**Knowledge Reasoning and Planning**

**Chapter -7 , 8, 9**

**Knowledge Representation, Reasoning, and Planning: Logic:**

**Propositional Logic, Propositional Theorem Proving, First Order Logic: Predicate Logic, Inference in First-Order Logic, Forward chaining, Backward chaining, Resolution.**

**Uncertain Knowledge and Probabilistic reasoning: Quantifying**

**uncertainty: Acting under uncertainty Basic probability notation, Inference using full joint distributions, independence, Bayes’ rule and its use, fuzzy logic.**

**What is Knowledge Representation?**

Humans are best at understanding, reasoning, and interpreting knowledge.

Human knows things, which is knowledge and as per their knowledge they perform various actions in the real world.

But how machines do all these things comes under knowledge representation and reasoning.

Hence we can describe Knowledge representation as following:

**PROPOSITIONAL LOGIC: A VERY SIMPLE LOGIC-**

We cover the syntax of propositional logic and its semantics—the way in which the truth of sentences is determined.

Then we look at entailment—the relation between a sentence and another sentence

that follows from it—and see how this leads to a simple algorithm for logical inference.

**The Wumpus World-**

The wumpus world is a cave consisting of rooms connected by passageways. Lurking somewhere in the cave is the terrible wumpus, a beast that eats anyone who enters its room. The wumpus can be shot by an agent, but the agent has only one arrow.

Some rooms contain bottomless pits that will trap anyone who wanders into these rooms (except for the wumpus, which is too big to fall in). The only mitigating feature of this bleak environment is the possibility of finding a heap of gold.

**Sensors**: The agent has five sensors, each of which gives a single bit of information:

– In the square containing the wumpus and in the directly (not diagonally) adjacent squares, the agent will perceive a Stench.

– In the squares directly adjacent to a pit, the agent will perceive a Breeze.

– In the square where the gold is, the agent will perceive a Glitter.

– When an agent walks into a wall, it will perceive a Bump.

– When the wumpus is killed, it emits a woeful Scream that can be perceived anywhere in the cave.

The percepts will be given to the agent program in the form of a list of five symbols; for example, if there is a stench and a breeze, but no glitter, bump, or scream, the agent program will get [Stench, Breeze, None, None, None].

**Syntax-**

The syntax of propositional logic defines the allowable sentences. The atomic sentences consist of a single proposition symbol.

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

We use symbols that start with an uppercase letter and may contain other

letters or subscripts, for example: P, Q, R, W1,3 and North.

The names are arbitrary but are often chosen to have some mnemonic value—we use W1,3 to stand for the proposition that the wumpus is in [1,3].

Complex sentences are constructed from simpler sentences, using parentheses and logical connectives. **There are five connectives in common use:**

**¬ (not).** A sentence such as ¬W1,3 is called the negation of W1,3.

**∧ (and).** A sentence whose main connective is ∧, such as W1,3 ∧ P3,1, is called a conjunction; its parts are the conjuncts.

**∨ (or).** A sentence using ∨, such as (W1,3∧P3,1)∨W2,2, is a disjunction of the disjuncts (W1,3 ∧ P3,1) and W2,2.

**⇒ (implies) or ->.** A sentence such as (W1,3∧P3,1) ⇒ ¬W2,2 is called an implication (or conditional). Its premise or antecedent is (W1,3 ∧P3,1), and its conclusion or consequent is ¬W2,2. Implications are also known as rules or **if–then statements.**

