Name :- Krutee Omprakash Lokiya

Class :- BEIT

Roll No :- 32

Sem :- VII th

Subject :- IS LAB

DOA	DOS	Marks	Signature

Assignment No :- 02

Page No.
Date

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.

Example 1 :-

1) Every child sees some witch no witch has both a black cat & a pointed hat.
2) Every witch is good witch ~~get~~ 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 following

1) ∃x ∀y (child (x), witch (y) → sees (x, y))
   ~∃y (witch (y) → has (y, black cat) ^ has (y, pointed hat))
2) ∃y (with (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) ∃x ∀y (child (x), witch (y) → sees (x, y))
→ ~ ∃y (witch (y) → has (y, black hat)
→ ~ ∃y (witch (y) → has (y, pointed hat)

Page No.
Date

2) ∀y (witch (y) → good (y))
   ∀y (witch(y) → bad (y))

3) Ɛx [(sees (x, y) → witch (y) → good(y)) → gets (x, candy)]

→ Ɛx [(sees) (x, good (y) → gets (x, candy)]

4) Ɛy [bad (y) → has (y, black hats)]

5) Ɛy [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 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)

gets (x, candy)   gets (x, candy)

Page No.	
Date	

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) → child (x))
2) ∀y (child (y) → gets (y, doll) or gets ~~(y, tra~~
~~y (~~ (y, train) or gets (y, coal).
3) ∀ w (boy (w) → ! gets (w, doll)).
4) for all z (child) (z) & bad (z)) → gets ~~(z,~~ (z, coal))
∀y child (y) → ! gets (y, train)
5) Child (ram) → gets (ram, coal)
to prove (child ram) → ~~to~~ 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 (ram, coal)
6) ! bad (ram).

Page No.
Date

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) ~~Hence~~ <a> get (ram, coal)
13) gets (ram, coal)
Hence, bad (ram) is proved.

Page No. | Date

Q.2) Differentiate between STRIPS & ADL.

STRIPS language	ADL language
(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 & negative literals. for eg: same sentence is expressed as ⇒ Stupid ∧ ¬ ugly.
(2) STRIPS stands for stand- and research institute problem solute.	(2) ADL stands for Action Description language.
(3) Makes use of closed world assumption (i.e) unmentioned literals are false.	(3) Make 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 & P2 in same place in eg blocks.
(5) Goals are conjuctions for eg:- (Intelligent ∧ Beautiful)	(5) Goals may involve conjuction & disjunction for eg:- (Intelligent ∧ (Beautiful ∧ Rich))

(5) effects are conjuction

(6) Conditional effects are allowed: when P:E means E is an effect only if P is satisfied

(6) Does not support equality.

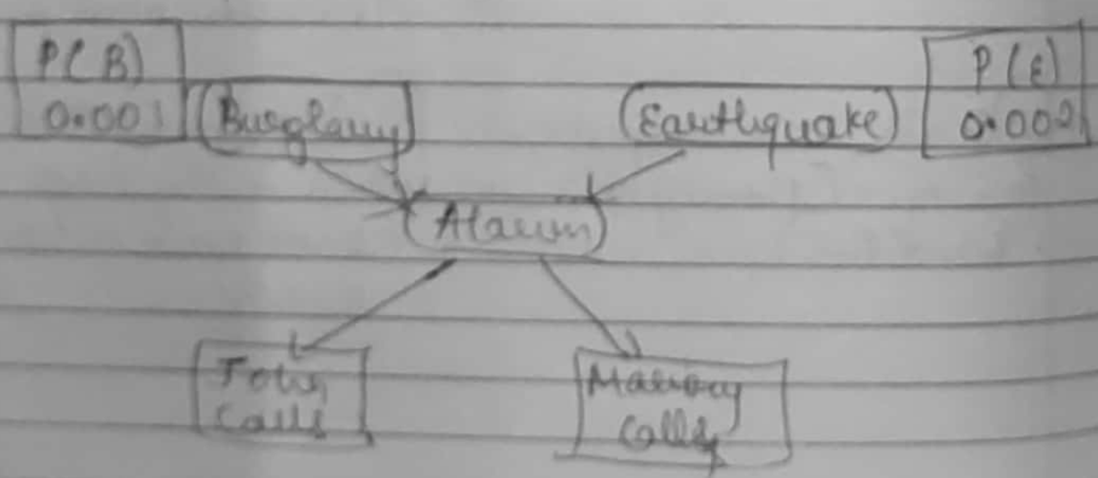
(7) Equality predicate (x=y) is build in.

(8) Does not have support for types.

(8) Support for types, for eg:- The variable p: person.

Q.4) You have two neighbours J & M, who have promised to call you at work when they here the alarm. J always call when he hoars the alarm, but sometimes confused telephone ringing with alarms together. Given the evidence of who has or has not called we would like to eliminate the probability of burglary. Draw a Bayesian Network for this domain with suitable probability table.

→



Page No.
Date

John calls | Marycalls

A	P(J)
T	0.09
F	0.05

A	P(M)
T	0.70
F	0.01

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

1) The topology of the network indicates that
- Burglary & earthquake affect the probability of the alarms going off.
- whether John & Mary call depends only on alarm.
- They do not perceive any burglaries directly they do not notice minor earthquakes & 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 implicity as uncertainty associated to calling at work.

3) The ~~ato~~ probability actualy summarize potentially infinite sets of circumstances.
- The alarm migh fail to go off due to high humidity, power failure, dead battery, cut wires, a dead mouse stuck inside the bell, etc.

Scanned with CamScanner

Page No.
Date

— John & 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 network gives probabilities 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 specific 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 $P(X_1 = x_1 \wedge \dots \wedge \dots X_n = x_n)$ abbreviated as $P(x_1, \dots, x_n) = \pi_{i-1,n} P(l, \text{parents}(x_i))$, where parents $s(x_i)$ denotes specific values of the variables parents (x_i)

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

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

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

= 0.000628

(12) Bayesian Network