



AUTOMATA, FORMAL LANGUAGES AND LOGIC

Dr Pooja Agarwal

Professor,
Department of Computer Science & Engineering

AUTOMATA, FORMAL LANGUAGES AND LOGIC

MODULE 5

Propositional Logic

Dr Pooja Agarwal

Department of Computer Science & Engineering

Outline

- ◆ Propositional logic - A very Simple Logic
 - Syntax and Semantics
 - A Simple Knowledge Base
 - A Simple Inference Procedure

Propositional Logic (Syntax and Semantics)

Propositions

Propositions are declarative sentence which is either universally true or universally false but not both. Usually denoted by P or Q or X or Y.

Propositions are **building blocks of logic**.

e.g. 1. The president of India is a woman. (P)

2. Toronto is the capital of Canada. (P)

3. Watch a movie. ($\sim P$)

4. What is your name? ($\sim P$)

Propositional Logic - Syntax

Syntax defines the sentences in the language.

- Atomic Sentences
- Complex Sentences

Atomic Sentences – consists of a *single proposition symbol*.

Each such symbol stands for a proposition that can be true or false.

Two proposition symbol with fixed meaning -

TRUE – Always true proposition

FALSE – Always false proposition

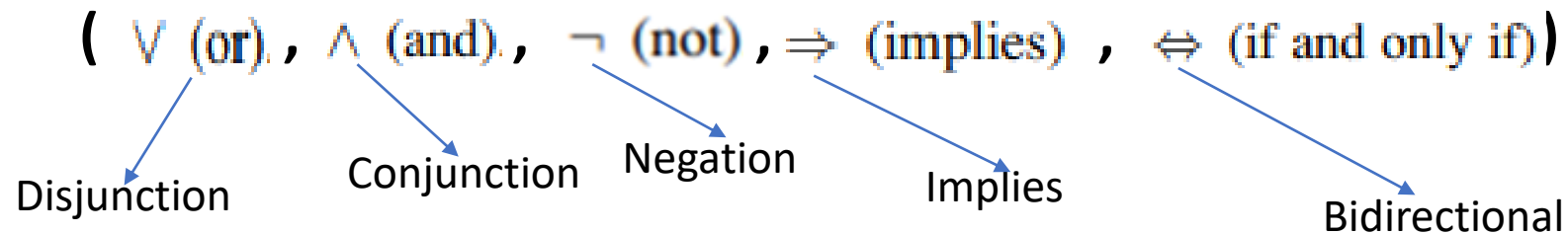
Propositional Logic - Syntax

Syntax defines the sentences in the language.

- Atomic Sentences
- Complex Sentences

Complex Sentences – are constructed from simple sentences.

There are *five logical connectives* in common use:-



Propositional Logic - Semantics

The Semantics defines the rules for determining the truth of a sentence *with respect to* a particular model.

Each model specifies true/false for each proposition symbol

E.g. $P_{1,2}$ $P_{2,2}$ $P_{3,1}$

One possible model will be

$m1 = \{P_{1,2} = \text{false}, P_{2,2} = \text{false}, P_{3,1} = \text{true}\}$

With these three proposition symbols, 8 possible models, can be enumerated automatically.

Propositional Logic - Semantics

Semantics for propositional logic must specify how to compute the truth value of any sentence, given a model m .

$\neg S$	is true iff	S is false
$S_1 \wedge S_2$	is true iff	S_1 is true and S_2 is true
$S_1 \vee S_2$	is true iff	S_1 is true or S_2 is true
$S_1 \Rightarrow S_2$	is true unless	S_1 is true and S_2 is false in m .
$S_1 \Leftrightarrow S_2$	is true iff	S_1, S_2 are both true or both false.

Propositional Logic - Semantics

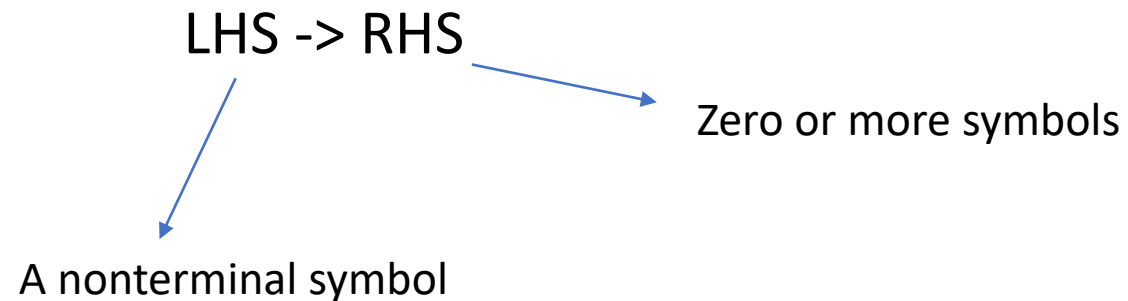
- The order from highest to lowest:

$\neg, \wedge, \vee, \Rightarrow$ *and* \Leftrightarrow

Propositional Logic - Semantics

BNF (Backus- Naur form) grammar, Like CFG has four components:-

- A set of terminal symbols
- A set of non-terminal symbols
- A start symbol
- A set of rewrite rules of the form



