

3) checkers:- (also called Reversi) 8x8 square grid.
dev by Rosenbloom in 1982) Grid has
& distinct of light. Each face represent a player.

Goal:- make more pieces of players to have.
of that colour free

4) Go 19x19 board.

Each player uses stones either black & white.
Goal is occupy larger part of the board.

5) Backgammon:- Developed by Babylonians in 1980 playing
pieces are moved by using Dice! player
must choose from numerous optimum.

UNIT-3 (Logic Concepts & programming)

1. Introduction
- 2. propositional Calculus.
- 3. propositional logic
- 4. Natural deduction System.
- 5. Axiomatic System.
- 6. Semantic Tableau System in propositional logic.
- 7. Resolution Refutation in propositional logic
- 8. predicate logic.

INTRODUCTION

* An effective way of solving a complex problem is to reduce it to simpler parts and solve each part separately. This method is called "Reduction".

* To implement the above process, we need to use some logic.

(i) Logic is used to distinguish correct reasoning from incorrect reasoning, which is concerned with the principle of drawing valid inferences from a given set of ~~two~~ statements.

(ii) Symbolic logic is divided into two types, they are:

- 1) propositional logic.
- 2) predicate logic.

Propositional logic deals with a collection of declarative statements which have a truth value. [TRUE/FALSE]

Predicate logic is an expression which consists of variables under a specific domain [its truth value depends on variables]

A predicate with variables can be made a proposition by either assigning a value to the variable or by quantifying the variable.

Calculus is a method to calculate reasoning.

Reasoning is the process of applying logical and critical

thinking to a problem; it involves identifying

the subject or area of study and then applying logical and critical thinking to form a conclusion or solution.

Logical reasoning involves

the logical relationships between different

concepts, such as $A \rightarrow B$, $A \wedge B$, $A \vee B$, $\neg A$, $A \leftrightarrow B$

and the use of logical rules to derive new conclusions.

Logical reasoning is a branch of mathematics that deals with the study of logical relationships between different concepts.

Some important properties of logic are:

- The law of identity: $A \rightarrow A$
- The law of non-contradiction: $\neg(A \wedge \neg A)$
- The law of excluded middle: $A \vee \neg A$

These laws are fundamental to the study of logic and are used to derive other logical relationships.

Logical reasoning is a branch of mathematics that deals with the study of logical relationships between different concepts.

2. PROPOSITIONAL CALCULUS

Propositional Calculus (PC) is a language of propositions in which a set of rules are used to combine simple propositions to form compound propositions using some logical operators.

- The logical operators are called connectives, they are.

\wedge (AND), \vee (OR), \sim (NOT), \rightarrow (Implies), \leftrightarrow (Equivalent)

- PC uses well formed formula, which is defined as symbol/string of symbols generated by formula grammar of a formal language.

- Some important properties of WFF in PC are,

• The smallest unit/atom is considered to be WFF

• If A is WFF, then $\sim A$ is also WFF

• If A & B are WFF, then $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, & $(A \leftrightarrow B)$ are also WFF

- A propositional expression is called WFF if & only if it satisfies above properties

TRUTH Table

- A Truth table is used to provide operational definitions of important logic operators
- The logical constants in PC are represented as True (T) & False (F) in a Truth Table.

Eg:- let us assume A, B as propositional symbols, then Truth table is as follows

A	B	$\neg A$	$A \wedge B$	$A \vee B$	$A \rightarrow B$	$A \leftrightarrow B$
T	T	F	T	T	T	T
T	F	F	F	T	F	F
F	T	T	F	T	T	F
F	F	T	F	F	T	T

Equivalence Laws:-

used to reduce / simplify WFF (or) to derive new formula.

Name of Relation	Equivalence Relation
1) Commutative law	$A \vee B \equiv B \vee A$, $A \wedge B \equiv B \wedge A$
2) Associative law	$A \vee (B \vee C) \equiv (A \vee B) \vee C$ $A \wedge (B \wedge C) \equiv (A \wedge B) \wedge C$
3) Double Negation	$\neg(\neg A) \equiv A$

1) Distributive Law

$$A \cup (B \cap C) \cong (A \cup B) \cap (A \cup C)$$

$$A \cap (B \cup C) \cong (A \cap B) \cup (A \cap C)$$

2) Demorgan's Law

$$\sim (A \cup B) \cong \sim A \cap \sim B$$

$$\sim (A \cap B) \cong \sim A \cup \sim B$$

3) Absorption Laws

$$A \cup (A \cap B) \cong A$$

$$A \cap (A \cup B) \cong A$$

$$A \cup (\sim A \cap B) \cong A \cup B, A \cap (\sim A \cup B) \cong A \cap B$$

4) Idempotence Law

$$A \cup A \cong A, A \cap A \cong A$$

5) Excluded Middle Law

$$A \cup \sim A \cong T \text{ (True)}$$

6) Contradiction Law

$$A \cap \sim A \cong F \text{ (False)}$$

10) Other Common Laws

$$A \cup F \cong A, A \cup T \cong T$$

$$A \cap F \cong F, A \cap T \cong A$$

$$A \leftrightarrow B \cong (A \rightarrow B) \cap (B \rightarrow A)$$

$$\cong (A \cap B) \cup (\sim A \cap \sim B)$$

3. PROPOSITIONAL LOGIC

Propositional Logic is used to check the validity, satisfiability, unsatisfiability of a formula using Equivalence laws.

A formula 'A' is said to be a 'tautology' if the value of 'A' is true for all 'interpretation'.

Problem:- Prove that the below argument is tautology

"If it is humid then it will rain and since it is humid today it will rain"

Sol:- First Symbolized the each part & above argument by using propositional atom

A: If it is humid

B: Then it will rain

The sentence, "If it is humid then it will rain and since it is humid today it will rain"

Can be written as

$$[(A \rightarrow B) \wedge A] \rightarrow B$$

The Truth Table for α is

A	B	$\neg A$	$(\neg A \vee B)$	$A \rightarrow B$	$\neg(A \rightarrow B)$	$\neg \neg$	$\neg \neg \neg$
T	T	F	T	T	F	T	T
T	F	T	T	F	T	T	T
F	T	F	T	T	F	T	T
F	F	T	T	T	F	T	T

Since α is true for all interpretations, The Sentence is a Tautology.

