SL.NO:CAA 156770

SRI SIVASUBRAMANIYA NADAR COLLEGE OF ENGINEERING

(An Autonomous Institution, Affiliated to Anna University, Chennai)
Rajiv Gandhi Salai (OMR), Kalavakkam – 603 110

THEORY EXAMINATIONS

Register Number	205001106		
Name of the Student	Siddharth. S		
Degree and Branch	B.E CSE	Semester	V
Subject code and Name	UCS1524 Logic programming.		
Assessment Test No.	1.	Date	22/9/22

Details of Marks Obtained

Part A		Part B				Part C			
Question No.	Marks	Question No.	(a) Marks	(b) Marks	Total Marks	Question No.	(a) Marks	(b) Marks	Total Marks
1	1	7			6	10			10
2	1								
3	1	8			6	11			
4	2					12			
5	1	9			6				
6	2					13			10
Total (A)	8	Total (B)			18	Total (C)			20

Grand Total (A+B+C)				Marks (in words)			
Signature of Faculty				46/50			

ssn

Part - A

1. Propositional logic forms have objects, predicate, quantifiers and functions. Predicate logic has objects, relations, functions and predicates. Each propositional logic Points to 1 context/variable. In predicate logic, functions are used to depict the logic.

Propositional :- $P \to Q$

Predicate :- $P(x,y) \to Q(x,y)$

2. $\neg P \wedge (P \to Q)$

$= \neg P \wedge (\neg P \vee Q)$

$= (\neg P \wedge \neg P) \vee (\neg P \wedge Q)$

$= \neg P$

P	Q	$\neg P$	$\neg P \vee Q$	$\neg\wedge$
0	0	1	1	1
0	1	1	1	1
1	0	0	0	0
1	1	0	1	0

3. (i) closed as all the variables are bounded due to quantifiers

(ii) open as u is not bounded by any quantifier.

4. A → there is rain ; B → climate is cool

$(A \to B) \wedge (B \to A)$

$= (\neg A \vee B) \wedge (\neg B \vee A)$

SSN

[illegible] is ~~[illegible]~~ restricted to Horn clause programming. The Horn clause states that there should be ~~atleast~~ atmost 1 ~~negative~~ positive literal.

This is need to break the big clauses by resolution using positive clauses. For resolution to occur, it needs horn clauses program to perform efficiently.

b. likes (adam, chocolates)
likes (adam, cake)
likes (vishnu, chocolates)
likes (vishnu, cake)

(i) likes (adam, x), likes (vishnu, x)

⇒ negate ⇒ ¬likes (adam, x) ¬likes (vishnu, x), answer (x)

¬likes (adam, x) ¬like (vishnu, x), answer (x) likes (adam, chocolate)

sub{x/chocolate}

likes (vishnu, chocolate) ¬like (vishnu, chocolate), answer (chocolate)

answer (chocolate)

¬likes(adam, x), ¬(vishnu, x), answer(x)   likes(adam, cake)

sub[x/cake]

likes(vishnu, cake)   ¬like(vishnu, cake), answer(cake)

answer(cake)

⇒ answer: {chocolate, cake}

(ii) likes(x, chocolate)

→ negate ⇒ ¬likes(x, chocolate), answer(x)

¬likes(x, chocolate), answer(x)   likes(adam, chocolate)

sub[x/adam]

answer(adam)

likes(vishnu, chocolate)

sub[x/vishnu]

answer(vishnu)

⇒ answer: {adam, vishnu}

SSN 5

Part-B

7. (i) $(A \vee (B \vee C)) \wedge (C \vee \neg A)$

(ii) $((B \wedge \neg A) \vee C)$

T.P:- (i) $\equiv$ (ii)

LHS:

$(A \vee (B \vee C)) \wedge (C \vee \neg A)$

(distribution)

$= (A \wedge C) \vee (B \wedge C) \vee (C \wedge C) \vee (A \wedge \neg A) \vee (B \wedge \neg A) \vee (C \wedge \neg A)$

$= (A \wedge C) \vee (B \wedge C) \vee C \vee F \vee (B \wedge \neg A) \vee (C \wedge \neg A)$ (inverse)

$= (B \wedge C) \vee C \vee (B \wedge \neg A) \vee (C \wedge (A \vee \neg A))$ (associative)

$= (B \wedge C) \vee (B \wedge \neg A) \vee C \vee (C \wedge T)$ (idempotent)

