical network, and so on can easily be understood through graphs. Here we begin with some basic graph terminologies and then discuss some important concepts in graph theory.

13.1 Basic Terminology

A graph G consists of two sets:

(i) A non-empty set V whose elements are called vertices, nodes or points of G. The set $V(G)$ is called the **vertex set** of G

(ii) A set E of edges such that each edge $e \in E$ associated with ordered or unordered pairs of elements of V. The set $E(G)$ is called the **edge set** of G

The graph G with vertices V and edges E is written as $G = (V, E)$ or $G(V, E)$.

If an edge $e \in E$ is associated with an ordered pair (u, v) or an unordered pair (u, v), where $u, v \in V$, then e is said to connect u and v which are called as end points of e. An edge is said to be incident with the vertices in joins. Thus, the edge e that joins the nodes u and v is said to be incident on each of its end points u and v. Any pair of nodes that is connected by an edge in a graph is called **adjacent nodes.**

In a graph a node that is not adjacent to another node is called an **isolated node.** A graph $G(V, E)$ is said to be **finite** if it has a finite number of vertices and finite number of edges. (A graph with a finite number of vertices must also have finite number of edges): otherwise, it is a **infinite** graph.

If G is a finite, $|V(G)|$ denotes the number of vertices in G and is called the **order** of G. Similarly if G is finite, $|E(G)|$ denotes the number of edges in G and is called the **size** of G. We shall often refer to a graph of order n and size m an (n, m) graph. If G be a (p, q) graph then G has p vertices and q edges.

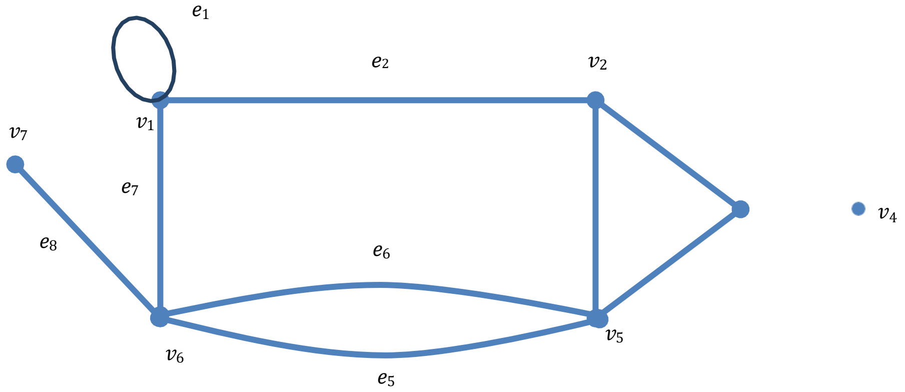


Fig 13.2 Graph with self-loop and parallel edges

Above figure represents a graph $G(V,\ E)$ having seven vertices and eight edges, the set of vertices and edges are defined as follows:

$V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$ and $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$

Degree of a Vertex

The degree of a vertex v is the number of edges incident on the vertex. It is a positive number and is denoted by $\deg(v)$. A loop is counted twice for the calculation of degree of a vertex. In above figure, deg $(v_2) = 3$

In a graph $G(V, E)$, the minimum and maximum degrees of a vertex are denoted by $\delta(G)\ \ and\ \Delta(G)$, respectively.

$$\delta(G) = Min(\deg(v) : v \in G(v))$$

$$\Delta(G) = Max(\deg(v) : v \in G(v))$$

Isolated Vertex and Pendent Vertex

A vertex is called an isolated vertex if no edge is incident on the vertex. The degree of an isolated vertex is zero for example v_4 in Fig 13.2.

A vertex is called a pendent vertex if the degree of the vertex is one for example, v_7 in Fig 13.2

Self-loop and parallel Edges

An edge starting and ending in the same vertex is called a self-loop. The edges e_1 forms a self-loop (fig13.2). Edges associated with the same pair of vertices are called parallel edges. As per above figure 13.2, the edges e_5 and e_6 are parallel edges.

Example 13.1

Define formally the graph given in figure 13.3 that is, the set of vertices, the set of edges and the degree of each vertex.

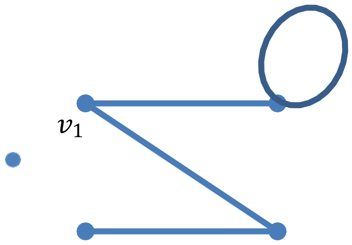


Fig 13.3

Solution

The graph $G(V, E)$ can be defined as follows:

$V = \{v_1, v_2, v_3, v_4, v_5\}$; $E = \{(v_1, v_2), (v_2, v_2), \ (v_1, v_3), \ (v_3, v_4)\}$; and

$$\deg(v_1) = 2, \deg\,(v_2) = 3, \deg(v_3) = 2, \deg(v_4) = 1 \ \ and \ \ \deg(v_5) = 0$$

Theorem13.1 Let $G(V, E)$ be a graph having n vertices e edges. Then show that

$$\sum_{i=1}^{n} \deg\,(v_i) = 2e$$

Proof: Consider a graph of n vertices $v_1, \ v_2, \ v_3, \ \ldots \ \ldots \ \ldots \ , \ v_n$ and e edges. Every edge is incident on two vertices; hence, every edge is counted twice for the calculation of the total degree of all vertices. . This fact leads to the conclusion that the sum of degrees of all vertices is twice the number of edges in the graph. That is,

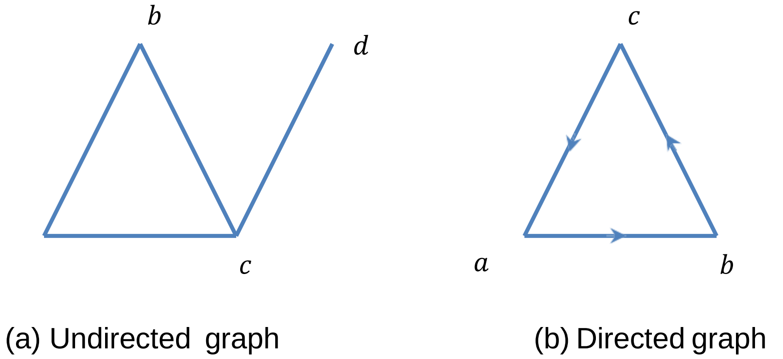
$$\sum_{i=1}^{n} \deg\,(v_i) = 2e$$

Undirected and Directed Graph

An undirected graph G consists of set V of vertices and a set E of edges such that each edge $e \in E$ is associated with an unordered pair of vertices.

The graph shown in fig 13.4(a) is an example of an undirected graph we can refer to an edge joining the vertex pair i and j as either (i, j) or (j, i).

A directed graph (or digraph) G consists of a set V of vertices and a set E of edges such that $e \in E$ is associated with an ordered pair of vertices. In other words, if each edge of the graph G has a direction then the graph is called directed graph. In the fig 1.3(b) each edge $e = (u, v)$ is represented by an arrow or directed curve from initial point u of e to the terminal point v



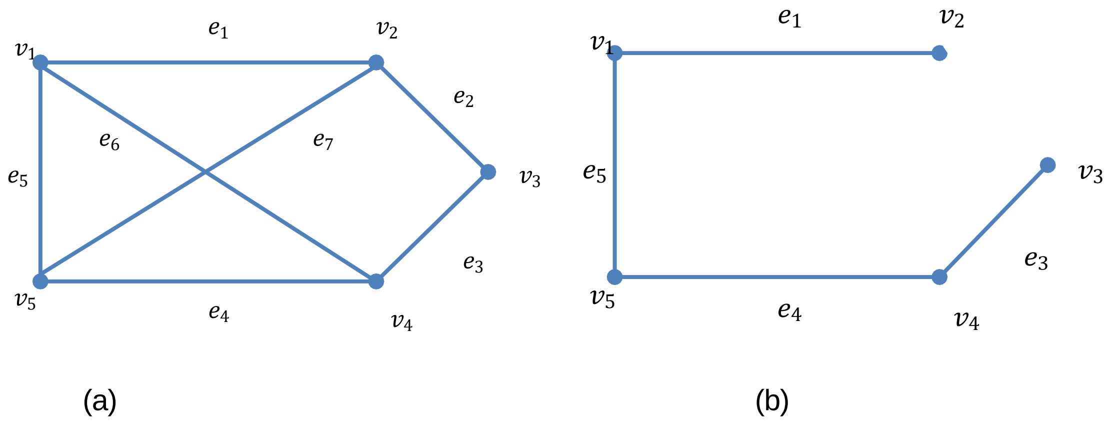
(a) Undirected graph

(b) Directed graph

Fig 13.4

Sub-graphs

A graph $G_1(V_1, E_1)$ is said to be a subgraph of a graph $G(V, E)$ if $V_1 \subseteq V$ and $E_1 \subseteq E_2$ and each edge has the same end vertices in G_1as in G



(a)

(b)

Fig. 13.5

Subgraph of a graph (a) $G(V, E)$ (b) $G_1(V_1, E_1)$

13.2 Types of Graphs

13.2.1 Simple Graph

A graph without self-loops and parallel edges is called a simple graph. A graph is called finite if it has a finite number of edges and finite number of vertices; otherwise, it is an infinite graph.

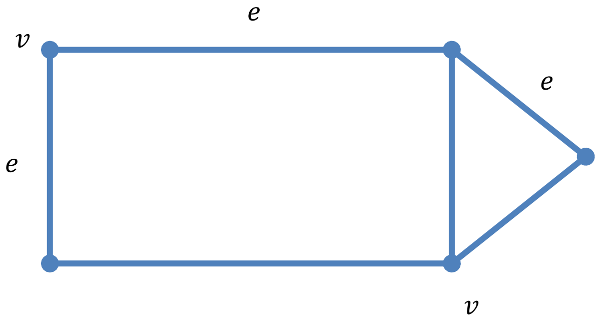


Fig. 13.6

13.2.2 Multi-graph, Trivial Graph and Null Graph

A graph having some parallel edges is called a multi-graph. A graph is called trivial graph if it has one vertex and no edges. A graph having finite vertices is called a null graph if it has no edges.

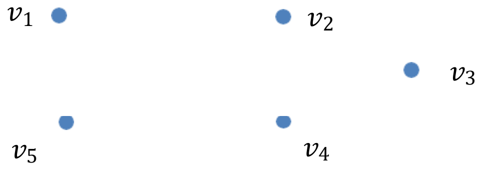


Fig 13.7

13.2.3 Pseudo-graph

A graph in which loops and multiple edges are allowed, is called a pseudo-graph. It may be noted that there is some lack of standardization of terminology in graph theory. Many words have almost obvious meaning, which are the same from book to book, but other is used differently by different authors.