4. NATURAL DEDUCTION SYSTEM

- A Natural deduction System minimizes the pattern of natural reasoning.

- Natural deduction System is based on a set of deductive Inference Rules

- Assume that A_1, A_2, \dots, A_n where $1 \leq k \leq n$, are set of atoms and α_j , j lies between $1 \leq j \leq n$ and β are well formed formulae, The Inference Rules are mentioned below.

Rule Name	Symbol	Rule	Description
Introducing \wedge	(I: \wedge)	If A_1, \dots, A_n Then $A_1 \wedge \dots \wedge A_n$	If A_1, \dots, A_n are true, Then Their conjunction $A_1 \wedge \dots \wedge A_n$ is also true.
Eliminating \wedge	(E: \wedge)	If $A_1 \wedge \dots \wedge A_n$ Then A_i ($1 \leq i \leq n$)	If $A_1 \wedge \dots \wedge A_n$ is true, Then any A_i is also true.
Introducing \vee	(I: \vee)	If any A_i ($1 \leq i \leq n$) Then $A_1 \vee \dots \vee A_n$	If any A_i ($1 \leq i \leq n$) is true, Then $A_1 \vee \dots \vee A_n$ is also true.
Eliminating \vee	(E: \vee)	If $A_1 \vee \dots \vee A_n, A_1 \rightarrow A, \dots, A_n \rightarrow A$ Then A	If $A_1 \vee \dots \vee A_n, A_1 \rightarrow A, \dots, A_n \rightarrow A$ are true; Then A is true.
Introducing \rightarrow	(I: \rightarrow)	If from $\alpha_1, \dots, \alpha_n$ infer β is proved $\alpha_1 \wedge \dots \wedge \alpha_n \rightarrow \beta$ is proved	If given that $\alpha_1, \alpha_2, \dots$ and α_n are true. and from those we deduce β Then $\alpha_1 \wedge \dots \wedge \alpha_n \rightarrow \beta$ is also true.

Eliminating \rightarrow	($E \rightarrow$)	If $A_1 \rightarrow A, A_1$, Then A	If $A_1 \rightarrow A$ and A_1 are true Then A is also true. This is called Modus ponens Rule.	
Introducing \leftrightarrow	($I \leftrightarrow$)	If $A_1 \rightarrow A_2, A_2 \rightarrow A_1$, Then $A_1 \leftrightarrow A_2$	If $A_1 \rightarrow A_2$ and $A_2 \rightarrow A_1$ are true. Then $A_1 \leftrightarrow A_2$ is also true.	*)
Eliminating \leftrightarrow	($E \leftrightarrow$)	If $A_1 \leftrightarrow A_2$ Then $A_1 \rightarrow A_2, A_2 \rightarrow A_1$	If $A_1 \leftrightarrow A_2$ is true. Then $A_1 \rightarrow A_2$ and $A_2 \rightarrow A_1$ are true.	*)
Introducing \neg	($I \neg$)	If from A infer $A, \neg A$, is proved Then $\neg A$ is proved	If from A (which is true), $\neg A$ is proved Then truth of $\neg A$ is also proved.	*)
Eliminating \neg	($E \neg$)	If from $\neg A$ infer $A, \neg A$ is proved Then A is proved	If from $\neg A$, a contradiction is found Then truth of A is also proved.	*)

Problem:- using NDS. (Natural deduction System) prove that $A \wedge (B \vee C)$ is deduced from $A \wedge B$.

Ans:- The above statement can be written in NDS as follows:
 from $A \wedge B$, infer $A \wedge (B \vee C)$.

	Description	Formula	Comment
1.	Theorem	from $A \wedge B$, infer $A \wedge (B \vee C)$	To be proved
2.	Hypothesis	$A \wedge B$	1
3.	$E: A$	A	2
4.	$E: \wedge(1)$	B	3
5.	$I: \vee(3)$	$B \vee C$	4
6.	$I: \wedge(2,4)$	$A \wedge (B \vee C)$	

5. AXIOMATIC SYSTEM

- x) An Axiom is a statement (and/or) proposition which is true.
- x) The Axiomatic System is based on a set of 3-axioms and one rule of deduction.
- x) Here \sim (not) and \rightarrow (implies) are only allowed to form a formula. i.e. \wedge , \vee & \leftrightarrow are to be expressed only by using \sim and \rightarrow .

EX:-

$$1) A \wedge B \approx \sim (\sim A \vee \sim B)$$

$$\approx \sim (A \rightarrow \sim B)$$

$$2) A \vee B \approx \sim (A \rightarrow \sim B)$$

$$3) A \leftrightarrow B \approx (A \rightarrow B) \wedge (B \rightarrow A)$$

$$\approx \sim [(A \rightarrow B) \rightarrow \sim (B \rightarrow A)]$$

- x) In axiomatic system there are 3 axioms which are always true with one rule modus (ponens's Rule)

Here α , β and γ are well formed formula of the axiomatic system.

- x) The 3 axioms and one rule of axiomatic system as

Axiom 1: $\alpha \rightarrow (\beta \rightarrow \alpha)$

Axiom 2: $[\alpha \rightarrow (\beta \rightarrow \gamma)] \rightarrow [(\alpha \rightarrow \beta) \rightarrow (\alpha \rightarrow \gamma)]$

Axiom 3: $(\neg \alpha \rightarrow \neg \beta) \rightarrow (\beta \rightarrow \alpha)$

1 Rule.

Modus ponem : Hypothesis : $\alpha \rightarrow \beta$ and α .