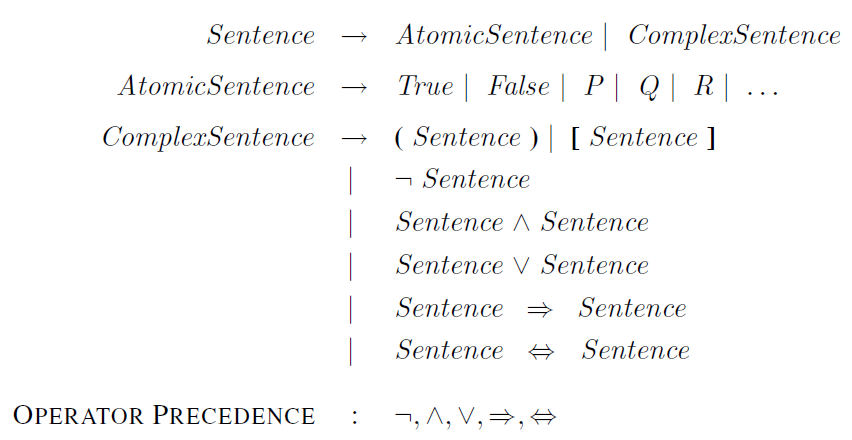
**Example**

**If it is raining, then the street is wet.**

**Let P= It is raining, and Q= Street is wet, so it is represented as P → Q**

**⇔ (if and only if).** The sentence W1,3 ⇔ ¬W2,2 is a biconditional. Some other books write this as ≡.

Example- P= I am breathing, Q= I am alive, it can be represented as P ⇔ Q.



For complex sentences, we have five rules, which hold for any sub-sentences P and Q in any model m (here “iff” means “if and only if”):

• ¬P is true iff P is false in m.

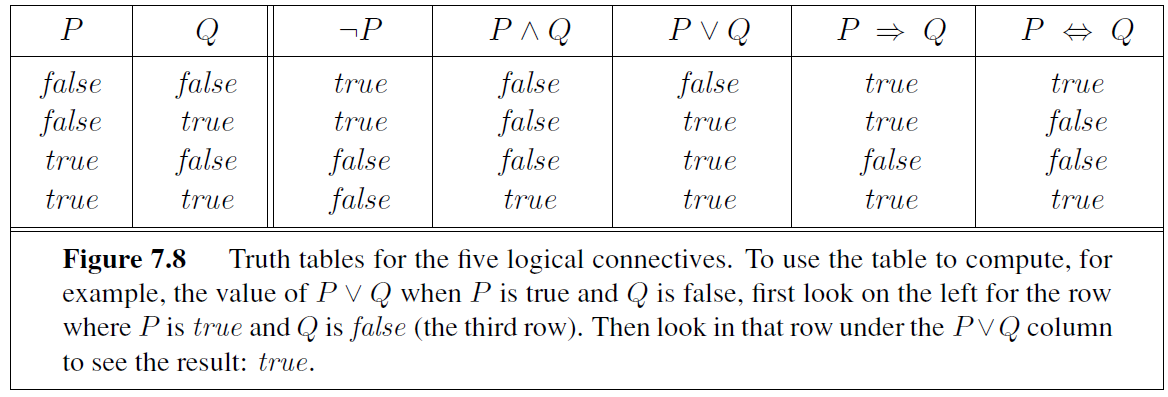
• P ∧ Q is true iff both P and Q are true in m.

• P ∨ Q is true iff either P or Q is true in m.

• P ⇒ Q is true unless P is true and Q is false in m.

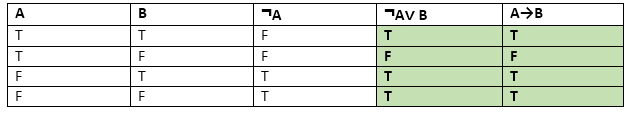
• P ⇔ Q is true iff P and Q are both true or both false in m.

The rules can also be expressed with truth tables that specify the truth value of a complex sentence for each possible assignment of truth values to its components



### Logical equivalence: Logical equivalence is one of the features of propositional logic. Two propositions are said to be logically equivalent if and only if the columns in the truth table are identical to each other.

Let's take two propositions A and B, so for logical equivalence, we can write it as A⇔B. In below truth table we can see that column for ¬A∨ B and A→B, are identical hence A is Equivalent to B



### Properties of Operators:

* **Commutativity:**
  + P∧ Q= Q ∧ P, or
  + P ∨ Q = Q ∨ P.
* **Associativity:**
  + (P ∧ Q) ∧ R= P ∧ (Q ∧ R),
  + (P ∨ Q) ∨ R= P ∨ (Q ∨ R)
* **Identity element:**
  + P ∧ True = P,
  + P ∨ True= True.
* **Distributive:**
  + P∧ (Q ∨ R) = (P ∧ Q) ∨ (P ∧ R).
  + P ∨ (Q ∧ R) = (P ∨ Q) ∧ (P ∨ R).
* **DE Morgan's Law:**
  + ¬ (P ∧ Q) = (¬P) ∨ (¬Q)
  + ¬ (P ∨ Q) = (¬ P) ∧ (¬Q).

|  |  |
| --- | --- |
|  |  |

* **Double-negation elimination:**
  + ¬ (¬P) = P.