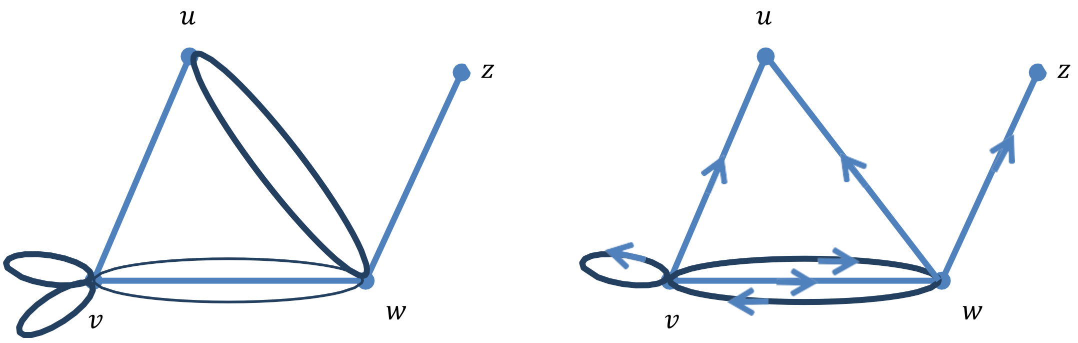
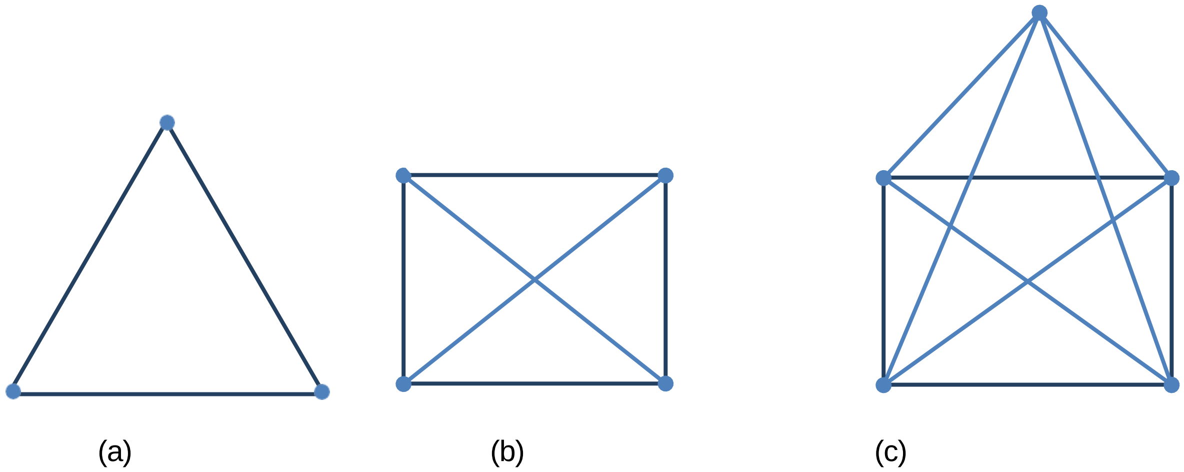


Fig 13.8

13.2.4 Complete Graph

A simple graph is said to be complete graph if there exists an edge between every pair of vertices. A complete graph having n vertices is denoted by K_n.



Complete graphs (a) K_3 (b) K_4 (c) K_5

Fig 13.9

13.2.5 Regular Graph

A simple graph is said to be regular if the degree of each vertex is the same. If the degree of each vertex of a regular graph equals r, the graph is said to be r –regular.

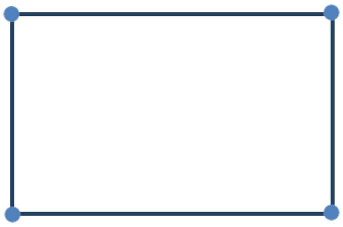


Fig 13.10

13.2.6 Bipartite Graph

A graph $G(V, E)$ is said to be a bipartite graph if there exists a partition of the set $V(G)$ into two disjoint sets $V_1(G)$ and $V_2(G)$ such that each edge of the graph has its one end in $V_1(G)$ and the other end in $V_2(G)$.

The graph shown in fig 13.9 here is a bipartite graph. In this graph, the set $V(G)$ is partitioned into two disjoint sets $V_1(G)$ and $V_2(G)$, where

$$V_1(G) = \{v_1, v_2\} \quad and \quad V_2(G) = \{v_3, v_4, v_5\}$$

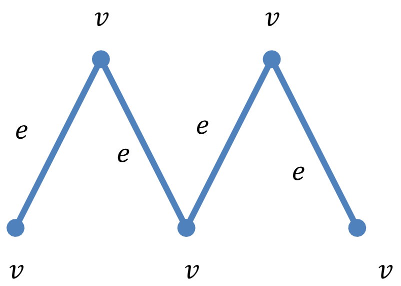


Fig 13.11

A bipartite graph is called complete bipartite if each vertex of $V_1(G)$ is joined to each vertex of $V_2(G)$ through an edge as per shown in the figure given below

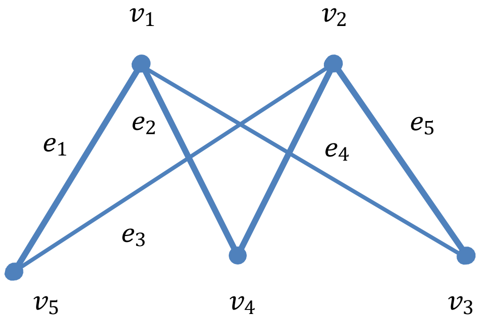


Fig. 13.12

13.2.7 Platonic Graph

These are of special interest among the regular graphs so-called Platonic graph, the graphs formed by the vertices and edges of the five regular (Platonic) solids – the tetrahedron, octahedron cube, dodecahedron and icosahedron. The graphs are shown in the following figures

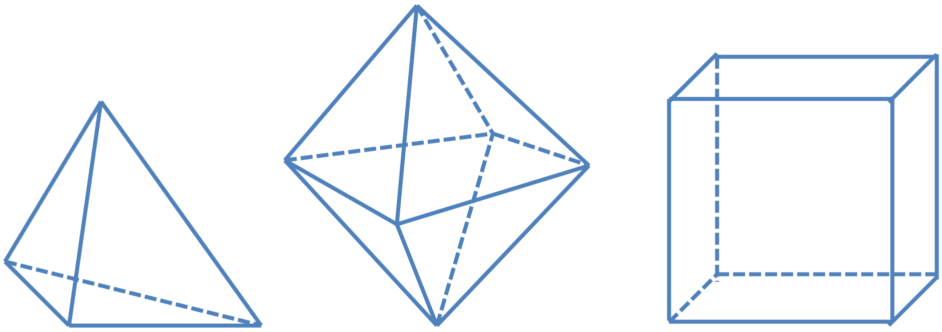


Fig. 13.13

13.2.8 Weighted Graph

A graph is called a weighted graph if all its edges have been assigned some positive real numbers (weights) to provide some additional information. For example, if we construct a graph of different places in a metropolitan city, then the traffic density (average number of vehicles per minute moving out through the route) of the different routes can be shown by defining weights to different edges

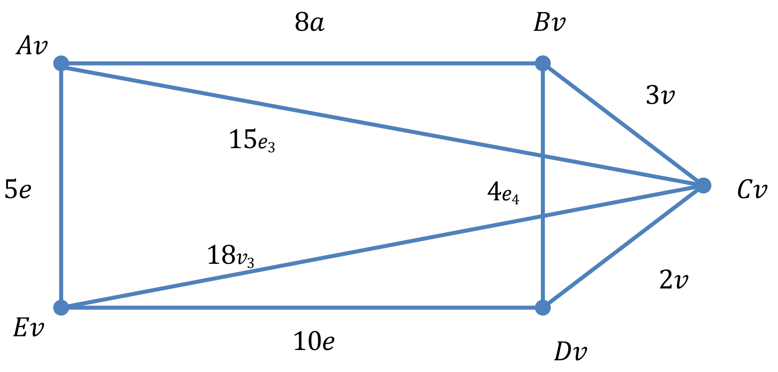


Fig 13.14

Walk

Walk in a graph is an alternating sequence of vertices and edges, starting and ending with vertices. The starting and ending vertices are called terminal vertices of the walk. In a walk the edges are not repeated but vertices may be repeated.

A walk with different terminal vertices is called open walk.

A walk with same terminal vertex is called closed walk.

Path

A path is an open walk in which no vertex repeated (or appears more than once). The length of a path is the number of edges in path. Two vertices are said to be reachable from each other it there exists a path between them. In fig 13.14 Vertex A is reachable from D.

13.3 Connected graphs & its components

A graph is said to be connected if every two vertices are reachable from each other. Otherwise the graph is said to be disconnected

Every disconnected graph can be partitioned into connected subgraphs and these connected subgraphs are called components.

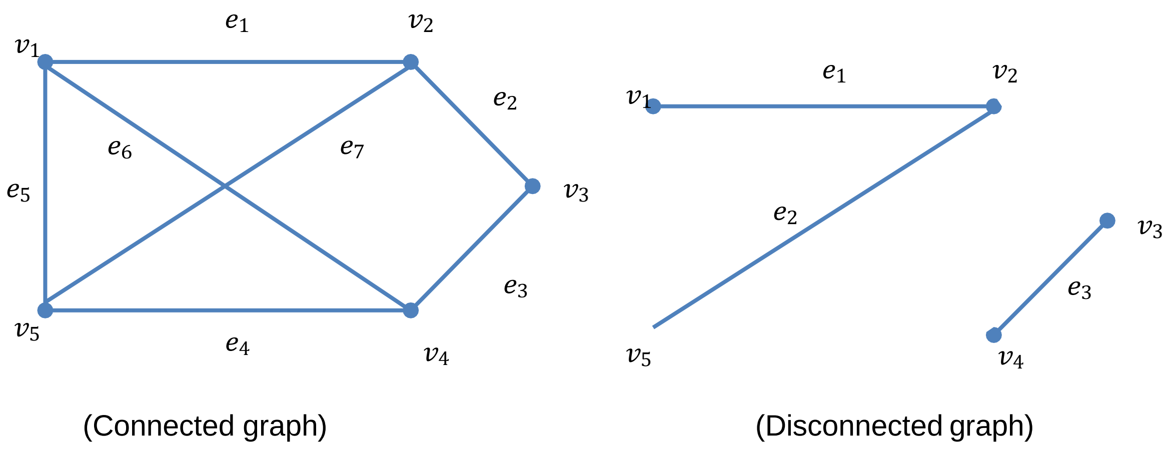


Fig. 13.15

Theorem13.2

A graph G is disconnected if and only if its vertex set V is partitioned into two non-empty, disjoint subsets V_1 and V_2 such that there exists no edge in G whose one end vertex is in V_1 and the other is in V_2

Proof: Let the graph G be disconnected. Every disconnected graph contains some components. Let V_1 and V_2 be the two components. Then $V_1 \cup V_2 = V$ and $V_1 \cap V_2 = \emptyset$. Thus, $\{V_1, V_2\}$ forms a partition of V such that there exists no edge in G whose one end vertex is in V_1 and other is in V_2.

Let the vertex set V be partitioned into two non-empty, disjoint subsets V_1 and V_2 such that there exists no edge in G whose one end vertex is in V_1 and the other is in V_2. Since the vertices of the set V_1 are not connected by the vertices of the set V_2 , the graph is disconnected.

This proves the theorem.

Theorem13.3

If a graph G (connected or disconnected) has exactly two vertices of odd degree, there must be path joining the two vertices.

