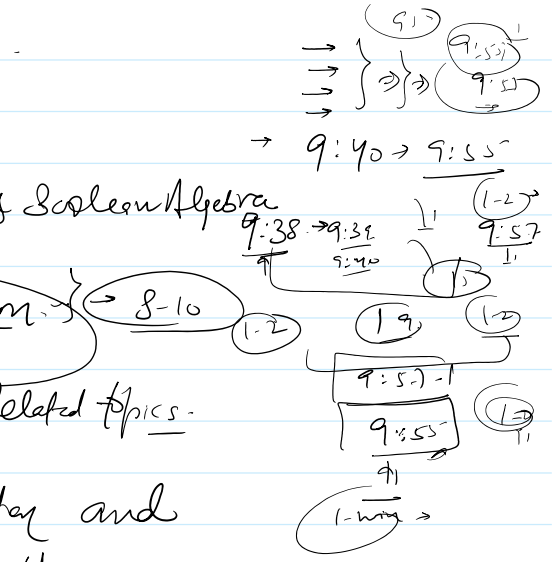
07 October 2020 08:59

Unit-4

- Unit-1 → Set-theory
- Unit-2 → Logic & proof
- Unit-3 → Lattice theory & Boolean Algebra
- Unit-4 → Algebraic System → 8-10
- Unit-5 → Rings, fields & Related topics
- Unit-6 → Recurrence Relation and Generating function → Advanced Counting techniques

Unit-4 = Unit-4 + Unit-5



07 October 2020 09:13

Unit-4. Algebraic System

→ Ref. book → Discrete mathematics by Trembley & Manohar

Ch-3.

Outline:- Algebraic System
→ Examples
→ properties ⇒ Abstract Algebraic System.
→ properties
→ axioms.
→ Imp. Results and Concepts e.g. Isomorphism

07 October 2020 09:18

Compiler ↔ TOC → Theory of Computing.
Theory of Computation

Epigram → gcc → MI Code

Semi-group → simplest algebraic structure.
→ closure, associativity.
→ Appⁿ → Sequential machines, formal language, Computer arithmetic and multiplication.

→ a=b+c

monoid: → identity property.
→ Syntactical Analysis and formal language.

Groups → inverse property.
→ fast adders and error Correcting Codes.

Extra Appⁿ: → Compilation of expressions in Polish notation, language and grammars, the theory of fast adders

Computer N/w → error Correcting and detecting Codes.

Binary Operation. →

07 October 2020 09:27

n-ary operation → n = 1, 2, 3 ....

Generally $f: X \to X$

$f: X^n \to X$, n = 1, 2, 3 ...

Defⁿ → X → set, $f: X \times X \to X$, then f is binary operation on X.

$f: X^n \to X$, → n-ary operation.

n = 1, Unary operation.

n → order of the operation.

$X \times X \to X$, → If operation produces images which also from the same set → closure.

X → ℤ → multiplication, addition, ... Binary.

X → E → Union, Intersection → Binary

X → E → Unary - Complementation.

X → E → Composition {fog gof}

07 October 2020 09:33

w.r.t. Operation }

w.r.t. Operation.

Binary operation → Composition table. w.r.t. Operation

$A = \{a, b\}$, $P(A) = (\{\phi\}, \{a\}, \{b\}, \{a, b\})$

$B_0 \quad B_1 \quad B_2 \quad B_3$.

Union →

Intersection →

$\langle P(A), \cup, \cap \rangle \rightarrow$

Note - If - down

$\cup$	B_0	B_1	B_2	B_3
B_0	B_0	B_1	B_2	B_3
B_1	B_1	B_1	B_3	B_3
B_2	B_2	B_3	B_2	B_3
B_3	B_3	B_3	B_3	B_3.

$\cap$	B_0	B_1	B_2	B_3
B_0	B_0	B_0	B_0	B_0.
B_1	B_0	B_1	B_0	B_1
B_2	B_0	B_0	B_2	B_2
B_3	B_0	B_1	B_2	B_3.

Binary Operation → $+, -, \circ, *, \Delta, \cup, \cap, \vee, \wedge,$ — etc.

$f\langle x, y \rangle \rightarrow xfy$ or $x*y$

$x+y$, $A \cup B$.

07 October 2020 09:41

General properties of binary operations:- $f<x,y>$ $x f y$ $x * y$.

$X \rightarrow$ set

1) Commutative: $f<x,y> = f<y,x>$ $\quad x * y = y * x$

2) Associative: $f<f<x,y>,z> = f<x, f<y,z>>$ $\quad (x*y)*z = x*(y*z)$

3) Distributive: $f: X * X \rightarrow X$, $\quad g: X * X \rightarrow X$

↳ $\rightarrow$ f is distributive over g.

$f<x, g<y,z>> = g<f<x,y>, f<x,z>>$

$f<x,y>$ multiplication is distributive over addition

$f \rightarrow *$, $g \rightarrow +$ then $x * y + z = (x*y) + (y*z)$

$\rightarrow$ addition, Comm., Associative. ← $5 \times 6 + 4 =$

Union, Intersection $\rightarrow$ $\rightarrow (5\times6) + (6\times4)$

Distributive: $\rightarrow$ multiplication over the Addition $\rightarrow$??