Consequent: β

* The modus ponem's rule says that, Given that $\alpha \rightarrow \beta$ and α are hypothesis (assume to be true), β is inferred (true) as a consequent

Problem :-

using axiomatic system, establish that $A \rightarrow C$ is a deductive consequent (K) of $\{A \rightarrow B, B \rightarrow C\}$ i.e.

$$\{A \rightarrow B, B \rightarrow C\} \vdash A \rightarrow C$$

\vdash deductive symbol

Solution :-

Prove that Theorem that

$$\{A \rightarrow B, B \rightarrow C\} \vdash A \rightarrow C$$

Description	Formula	Comment
Theorem	$(A \rightarrow B, B \rightarrow C) \vdash A \rightarrow C$	To be proved.
Hypothesis 1	$A \rightarrow B$	1
Hypothesis 2	$B \rightarrow C$	2
Instance of Axiom 1	$B \rightarrow C \rightarrow [A \rightarrow (B \rightarrow C)]$	3
Modus ponem's (2,3)	$A \rightarrow (B \rightarrow C)$	4
Instance of Axiom 2	$[A \rightarrow (B \rightarrow C)] \rightarrow [(A \rightarrow B) \rightarrow (A \rightarrow C)]$	5
Modus ponem's (4,5)	$(A \rightarrow B) \rightarrow (A \rightarrow C)$	6
Modus ponem's (1,6)	$A \rightarrow C$	Hence proved.

6. Semantic Tableau System In propositional Logic

* Both 'Natural Deduction' System & 'Axiomatic System' uses 'forward changing approach' but 'Constructing proofs and derivations'.

* The other two approaches which follows 'Backward changing' are.

- Semantic Tableau
- Resolution Refutation methods.

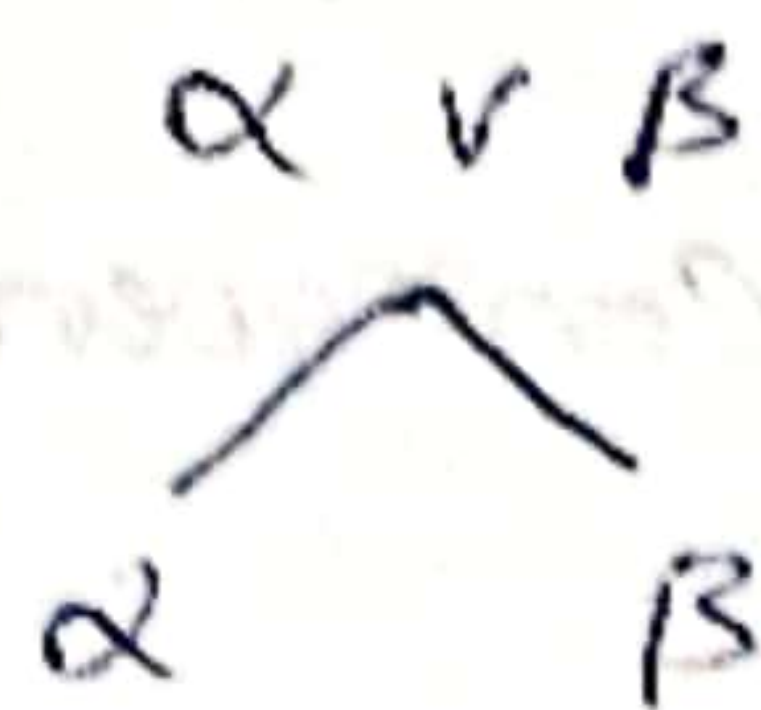
* In Semantic tableau method, a set of rules are applied systematically in a formula or a set of formulae in order to establish consistency (or) inconsistency.

* Semantic tableau is a binary tree which is constructed by using semantic tableau rules with a formulae as a root.

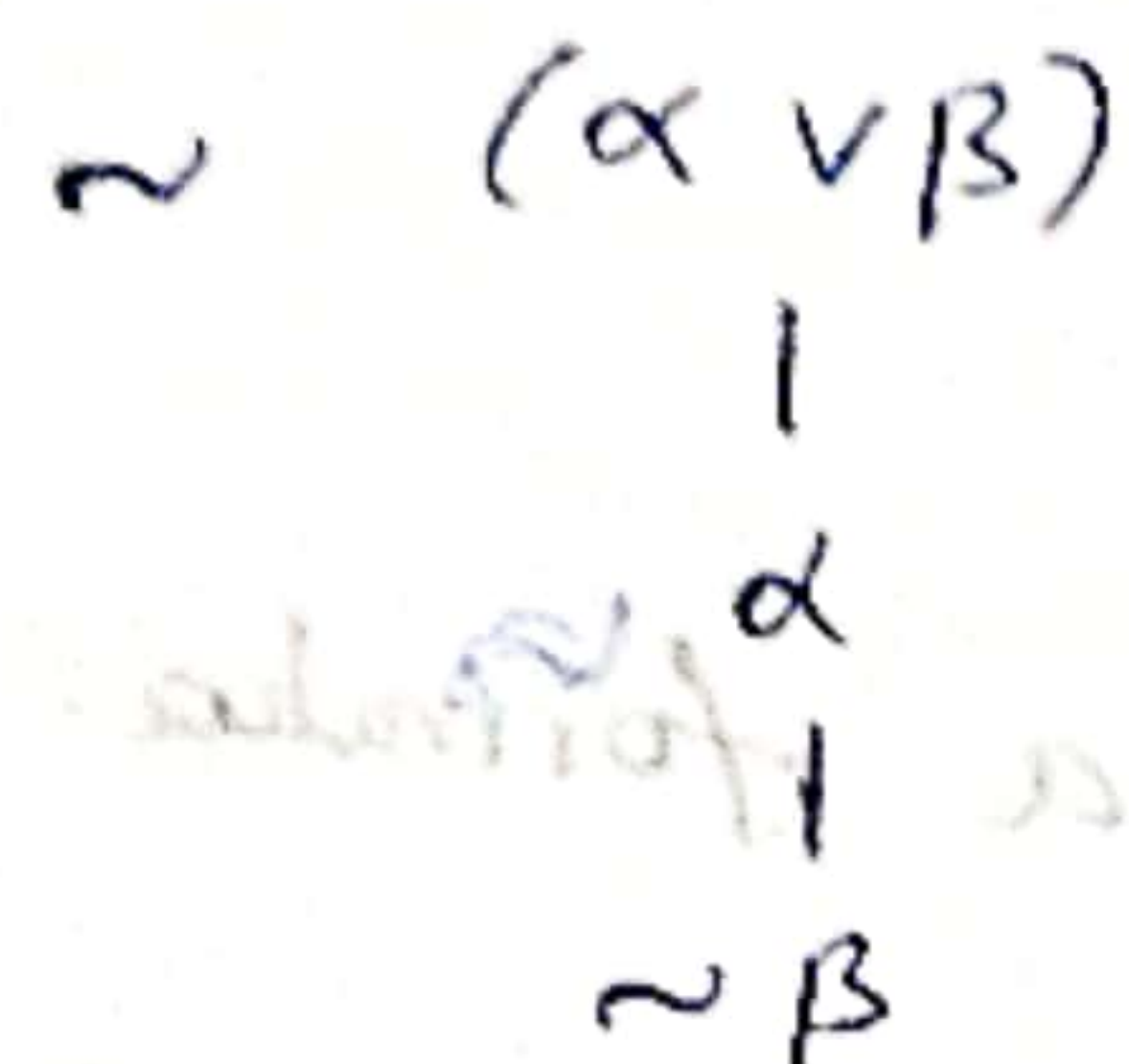
Rule No.	Tableau Tree.
Rule 1	$\alpha \wedge \beta$ $ $ α $ $ β
Rule 2	$\sim (\alpha \wedge \beta)$ $\swarrow \quad \searrow$ $\sim \alpha \quad \sim \beta$