Proof:

Let the graph G be connected. Then there exists a path between each pair of vertices. Thus, if the graph G is connected and has exactly two vertices of odd degree, there will be a path between the two vertices.

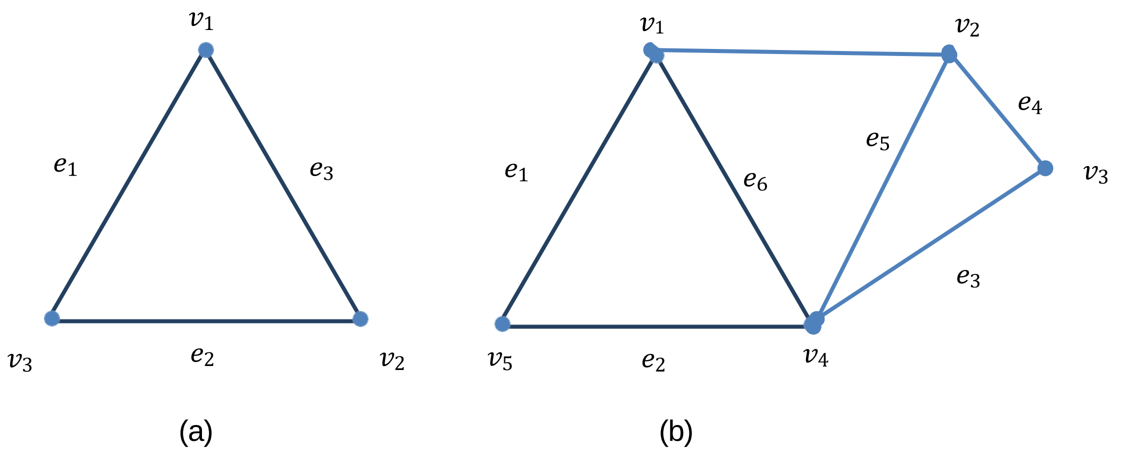
Let the graph G be disconnected and let it have exactly two vertices of odd degree. We know that every disconnected graph contains components, and a component is a sub-graph of the graph G and hence forms a graph itself. Since the number of vertices having odd degree is two, the two vertices must be a part of one component. As every component is a connected graph, there exists a path between the two vertices.

This proves the theorem

13.4 Euler Graph

The concept of Euler graph came from the question in what type of graph G is it possible to find a closed walk passing through every edge of G, which Euler has described in his paper dealing with the $K\ddot{o}nigsberg$ bridge problem.

A closed walk that contains all edges of a graph is called an Euler line, and a graph that contains an Euler line is called an Euler graph. We know that a walk traces each edge exactly once and it is connected. Since an Euler graph contains all edges of a graph, it is always connected and hence Euler graphs do not have isolated vertices. An open walk that includes all edges of a graph is called a unicursal line or an open Euler line. A connected graph that has a unicursal line is called a unicursal graph



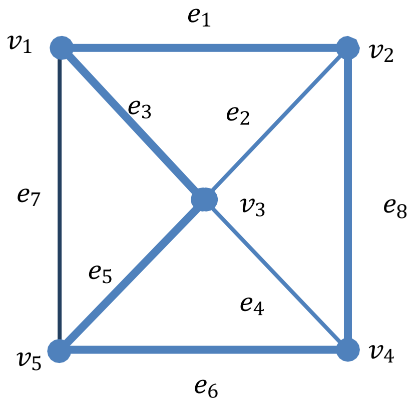
Euler and unicursal graphs (a) Euler graph (b) Unicursal graphs

Fig. 13.16

From the definition of a unicersal line, it is clear that by adding an edge between the initial and final vertices of a unicursal line, we will get an Euler line. Thus, a connected graph is unicursal if and only if it has exactly two vertices of odd degree.

13.5 Hamiltonian path and Circuit

A Hamiltonian circuit is a closed walk that traverses each vertex of a graph G exactly once except the starting vertex at which the walk also terminates. In the graph shown below is a Hamiltonian circuit (shown in bold lines) $(v_1\ e_3\ v_3\ e_5\ v_5\ e_6\ v_4\ e_8\ v_2\ e_1\ v_1)$



(Hamiltonian circuit)

Fig. 13.17

Not every circuit in a graph is a Hamiltonian circuit. A circuit in a graph is said to be Hamiltonian if it includes all vertices of the graph. Hence, a Hamiltonian circuit in a graph of n vertices contains exactly n vertices and n edges.

It should be remembered that every connected graph need not contain a Hamiltonian circuit. There is no criterion or condition through which we can determine the existence of a Hamiltonian circuit in a graph.

A path obtained by removing one edge from a Hamiltonian circuit is called a Hamiltonian path. Thus, a Hamiltonian path contains all vertices of the graph and the length of the Hamiltonian path in a graph of n vertices is $n - 1$. Every graph that has a Hamiltonian circuit also has a Hamiltonian path but its converse is not true.

Self-loops and parallel edges cannot be included in a Hamiltonian circuit(path) as a Hamiltonian circuit (path) traverses each vertex exactly once. Therefore, in searching for the existence of a Hamiltonian circuit (path) in a given graph, the graph can be made a simple graph by removing all self-loops and parallel edges. Each member of a family of complete graphs having three or more vertices contains a Hamiltonian circuit.

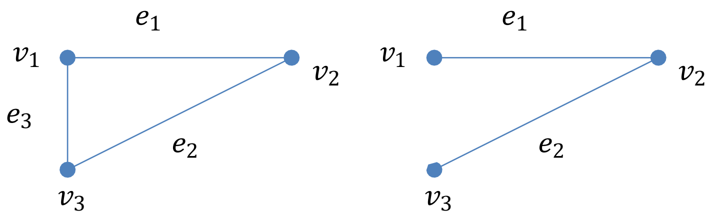
A given graph may have more than one Hamiltonian circuit. As regards the presence of edge disjoint Hamiltonian circuit, the determination of the exact number of edge disjoint Hamiltonian circuits in a graph is also an unsolved problem. However, in a complete graph with odd number of vertices, the number of edge disjoint Hamiltonian circuits can be calculated.

13.6 Graph Coloring & Chromatic number

Colouring of a graph is the problem associated with the assignment of colours to the elements (vertices, edges, regions) of the graph such that no two adjacent elements have the same colour. The colouring of vertices so that no two vertices have the same colour is called vertex colouring; edge colouring and region colouring are similarly defined. Vertex colouring is the initial point of colouring of graphs, and other colouring problems can be transformed to vertex colouring.

Chromatic Number

Assigning colours to all vertices of a graph such that no two vertices have the same colour is called proper colouring, and the graph whose vertices are coloured in such a way is called a properly coloured graph. The minimum number of colours required to colour a graph properly is called the chromatic number of the graph, denoted by $k(G)$. For example, the chromatic number of the graph given in fig 13.18 is three as minimum three colours are required to colour the graph properly, but the chromatic number of the graph in given figure 1.16 is two as the graph can be properly coloured with only two colours. The vertices v_1 and v_3 can be assigned the same colour.



(a) 3 – chromatic graph (b) 2 – chromatic graph

Fig. 13.18

In discussion involving colouring of graphs, a connected graph is usually considered because the colouring of one component of the graph has no effect on the colouring of the other components. Self-loops can be discarded and parallel edges may be replaced by a single edge for colouring of graphs, as colouring of vertices is not affected by this process. If a graph is a null graph, then its chromatic number is one, and if the graph is a complete graph of n vertices, then its chromatic number is n.

Problems for Exercise

1. Draw a diagram for each of the following graphs $G = G(V, E)$
   a. $V = \{A, B, C, D\},\ E = [\{A, B\}, \{D, A\}, \{C, A\}, \{C, D\}]$
   b. $V = \{a, b, c, d, e, f\}, E = [\{a, d\}, \{a, f\}, \{b, c\}, \{b, f\}, \{c, e\}]$
2. Draw a diagram of each of the following multigraphs $G(V, E)\ where\ V = \{P_1, P_2, P_3, P_4, P_5\}$ and
   a. $E = [\{P_2, P_4\}, \{P_2, P_3\}, \{P_3, P_5\}, \{P_5, P_4\}]$
   b. $E = [\{P_1, P_1\}, \{P_2, P_3\}, \{P_2, P_4\}, \{P_3, P_2\}, \{P_4, P_1\}, \{P_5, P_4\}]$
3. Determine whether or not each of the following multigraphs $G(V, E)$ is a graph where $V = \{A, B, C, D\}$ and
   a. $E = [\{A, B\},\ \{A, C\}, \{A, D\},\ \{B, C\},\ \{C, D\}]$
   b. $E = [\{A, B\},\ \{B, B\},\ \{A, D\}]$
   c. $E = [\{A, B\},\ \{C, D\},\ \{A, B\},\ \{B, D\}]$
   d. $E = [\{A, B\},\ \{B, C\},\ \{C, B\},\ \{B, B\}]$
4. Suppose $G = G(V, E)$ has five vertices. Find the maximum number m of edges in E if:
   a. G is a graph,
   b. G is multigraph
5. A graph has 12 edges, two vertices of degree 3, two vertices of degree 4, and other vertices of degree 5. Find the number of vertices in the graph.
6. Let e denote the number of edges in a complete bipartite graph $K\_(m, n)$.
7. Consider a graph (multigraph) G. Define a closed path and a cycle in G.
8. Consider the graph G where

   $$V(G) = \{A, B, C, D\} \quad and \quad E(G) = [\{A, B\}, \{B, C\}, \{B, D\}, \{C, D\}]$$

   Find the degree of each vertex in G.
9. Find the connected components of G where $V(G) = \{A, B, C, X, Y, Z\}$ and
   a. $E(G) = [\{A, X\}, \{C, X\}]$
   b. $E(G) = [\{A, Y\}, \{B, C\}, \{Z, Y\}, \{X, Z\}]$.
10. Find the connected components of G where $V(G) = \{A, B, C, P, Q\}$ and
   a. $E(G) = [\{A, C\}, \{B, Q\}, \{P, C\}, \{Q, A\}]$
   b. $E(G) = \emptyset$, the empty set.

Block V: Unit II Trees

Learning Objectives
After completing this unit, the learner shall be able to:

- Define a Tree;
- Explain the properties of a Tree;
- Define various types of Trees;
- Define Binary Trees;
- Find the preorder, inorder and post order traversal of a given Tree;
- Explain Binary Search Tree;
- Create a Binary Search Tree;
- Use a binary tree to sort the given list of numbers;

The word tree suggests branching out from a root and never completing a cycle. Trees form one of the most widely used subclass of graphs. This is due to the fact that many of the applications of graph theory, directly or indirectly, involve trees. Trees occur in situations where many elements are to be organized into some short of hierarchy. In computer science, trees are useful in organizing and storing data in a database.

Here in this session the discussion will around the basic terminology of tree, their types, and properties, searching trees and traversing.

14.1 Definition

A tree is a connected acyclic graph i.e. a connected graph having no cycle. Its edges are called branches. Following are examples of trees with at most five vertices.