Propositional Logic - Semantics

- BNF Grammar for simple arithmetic expressions:-

Expr \rightarrow Expr operator Expr | (Expr) | Number

Number \rightarrow Digit | Number Digit

Digit \rightarrow 0|1|2|3|4|5|6|7|8|9

Operator \rightarrow +|-|/|*

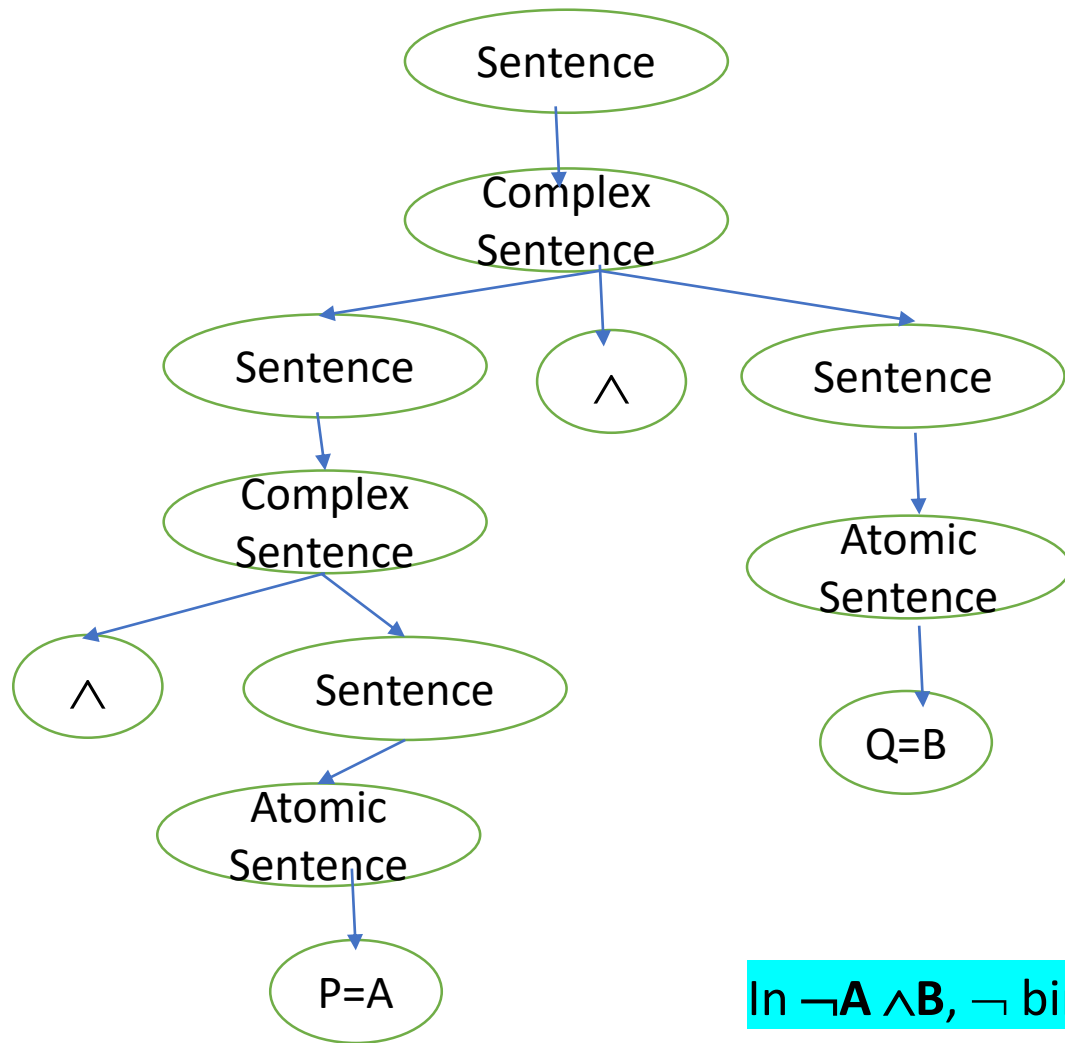
Propositional Logic - Semantics

BNF grammar of sentences in propositional logic, along with operator precedence, from highest to lowest

$$\begin{aligned} \text{Sentence} &\rightarrow \text{AtomicSentence} \mid \text{ComplexSentence} \\ \text{AtomicSentence} &\rightarrow \text{True} \mid \text{False} \mid P \mid Q \mid R \mid \dots \\ \text{ComplexSentence} &\rightarrow (\text{Sentence}) \mid [\text{Sentence}] \\ &\mid \neg \text{Sentence} \\ &\mid \text{Sentence} \wedge \text{Sentence} \\ &\mid \text{Sentence} \vee \text{Sentence} \\ &\mid \text{Sentence} \Rightarrow \text{Sentence} \\ &\mid \text{Sentence} \Leftrightarrow \text{Sentence} \end{aligned}$$

OPERATOR PRECEDENCE : $\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow$

Example: $\neg A \wedge B$



BNF grammar of sentences in propositional logic

Sentence \rightarrow AtomicSentence | ComplexSentence

AtomicSentence \rightarrow True | False | P | Q | R | ...