Rule
Rule
Rule
Rule
Rule 8
Rule 9
Ty
pu

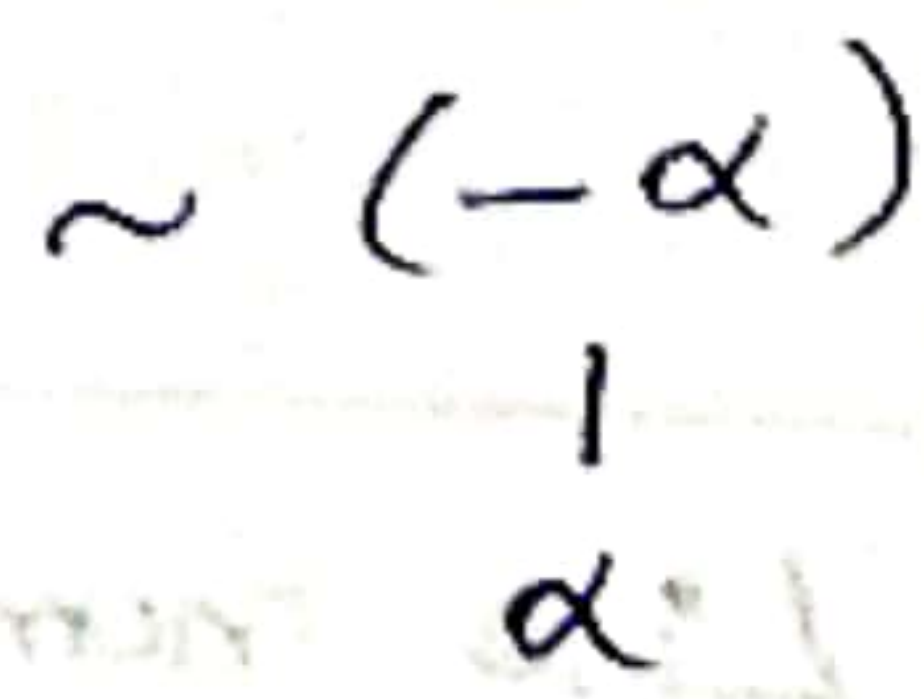
Rule 3



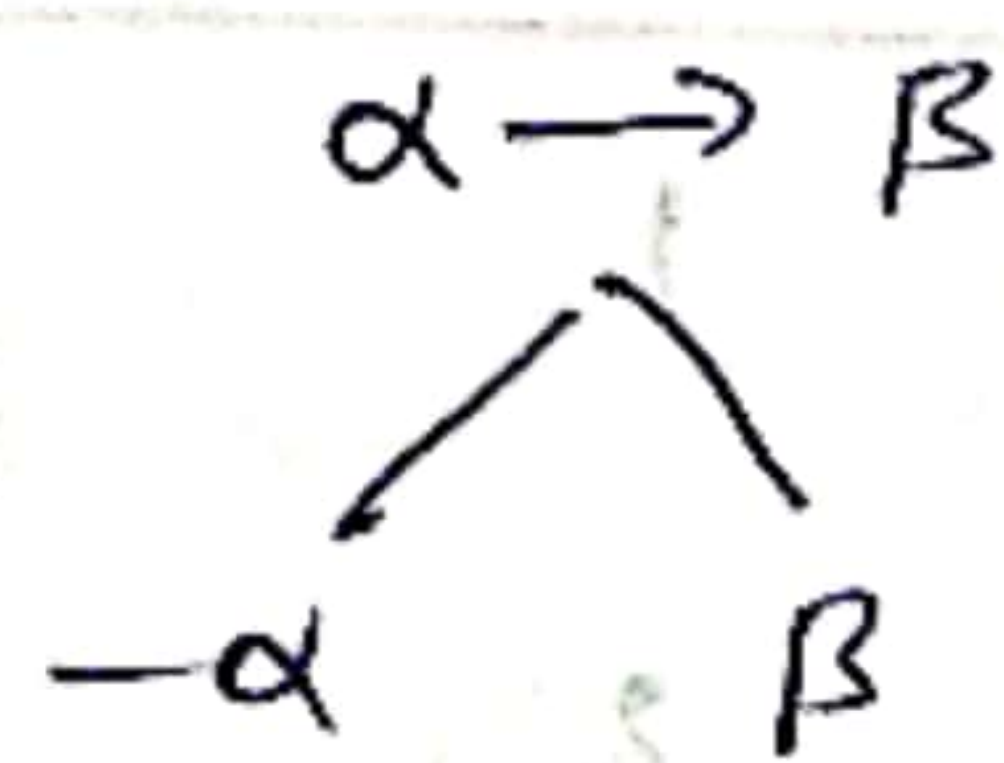
Rule 4



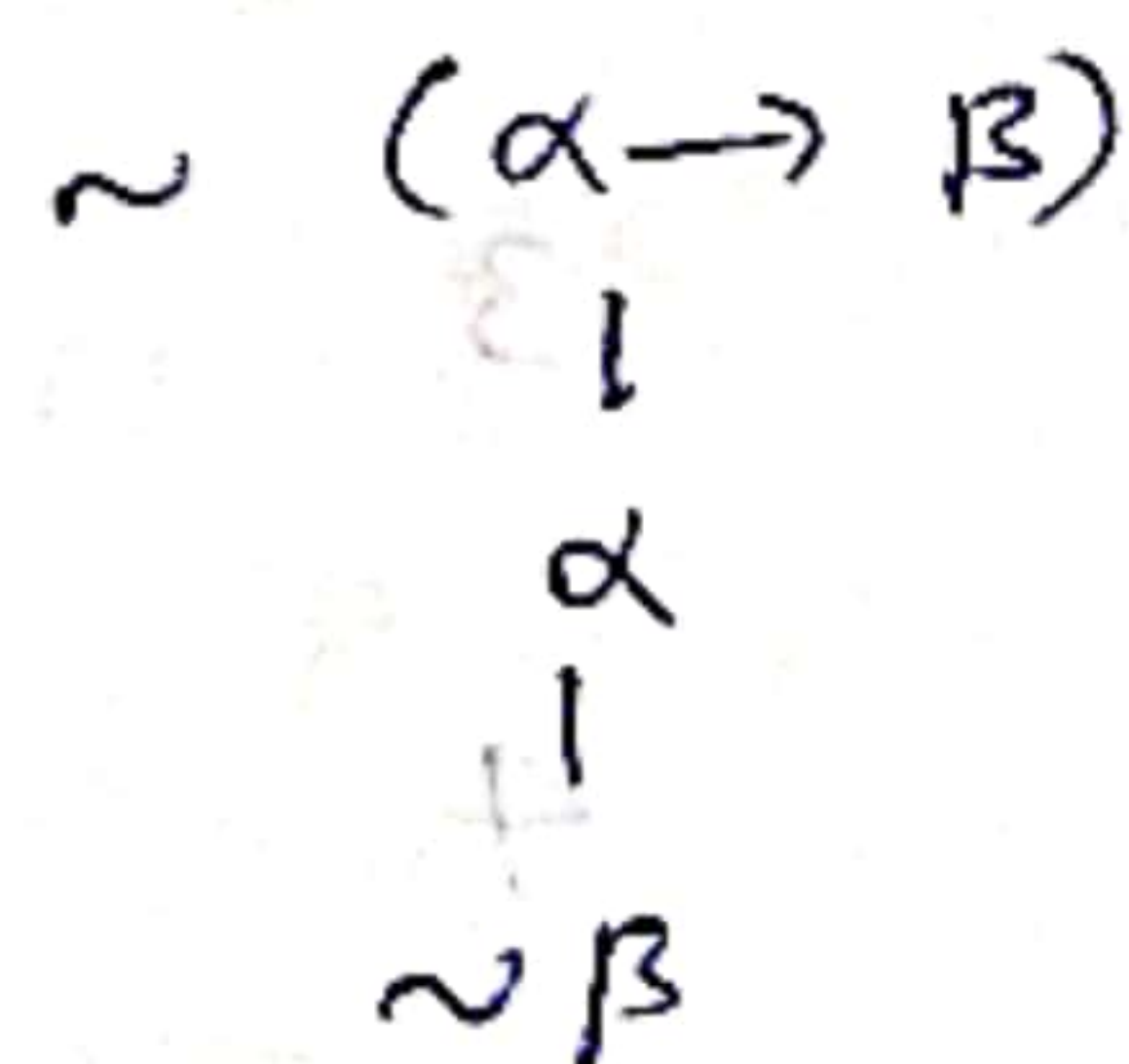
Rule 5