A tree with only one vertex is called a $\boldsymbol{trivial\ tree}$ otherwise T is a $\boldsymbol{nontrivial\ tree.}$

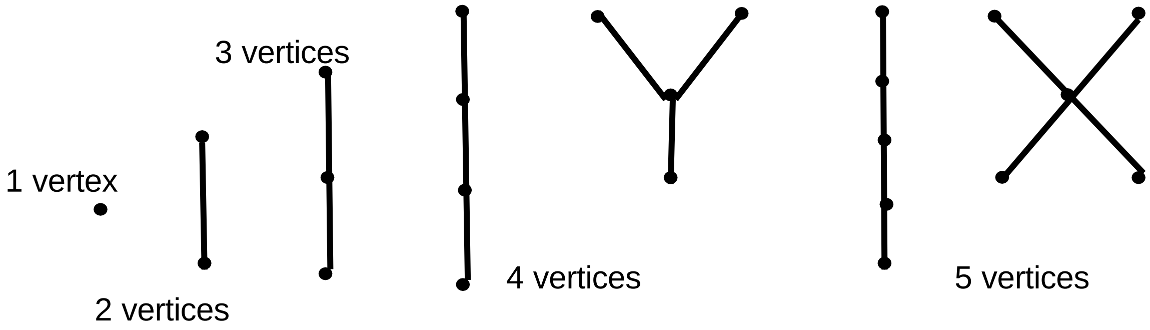


Fig 14.1

14.2 Properties of Trees

Characterization

Trees have many equivalent characterizations, any of which could be taken as the definition. Such characterization are useful because we need only verify that a graph satisfies any one of them to prove that it is a tree, after which we can use all other properties. A few simple and important theorems on the general properties of trees are given below

Theorem 14.1: Prove that there is one and only one path between every pair of vertices in a tree T.

Proof: We know that between every pair of vertices, there exists a path connecting them because tree is a connected graph.

Let us consider for now that there exists more than one path between a pair of vertices. These two paths will create a circuit and it is a contradiction to the definition of a tree.

Hence, there is one and only one path between every pair of vertices in a tree T.

Theorem 14.2 Prove that there are $n-1$ edges in a tree with n vertices.

Proof: we shall use the principle of mathematical induction to prove the theorem. Let us consider a tree with only one vertex only, that is, $n = 1$. A tree with one vertex has no edge. Similarly, a tree with two vertices, that is, for $n = 2$, has only one edge. Hence, the statement is true for $n = 1$ and $n = 2$.

Let the statement be true for $n = k$; that is, a tree with k vertices has $k - 1$ edges. We shall prove the statement for $n = k + 1$. If we insert a vertex in a tree of k vertices, then the vertex must be joined by any of the k vertices through one edge only, because a tree is a connected graph with no circuit. Thus, increase in the number of vertices by one, the number of edges also increased by one. $Total\ number\ of\ vertices = k + 1$, $Total\ number\ of\ edges = k - 1 + 1 = k$. This implies that the statement is true for $n = k + 1$, and hence, it is true for all natural numbers.

Theorem 14.3 For any positive integer n, if G is a connected graph with n vertices and $n-1$ edges, then G is tree.

Proof: Assume n be a positive integer and Let G is a specific but randomly chosen graph that is connected and has n vertices and $n-1$ edges. It is proved with the theorem discussed before that a tree is a connected graph without cycles or circuits. We have also proved in recent theorem that a tree has $n-1$ edges.

Now, we will prove the converse that if G has no circuit and $n - 1$ edges, then G is connected. We split G into G_1, G_2, ... G_k . Each G is connected and it has no circuit, since G has no circuits. Hence, each G_k is a tree. Now, edge $e_1 = n_1 - 1$ and

$$\sum_{i=1}^{k} e_i = \sum_{i=1}^{k} (n_i - 1) = n - k \ or \ e = n - k$$

Therefore, $k = 1$ or G has only one component, Hence, G is a tree.

Theorem 14.4 A graph is a tree if and only if it is insignificantly connected

Proof: Assume that G is a graph with n vertices and let G is a tree with n vertices. We know that if a tree with has n number of vertices then it has $(n - 1)$ number of edges. If an edge is removed from G, then it has $(n-2)$ edges and G becomes disconnected.

Therefore G is an insignificantly connected graph.

On the other hand, let G be a insignificantly connected graph with n number of vertices. The number of edges of $G \geq n - 1$ as G is a connected graph. If possible, assume that G is not a tree. Then there exists a circuit in G and G is still connected if an edge of this circuit is deleted from G. This controverts hypothesis that G is a insignificantly connected graph. Therefore, G is a tree.

On the basis of the results from earlier theorems following are different but similar definitions of tree. A graph having n vertices is called a tree if

1. G is connected and has no circuits
2. G is connected and has $n-1$ edges
3. G is a acyclic and has $n-1$ edges
4. There exists a unique path between every pair of vertices in G
5. G is a insignificantly connected graph.

Thus, an undirected simple graph T is a tree that satisfies any of the following corresponding conditions:

- T is connected and has no circuits.
- T has no circuits, and a circuit is formed if any edge is added to T.
- T becomes disconnected if any edge is removed from T.

- Every two vertices of T are connected by a unique path.

14.3 Types of Trees (Rooted, Binary)

14.3.1 Rooted Trees

A tree in which a specific vertex can be distinguished from other vertices of a tree then such a tree is called as rooted trees. It is not similar to the natural trees in which they have their roots at the bottom whereas in the graph theory rooted trees are usually drawn with their roots at the top. Initially, we keep the root at the top. Beneath the root and on the same level, we place the vertices that can be extended from the root on a simple path of length 1. Same process is done with these vertices also on the same level but simple path of length 2. We repeat it until the entire tree is drawn. Definitions of some of the related terms can be given as.

1. The number of edges laterally on unique path amongst vertex and the root can be called as level of vertex. The level of the root can be called as 0. The vertices immediately under the root are said to be in level 1 and so on.
2. The maximum level to any vertex of the tree is called as height of that rooted tree. The deepness of a vertex v in a tree is the distance of the path from the root to v.
3. Given any interior vertex v of a specific rooted tree, the children of v are all those vertices that are nearby to v and are one level further away from the root as compared to v. If w is a child of v, the v is called the parent of w, and two vertices that are both children of the same ancestor are called **siblings.**
4. If the vertex has no children, then that vertex is called as leaf (or a terminal vertex). If any vertex has either one or two children, then that vertex is called an internal vertex.
5. The offspring of the vertex u is the set consisting of all the children of u together with ancestry of those children. Given vertices v and w, if v lies on the unique path between w and the root, then v is an ancestor of w and w is a offspring of v.

These terms are illustrated in following figure 14.2.

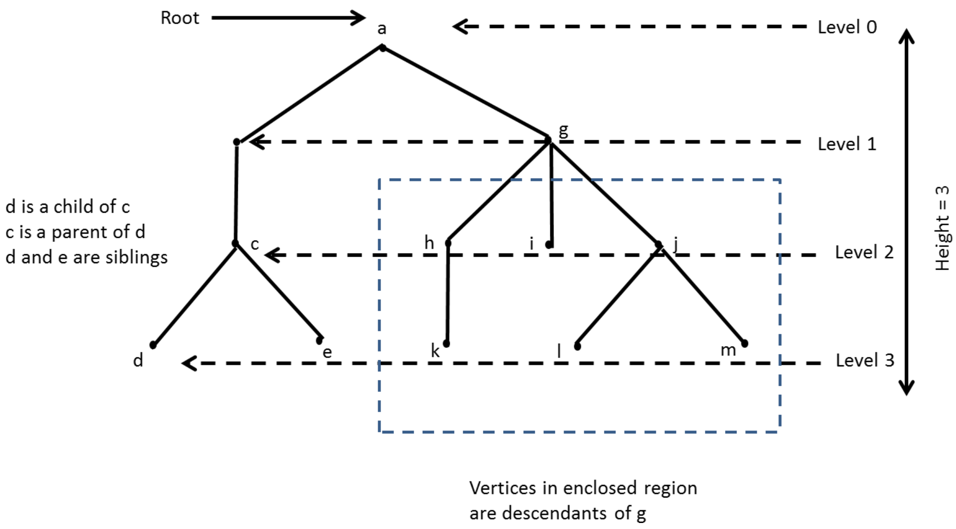


Fig 14.2

Example 14.1

Consider the rooted tree in following figure

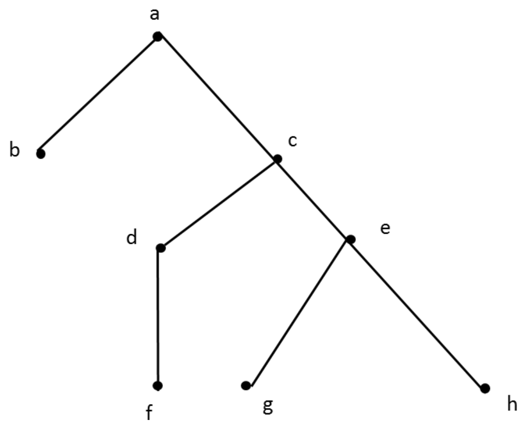


Fig 14.3

(a) What is the root of T?

(b) Find the leaves and the internal vertices of T.

(c) What are the levels of c and e.

(d) Find the children of c and e.

(e) Find the descendants of the vertices a and c.

Solution:

(a) Vertex a is distinguished as the only vertex located at the top of the tree. Therefore a is the root.

(b) The leaves are those vertices that have no children. These are b, f, g and h. The internal vertices are c, d and e.

(c) The levels of c and e are 1 and 2 respectively.

(d) The children of c are d and e and of e are g and h.

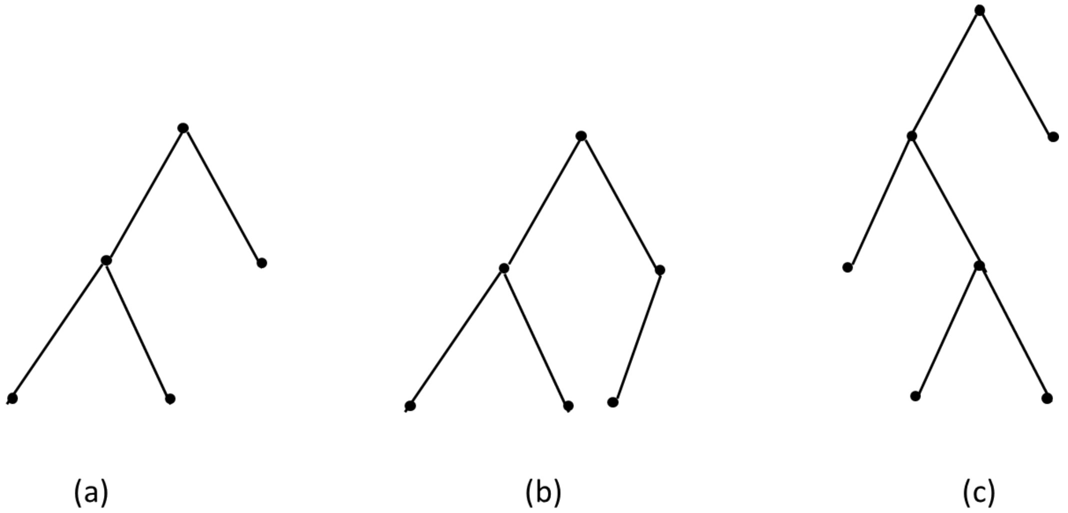
(e) The descendants of a are b, c, d, e, f, g, h. The descendants of c are d, e, f, g, h.

