



Q.1 Solve the following with forward chaining or backward chaining or resolution (any one) use predicate logic as language of knowledge representation clearly specify the facts & inference rule used.

Q.1) Example 1:

- 1) Every child sees some witch. No witch has both a black cat & a pointed hat.
- 2) Every witch is good or bad.
- 3) Every child who sees any good witch gets candy.
- 4) Every witch that is bad has a black cat.
- 5) Every witch that is seen by any child has a pointed hat.
- 6) Prove: every child gets candy.

→ A) facts into FOL.

- 1)  $\exists x \forall y (\text{child}(x), \text{witch}(y) \rightarrow \text{sees}(x, y))$   
 $\neg \exists y (\text{witch}(y) \rightarrow \text{has}(y, \text{black cat}) \wedge \text{has}(y, \text{pointed hat}))$
- 2)  $\exists y (\text{witch}(y) \rightarrow \text{good}(y) \vee \text{bad}(y))$
- 3)  $\forall x (\text{sees}(x, y) \rightarrow (\text{witch}(y) \rightarrow \text{good}(y)) \rightarrow \text{get}(x, \text{candy}))$
- 4)  $\forall y ((\text{witch}(y) \rightarrow \text{bad}(y)) \rightarrow \text{has}(y, \text{black hat}))$
- 5)  $\forall y (\text{sees}(x, y) \rightarrow \text{has}(y, \text{pointed hat}))$

B) FOL into CNF

- 1)  $\exists x \forall y (\text{child}(x), \text{witch}(y) \rightarrow \text{sees}(x, y))$   
 $\rightarrow \neg \exists y (\text{witch}(y) \rightarrow \text{has}(y, \text{black hat}))$   
 $\rightarrow \neg \exists y (\text{witch}(y) \rightarrow \text{has}(y, \text{pointed hat}))$

2)  $\forall y \ [ \text{witch}(y) \rightarrow \text{good}(y) ]$

3)  $\forall y \ [ \text{witch}(y) \rightarrow \text{bad}(y) ]$

3)  $\exists x \ [ (\text{sees}(x, y) \rightarrow \text{witch}(y) \rightarrow \text{good}(y)) \rightarrow \text{gets}(x, \text{candy}) ]$

$\Rightarrow \exists x \ [ (\text{sees}(x, \text{good}) \rightarrow \text{gets}(x, \text{candy})) ]$

4)  $\exists y \ [ \text{bad}(y) \rightarrow \text{has}(y, \text{black hats}) ]$

5)  $\exists y \ [ \text{seen}(x, y) \rightarrow \text{has}(y, \text{pointed hat}) ]$

$\Rightarrow \forall y \ [ \text{seen}(x, y) \rightarrow \text{has}(y, \text{black hat}) ]$

c)

$\text{sees}(x, y)$

$\text{witch}(y) \vee \text{sees}(x, y)$

$\{\text{good} \vee \text{bad}\}$

$\neg \text{seen}(x, \text{good}) \wedge \text{seen}(x, \text{bad})$

$\text{has}(y, z)$

$\{y/\text{good} \vee \text{bad}\}$

$\{z/\text{black hat} \vee$

$\text{pointed hat}\}$

$\text{seen}(x, \text{good}) \vee \text{seen}(x, \text{bad})$

$\text{has}(\text{good}, \text{pointed}$

$\text{hats} \vee \text{get}(x, \text{candy})$

$\text{seen}(x, \text{good}) \vee \text{has}(\text{good},$

$\text{pointed hat}) \vee \text{get}(x,$

$\text{candy})$

$\text{seen}(x, \text{good}) \vee$

$\text{get}(x, \text{candy})$

$\text{get}(x, \text{candy})$

$\text{get}(x, \text{candy})$

2) Example 2:

- 1) Every boy or girl is a child.
- 2) Every child gets a doll or a train or a lump of coal.
- 3) No boy gets any doll.
- 4) Every child who is bad gets any lump of coal.
- 5) No child gets a train.
- 6) Ram gets lump of coal.
- 7) Prove: Ram is bad.



- 1)  $\forall x (\text{boy}(x) \text{ or } \text{girl}(x) \rightarrow \text{child}(x))$
  - 2)  $\forall y (\text{child}(y) \rightarrow \text{gets}(y, \text{doll}) \text{ or } \text{gets}(y, \text{train}) \text{ or } \text{gets}(y, \text{coal}))$
  - 3)  $\forall w (\text{boy}(w) \rightarrow \neg \text{gets}(w, \text{doll}))$
  - 4) For all  $z (\text{child}(z) \text{ and } \text{bad}(z) \rightarrow \text{gets}(z, \text{coal}))$   
 $\forall y \text{ child}(y) \rightarrow \neg \text{gets}(y, \text{train})$
  - 5)  $\text{child}(\text{ram}) \rightarrow \text{gets}(\text{ram}, \text{coal})$
- To prove  $(\text{child}(\text{ram}) \rightarrow \text{bad}(\text{ram}))$