### 

### Limitations of Propositional logic:

* We cannot represent relations like ALL, some, or none with propositional logic. Example:
  1. **All the girls are intelligent.**
  2. **Some apples are sweet.**
* Propositional logic has limited expressive power.
* In propositional logic, we cannot describe statements in terms of their properties or logical relationships.

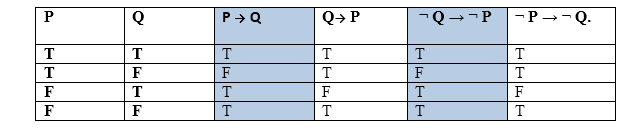
Inference and proofs-

This section covers inference rules that can be applied to derive a proof—a chain of conclusions that leads to the desired goal.

Following are some terminologies related to inference rules:

1. **Implication:** It is one of the logical connectives which can be represented as P → Q. It is a Boolean expression.
2. **Converse:** The converse of implication, which means the right-hand side proposition goes to the left-hand side and vice-versa. It can be written as Q → P.
3. **Contrapositive:** The negation of converse is termed as contrapositive, and it can be represented as ¬ Q → ¬ P.
4. **Inverse:** The negation of implication is called inverse. It can be represented as - ¬ P → ¬ Q.

From the above term some of the compound statements are equivalent to each other, which we can prove using truth table:



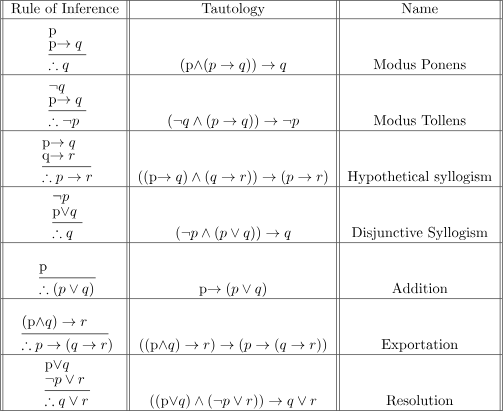
Hence from the above truth table, we can prove that P → Q is equivalent to ¬ Q → ¬ P, and Q→ P is equivalent to ¬ P → ¬ Q.

## 

## Types of Inference rules:

Refefence

1. <https://www.javatpoint.com/rules-of-inference-in-artificial-intelligence>
2. <https://www.geeksforgeeks.org/mathematical-logic-rules-inference/?ref=lbp>



syllogism

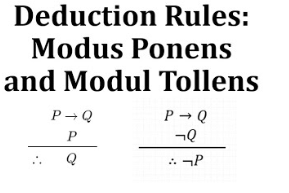
ˈसिलजिज़म्

a form of arguing in which two statements are used to prove that a third statement is true, for example, ‘All humans are mortal; I am a human, therefore I am mortal.’

1. The best-known rule is called **Modus Ponens** (Latin for

mode that affirms) and is written-

One of the most essential laws of inference is the Modus Ponens rule, which asserts that if P and P → Q are both true, we can infer that Q will be true as well. It's written like this:



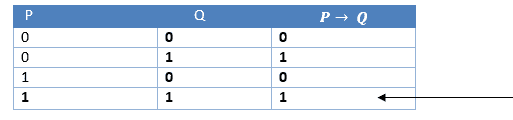
Example:

Statement-1: "If I am sleepy then I go to bed" ==> P → Q

Statement-2: ""I am sleepy" ==> P"

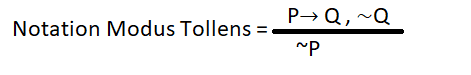
Conclusion: "I go to bed." ==> Q.