Rooted trees with the property that all of their internal vertices have the same number of children are used in many problems involving searching, sorting and coding.

14.3.2 Binary Trees

A tree in which a vertex can have at most two children then such a tree is called as binary tree. In the area computer science specifically dealing with algorithms, a binary tree having vertices of zero or two children is very useful. Thus, a specific class of binary can defined as follows:

A full binary tree has exactly one vertex of degree two and other vertices of degree three or one. A complete binary tree is completely filled (every vertex has left as well as right child) at every level, except possibly the last, and all nodes are as far left as possible.



(a) Full and Complete Binary tree

(b) Complete but not full binary tree

(c) Full but not complete binary tree

Fig 14.4

From the definition of a binary tree, the following properties can be observed.

Theorem 14.5 The total number of vertices n in a full binary tree is always odd.

Proof: Let us consider that n be the vertices of a binary tree. As per the definition the degree of one vertex is two, and and on the other hand the remaining $n - 1$ vertices is either one or three. We know that there are even numbers of vertices having an odd degree in a graph. Since, $n - 1$ is an even number and therefore n is an odd number.

Theorem 14.6 The total number of pendent vertices in a fully binary tree with n vertices is $\frac{n+1}{2}$.

Proof: Let p be the number of pendent vertices. The degree of one vertex in a full binary tree is two and that for the remaining vertices is three. We know that as per the earlier theorem a tree with n vertices contains $n - 1$ edges. Thus, using the equation

$$\sum_{i=1}^{n} \deg(v_i) = 2e$$

We have $1.p + 2.1 + 3.(n - p - 1) = 2(n - 1)$

$$\Rightarrow p + 2 + 3n - 3p - 3 = 2n - 2$$

$$\Rightarrow -2p = -n - 1$$

$$\Rightarrow p = \frac{n+1}{2}$$

14.4 Tree Traversing

A traversal of a tree is a process to traverse a tree in a systematic way so that each vertex is visited exactly once. Three commonly used traversals are preorder, postorder and inorder. We describe here these three process that may be used to traverse a binary tree.

Preorder Traversal

The recursive definition of preorder traversal of a binary tree is given as follows.

(i) Initially start with the root
(ii) Navigate to the left subtree in preorder
(iii) Navigate to the right subtree in preorder

Postorder Traversal

The recursive definition of postorder traversal of a binary tree is given as follows

(i) Navigate to the left subtree in postorder
(ii) Navigate to the right subtree in postorder
(iii) Finally end up at the root

Inorder Traversal

The recursive definition of inorder traversal of a binary tree is given as follows

(i) Navigate in inorder to the left subtree
(ii) Visit through the root
(iii) Navigate in inorder to the right subtree

Example 14.3 Find the preorder, inorder and post order traversal of the following tree T

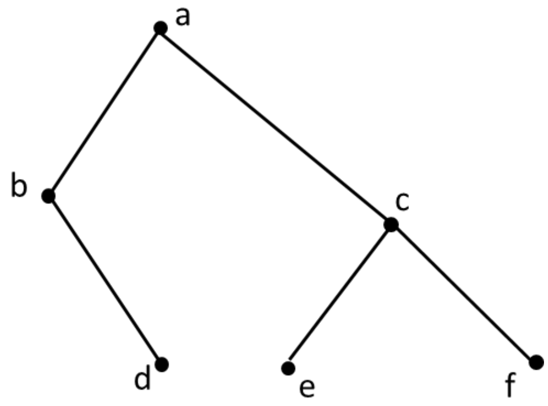


Fig 14.5

Preorder Traversal

1. First visit the root $\boldsymbol{a}$
2. Traverse the left subtree and visit the root $\boldsymbol{b}$**.** Now traverse the left subtree with b as the root; it is empty. The traverse the right subtree of $\boldsymbol{b}$ and visit the root $\boldsymbol{d}$**.** There is no subtree of $\boldsymbol{d}$**.**
3. Then back to the root $\boldsymbol{a}$**,** traverse the right subtree and visit the root $\boldsymbol{c}$**.** Traverse the left subtree with c as the root and visit the root e. The subtree of e is empty. Then traverse the right subtree of $\boldsymbol{c}$ and visit the root f. The root f has no subtree.

All the vertices have been covered. Hence output is $\boldsymbol{a \;\; b \;\; d \;\; c \;\; e \;\; f}$

Inorder Traversal

1. First traverse the left subtree with root $\boldsymbol{a}$ in inorder. Again traverse the left subtree with root b in inorder. It is empty, so visit b. Now traverse right subtree of b. Then traverse left subtree of d which is empty, visit d. The right subtree of d is empty.
2. Back to $\boldsymbol{a}$ and visit $\boldsymbol{a}$
3. Traverse the right subtree of $\boldsymbol{a}$ in inorder. Again traverse the left subtree of c inorder. Then the left subtree of e in inorder; it is empty. So visit e. Traverse the right subtree of c. There is no subtree of f, so visit f, so visit f. Back c and visit c.

All the vertices have been covered. Hence output is $\boldsymbol{b\ \ d\ \ a\ \ e\ \ f\ \ c}$**.**

Postorder Traversal

1. Traverse the left subtree with root $\boldsymbol{a}$ in post order.
2. Traverse the left subtree with root b in postorder. The left subtree of b is empty. Traverse the right subtree. Since d has no subtree, visit d. Then back to b and visit b.

   Then back to $\boldsymbol{a}$ and traverse the right subtree of **a.** Traverse the left subtree with c as root. Since e has no subtree, visit e. Traverse the right subtree with c as root. Since f has no subtree, visit f. Then back to c and visit c.

3. Back to the root $\boldsymbol{a}$ and visit $\boldsymbol{a}$**.**

All the vertices have been covered. Hence output is $\boldsymbol{d\ b\ e\ f\ c\ a}$

Given an order of traversal of a tree it is possible to construct a tree. For example consider the following order:

Inorder = $\boldsymbol{d\ \ b\ \ e\ \ a\ \ c}$

We can construct the binary tree shown below using this order of traversal:

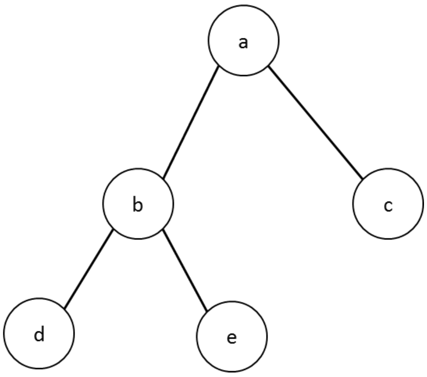


Fig 14.6

14.5 Binary Search Tree

A binary search tree is basically a binary tree, and therefore it can be traversed in preorder, postorder, and inorder. If we traverse a binary search tree inorder and print the identifiers contained in the vertices of the tree, we get a sorted list of identifiers in the ascending order.

Binary trees are used extensively in computer science to store elements from an ordered set such as a set of numbers or a set of strings. Suppose we have a set of strings and numbers. We call them as keys. We are interested in two of the many operations that can be performed on this set.

1. Ordering (or sorting) the set.
2. Searching the ordered set to locate a certain key and, in the event of not finding the key in the set, adding it at the right position so that the ordering of the set is maintained.

Definition

A binary tree T in which data are linked with the vertices is called as binary search tree. The data are organized so that, for each vertex v in T, each data item under the left subtree of v is less than that of data item in v itself and each data item under the right subtree of v is greater than that of data item in v itself. Therefore, A binary search tree for a finite set S is a labled binary tree where each vertex v is associated with an element $l(v) \in S$ such that

1. for each vertex u under the left subtree of v, $l(u) < l(v)$,
2. for each vertex u under the right subtree of v, $l(u) > l(v)$, and
3. for each element $a \in S$, there is a unique vertex v such that $l(v) = a$

The binary tree T is a binary search tree since each vertex in T is greater than each number under its left subtree and is less than each number under its right subtree.

Creating a Binary Search Tree

In order to form a binary tree for a set of items a recursive procedure can be adopted as follows.

At the start, we form a vertex and keep the first item in the list in this vertex and assign it as the key of the root of that tree. In order to add a new item to the tree, first we will compare it with the keys of vertices already in the tree, starting from the root and traversing to the left, if the item is less than that of the key at respective vertex and has a left child, or moving to the right if the item is greater than that of the key at the respective vertex and vertex has a right child. When the item is found less than the key at respective vertex with no child then we create a new child to the left of that vertex. Similarly, when the item is found greater than the key at respective vertex having no right child, then we create a new right child and assign the item to it. In this manner, all the items in the list can be stored in the tree and thus a search tree is created.

Example 14.2 Form a binary search tree for the data $16, 24, 7, 5, 8, 20, 40, 3$ in the given order.

Solution: We start by choosing the number 16 to be the root of the tree. Since the number 24 is greater than 16, add a right child to the root and assign 24 to it. Similarly, next element in the list is 7 and again we need to start from the root and equate it with 16. Here 7 is less than the 16, add a left child to the root and assign 7 to it. Further we have 5 in the list which is less than 16 and 7 as well, then we move further down to the left child of 7 and assign the vertex to 5. Similar procedure is followed for remaining numbers in the list. The resultant binary search tree is given as below.

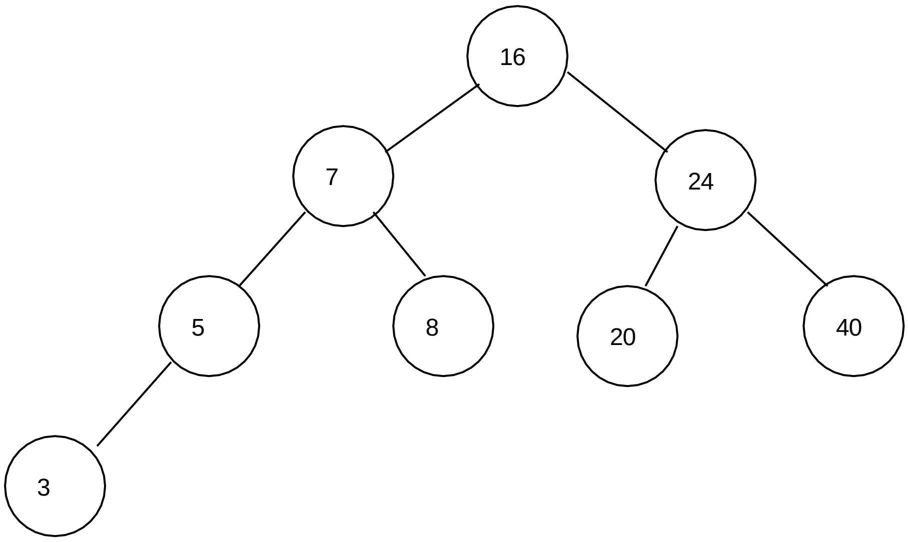


Fig 14.7

Problems for Exercise:

