Date: Page:

Unit 1 :- Basic foundations (3 Hrs)

Review of mathematical preliminaries:-

Sets:- 1.1 Review of set theory, Logic, functions, proof

The term set is the "collection of ~~objects~~ well defined objects", Which are called the elements of the sets.

Some examples of sets are given below.
- Set of vowels in English alphabet
- Set of odd numbers less than 20
- The set of natural numbers Less than 10 etc.

If x is an element of A then we write $x \in A$
If x is not an element of A then we write $x \notin A$

eg:-
Set of binary digits is $\{0, 1\}$
set of decimal digits is $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$

We basically use the capital letters (A, B, C ...) ~~for~~ for denoting the sets and smaller letters (a, b, c, ...) to denote the elements of any sets.

eg $A = \{a, b, c, \ldots\}$

Methods of describing a set.

a) Enumeration method – Listing
b) Standard method or description method.
c) Set builder method.

a) Enumeration method :-

- Elements of a set are enumerated or listed. We enclose them in braces.

eg:- $A = \{1, 2, 3, --- 100\}$
$B = \{a, b, c, ---- z\}$
$C = \{a, e, i, o, u\}$

b) Standard method :-

- Frequently used sets are usually given symbols that are reserved for them alone.

for eg:-
N (Natural numbers) $= \{1, 2, 3, ---- n\}$
Z (the integers) $= \{--- 2, -1, 0, 1, 2, 3 ---\}$
Q (rational numbers)
R (real numbers) etc.

c) Set builder method :-

- Another way of describing sets is to use the set builder notation.

eg:-
We could define the rational numbers as.
$Q = \{x/y : x \in Z, y \in Z, y \neq 0\}$

Date: Page:

Note that in the set builder description of rational numbers
a) x/y indicates that a typical element of the set is "fraction"
b) ':' (Colon) is read "such that" or "where" used interchangeably with vertical line '|'.
c) $x, y \in Z$ abbreviates $x \in Z$ and $y \in Z$
d) all commas are read as "and".

Types of sets:-

a) Empty set :- A set which has no elements aka null set or void set. denoted by $\{\emptyset\}$
b) Singleton set :- A set which has a single element
eg. $S = \{1\}$
c) Disjoint sets :- Two or more sets are said to be disjoints, if there are no common elements among.
$A = \{a, b, c\}$
$B = \{e, f, g\}$
d) Overlapping sets :- Two or more sets are said to be Overlapping set if there are at least one common element among.
e) Finite sets : A set having ~~havi~~ specified number of elements
$A = \{1, 2, 3\}$
f) Infinite sets : A set having non finite elements
$A = \{1, 2, 3, ---\}$

Date: Page:

- Universal set :- The set of all objects or things under consideration is called universal set denoted by U.

Basic Set Operations :-

Union :-

eg Union of $(A \cup B) = \{x \mid x \in A \text{ or } x \in B\}$

Intersection :-

eg: Intersection of A and B:

$A \cap B = \{x \mid x \in A \text{ and } x \in B\}$

Difference :-

eg:- Complement of B in A:

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

Date: Page:

Logic:

Some of the basic concepts of mathematical Logic Includes,

Propositions

- A proposition is a sentence to which one and only one of the terms "true" or "false" can be meaningfully applied.

eg:-
"five is odd" → True
"$40 \geq 20$" → True
"$5 \in \{1,2,3\}$" → false

Logical operators:

- If p and q are propositions then their
- Conjunction (AND) 'p and q' (denoted by $p \wedge q$)
- Disjunction (OR) $\vee$
  $p \vee q$
- Negation (NOT) ~ OR ¬ ($\sim p$, $\neg p$)
- Conditional operator (implication) $p \to q$, if p then q
- Biconditional operator (If and only if) $p \leftrightarrow q$

Conditional

p	q	$p \to q$
0	0	1
0	1	1
1	0	0
1	1	1

false when condition (hypothesis) is true & conclusion is false

Biconditional

p	q	$p \leftrightarrow q$
0	0	1
0	1	0
1	0	0
1	1	1

$p \leftrightarrow q$ is true when p & q have same truth values

Date:

Tautology:-

Expression involving variables that is true in all cases, is called tautology.

eg $p \vee \sim p$, $(p \wedge q) \rightarrow p$ etc.

Contradictions:-

Expression involving logical variables that is false for all cases of its truth table is called contradiction.

eg $p \wedge \sim p$; ϕ

~~Relati~~ Functions:-

⇒ A function is a correspondance relation between two sets (called the domain and range) such that to each element of the domain, there is assigned exactly one element of the range.