$= (B \wedge C) \vee (B \wedge \neg A) \vee C \vee C$ (identity)

$= (B \wedge C) \vee C \vee (B \wedge \neg A)$

$= (C \wedge (B \vee T)) \vee (B \wedge \neg A)$ (Associative)

$= (C \wedge T) \vee (B \wedge \neg A)$ (True law)

$= C \vee (B \wedge \neg A)$ (Idempotent)

$=$ (ii) $=$ RHS

$\Rightarrow$ Hence proved //

SSN 7

8 (i) $(P \to Q) \to R$

$= (\neg P \vee Q) \to R$ (Removing $\to$) (implication law)

$= \neg(\neg P \vee Q) \vee R$ ~~double negation, distributive~~ (demorgan's law)

$= (P \wedge \neg Q) \vee R$ (distribution)

$= (P \vee R) \wedge (\neg Q \vee R)$

6

(ii) $(R \to S) \to \neg(S \to Q)$

$= (\neg R \vee S) \to \neg(\neg S \vee Q)$ (implication law)

$= (\neg R \vee S) \to (S \wedge \neg Q)$ (implication)

$= \neg(\neg R \vee S) \vee (S \wedge \neg Q)$ (demorgan)

$= (R \wedge \neg S) \vee (S \wedge \neg Q)$

$= (R \vee S) \wedge (S \vee \neg S) \wedge (\neg Q \vee R) \wedge (\neg S \vee \neg Q)$ (distribution)

$= (R \vee S) \wedge T \wedge (\neg Q \vee R) \wedge (\neg Q \vee \neg S)$

$= (R \vee S) \wedge (\neg Q \vee R) \wedge (\neg Q \vee \neg S)$

9. Answer generation technique is a technique where an answer clause is assumed and on resolving the clauses, the answer to the question is brought in the answer clause instead of arriving at a contradiction / empty clause. It can fetch both single as well as multiple answers.

eg. likes (adam, apple)
likes (eve, apple)
likes (eve, cofee)

Let question be :-
(i) who likes coffee
(ii) who likes apples

(i) inference :- likes (x, coffee)
clause : ¬likes (x, coffee), answer (x) → The answer clause.

Resolving takes place,

¬likes (x, coffee), answer (x) likes (eve, coffee)

sub [x / eve]

answer (eve)

⇒ It can be infered that eve likes coffee

(ii) Inference :- likes (x, apples)
clause :- ¬ likes (x, apple), answer (x)

Resolving,

¬ likes(x, apples), answer(x)   likes (adam, apple)

sub[x/adam]

answer (adam)

sub [x/eve]   likes (eve, apple)

answer (eve)

⇒ It can be inferred that adam and eve likes apples.

Part-C

10. (i) $\exists x$ Kitten(x) $\wedge$ Has(Anu, x)

Eliminate existential, $\wedge$ can be replaced by ,

$\Rightarrow$ {Kitten(x), Has(Anu, x)}

(ii) $\forall x(\exists y$ Kitten$(y) \wedge$ Has$(x,y)) \rightarrow$ Animal_lover(x)

* Bring $\exists y$ Outside and eliminate (prenex, skolemisation)
* Eliminate universal quantifiers

$\Rightarrow$ Kitten$(y) \wedge$ Has$(x,y) \rightarrow$ Animal_lover(x)

* Convert to CNF

$\Rightarrow$ $\neg$ Kitten(y) $\vee$ $\neg$ Has(x,y) $\vee$ Animal_lover(x)

(iii) $\forall x$ Animal-lover$(x) \rightarrow \forall y$ Animal$(y) \rightarrow \neg$ Hurts(x,y)

* Eliminate universal quantifiers * Convert to CNF

$\Rightarrow$ $\neg$ Animal_lover(x) $\vee$ $\neg$ Animal(y) $\vee$ $\neg$ Hurts(x,y)

(iv) Hurst(Anu, Ram) $\vee$ Hurts(Sanjay, Ram)

(v) Puppy(Ram)

(vi) $\forall x$ Puppy$(x) \rightarrow$ Animal(x)

* Eliminate universal quantifiers * Convert to CNF

$\Rightarrow$ $\neg$ Puppy(x) $\vee$ Animal(x)

(vii) Inference $\Rightarrow$ Hurst(Sanjay, Ram)