1. Define a rooted tree T, with an example and show how such a tree may be viewed as directed graph.
2. Define the terms leaves and branches in a rooted tree T and give examples.
3. Let v be a vertex in a rooted tree T with root$R$. Define the level or depth of v.
4. Let T be a rooted tree with root R
    a. Explain what it means for a vertex to precedes or follow another vertex in T.
    b. Define the notion of parent, children, and siblings in T
5. A tree is a useful device for enumerating all the logical possibilities of sequence of event can occur in a finite number of ways. Use a tree to identify all the possible results of a tennis match between two players, Mahesh and Ajay, where the winner is the first player who wins two sets in a row or the first player who wins a total of three sets.
6. Define a binary tree T and give an example.
7. Define right and left successors of a root node of a binary tree. Explain it with an example.
8. Draw all possible non-similar binary trees with 3 nodes.
9. A binary tree T has 9 nodes. The inorder and preorder traversals of T yield the following sequences of nodes:
    a. Inorder: $E\ \ A\ \ C\ \ K\ \ F\ \ H\ \ D\ \ B\ \ G$
    b. Preorder: $F\ \ A\ \ E\ \ K\ \ C\ \ D\ \ H\ \ G\ \ B$
10. Use a binary tree to sort the following list of numbers

$$15, 7, 24, 11, 27, 13, 18, 19, 9$$

Block V: Unit III Finite Automata

Learning Objectives
After completing this unit, the learner shall be able to:

- Explain the basic concepts of Automation theory;
- Find a phrase structure grammar to generate the set;
- Explain Chomsky Hierarchy;
- Define Deterministic Finite Automaton (DFA);
- Design a DFA that accepts a given string;
- Define Non-deterministic Finite Automaton (NFA);
- Differentiate between Deterministic Finite Automaton and Non-Deterministic Finite Automaton;
- Reduce the given DFA into minimal state DFA;
- Design a Mealy machine;
- Design a Moore machine;

Words in a language can be combined in various ways. The grammar of a language tells us whether a combination of words is a valid sentence. The syntax of a natural language (body of words and methods of combining words used and understood by considerable community), that is, a spoken language, such as English, Hindi, Bengali or Tamil is extremely complicated. Research in the automatic translation of one language to another has led to the concept of formal language which is specified by a well-defined set of rules of syntax. We will describe the sentences of a formal language using a grammar which is an algebraic system describing the process by which instances of a language can be constructed.

Formal languages are different from natural languages in that formal languages are designed for specific applications. They play an important role in programming languages in computer science.

15.1 Basic concepts of Automation theory

An automation is abstract model of a computer, which accepts some input (a string), produces output (yes / no or a string), and may have internal storage (usually a stack or tape). Three fundamental ideas are the major themes of this session: language, grammars and automata.

15.1.1 Alphabet & Words

An alphabet is a finite and non-empty set of symbols. The symbol $\sum$ is used to denote an alphabet. The elements of an alphabet are known as letters. For example

$$\sum = \{a, b, c\} \; or \; \{0, 1\}$$

A word (string) is a finite sequence of the symbols of an alphabet. Let $\sum = \{a, b, c\}$. Then $abbc,$ $accb,$ and $bbac$ are words over the alphabet $\sum$. The length of a word w, denoted by $|w|$, is the number of symbols in the word. Let $w = abbc$ be a word over the alphabet $\sum = \{a, b, c\}$. Then $|w| = 4$. A word with zero occurrences of symbols ($|w| = 0$) is called an empty word, denoted by λ.

The concatenation of two words u and v is the word uv obtained by juxtaposing u and v. If $u = abbc$ and $v = baac$, then $uv = abbcbaac$.

The reverse of a word w, denoted by w^{-1} or w^R, is the word having all letters of the word w in the reverse order. If $w = acbb$, then $w^{-1} = bbca$.

If $\sum$ is an alphabet, then $\sum*$ is used to denote the set of all possible words generated over the alphabet $\sum$. The operator $*$ is called the Kleeny closure. The set $\sum*$ always contains the empty word. If we exclude the empty word from $\sum*$, then we have a set of non-empty words over the alphabet $\sum$ denoted by $\sum^+$. Thus,

$$\sum^+ = \sum* - \{\lambda\} \quad or \quad \sum* = \sum^+ \cup \{\lambda\}$$

The symbol $\sum k$ is used to denote the set of words of length k over the alphabet $\sum$. If $\sum = \{a, b\}$, then $\sum^2 = \{aa, ab, ba, bb\}$.

Using the definition of $\sum k$, we can define $\sum*$ as follows:

$$\sum* = \sum^0 \cup \sum^1 \cup \sum^2 \ldots\ldots \cup \sum^k \cup \sum^{k+1} \cup \ldots\ldots\ldots$$

15.1.2 Language

A language over an alphabet $\sum$ is a subset of $\sum*$. Let $\sum = \{a, b\}$. Then the set $\{a, aa, aaa\}$ is a language over $\sum$. This language has a finite number of words; hence, it is a finite language.

Example 15.1 Let $L = \{a^n b^n : n \geq 0\}$. Find the words of the language.

Solution: $L = \{\lambda, ab, aabb, aaabbb, \ldots \ldots\}$. The language L consists of words in which the number of $a's$ is equal to that of $b's$ and all occurrences of $b's$ are followed by the occurrences of $a's$.

15.1.3 Grammars

Grammar is a set of rules to define a valid sentence in any language. Before giving a technical definition of grammar, we shall study, for the sake of simplicity, two sentences in English with a view of formalizing of construction of these sentences. A typical rule of English grammar specify that :

1. A ***sentence*** is made up of a ***noun phrase*** followed by a ***verb phrase***.
2. A ***noun phrase*** is made up of an ***article*** followed by an ***adjective*** followed by a ***noun***, or
3. A ***noun phrase*** is made up of an ***article*** followed by a ***noun***;
4. A ***verb phrase*** is made up of a ***verb*** followed by an ***adverb***
5. A ***verb phrase*** is made of a ***verb***;
6. An article is a, or
7. An article is the;
8. An adjective is large, or
9. An adjective is hungry;
10. A noun is rabbit, or

11. A noun is mathematician;

More concisely, we write this as

$$< Sentence > \rightarrow < Noun\ Phrase >< Verb\ Phrase >$$

$$< Noun\ Phrase > \rightarrow < Article >< adjective >< Noun >$$

$$< Noun\ Phrase > \rightarrow < Article >< Noun >$$

$$< Verb\ Phrase > \rightarrow < Verb >< Adverb >$$

$$< Verb\ Phrase > \rightarrow < Verb >$$

$$< Sentence > \rightarrow < Article >< Noun >< Verb >< Adverb >$$

Now, we associate the actual words,

$$< Article > \rightarrow \ \ a$$

$$< Article > \rightarrow the$$

$$< Noun > \rightarrow boy$$

$$< Noun > \rightarrow dog$$

$$< Verb > \rightarrow runs$$

$$< Verb > \rightarrow walks$$

$$< adverb > \rightarrow quickly$$

$$< adverb > \rightarrow slowly$$

Then some sentences in the language are

$$A\ boy\ runs\ quickly.$$

$$The\ dog\ walks\ slowly.$$

This example illustrates the definition of a general concept in terms of simpler ones. With this background we can give definition of a grammar. This definition is due to Noam Chomsky.

15.1.4 Definition of Grammar

A phrase-structure grammar (or, simply, grammar) G is defined by a 4 – tuple $G = (V_N, V_T, S, P)$ where

(i) V_N is a finite set of non-terminal symbols.

(ii) V_T is a finite set of terminal symbols

(iii) Among all the non-terminals in V_N, there is a special non-terminal $S \in V_N$, called the start symbol.

(iv) P is a finite set where elements are $\alpha \rightarrow \beta$ where α and β are strings on $V_N \cup V_T$, has at least one symbol from V_N . Elements of P are called productions or production rules.

Set of production is the heart of a grammar and language specification.

The nonterminal in V_n are intermediate symbols used to describe the structure of the sentences.

In the above example,

$$V_N = \{< sentence >, < article >, < noun >, < verb >, < adverb >\}$$

The terminals in V_T are symbols used to make up sentences in the language. In the above example,

$$V_T = \{a, the, boy, dog, runs, walks, quickly, slowly\}$$

The starting symbol S is a special nonterminal that begins the generation of any sentence in the language. In the above example, $S =< sentence >$.

The productions P are grammatical rules that specify how sentences in the language can be made up.

Example 15.2 Find a phrase structure grammar to generate the set $\{0^m1^n :$ $m\ \ and\ n\ are\ non\ negative\ integers\}$

Solution: we will give two grammars G_1 and G_2 that generate this set. This will illustrate that two grammars can be generate the same language.

The grammar G_1 has alphabet $V_n = \{S\}$, terminals $V_T\ = \{0, 1\}$, and productions $S \rightarrow 0S,\ S \rightarrow S1,\ and\ S \rightarrow \lambda$. G_1 generates the correct set, since using the first production m times puts m $0s$ at the beginning of the string, and using the second production n times puts n times puts $n\ \ 1s$ at the end of the string. The details of this verification are left to the reader.

The grammarG_2 has alphabet $V_n = \{S, A\}$, terminals $V_T = \{0,\ 1\}$ and productions $S \rightarrow 0S,\ S \rightarrow 1A,\ S \rightarrow 1,\ A \rightarrow 1A,\ A \rightarrow 1,\ S \rightarrow \lambda$. The details that this grammar generates the correct set are left as exercise for the reader.

15.1.5 Types of Phrase structure grammar

Phrase structure grammars can be classified according to the types of productions that are allowed.

We will see the different types of languages defined in this scheme correspond to the classes of the languages that can be recognized using different models of computing machines.

A $\boldsymbol{type\ \ 0}$ grammar has no restrictions on its productions. A $\boldsymbol{type\ \ 1}$ grammar can have productions only of the form $w_1\ \rightarrow\ w_2$ and $w_1\ \rightarrow\ \lambda$. A $\boldsymbol{type\ \ 2}$ grammar can have productions only of the form $w_1 \rightarrow w_2$, where w_1 is a single symbol that is not a terminal symbol. A $\boldsymbol{type\ 3}$

grammar can have productions only of the form $w_1 \rightarrow w_2$ with $w_1 = A$ and $w_2 = aB \ or \ w_2 = a$, where $A \ \ and \ \ B$ are non-terminal symbols and a is a terminal symbol, or with $w_1 = S$ and $w_2 = \lambda$.