Classes of functions:-

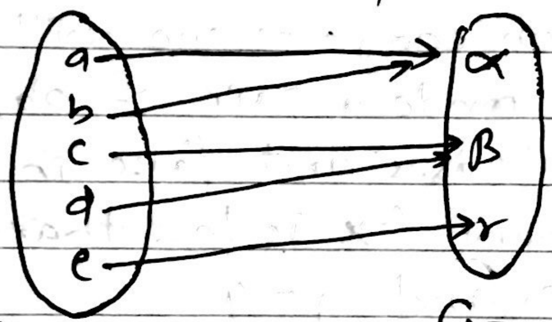
On-to-functions:- (~~Injective~~ Surjective) (~~one to many~~)

⇒ A function from set S_1 to S_2 is said to be an on-to function if every element of S_2 is the image of one or more element of S_1.

let $S_1 = \{a, b, c, d, e\}$

$S_2 = \{\alpha, \beta, \gamma\}$

here, f is defined as follows.



(Domain) (range)

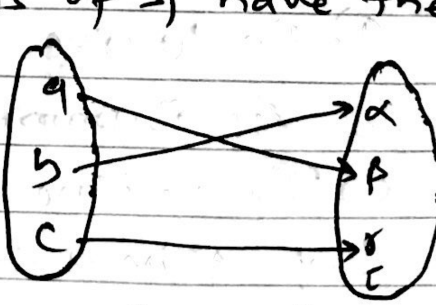
every element of co-domain is mapped to by <u>at least</u> one element of domain.

One-to-one function:- (Injective):-

⇒ A function from S_1 to S_2 is said to be a one to one function if no two elements of S_1 have the same image in S_2.

$S_1 = \{a, b, c\}$

$S_2 = \{\alpha, \beta, \gamma, I\}$



(Domain) (range)

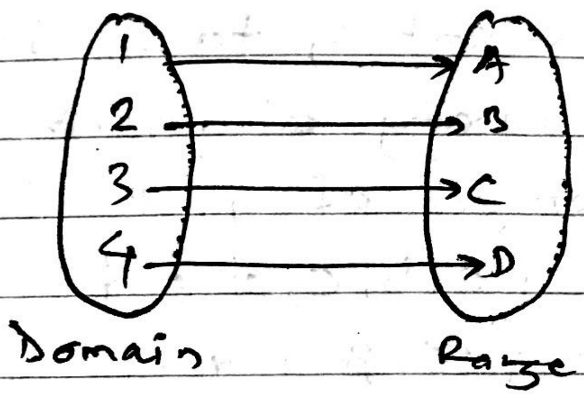
co-domain is mapped <u>atmost</u> one element in domain

Date:______ Page:______

One to one On-to function:- (bijective)

- A function from set S_1 to S_2 is said to be one to one onto function if it is both onto and one to one function.

$S_1 = \{1, 2, 3, 4\}$
$S_2 = \{A, B, C, D\}$



element of co-domain is mapped to by exactly one element of domain.

new 1.1

Proof Techniques:-

- A proof is a valid arguement that establishes the truth of a statement.
- While establishing the truth of a statement, theorem; different rules, axioms and already proven facts are used.

Two types of proof techniques.

1. formal proof
   - have to follow certain defined rules.
2. Informal proof
   - No fixed rules or predefined rules

   a) Direct proof

   b) Indirect proof
   - → Proof by contradiction
   - → proof by Contra-positive

hypothesis conclusion.
$p \rightarrow q$
p is true
q is true.

Date: Page:

(a) **Direct proof:-**

let $p \rightarrow q$ be an implication. Then in direct proof, we assume that the hypothesis of the implication is true i.e. p is true, using different theorems, axioms and already proven facts. We conclude that ~~conclude~~ the conclusion of the implication is true i.e. q is true.

- The idea behind the direct proof is that if true hypothesis leads to true conclusion then the implication is also true.

eg:- Show that if n is even, then n^2 is also even using direct proof.

⇒ let, n is even.

Then by definition of even integer

$n = 2k$, for some integer k.

$n^2 = 4k^2$ (squaring both sides)

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

by definition of even integers, 2k is even.

(b) **Indirect proof.**

~~Proof~~ by contrapositive.

⇒ We assume that the negation of conclusion is true. Using different rules, axioms & already proven ~~theorem~~ facts, we conclude that the negation of hypothesis is true.

$$p \rightarrow q$$

$$\neg q \rightarrow \neg p$$

Proof by contradiction:-

p: Ram is a student
$\neg p$ बार छुट (assume)

⇒ following cases may arise,

① If the statement is not implication,

② If the statement is implication. (just like contrapositive)

- If the statement is not implication, we assume the negation of the statement is true and by processing we reach to a contradiction. Hence, we conclude that our assumption is false and the given statement is ~~false~~ true.

Other proof techniques:-

Proof by resolution:-

$$
\begin{array}{l}
p \lor \neg q \\
q \lor r \\
\hline
\therefore p \lor r
\end{array}
$$

Mathematical Induction:-

⇒ To prove using mathematical indution, we first show that it is true for some initial value of n.

- Then assume that the statement is true for any arbitrary value 'k' and show that it is true for k+1.