CNF clauses

- 1)  $\neg \text{boy}(x) \text{ or } \text{child}(x)$
- 2)  $\neg \text{girl}(x) \text{ or } \text{child}(x)$
- 3)  $\neg \text{child}(y) \text{ or } \text{gets}(y, \text{doll}) \text{ or } \text{gets}(y, \text{train}) \text{ or } \text{gets}(y, \text{coal})$
- 4)  $\neg \text{boy}(w) \text{ or } \neg \text{gets}(w, \text{doll})$
- 5)  $\neg \text{child}(z) \text{ or } \neg \text{bad}(z) \text{ or } \text{gets}(z, \text{coal})$
- 6)  $\neg \text{child}(\text{ram}) \rightarrow \text{gets}(\text{ram}, \text{coal})$
- 7)  $\text{bad}(\text{ram})$

Resolution

4) ! child (z) or ! bad (z) or get (z, coal)  
6) bad (ram)

7) ! child (ram) or gets (ram, coal)

Substituting z by ram

1) (a) ! boy (x) or child (x)  
boy (ram)

2) child ram (substituting x by ram)

7) ! child (ram) or gets (ram, coal)

8) child (ram)

9) gets (ram, coal)

2) ! child (y) (or gets (y, doll) or gets (y, train) or  
gets (y, coal))

8) child (ram)

10) gets (ram, doll) or gets (ram, train) or gets  
(ram, coal)

(substituting y by ram)

9) gets (ram, coal)

10) gets (ram, doll) or gets (ram, train) or gets  
(ram, coal)

11) gets (ram, doll) or gets (ram, coal)

3) ! boy (w) or ! gets (w, doll)

5) boy (ram)

12) ! get (ram, doll) (substituting w by ram)

11) gets (ram, doll) or gets (ram, train)

12) ! gets (ram, doll)

13) gets (ram, coal)

6) <a> get (ram, coal)

13) gets (ram, coal)

Hence, bad (ram) is proved.

Q2) Differentiate between STRIPS and ADL

### STRIPS language

- 1) Only allow positive literals in the states.  
for eg. :- A valid sentence is STRIPS is expressed as  
 $\Rightarrow \text{Intelligent} \wedge \text{Beautiful}$

2) STRIPS stand for Standard Research Institute Description Language Problem solver.

3) makes use of closed world assumption (i.e.) unmentioned literals are false

4) We only can find ground literals in goals -  
for eg. :- Intelligent  $\wedge$  Beautiful

5) Goals are conjunctions  
for eg. :- (Intelligent  $\wedge$  Beautiful)

### ADL

- 1) Can support both positive & negative literals.  
for eg. :- same sentence is expressed as  $\Rightarrow \text{stupid} \wedge \neg \text{ugly}$

2) stands for Action

3) makes use of open world Assumption (i.e.) unmentioned literals are unknown.

4) We can find qualified variables in goal  
for eg. :-  $\exists x \text{At}(P1, x)$

$\wedge \text{At}(P2, x)$  is the goal of having  $P1 \wedge P2$  in the same place in the example of blocks

5) Goals may involves involve conjunctions of disjunctions for eg -  
(Intelligent  $\wedge$  (Beautiful  $\vee$  Rich))

6) Effects are conjunctions

7) Does not support equality.