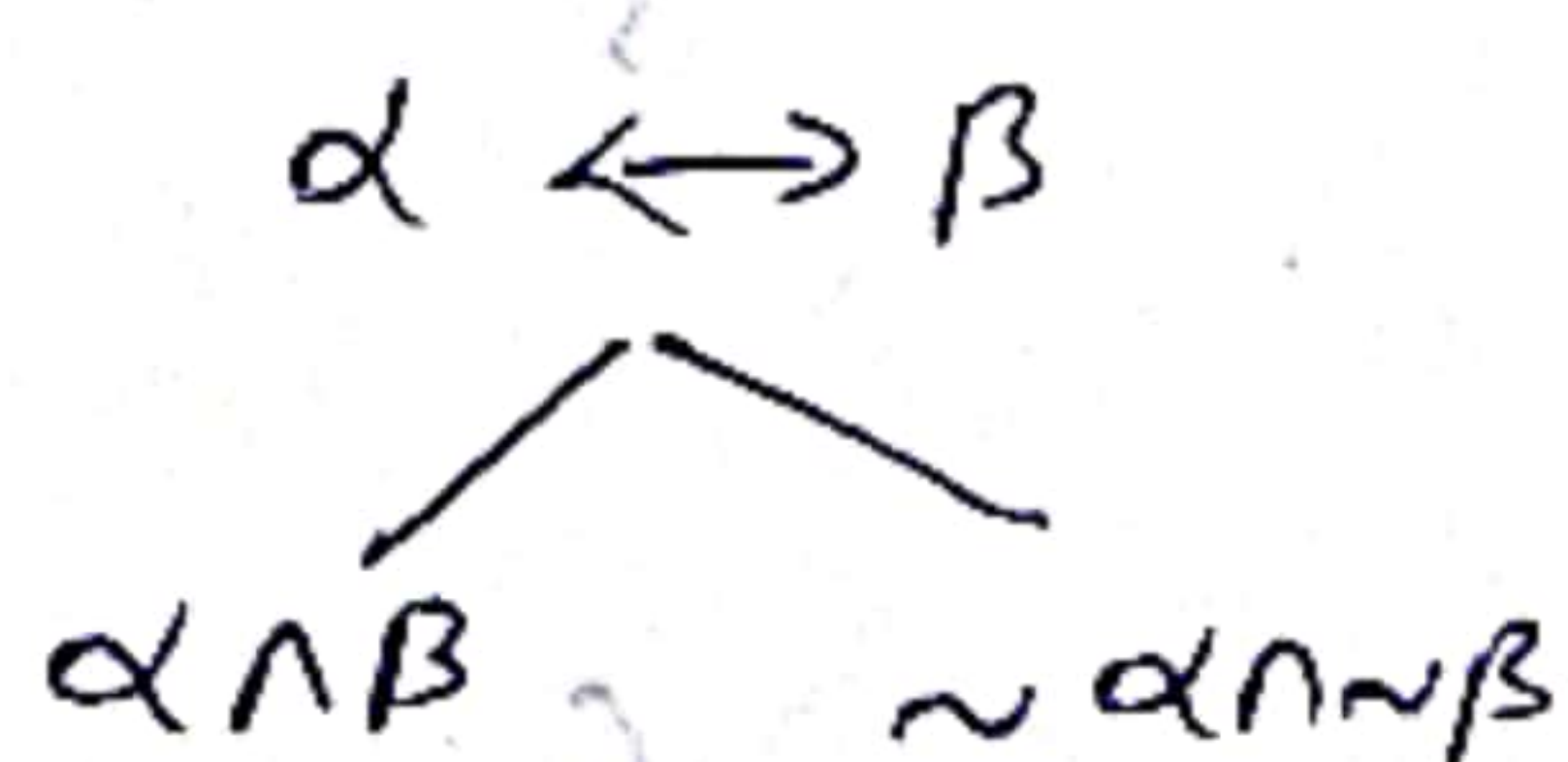
Rule 6



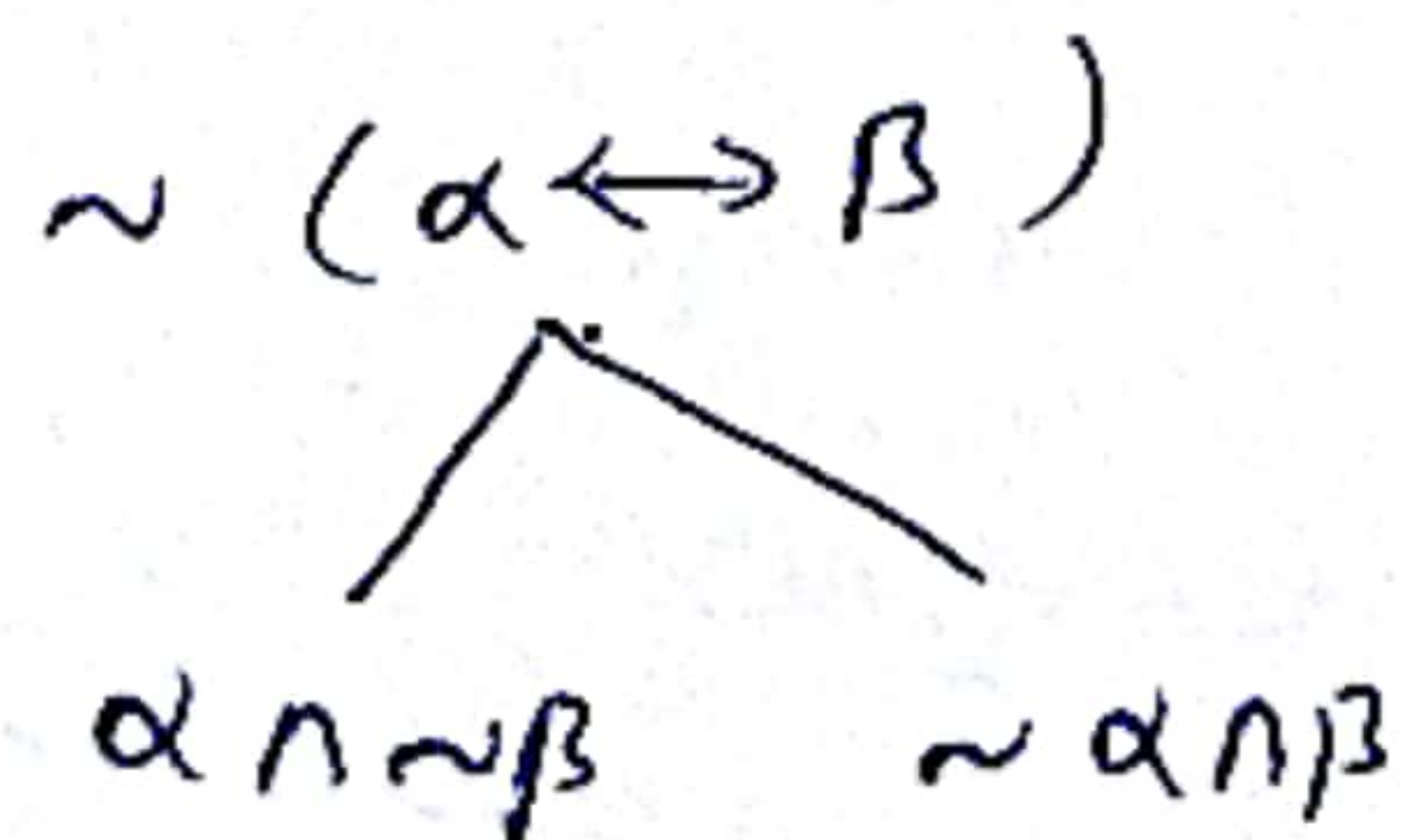
Rule 7



Rule 8



Rule 9



Types of Semantic tableau :-

problems :-

1. Construct a semantic tableau from formula.
2. Show that a formula is unsatisfiable/satisfiable using tableau section

Show That a formula is consistent

Show That as axiom is a logical consequence of formula.