Hence, we can say that, if P → Q is true and P is true then Q will be true.



2. Modus Tollens:

According to the Modus Tollens rule if P→ Q is true and ¬ Q is true, then ¬ P will also true. It can be represented as:

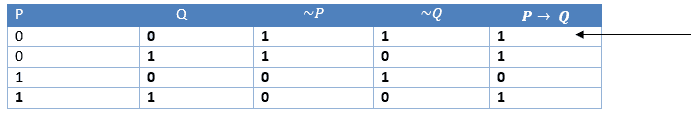


Example:

Statement-1: "If I am sleepy then I go to bed" ==> P→ Q

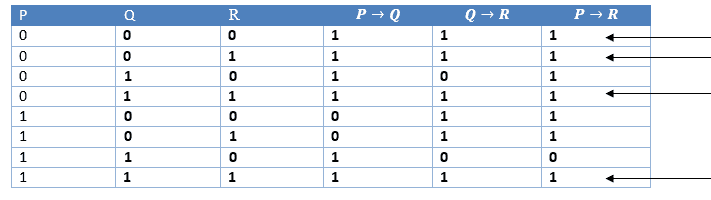
Statement-2: "I do not go to the bed."==> ~Q

Statement-3: Which infers that "I am not sleepy" => ~P



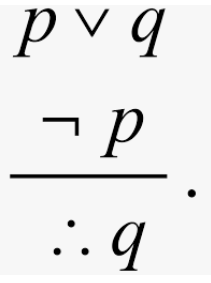
### 3. Hypothetical Syllogism:

According to the Hypothetical Syllogism rule if P→R is true whenever P→Q is true, and Q→R is true. It can be represented as the following notation:  
Example:  
Statement-1: Statement-1: If you have my home key then you can unlock my home. P→Q  
Statement-2: Statement-2: If you can unlock my home then you can take my money. Q→R  
Statement-3: Conclusion: If you have my home key then you can take my money. P→R



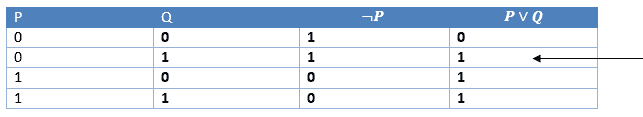
### 4. Disjunctive Syllogism:

According to the Disjunctive syllogism rule if P∨Q is true, and ¬P is true, then Q will be true. It can be represented as:



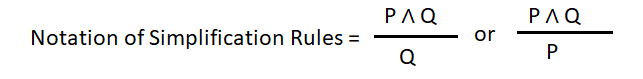
Example:  
Statement-1:Today is Sunday or Monday. ==>P∨Q  
Statement-2:Today is not Sunday. ==> ¬P  
Conclusion: Today is Monday. ==> Q

#### Proof by Truth table:



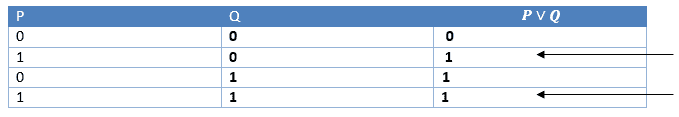
### 5. Addition:

According to the Addition rule which is one of the common inference rule, If P is true, then P∨Q will be true.



Example:  
Statement-1: I have a vanilla ice-cream. ==> P  
Statement-2: I have Chocolate ice-cream.  
Conclusion: I have vanilla or chocolate ice-cream. ==> (P∨Q)

#### Proof by Truth table:



Also called **And-Elimination**,

### 6. Simplification:

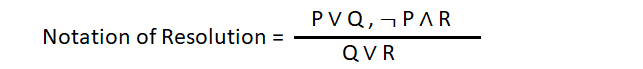
According to the simplification rule if P∧ Q is true, then Q or P will also be true. It can be represented as:

#### Proof by Truth table:Rules of Inference in AI12 in Artificial Intelligence (AI)

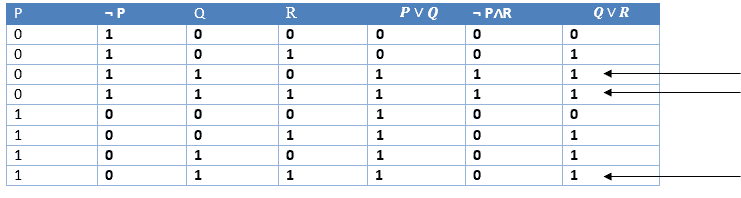
### 

### 7. Resolution:

According to the Resolution rule if P∨Q and ¬ P∧R is true, then Q∨R will also be true. It can be represented as



#### Proof by Truth table:



Let’s see how **Rules of Inference** can be used to deduce conclusions from given arguments or check the validity of a given argument.