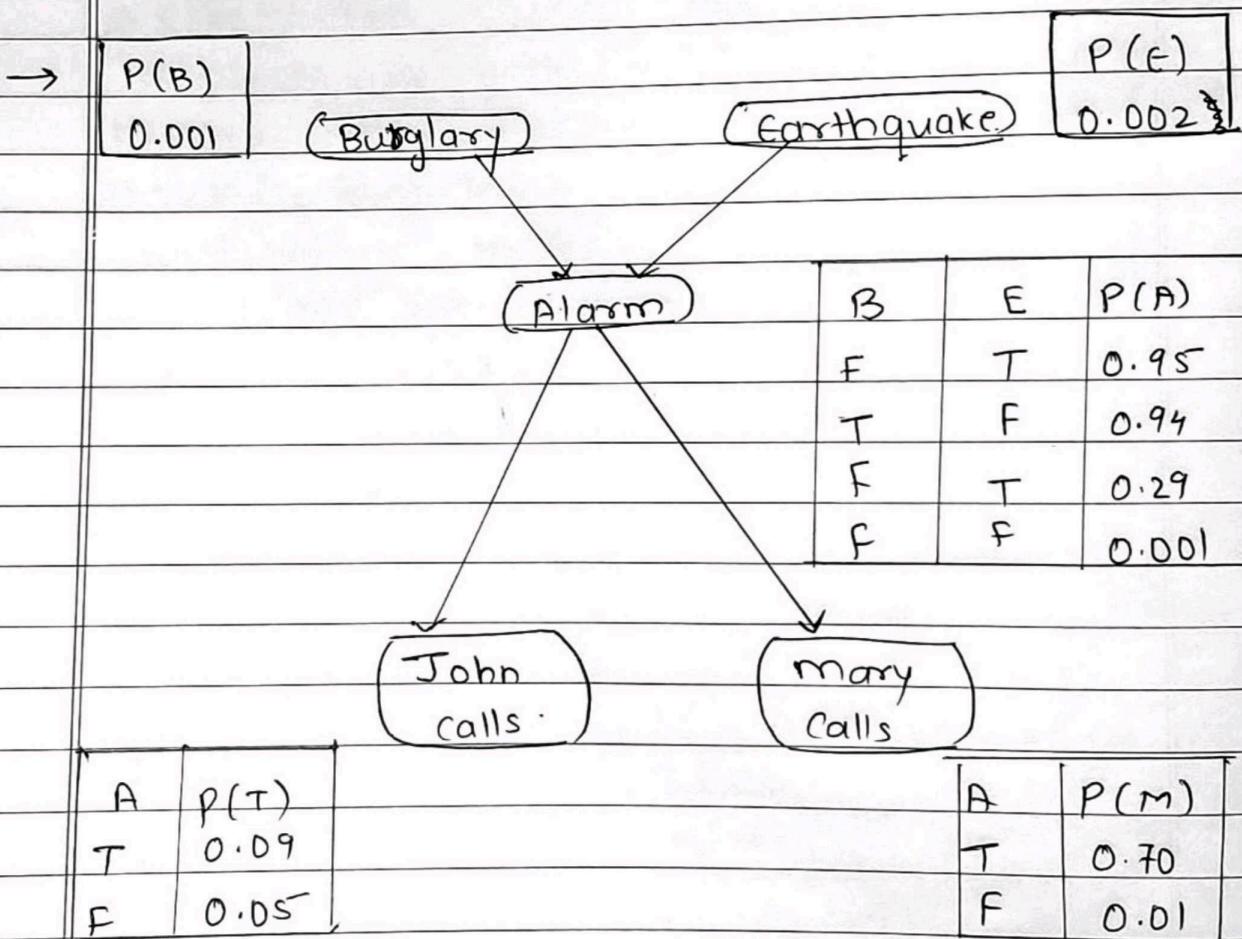
8) Does not have support for types.

6) conditional effects are allowed : when  $P : E$  means  $E$  is an effect only if  $P$  is satisfied.

7) Equality predicate ( $x = y$ ) is build in.

8) Support for types  
for eg: The variable  
 $P$ : person

Q4) You have two neighbors J and M, who have promised to call you at work when they hear the alarm. J always calls when he hears the alarm, but sometimes confused telephone ringing with alarms if calls them too. M likes loud music and sometimes misses the alarm together. Given the evidence of who has or has not called we would like to estimate the probability of burglary. Draw a Bayesian network for this domain with suitable probability table.



- ① The topology of the network indicates that - Burglary and earthquake affect the probability

- of the alarms going off.
- whether John and Mary call depends only on alarm.
  - They do not perceive any burglaries directly they do not notice minor earthquakes and they do not confer before calling.
- 2) Mary listening to loud music & John confusing phone ringing to sound of alarm can be read from network only implicitly as uncertainty associated to calling at work.
- 3) The probability actually summarize potentially infinite sets of circumstances.
- The alarm might fail to go off due to high humidity, power failure, dead battery, cut wires, a dead mouse stuck inside the bell, etc.
  - John and Mary might fail to call and report of alarm because they are out to lunch, on vacation, temporarily deaf, passing helicopter, etc.
- 4) The condition probability tables in PLN gives probability for values of random variables depending on combination of values for the parent nodes.
- 5) Each row must be sum. to 1, because entries represent exhaustive set of cases for variable.
- 6) All variables are Boolean.
- 7) In general, a table for a Boolean variable with  $k$  parents contains  $2^k$  independently specified probabilities.

8) A variable with no parents has only one row, representing prior probabilities of each possible value of the variable.

9) Every entry in full joint probability distribution can be calculated from information in Bayesian network.

10) A generic entry in joint distribution is probability of a conjunction of particular assignments to each variable  $P(x_1 = x_1 \wedge \dots \wedge x_n = x_n)$  abbreviated as  $P(x_1, \dots, x_n)$

11) The value of this entry is  $P(x_1, \dots, x_n) = \prod_{i=1}^n P(s_i | \text{parents}(x_i))$ , where  $s_i(x_i)$  denotes the specific values of the variables  $\text{parents}(x_i)$

$$= P(j|a) P(m|a) P(g|ublue) P(u|b) e(|ue)$$

$$= 0.09 \times 0.07 \times 0.001 \times 0.999 \times 0.998$$

$$= 0.000628$$

12) Bayesian network.