$\Rightarrow$ clause = $\neg$ Hurts(Sanjay, Ram)

F = {Kitten(x), Has(Anu, x), [$\neg$ Kitten(y) $\vee$ $\neg$ Has(x,y) $\vee$ Animal lover(x)], [$\neg$ Animal_lover(x) $\vee$ $\neg$ Animal(y) $\vee$ $\neg$ Hurts(x,y)], [Hurts(Anu, Ram) $\vee$ Hurts(Sanjay, Ram)], Puppy(Ram), [$\neg$ Puppy(x) $\vee$ Animal(x)], [$\neg$ Hurts(Sanjay, Ram

Puppy(Ram) ¬puppy(x) ∨ Animal(x)

sub[x/Ram]

Animal(Ram) ¬Animal_lover(x) ∨ ¬Animal(y) ∨ ¬Hurts(x,y)

sub[y/Ram]

¬Animal_lover(x) ∨ ¬Hurts(x,Ram) Hurts(Anu,Ram) ∨ Hurts(Sanjay,Ram)

sub[x/Anu]

¬Animal_lover(Anu) ∨ Hurts(Sanjay, Ram) ¬Hurts(Sanjay, Ram)

¬Animal_lover(Anu) ¬Kitten(y) ∨ ¬Has(x,y) ∨ Animal lover(x)

Sub[x/Anu]

¬Kitten(y) ∨ ¬Has(Anu,y) Has(Anu,x)

sub[y/x]

¬Kitten(x) Kitten(x)

□ → Hence proved//

10

This resolution follows a N-resolution as each intermediate has atleast 1 negative clause.
N-resolution is used so that a positive clause might cancel the negative clause in each iteration until □ is obtained.

↑
empty clause

13. A → ToC book
B → AI book
C → reads
d → write exam

a) $A \wedge C$
b) $\cancel{\neg B \wedge C}$ $B \wedge \neg C$
c) $(A \wedge C) \rightarrow d$
d) $(A \wedge C \wedge d) \rightarrow (B \wedge C)$

13 A → reads ToC
B → reads AI
C → write exam

a) A
b) $\neg B$
c) $A \rightarrow C$
d) $A \wedge C \rightarrow B$

Horn clauses :-

a) $\cancel{C \wedge A}$ $1 \rightarrow A$
b) $B \rightarrow 0$
c) $A \rightarrow C$
d) $A \wedge C \rightarrow B$

$F = \{(1 \to A), (B \to 0), (A \to C), (A \land C) \to B\}$

checking satisfiability,

→ mark all one's

$F = \{(\underline{1} \to A), (B \to 0), (A \to C), (A \land C) \to B\}$

→ mark all clauses implied by the one's (mark A)

$F = \{(\underline{1} \to \underline{A}), (B \to 0), (\underline{A} \to C), (\underline{A} \land C \to B)\}$

While loop,

mark all clauses implied by the marked clauses,

i.e $A \to C$, as A is marked, mark C

$F = \{(\underline{1} \to \underline{A}), (B \to 0), (\underline{A} \to \underline{C}), (\underline{A} \land \underline{C} \to B)\}$

$\underline{A} \land \underline{C} \to B$, as A and C are marked, mark B

$F = \{(\underline{1} \to \underline{A}), (\underline{B} \to 0), (\underline{A} \to \underline{C}), (\underline{A} \land \underline{C} \to \underline{B})$

As $B \to 0$, B is marked, we have to mark the 0 (or)

Contradiction.

The horn satisfiability states that if $\perp$ (or) 0 is marked,

then unsatisfiable, else satisfiable.

⇒ $F = \{(\underline{1}, \underline{A}), (\underline{B} \to \underline{0}), (\underline{A} \to \underline{C}), (\underline{A} \land \underline{C} \to \underline{B})$

⇒ Unsatisfiable.

P Q R S Satisfiability che

A	B	C	¬B	A→C	A∧C	A∧C→B	P∧Q∧R∧S
0	0	0	1	1	0	1	0
0	0	1	1	1	0	1	0
0	1	0	0	1	0	1	0
0	1	1	0	1	0	1	0
1	0	0	1	0	0	1	0
1	0	1	1	1	1	0	0
1	1	0	0	0	0	1	0
1	1	1	0	1	1	1	0

As there are no 1's in P∧Q∧R∧S,

⇒ Unsatisfiable.

10