**Example :**

Show that the hypotheses “It is not sunny this afternoon and it is colder than yesterday”, “We will go swimming only if it is sunny”, “If we do not go swimming, then we will take a canoe trip”, and “If we take a canoe trip, then we will be home by sunset” lead to the conclusion “We will be home by sunset”.

The first step is to identify propositions and use propositional variables to represent them.

|  |  |
| --- | --- |
| p-“It is sunny this afternoon”  q- “It is colder than yesterday”  r- “We will go swimming”  s-“We will take a canoe trip”  t- “We will be home by sunset”  The hypotheses are –  and s->t |  |

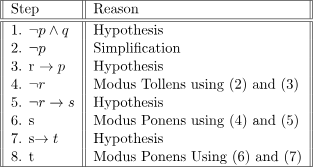
The conclusion is = t

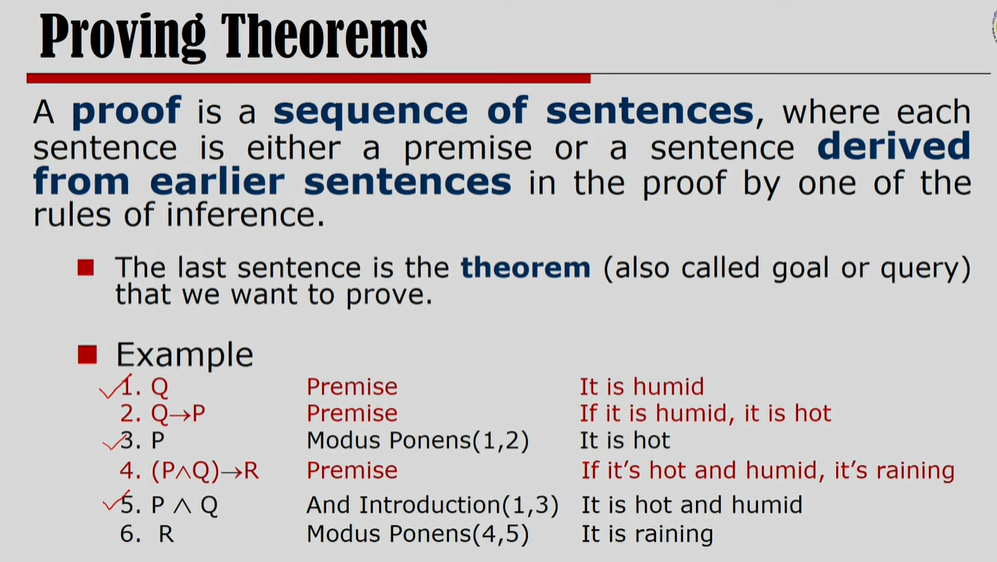
To deduce the conclusion we must use Rules of Inference to construct a proof

and s->t

The conclusion is = t

using the given hypotheses.





<https://www.geeksforgeeks.org/difference-between-propositional-logic-and-predicate-logic/>

Questions-

Prove equivalence-

1. DeMorgan’s law
2. Implies equivalence to contrapositive
3. Converse is equivalent to inverse

Rules of Inference

Proof by resolution

|  |  |  |
| --- | --- | --- |
| Q1 unit 1 | A,b, - any one | 5 |
| Q2 unit 2 | A,b, - any one | 5 |
| Q3 unit 3 | A,b, - any one | 5 |
| Q4 unit 4 | A,b,c,d - any one | 5 |
| Q5 all units | Combination of 4 units. One question from each unit | 10 |
|  |  |  |
|  |  |  |