Show That a formula is valid.

Construct a Semantic tableau for a formula.

$$(A \wedge \sim B) \wedge (\sim B \rightarrow C)$$

Rule applied	Formula	Line number.
Tableau root	$(A \wedge \sim B) \wedge (\sim B \rightarrow C)$	1
Rule 2 (1)	$A \wedge \sim B$	2
	$\sim B \rightarrow C$	3
Rule 1 (2)	A	4
	$\sim B$	5
	<div style="display: flex; justify-content: space-between;"> <div style="width: 45%;"> $\sim(\sim B)$ $$ B \times (Closed) </div> <div style="width: 45%;"> (Open) Since no Complementary atoms Since, we have Complementary atoms $\{ B, \sim B \}$ </div> </div>	6

7. Resolution Refutation in Propositional Logic

- (1) To prove a formula or derive a goal from a given set of clauses by contradiction is done by "Resolution Refutation" method.
- * clause is used to denote special formula containing operations \neg and \vee .
 - * Any formula can be converted to set of clauses
 - * RR uses single inference rules, which is known as resolution based on 'modus ponens' inference rule.
 - * During resolution, we need to identify 2 clauses - one with positive atom (P) and other with negative atom ($\neg P$)

Conversion of a formula into a set of clauses :-

- * A function is said to be its normal form if it is constructed using only natural connectives (\wedge, \vee, \neg)
- * Normal forms can be DNF (disjunction) or CNF (conjunction)
- * CNF is a conjunction of Disjunction whereas DNF is a disjunction of conjunctions as

Rules to convert formula to 2IS CNF.

Ex:
Ans:

- $\sim(\sim A) \approx A$
- $\sim(A \wedge B) \approx \sim A \vee \sim B, \sim(A \vee B) \approx \sim A \wedge \sim B$
- $A \vee (B \wedge C) \approx (A \vee B) \wedge (A \vee C)$
- $A \rightarrow B \approx \sim A \vee B$
- $A \leftrightarrow B \approx (A \rightarrow B) \wedge (B \rightarrow A)$

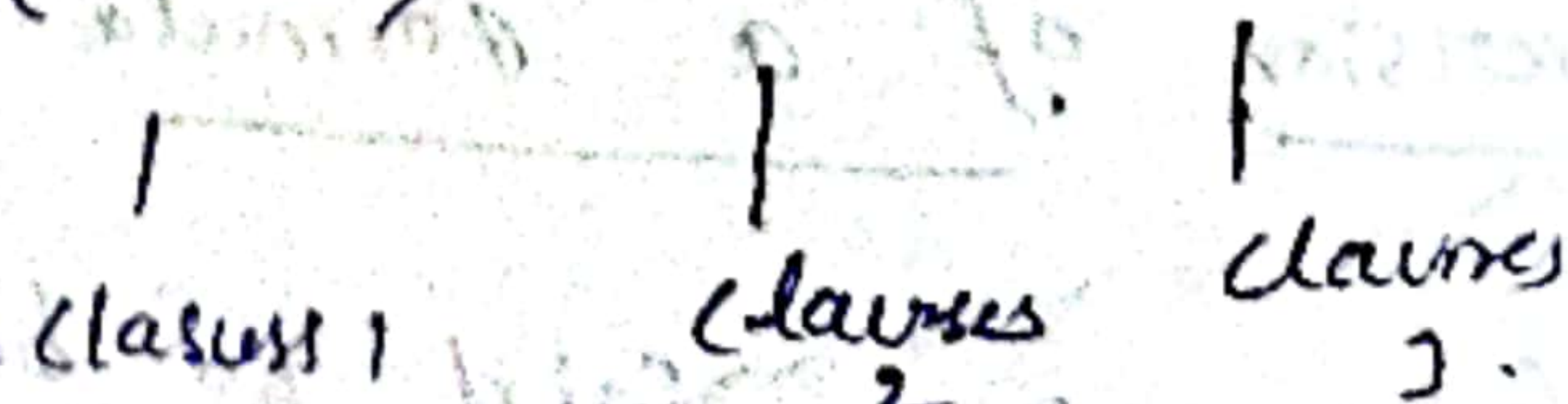
problem:- Convert The problem formula $(\sim A \rightarrow B) \wedge (C \wedge \sim A)$ into CNF

Sol:-

$$(\sim A \rightarrow B) \wedge (C \wedge \sim A) \approx (\sim(\sim A) \vee B) \wedge (C \wedge \sim A)$$