15.1.6 Chomsky Hierarchy

From the definitions of different types of grammars we see that every type3 grammar is type 2 grammars, every type 2 grammar is a type 1 grammar, and every type 1 grammar is a type 0 grammar. Type 2 grammar are called $\boldsymbol{context \ free \ grammars}$ since a non-terminal symbol that is the left side of a production can be replaced in a string whenever it occurs, no matter what else is in the string. A language generated by a type 2 grammar is called $\boldsymbol{context \ free \ language}$**.** When there is a production of the form $lw_1r \rightarrow lw_2r$ (but not of the form $w_1 \rightarrow w_2$), the grammar is called type 1 or $\boldsymbol{context \ sensitive}$ since $w_1$can be replaced by w_2 only when it is surrounded by a regular grammar called $\boldsymbol{regular}$**.**

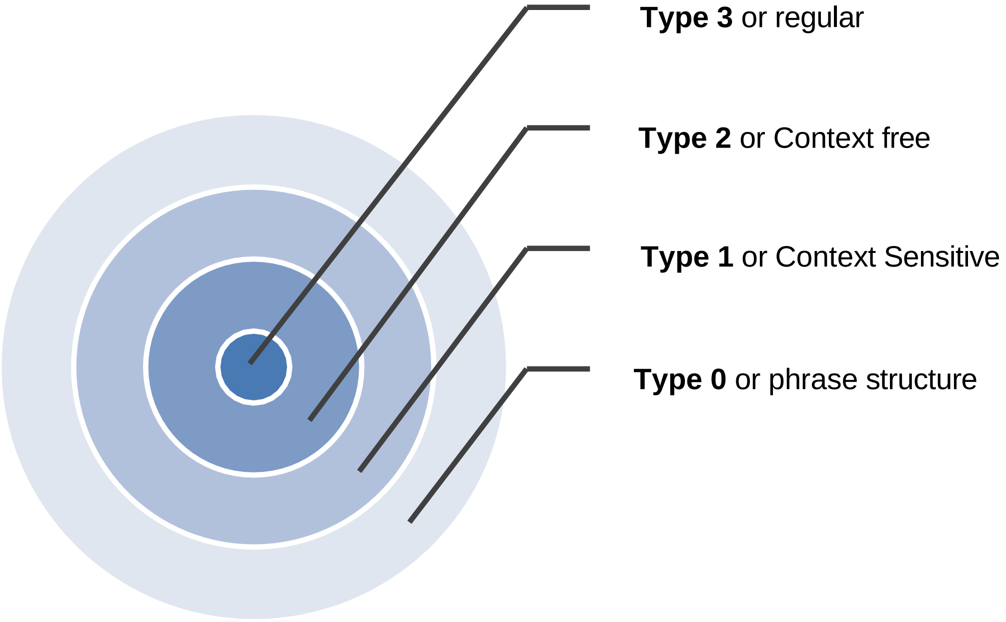

Type 3 or regular
Type 2 or Context free
Type 1 or Context Sensitive
Type 0 or phrase structure

Fig 15.1

Example 15.3

1. $\{0^m1^n : m, n = 0, 1, 2, \dots\}$ is a regular language, since it can be generated by a regular grammar.
2. $\{0^n1^n : n = 0, 1, 2, \dots\}$ is a context free language, since the productions in this grammar are $S \rightarrow 0S1$ and $S \rightarrow \lambda$. However, it is not a regular language
3. The set $\{0^n1^n2^n : n = 0, 1, 2, \dots\}$ is a context sensitive language, since it can be generated by a type 1 language, but not by any type 2 language.

Table 15.1 The Chomsky Hierarchy

Class of Grammar, G_1	**Grammatical Characterization**	**Machine Characterization**
Type 0	Unrestricted (or phrase structure)	Turing machine
Type 1	Context-sensitive	Linear Bounded Automation
Type 2	Context-free	Pushdown Automation
Type 3	Regular (or right linear)	Finite state machine

15.2 Deterministic Finite State Automata

A deterministic finite automaton (DFA) is a finite state machine where for each pair of state and input symbol, there is one and only one transition to the next state as defined by transition function. Formally, DFA is defined by a five – tuple $M\_D = (Q, \sum, \delta, q_0, F)$, where Q is a finite set of internal states, $\sum$ is a finite set of input symbols, called the alphabet, $\delta: Q \times \sum \rightarrow Q$ is the transition function, $q_0 \in Q$ is the initial state, and $F \subseteq Q$ is the set of final states. Moreover, $L(M_D)$ is the language accepted by the machine M_D.

Transition graphs are used to visualize and represent a finite automaton. A transition graph has the following components:

A state q is represented by a circle

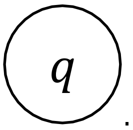


Fig 15.2

The initial (starting) state is given by a circle with an arrow

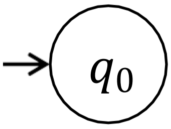


Fig 15.3

The final state is represented by a double circle

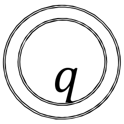


Fig 15.4

The transition between states is denoted by an arc between them with an input symbol and the direction is marked by an arrow

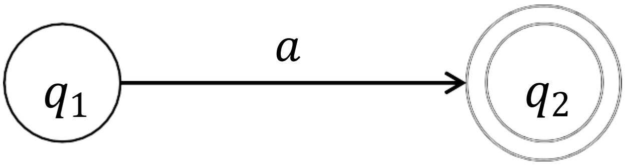


Fig 15.5

A deterministic automaton works as follows: Initially, it is assumed that the automaton is in the initial state q_0. The automaton reads the strings from left to right. Each transition consumes one input symbol and the transition function decides the next state. Finally, at the end of the string, the string is accepted if the automaton is in final state and is otherwise rejected.

A state is said to be a dead state if it is not accepting state and for each input symbol the transition is defined to itself.

Example 15.4 Let $\sum = \{a, b\}$. Design a DFA that accepts all the strings containing exactly one a.

Solution: Since the set of strings contains exactly one a, it can start and terminate with any number of $b's$. Let the starting state be q_0; that is, initially the automaton will be at the state q_0. If the input symbol is b, then the state will remain the same; if the input symbol is a, the automaton will move to the next state q_1. As the condition of exactly one a has been fulfilled, the state q_1 will be the final state. At q_1 if the next input symbol is b, then the state will remain the same as b can appear any number of times after one a. However, if the next input symbol is a, then the automaton will move to the next non-accepting state q_2, and for any other input symbol the state will remain the same.

The DFA $M\_D = (Q, \sum, \delta, q_0, F)$ consists of the following sets:

$Q = \{q_0, q_1, q_2\}$ and $F = \{q_1\}$

The transition function can be defined in the form of the transition table

Table 15.2 Transition table

Present State	**Transition state for input symbols**	
	$\boldsymbol{a}$	$\boldsymbol{b}$
q_0	q_1	q_0
q_1	q_2	q_1
q_2	q_2	q_2

The transition graph is given as,

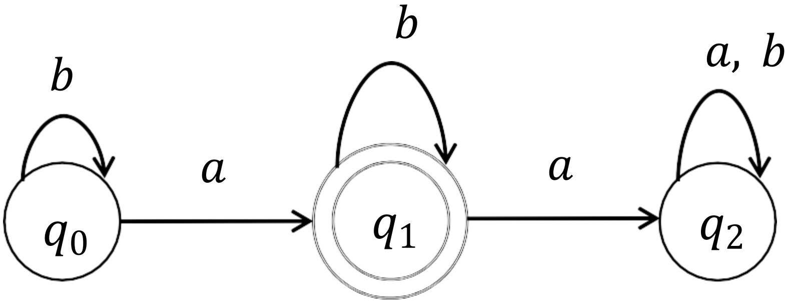


Fig 15.6

Extended Transition Function

The behavior of transition of a transition function over an arbitrary string needs to be determined. An extended transition function is defined as $\delta *: Q \times \sum * \rightarrow Q$, which reads a string in place of an input symbol and defines a new transition state after reading the string. For example, if $\delta(q_0, a) = q_1$ and $\delta(q_1, b) = q_2$, then $\delta * (q_0, ab) = q_2$.

15.3 Non-deterministic Finite Automata

A non-deterministic finite automaton (NFA) allows a set of moves for each situation rather than a fixed choice as in the case of deterministic automata. Formally, NFA is defined by a five-tuple $M_N = (Q, \sum, \delta, q_0, F)$, where Q is a finite set of internal states, $\sum$ is a finite set of input symbols, called the alphabet, $\delta.\ Q \times (\sum \cup \{\lambda\}) \rightarrow P(Q)$ is the transition function, $P(Q)$ is the power set of $Q, q_0 \in Q$ is the initial state, and $F \subseteq Q$ is the set of final states. Moreover, $L(M_N)$ is the language accepted by the machine M_N.

Differences between Deterministic Finite Automaton and Non-Deterministic Finite Automaton

1. In NFA, the range of δ is the power set $P(Q)$, which describes a set of transition states for a given input symbol and the existing state. For example, for a current state q_0 and an input symbol $a, \delta(q_0,\ a) = \{q_1, q_2\}$. In DFA, each transition defines a unique state.
2. NFA can make a transition without consuming an input symbol. It is defined as $\lambda - transition$. For example, $\delta(q_0,\ \lambda) = q_1$. It is not possible in DFA.
3. In NFA, there may be no transition defined for a specific state. For example, $\delta(q_0, a) = \emptyset$. It is not possible in DFA.

Although there are some differences between the definitions of DFA and NFA, it may be shown in formal theory that they are equivalent. For any given NFA, one may construct an equivalent DFA, and vice versa.

In NFA, where several moves are possible, we start with one and move forward to check whether the given string is accepted or not. In case of non-acceptance, we move backwards and explore other choices. This process is called $backtracking$.

Example 15.5 Let $\sum = \{a, b\}$. Construct an NFA that accepts the set of all strings containing ab as a substring.

Solution: Since we need ab as a substring, in order to get $ab,$ we need two states after the initial state q_0

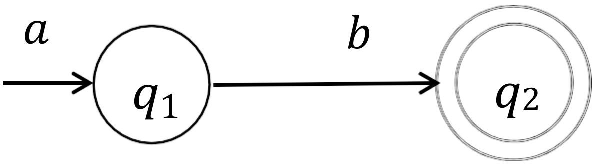


Fig 15.7

Now, before and after the string ab, any number of a and b can appear. Thus, the NFA can be constructed as shown below,

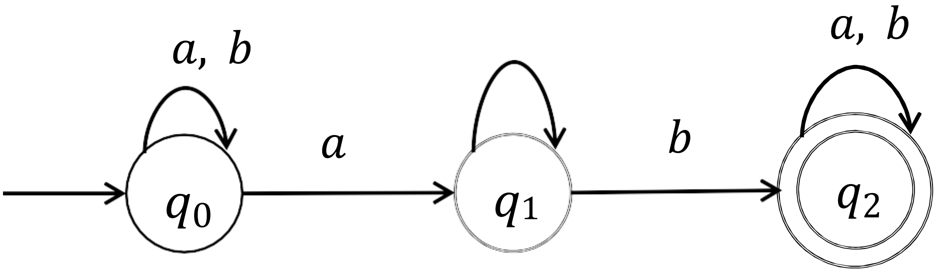


Fig 16.8

15.4 Minimization of Finite Automata

For a given language, there may be more than one DFA that accepts the same language. The two or more deterministic finite automata that accept the same language will differ only in the number of states. Thus, the number of states in a DFA can be reduced without affecting the nature of the automata. The reduction procedure can be understood with the help of the following definitions:

Two states q_i and q_j are called indistinguishable if $\delta * (q_i, w) \in F \rightarrow \delta * (q_j, w) \in F$ and $\delta * (q_i, w) \notin F \rightarrow \delta * (q_j, w) \notin F$ for all $w \in \sum^*$.