ComplexSentence \rightarrow (Sentence) | [Sentence]
| \neg Sentence
| Sentence \wedge Sentence
| Sentence \vee Sentence
| Sentence \Rightarrow Sentence
| Sentence \Leftrightarrow Sentence

In $\neg A \wedge B$, \neg binds most tightly as $(\neg A) \wedge B$ rather than $\neg(A) \wedge B$

Truth table of Five Logical Connectives

Negation \sim Or \neg

P	$\sim P$
T	F
F	T

e.g.

1. P: Ram is a good boy

$\neg P$: Ram is not a good boy.

2. P: $8+1 = 5$

$\neg P$: $8+1 \neq 5$

Truth table of Five Logical Connectives

Conjunction \wedge

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

Disjunction \vee

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

Truth table of Five Logical Connectives

Conjunction \wedge

e.g.

P: Today is Friday.

Q: It is raining today.

$P \wedge Q$: Today is Friday and it is raining today.

Disjunction \vee

e.g.

1. P: Today is Friday.

Q: It is raining today.

$P \vee Q$: Today is Friday or it is raining today.

2. P: Sam is an Architect.

Q: Sam is a draftsman.

$P \vee Q$: Sam is an Architect or a draftsman.

Truth table of Five Logical Connectives

Conditional Statement (Implication/ Implies) \Rightarrow
(If..then)

$P \Rightarrow Q$

Antecedent

Consequent

P	Q	$P \Rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

is true **unless** P is true and Q is false in model.

e.g.

1. P: Emma learns AI.

Q: Emma will find a good job.

$P \Rightarrow Q$: If Emma learns AI, then she will find a good job.

2. If you do hard work, then you **would** get 90% on final.

If you do not do hard work, then you **may or may not** get 90% on final.

Truth table of Five Logical Connectives

Conditional Statement (Implication/ Implies) \Rightarrow

(If..then)

$P \Rightarrow Q$

Antecedent

Consequent

If it is raining, then home team wins

is true **unless** P is true and Q is false in model.

P implies Q also means

- P only if Q
- Q whenever P
- Q is necessary condition for P
- Q follows from P
- Q unless $\neg P$

The home team wins **whenever** it is raining

The home team wins **unless** it is not raining

P	Q	$P \Rightarrow Q$
T	T	T
T	F	F
F	T	T
F	F	T

Truth table of Five Logical Connectives

Biconditional Operator (If and Only If) \Leftrightarrow

P	Q	$P \Rightarrow Q$	$Q \Rightarrow P$	$P \Leftrightarrow Q$
T	T	T	T	T
T	F	F	T	F
F	T	T	F	F
F	F	T	T	T

$P \Leftrightarrow Q$ means $P \Rightarrow Q$ and $Q \Rightarrow P$

$P \Leftrightarrow Q$ is a conjunction of $P \Rightarrow Q$, $Q \Rightarrow P$

$P \Leftrightarrow Q \equiv P \Rightarrow Q \wedge Q \Rightarrow P$

e.g.

P: You can take the flight

Q: You buy a ticket.

Then

$P \Leftrightarrow Q$: You can take the flight if and only if you buy a ticket .

Truth table of Five Logical Connectives

P	Q	$\neg P$	$P \wedge Q$	$Q \vee P$	$P \Rightarrow Q$	$P \Leftrightarrow Q$
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

Simple recursive process evaluates an arbitrary sentence, e.g.,

$$\neg P_{1,2} \wedge (P_{2,2} \vee P_{3,1}) = \text{true} \wedge (\text{true} \vee \text{false}) = \text{true} \wedge \text{true} = \text{true}$$

Example

- A square is breezy *if* a neighboring square has a pit.

- A square is breezy *only if* a neighboring square has a pit.

=> means the presence of pits if there is a breeze.

Biconditional,

$$B_{1,1} \Leftrightarrow P_{1,2} \text{ or } P_{2,1}$$

- \Leftrightarrow means (requires) the absence of pits if there is no breeze.

Tautology

- A tautology is when you use different words to repeat the same idea.

e.g. 1. The phrase

“It was adequate enough” is a tautology. The words adequate and enough convey the same meaning.

2. $P \Rightarrow Q \equiv \neg P \vee Q$

Example

1. Translate the following English sentence into a logical Expression.

“You can access the Internet from campus ONLY IF you are a computer Science major OR you are not a fresher.”

2. Prove the logical Equivalence of

$$P \Rightarrow Q \equiv \neg P \vee Q$$

3. Prove the logical Equivalence for Biconditional

$$P \Leftrightarrow Q \equiv (P \wedge Q) \vee (\neg P \wedge \neg Q)$$

P	Q				



THANK YOU

Dr Pooja Agarwal

Department of Computer Science & Engineering

poojaagarwal@pes.edu