$$\approx (A \vee B) \wedge (C \wedge \sim A)$$

$$\approx (A \vee B) \wedge C \wedge \sim A$$



The Set of Clauses are: $A \vee B, C, \sim A$ which can be written as

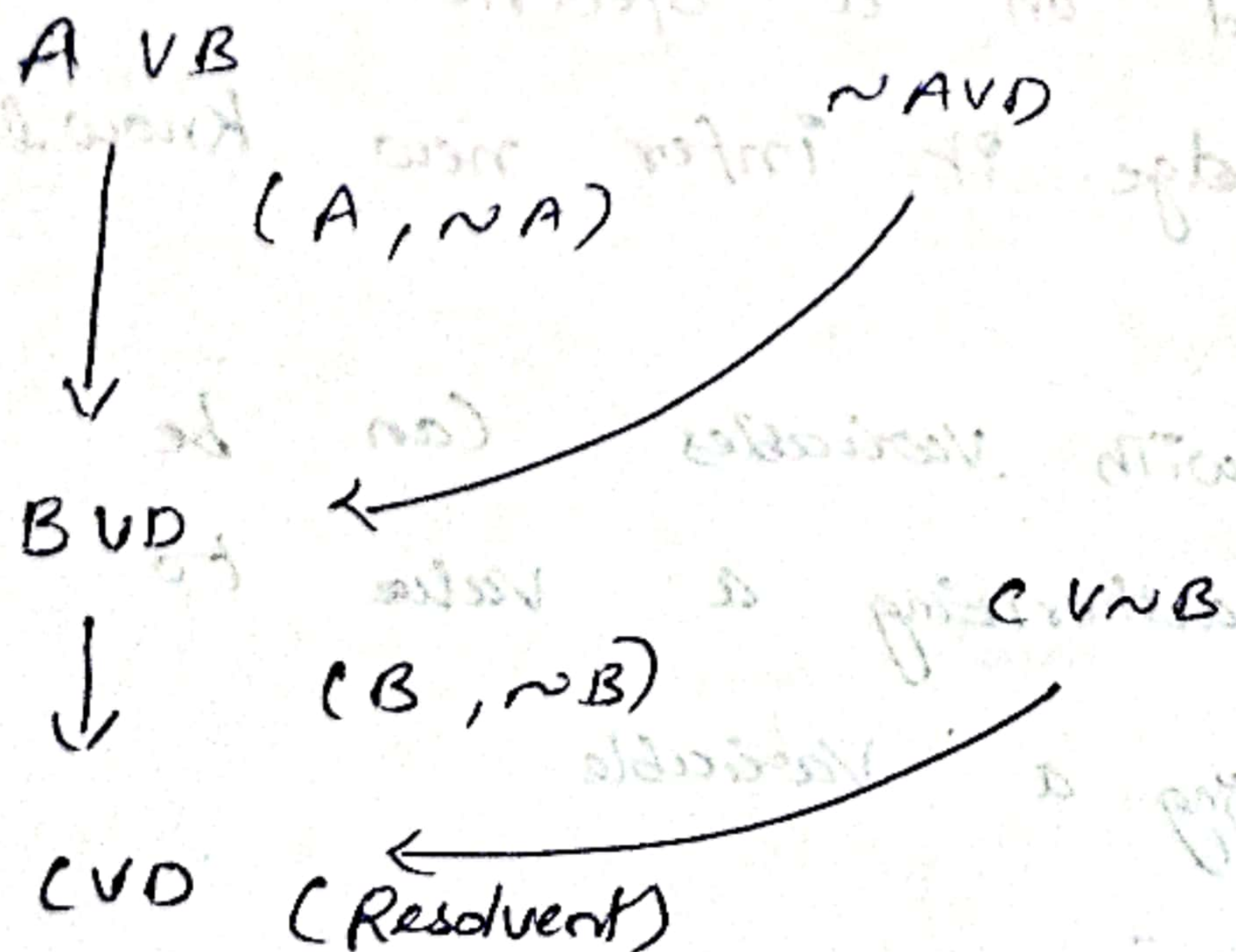
$$\{(A \vee B), C, \sim A\}$$

Resolution of clauses:-

Two clauses can be resolved by eliminating completing pair of literals, from a new clause is constructed by division of remaining literals on both.

Ex:- Find The resolvent of The clause in The set $\{A \vee B, \neg A \vee D, C \vee \neg B\}$

Ans:



from The figure, $C \vee D$ is Resolvent

8. predicate Logic

- A predicate is an expression of one or more variables determined on a specific domain (i.e. we store new knowledge. It infer new knowledge.

- A predicate with variables can be made a proposition by authorizing a value to a variable or by quantifying a variable.

- There are 2 types of quantifier in predicate logic.

- Existential quantifier, denoted by \exists (There exists)
- Universal quantifier, denoted by \forall (for all)

Eg:-

Existential Quantifier: (\exists)

• we use 'and' (conjunction) / AND

• if $P(x)$ is a proposition over universe. U is denoted as $\exists x P(x)$ (There exists at least one value in the universe of variable x such that $P(x)$ true)

Universal Quantifier (\forall)

• If $P(x)$ is a proposition over the universe U is denoted as $\forall x P(x)$ [for every $x \in U$, $P(x)$ is true]

First-order predicate (FOP) Logic / predicate logic

* FOP is another extension of propositional logic which is more efficient than above. Previous mechanisms

FOP = { predicate calculus inference rules }

* FOP not only assumes facts but also assumes things like.

- objects
- relations
- functions

* FOP has 2 main parts

- Syntax
- Semantics.

* Syntax in FOP is a allocation of symbols in a logical expression. These symbols are also called as "Syntactic elements".

* The various categories of syntactic elements / symbols are.

Variables x, y, z

Constant - India, Deepa, Aadhya, mo

predicates Father, Mother, Brother, Sister

function power, sort

Connectives $\wedge, \vee, \neg, \rightarrow, \leftrightarrow$

Equality $=$

Quantifiers \exists, \forall

* Sentences in FOPL are of 2 types

- Atomic Sentence
- Complex Sentence

* Atomic Sentences are basic sentences which are formed from a predicate symbol followed parenthesis with a sequence of terms

Syntax: - predicate symbol (term₁, term₂, ..., term_n)

Ex 5-

1. Ravi & Ajay are brothers. Can be written as Brothers
(Ravi, Ajay)

2. Cat is a cat
Cat (Catty)

x) Complex sentences are formed by combining atomic sentences using connectives

* FOPL statements can be divided into 2 parts

- Subject (main part)
- predicate (relation which binds 2 atoms together)

Eg:-

x (Subject)

is an integer (predicate)

Subject & predicate :- x is an integer

↓ ↓
Sub. predicate

FOPL

Quantifier Eg's :- Symbol Connectives.

Existential :- (\exists) , ' \wedge '

" Some students are intelligent.

Ex :- Students (x) \wedge intelligence \wedge boy

There are some x where x is a student who is

2. Universal: - (\forall) \rightarrow

"All students drinks milk"

$\forall x$ students $(x) \rightarrow \text{drink}(x, \text{milk})$

for all x where x is a student will drink milk.

Note:

Above students cannot be represented in propositional logic, which is overcome by predicate logic.