Two states q_i and q_j are called distinguishable by a string w if there exists a string $w \in \sum^*$ such that $\delta * (q_i, w) \in F$ and $\delta * (q_j, w) \notin F$

The relation indistinguishable on the set of states of a DFA forms an equivalence relation. As every equivalence relation generates a partition on the given set in which the relation is defined, the relation indistinguishable will also generate a partition on the set of states. Let $M\_D = (Q, \sum, \delta, q_{R0}, F)$ be a DFA and $M_R = (Q_R, \sum, \delta_R, q_{R0}, F_R)$ be the corresponding reduced DFA. The reduction procedure can be defined as follows:

1. If a state is not accessible from the initial state through any path, then remove the state.

2. Check every pair of states (q_i, q_j) for being distinguishable. For this purpose, if $q_i \in F$ and $q_j \notin F$ or vice versa, then identity (q_i, q_j) as distinguishable.
3. For all remaining pairs (q_i, q_j) and all $\alpha \in \sum$, compute $\delta(q_i, \alpha)$ and $\delta(q_j, \alpha)$. Let $\delta(q_i,\ \alpha)\ =\ q_m$ and $\delta(q_j,\ \alpha)\ =\ q_N$. If $(q_m,\ q_N)$ is identified as distinguishable through step 2, then identify (q_i, q_j) also as distinguishable.
4. Repeat step 3 until the chain of identification of distinguishable pairs terminates. This procedure will give all distinguishable pairs of states.
5. Find the remaining indistinguishable pairs of states. This will generate a partition on the set of states. Hence, find the partition generated by the pair of indistinguishable states.
6. For each set of partition $\{q_i, q_j, \ldots \ldots, q_k\}$, create a state $\{q_i, q_j, \ldots \ldots, q_k\}$ in M_R.
7. For each state $\{q_i, q_j, \ldots \ldots, q_k\}$, the transition rule can be made as follows:
   Let $q_x \in \{q_i, q_j, \ldots \ldots, q_k\}$, and $q_y \in \{q_l, q_m, \ldots \ldots, q_N\}$, such that $\delta(q_x, \alpha) = q_y\ \ in\ \ M_D$.
   Then $\delta_R(\{q_i, q_j, \ldots \ldots, q_k\},\ \alpha)\ =\ \{q_l, q_m, \ldots \ldots, q_N\}$ in M_R.
8. The initial and final states of M_R are the states that include q_0 and $q_i \in F$, respectively.

Example 15.6 Reduce the DFA shown in the given figure in to minimal state DFA.

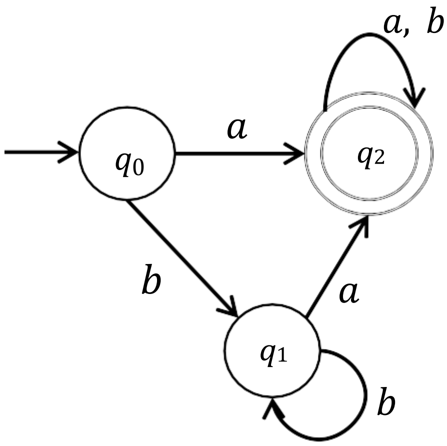


Fig 15.9

Solution

(a) Since $q_0 \notin F,\ q_1 \notin F\ \ and\ \ q_2 \in F$, it can be seen that (q_0, q_1) and (q_1, q_2) are distinguishable pairs of states.

(b) Now the remaining pair is (q_0, q_1). We shall check whether the pair (q_0, q_1) is distinguishable or not. Since $\delta(q_0, a) = q_2,\ \delta(q_1, a) = q_2$ and (q_2, q_2) is not marked as distinguishable pair of states as both states are same in the pair thus (q_0, q_1) cannot be marked as distinguishable. Similarly $\delta(q_0, b)\ =\ q_1$ and $\delta(q_1, b)\ =\ q_1,\ (q_0, q_1)$ cannot be marked as distinguishable. From this, it can be identified as indistinguishable.

(c) The indistinguishable pair of states is (q_0, q_1). The pair (q_0, q_1) generates the partition $\{\{q_0, q_1\}, \{q_2\}\}$. Therefore, there will be two states in the reduced DFA, namely $\{q_0, q_1\}\ and\ \{q_2\}$.

(d) Since $\delta(q_0, a) = q_2$ and $\delta(q_1, a) = q_2$, we have $\delta_R(\{q_0, q_1\}, a) = \{q_2\}$. Similarly, $\delta(q_0, b) = q_1$ and $\delta(q_1, b) = q_1$ and therefore, $\delta_R(\{q_0, q_1\}, b) = \{q_0, q_1\}$. Moreover, $\delta_R(\{q_2\}, a) = \{q_2\}$ and $\delta_R(\{q_2\}, b) = \{q_2\}$.

(e) The state $\{q_0, q_1\}$ will be the initial state and the state $\{q_2\}$ will be the final state.

The transition graph of the reduced DFA can be drawn as below

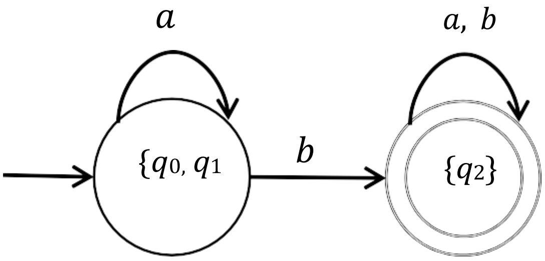


Fig 15.10

So far, we have discussed the finite state machines that accept or reject a given input string. Now we shall study a form of automaton that does not provide the decision of acceptance or rejection of a string, but provides an output in the form of another string. An automaton that accepts input strings and translates them into output strings is called an automaton with an output or a transducer.

Let $\sum$ be a finite set of input symbols and $\prod$ be the finite set of output symbols. Then a transducer T can be defined as $T : \sum^* \rightarrow \prod^*$.

In the case of a transducer, we get an output string for each input string. Thus, the concept of final state is meaningless in transducers. There are two types of transducers:

1. Mealy machine
2. Moore machine

15.5 Mealy Machine

A Mealy machine is defined as a six-tuple $M\_e\ (Q, \sum, \prod, \delta, \gamma, q_0)$, where Q is a finite set of internal states, $\sum$ is a finite set of input symbols, $\prod$ is a finite set of output symbols, $\delta :$ $Q \times \sum \rightarrow Q$ is the transition function that maps a state into another state for a given input symbol for a given state, and $q_0 \in Q$ is the initial state.

In a Mealy machine, the output is given over the transition arc. The representation of a Mealy machine is similar to that of DFA, except the label of edges where a pair of symbol is assigned to each transition arc that shows the input symbols and the corresponding output symbols. In a Mealy machine, the length of the output string is the same as that of the input string.

Example 15.7 Design a Mealy machine that generates the complement of a binary number.

Solution Here, $\sum = \{0,1\}$ and $\prod = \{0, 1\}$. For the input 0, the output is 1, and for the input 1, the output is 0. The transition graph of the Mealy machine is shown below

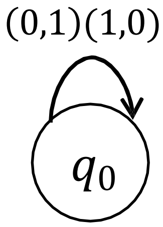


Fig 15.11

Table 15.3 Transition table

Present State	**Transition output**			
	Input = 0		**Input = 1**	
	Next State	**Output Symbol**	**Next state**	**Output symbol**
q_0	q_0	1	q_0	0

15.6 Moore Machine

A Moore machine is defined as a six – tuple $M_o(Q, \sum, \prod, \delta, \gamma, q_0)$, where Q is a finite set of internal states, $\sum$ is a finite set of input symbols, $\prod$ is a finite set of output symbols, $\delta : Q \times \sum \rightarrow Q$ is the transition function that maps a state into another state for a given input symbol, $\gamma : Q \rightarrow \prod$ is the output function that maps a state into an output symbol, and $q_0 \in Q$ is the initial state.

In a Moore machine, the output is given by the state itself. The representation of a Moore machine is similar to that of a DFA, except the state representation where we assign an output string is one more than that of the input string. The first symbol in the output string always specifies the start state.

Example 15.8 Design a Moore machine that generates the complement of a binary number.

Solution: Here, $\sum = \{0,1\}$ and $\prod = \{0,1\}$

For the input 0, the output is 1, and for the input 1, the output is 0. The transition graph of the Moore machine is shown below

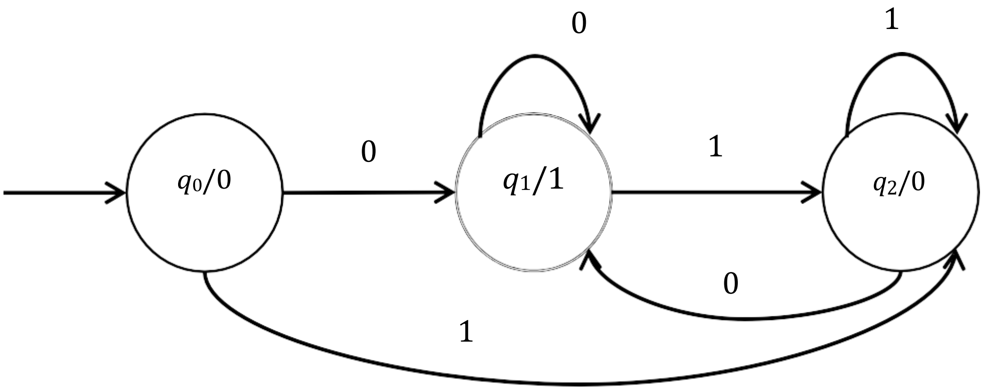


Fig 15.12

Table 15.4 Transition table of Moore machine

Present State	Transition Output		
	Next state		Output Symbol
	Input =0	Input = 1	
q_0	q_1	q_2	0
q_1	q_1	q_2	1
q_2	q_1	q_2	0

Problems for Exercise:

1. Define a word or string from a set A, and give an example.
2. Define the operation of concatenation, and give an example.
3. Consider the words $u = a^2ba^3b^2$ and $v = bab^2$. Find
    a. uv
    b. vu
    c. v^2
4. Define a language over a set A.
5. Let K and L be languages over an alphabet A. Define the language KL over A
6. Consider the language $L = \{ab, c\}$ over $A = \{a, b, c\}$ find
    a. L^0
    b. L^3
    c. L^{-2}
7. Define the finite state automaton (FSA)
8. Let $A = \{a, b\}$. Construct an automaton M which will accept those words from A where the number of $b's$ is divisible by three.
9. Define the state diagram $D = D(M)$ of a finite state automaton M, and give an example. Usually, an automaton M is defined by means of its state diagram rather than by listing its five parts.
10. Construct a deterministic automaton equivalent to

$$M = (\{q_0, q_1\}, \{0, 1\}, \delta, q_0, \{q_0\})$$

Video Lectures of this course is available at:

http://elearning.uou.ac.in/course/view.php?id=112

Steps to access the course

Step 1: open the link: elearning.uou.ac.in

Step 2: Create a login ID at: http://elearning.uou.ac.in/login/signup.php?

Step 3: Verify your credentials by clicking on the link sent to your registered email.

Step 4: Login to elearning.uou.ac.in

Step 5: Click Online Course Tab on the top of the page.

Step 6: Select Course on "*Discreet Mathematics*"

Step 7: Click on *Enroll Now* button

Step 8: Access the contents