Assignment No: 2

Name :- Jyoti Mukesh Giri

Roll. No :- 19

Class :- BE-IT

Sem :- VII

Sub :- JS Lab

DOP	DOC	Marks	Sign

Assignment No : 1

Q1. Solve the following with forward chaining or backward chaining or resolution (only one) use predicate logic as language of knowledge representation. Clearly specify the facts and inference rule used.

Q1. Example 1 :-

1. Every child sees some witch No witch has both a black cat and a pointed hat
2. Every ~~with~~ 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. $\exists x \forall y$ (Child (x) , Witch (y) → sees (x,y))
   ~ $\exists y$ (witch (y) → has (y, black cat) ∧ has (y, pointed hat)
2. $\exists y$ (witch (y) → good (y) ∨ bad (y)
3. $\forall x$ (sees (x,y) → witch (y) → good (y)) → get (x, candy)
4. $\forall y$ ((witch (y) → bad (y) → has (y → black hat))
5. $\forall y$ (sees (x,y) → has (y, pointed hot)

B. FOL into CNF

1. $\exists x \forall y$ (Child (x), witch (y) → sees (x,y)
   → ~$\exists y$, (witch (y) → has (y, black hat)
   → ~$\exists y$ (with (y) → has (y, pointed hat)

Assignment

2. $\forall y$ (witch (y) $\rightarrow$ good (y))
$\forall y$ (witch (y) $\rightarrow$ bad (y))
3. Ex (sees (x,y) $\rightarrow$ witch (y) $\rightarrow$ good (y)) $\rightarrow$
gets (x, candy)
$\rightarrow$ Ex (<sees (x, good (y) $\rightarrow$ gets (x, candy))
4. $\forall y$ [band (y) $\rightarrow$ has (y, black hats)]
5. $\forall y$ [seen (x,y) $\rightarrow$ has (y, pointed hat)]
$\rightarrow$ ~$\forall y$ [seen (x,y) $\rightarrow$ has (y, black hat)]

C. sees (x,y)

witch (y) v sees (x,y)
{good v bad / y}

~seen (x, (good) ∧ sees (x, bad)

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 hat) v gets (x, candy)

seen (x, good) v gets (x, candy)

get (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. $\forall x$ (boy (x) or girl (x) $\longrightarrow$ Child (x))
2. $\forall y$ (child (y) $\longrightarrow$ gets (y, doll) or gets (y, train) or gets (y, coal))
3. $\forall w$ (boy (w) $\longrightarrow$! gets (w, doll))
4. For all z (Child (z) and bad (z)) $\longrightarrow$ gets (z, coal))) $\forall y$ Child (y) $\longrightarrow$! gets (y, train)
5. Child (ram) $\longrightarrow$ gets (ram, coal)

To prove (child (ram) $\longrightarrow$ 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) $\longrightarrow$ gets (ram, coal)
6. bad (ram)

Resolution

0. ! Child (z) or ! bad(z) or get (z, coal)
6. bad (ram)
7. ! Child (ram) or get (ram, coal)
subsituting z by ram

1. (a) ! boy (x) or child (x)
boy (ram)
Child ram (subsituting x by ram)
3 ! Child (ram) or get (ram, coal)
7 Child (ram)
8 get (ram, coal)
9. ! Child (y) (or get (y, doll) or get (y, train) or get (y, coal).
2. Child (ram).
10. get (ram, doll) or get (ram, train) or get (ram, coal)
~~11. get (ram, doll) or get (ram, train or get (~~
0. get (ram, coal)
10 get (ram, doll) or gets (ram, train) or get (ram, coal
11. gets (ram, doll) or gets (ram, coal)
13 ! boy (w) or ! gets (w, doll)
5 boy (ram)
12. ! get (ram, doll) (subsituting w by ram)
11. get (ram, doll) or get (ram, train)
12. get (ram, doll)
13. gets (ram, coal)
16 <a> get (ram, coal)
13 get (ram, coal)

Hence, bad (ram) is proved.

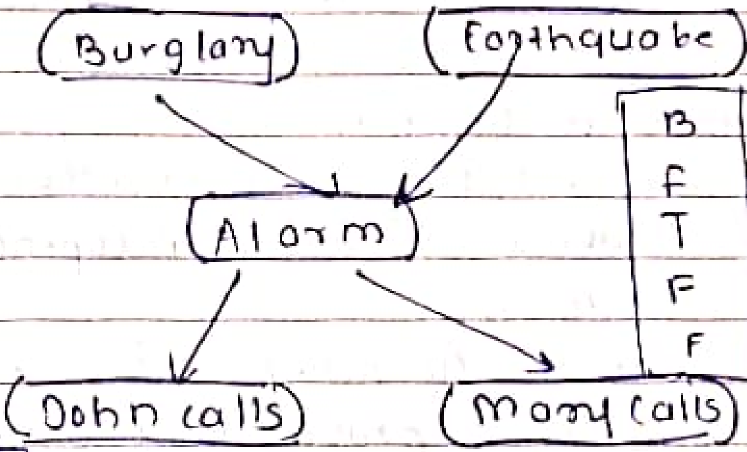
Q2 Differentiate between STRIPS and ADL

STRIPS	ADL
1. Only allow positive literals in the states. for eg:- A valid sentence is STRIPS is expressed as:- ⇒ Intelligent ∧ Beautiful.	1. Can support both positive and negative literals. For eg:- same sentence is expressed as ⇒ stupid ∧ ¬ugly.
2. STRIPS stand for Standard Research Institute Problem solver.	2. Stands for Action Description language.
3. Make use of closed world assumption (i.e) unmentioned literals are false.	3. Makes use of open world assumption (i.e) unmentioned literals are unknown.
4. We only can find ground literals in goals for eg:- Intelligent ∧ Beautiful	4. We can find qualified variables in goal for eg: $\exists x\ At(P1, x) \wedge At(P2, x)$ is the goal of having P1 and P2 in same place in example of blocks.
5. Goals are conjunctions for eg:- (Intelligent ∧ Beautiful).	5. Goals may involve conjunctions and disconjunctions. For eg:- (Intelligent ∧ (Beautiful ∧ Rich))

Q.6 a) You have two neighbours 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 alarm and 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 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	T	0.29
F	F	0.001

A	P(J)
T	0.09
F	0.05

A	P(M)
T	0.70
F	0.01

1. The topology of network indicates that Burglary and earthquake affect the probability of alarm going off.
2. Whether John and Mary call depends only on alarm.

2. mary listening to loud music and John watching 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 set of circumstances:
- John and mary might fail to call and report and alarm because they are out to lunch, on vacation, temporarily deaf, passing helicopter, etc.
4. The condition probability tables in network gives probability for values of random variables depending on combination of values of 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 contain 2^k independently specific probabilities.
8. A variable with no parents has only one row, representing prior probabilities of each possible value of variable.
9. A generic entry in joint distribution is probability of a conjunction of particular assignments to each variable $P(X_i = x_1 \wedge \ldots \wedge \ldots X_n = x_n)$ abbreviated as $(P\ x_1, \ldots, x_n)$
0. The value of entry is $P(x_1, \ldots x_n) = x_i - 1, \Pi P(1, \text{parents}(x_i))$, where parents$(x_i)$ denote the specific value of variables parents (x_i)

$- p(j \wedge m \wedge a \wedge a \wedge b \sim e)$

$= p(j|a)\ p(m|a)\ p(a|\sim b \cap \sim e)\ p(\sim b)\ e(\sim e)$

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

$= 0.000628$

18. Baysian network