Assignment No:-2.

Name :- Rutuja Kamlakar Sawant

Roll no :- 57

Class :- BE/IT

Sem :- VII

Sub :- IS Lab.

DOP	DOC	Remark	Sign

Q1. 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.

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 ~~which~~ witch that is 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) ∃xAy (child (x), witch (y) → sees (x,y))
   ¬∃y (witch (y) → has (y, black cat) ∧ has (y, pointed hat).
2) ∃y (witch (y) → good (y) v bad (y))
3) Ex ((sees (x,y) → (witch (y) → good (y)) → get (x, candy)
4) Ey ((witch (y) → bad (y)) → has (y → black hat))
5) ey (sees (x,y) → has (y, pointed hat)

B) FOL into CNF

1) ∃ xAy (child (x), witch (y) → sees (x,y)).

→ ¬∃y, (witch (y) → has (y, black hat).

→ ¬∃y (witch(y) → has (y, pointed hat))

2) ∀y (witch(y) → good(y))

3) ∀y (witch(y) → bad(y))

3) ∃x [(sees (x, y) → witch(y) → good(y)) → gets (x and y)]

→ ∃x [(sees [x, good (y) → gets (x and y)].

4) ∃y [bad(y) → has (y, black. hat)].

5) ∃y [seen (x, y) → has (y, pointed hat)]

→ ¬∀y [seen(x, y) → has (y, black hat)]

c)

sees(x, y) — witch(y) v sees (x, y) {good v badly }

¬seen(x, good) ∧ sees (x, ba). — has (y, z) {y/good v bad} {z/black cat v pointed hat)}

seen(x, good) v seen(x, bad) — has good pointed hats v get (x candy)

seen(x, good) v has (good, pointed hats) v gets (x, candy) — seen(x, god) v gets (x candy)

gets (x, candy) — gets (x 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 a 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) ∀ x (boy (x) or girl (x) → chid (x))
2) ∀ y (chid (y) → gets (y, doll) or gets (y, train) or gets (y, coal).
3) ∀ w (boy (w) → ! gets (w, doll))
4) For all z (child (z) and bad (z)) → gets (z, coal).
∀y child (y) → ! gets (y, tram).
5) child (ram) → gets (ram, coal).
To prove (child ram) → bad (ram)

CNF clauses
1) ! boy (x) or child (x).
! girl (x) or child (x).
2) ! child (y) or gets (y, doll) or gets (y ; train) or gets (y, coal).
3) ! boy (w) or ! gets (w, doll)
4) ! child (z) or ! bad (z) or gets (z, coal)
5) ! child (ram) → gets (room, coal)
6) bad (ram)

Resolution :

4) !child (z) or ! bad (z) gets(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).

8) 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 (or gets (y, doll) or gets (y, train 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> gets (ram, coal).

13) gets (ram, coal)

hence bad (ram) is proved.

Q2. Differentiate between STRIPS and ADL.

STRIPS language	ADL.
1) Only allow positive literals in the states for eg: A valid sentence is STRIP is expressed as ⇒ Intelligent ∧ Beautiful	1) Can support both positive & negative literals. For eg :- Some sentence is expressed as ⇒ stupid ∧ ¬ugly.
2) STRIPS stand for standard research Institute problem solver.	2) Stands for Action Description language.
3) Makes use of closed world assumption (i.e.) an mentioned literals are false.	3) Makes use of open world Assumption (i.e) unmentioned literals are unknown.
4) We only can find ground literal in goals For eg; Intelligent ∧ Beautiful.	4) We can find qualified variable in goal. For eg: $\exists x \; At(P_1, x) \wedge At(P_2, x)$ is the goal of having P1 & P2 in the some place in the example of blocks.
5) Goals are conjuctions For eg :- Intelligent ∧ Beautiful)	5) Goals may involve conjucations & disjunctions For eg :- (Intelligent ∧ (Beautiful ∧ Ric)).