Addition over the multiplication $\rightarrow$

$5+4 \times 6 \Rightarrow$ ~~$5+6$~~ $(5+4) \times (4+6)$ $\rightarrow$ Composition of two functions $\rightarrow$ $fog \neq gof$ ✗

not Commutative

07 October 2020 09:56

Some special elements:

Any $y * x$

Def^n: Let $*$ → binary operation, X → set

If $e_l \in X$ s.t. $e_l * x = x$ for every $x \in X$, then e_l → left identity w.r.t. $*$

If $e_r \in X$ s.t. $x * e_r = x$ for ——— e_r → right identity w.r.t. $*$

$\{e_l = e_r\}$

Theorem:

X →
e_l → then $e_l = e_r = e$ (say) s.t. $x * e = e * x = x$ for every $x \in X$.
e_r → C

then $e \in X$ is unique and called the identity (i.d.) w.r.t. $*$

Pf →

$e_l = e_r = e$.

→ $(\mathbb{I}, +) \to 0$ $\quad E$, Union → ϕ
$(\mathbb{Z}, \times) \to 1$ $\quad$ Intersection → E. }

Propositional logic → disjunction → Contradiction — Identity.
→ Conjunction → tautology.

Def^{n} :→ $*$ → binary operation on X.

If $0_l \in X$ s.t. $0_l * n = 0_l$ for every $n \in X$ → 0_l → left zero w.r.t. n.
If $0_r \in X$ s.t. $0_r * n = 0_r$ — $n \in X$ → 0_r → right zero — —

Theorem:- X →
$*$ → then $0_l = 0_r = 0$ s.t. $0 * n = n * 0 = 0$ for all $n \in X$.
0_l →
0_r →

$0 \in X$ is also unique. also
called zero w.r.t. $*$.

→ zero
$(I, X) \to 0$
$(E, \wedge) \to \phi$

Def^n:- Let $*$ on X. $a \in X$ is idempotent w.r.t. $*$ if $a * a = a$

identity → e
zero → z

Def^n:- $X \to$

$*$ → $a \in X$ is said to be left-invertible if there exist

$x_L \in X$ s.t. $x_L * a = e$ (e: identity) → a → left-invertible, x_L → left-inverse of a w.r.t. $*$

$x_R \in X$ s.t. $a * x_R = e$ → a → right-invertible.
x_R → right-inverse of a w.r.t. $*$.

if a is both left and right invertible then → a is invertible

$*$ on X → Commutative. a is left-invertible & right-invertible

07 October 2020 10:15

Theorem. $*$ → Associative, and identity $e \in X$.

If any element $a \in X$ is invertible, then its both x_l, x_r equal. Such an element is called the inverse of a because it is unique. (a^{-1}).

x_l

x_r

$x_l * a = a * x_l = e$

$a * a^{-1} = a^{-1} * a = e$

$x_l = x_r = a^{-1}$ ⇒

identity → idempotent

$1 * 1 = 1$

identity element → exist → Unique

↳ invertible ??

In any binary operation, the identity element, if exists, is invertible.

A zero element w.r.t. $*$ is invertible — Yes / No

identity
zero
idempotent
inverse

Addition $(\mathbb{Z}, +)$ → $a^{-1} = -a$, $a \in \mathbb{R}$.

$(\mathbb{R}, \times)$ → $a^{-1} = \frac{1}{a}$, $a \in \mathbb{R}$

07 October 2020 10:21

An element $a \in X$ is cancellable w.r.t a binary operation $*$ on X, if for every $x, y \in X$

$$(a * x = a * y) \vee (x * a = y * a) \Rightarrow x = y.$$

If $*$ is associative, $a \in X$ is invertible, then a is cancellable w.r.t $*$.

Reverse is not true. - it means ??

There are some cases where an element is cancellable but not necessarily invertible.

07 October 2020 10:30

→ closure

→ X $x, y \in X \rightarrow x * y$

→ * on X → $x * y \in X$ → again apply → other operations: + , −

*, +

( ) parentheses →

$(x*y)+z$ $z*(x*y)$ $(x*y)+z$ $z+(x*y)$

*, + → precedence 3 ( )

e.g. * → associative, then $(x*y)*z = x*(y*z)$ → no need to put → drop the pain

FORTRAN, → A+B+C → (A+B)+C ← Assumed.

Rounding off

$(A+B)+C \approx A+(B+C)$ → $6.5+(2.4+2.2)$

$(6.5+2.4)+2.2$

$13+2.2$

$15.2 \rightarrow 15$

$6.5+10.1$

$16.6 \rightarrow 6.5+10$

16

Left- associative & Right- associative.

$A + B + C$

$\Downarrow$ left

$(A + B) + C$

$(A' + C)$

$\leftarrow$ assign

$A \leftarrow (B \leftarrow C)$

$\sim\sim p$

$\sim(\sim p)$

$(A - B) - C = A - (B - C)$ X

FORTRAN

$A - B - C \Rightarrow (A - B) - C$ → left- associative

→ left- associative
right- associative

Programming language

07 October 2020 10:39

Binary operations and their properties

Algebraic Systems

Examples

Ch. 3 ⇒

Groups & Monoids

Isomorphism