Date: Page:

eg:- Using mathematical induction, show that

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

Sol^n: **Basic step :-**

~~for n = 1~~

for $n = 1$

$$\frac{1(1+1)}{2} = 1, \text{ which is true}$$

Inductive step:-

- We assume that it is true for any arbitrary k.

i.e. $1+2+3+\cdots+k = \frac{k(k+1)}{2}$ is true

- Now we have to show that it is true for $(k+1)$.

i.e.
$$\underbrace{1+2+3+\cdots+k}_{} + k+1 = \frac{k(k+1)}{2} + k+1$$

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

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

$$\underset{\text{like}}{=} \frac{(\underset{n}{k}+1)((\underset{n}{k}+1)+1)}{2}$$

which is true.

1.2:

Automata, Computability & Complexity:-

⇒ Theory of computation can be divided into the following 3 areas:

1) Automata theory
2) Computability theory
3) Complexity theory.

Automata theory:-

- It is the study of abstract machines (or more appropriately, abstract 'mathematical' machines or systems) for modeling computations.

why abstract machine?

⇒ Abstract machine allows us to model the essential parameters and ignore the non-essential parameters.

- Examples of the computational models are:

*) Finite Automata.
  - These are used in text processing, compilers, and hardware design.

*) Context-free Grammars. (PDA)
  - These are used to define programming languages and in Artificial Intelligence.

*) Turing Machines.
  - These form a simple abstract model of a "real" computer, such as your PC at home.

Date:________ Page:________

Computability theory :-

* What is computability ? What is computable?
  - Add 2 numbers
  - find the roots of a quadratic equation
  - Multiply 2 matrices & so on.

  Note that : all the above have algorithms

- First of all you must know it (ie. there must be an algorithm) then only you can instruct the computer coding that algorithm into some suitable language.
- We can compute those problems those who have algorithms suppose you do not know how to multiply 2 matrices then the computers also cannot do it.

* What is not computable?
- There are many problems which cannot be computed even using very powerful computers.
- It's because there are no algorithms for that. (ie. If we do not know how to do it then no computer is able to do that.

* Central Question in computability theory : Classify problems as being solvable or unsolvable.

Date: ______ Page: ______

Complexity Theory:-

- Various kinds of computer problems may be easy or hard.
- Informally, a problem is called "easy", if it is effeciently solvable.

  example:-- Sorting a sequence of say 1,000,000 numbers.

  - Searching for a name in a telephone directory.
- On the other hand, a problem is called "hard", if it cannot be solved effeciently or if we don't know whether it can be solved efficiently.

  eg:-

  - Time table scheduling for all courses at DWIT.

* eg:- Time complexity (eg. big O)

  space complexity.

1.3 Basic concept of Automata Theory

Languages:-

Symbol < 0, a

Alphabets:- (symbols make alphabet)

The symbols are generally letters and digits. Alphabets are defined as a finite set of symbols. It is denoted by 'Σ' — symbol.

eg:- An alphabet of set of decimal numbers is given by $\Sigma = \{0, 1, 2, --- 9\}$
An alphabet of binary number is
$$\Sigma = \{0, 1\}$$

Strings:-

- A string or word is a finite sequence of symbols selected from some alphabets.

Eg:- If $\Sigma = \{a, b\}$ then
'abab' is a string over Σ.

- A string is generally denoted by 'w'.
- Empty string is denoted by ϵ represents string with 0(zero) occurance of symbols.

$\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \ldots$

Date: Page:

Closure of an alphabet :- (kleen star)

⇒ Closure of an alphabet is defined as the set of all strings over an alphabet Σ including empty string and is denoted by Σ^*

eg:-

let $\Sigma = (0,1)$ then

$\Sigma^* = \{\epsilon, 0, 1, 00, 01, 10, 000, ----\}$

power of Alphabet → $\Sigma^1 = \{0, 1\}$

$\Sigma^2 = \{00, 01, 10, 11\}$

$\Sigma^0 = \{\epsilon\}$

$\left.\begin{array}{l}\Sigma^1 \to \text{set of strings} \\ \Sigma \to \text{alphabet (its members are symbol)}\end{array}\right\}$ difference.

$\Sigma^+ = \{0, 1, 00, \ldots\}$

+ve closure

↳ not having ϵ

$\Sigma^* - \epsilon$

Concatenating of string :-

⇒ let w_1 and w_2 be two strings, then $w_1 w_2$ denotes the concatenation of w_1 and w_2.

eg.

$w_1 = abc$ $w_2 = xyz$ then,

$w_1 w_2 = abcxyz$

Substring :

Date:______ Page:______

Language:-

- A set of strings all of which are chosen from Σ^*, where 'Σ' is a particular alphabet, is called a language.

eg:- The Language of all strings consisting n 0s followed by n 1s $(n \geq 0)$:

$\{\epsilon, 01, 0011, 000111, \ldots\}$

- The set of strings of 0s and 1s with equal number of each:

$\{\epsilon, 01, 10, 0011, 0101, 1001, \ldots\}$

Scanned with CamScanner