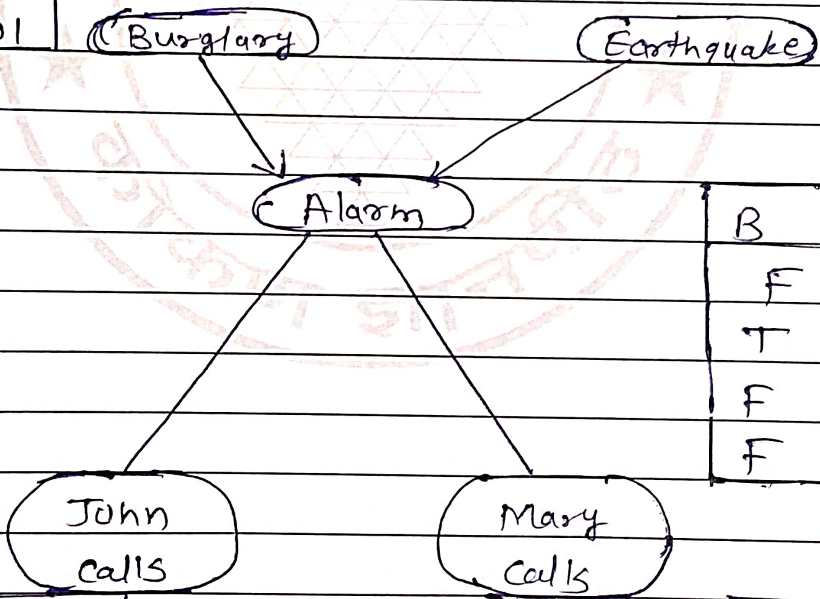
6) Effects are conjucations	6) Conditional effects are allowed: when P: E means E is an effect only if P is satisfied
7) Does not support equality.	7) Equality predicate (x = y) is build in.
8) Does not have support for types	8) suppor for types for eg: The variable P: person.

Q4. you have too neighbors J and M, who have promised to call you at work when they here the alarm J always calls when he hears the alarm, but sometimes confused telephone ringing with alarms & calls then 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 Byes'an networks for this domain with suitable probability table.

→

P(B)
0.001

P(E)
0.002



B	E	P(A
F	T	0.95
T	F	0.94
F	~~F~~T	0.29
F	F	0.001

A	P(J)
T	0.09
F	0.05

A	P(M
T	0.90
F	0.01

① The topology of the network indicates that - Burglary and earthquake affect the probability of the alarm going off.

-Wheather John and Mary call depends only on alarm
-They do not percieve any burglaries directly they do not notice minor earthquakes and they do not confer befor calling.

2) Mary listening to loud music & John confusing phone ringing to sound of alarm can be read from network only implicity as uncertainty associated to calling at work.

3) The probability actually summarize potentially infinite set of circumstance.
-The alarm might fail to go off due to high humidity, power faliure dead battery, cut wires a dead mouse stuck inside the bell, etc.
-John and mary might fail to call report & alarm because they are out to lunch on vacation temporarily deaf, passing helicopter etc.

4) The condition probability tables in n/w gives probability for values of random variable depeding on combination of values for the parent nodes.

5) Each row must be sum to 1 because entires represent exhustive set of cases of variable.

6) All Variable are Boolean.

7) In general a table for a boolean Variable with k parents contain 2 Independently specific probabilities.

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

10) A generic entry in joint distribution is probability of a conjucation of particular assignments to each variable $p(x_1 = x_1 \cap \ldots \cap \ldots x_n = x_n)$ abbreviated as $P(x_1, \ldots, x_n)$.

11) The value of this entry is $(P x_1, \ldots, x_n) = \pi_{i-1}, n P(1, \text{parents}(x_i))$, where parents $s(x_i)$ denotes the specific values of the variable parents (x_i).

$= P(j \wedge m) \cap a \cap \neg b \wedge \neg e)$

$= P(j|a)\, p(m|a)\, P(a|\neg b \wedge \neg e)\, P(\neg b) e(\neg e)$

$= 0.09 \times 0.07 \times 0.001 \times 0.999 \times 0.9998$

$= 0.00628$

12) Bayession network.