GALKINA, Daria (dg4217)

Imperial College London

Department of Computing
Academic Year **2020-2021**
Page created Tue Nov 3 23:15:05 GMT 2020

CATE

70051 rac101 2
t5 dg4217 v1

0 0 1 2 0 0 0 1 5 3 2 1

Electronic submission
Tue - 03 Nov 2020 18:40:16

dg4217

Exercise Information

Module: 70051 Introduction to Symbolic Artificial Intelligence (MSc AI)
Exercise: 2 (CW)
Title: Logic
FAO: Craven, Robert (rac101)

Issued: Tue - 20 Oct 2020
Due: Tue - 03 Nov 2020
Assessment: Individual
Submission: Electronic

Student Declaration - Version 1

- I declare that this final submitted version is my unaided work.

Signed: (electronic signature) Date: 2020-11-03 18:36:46

For Markers only: (circle appropriate grade)

GALKINA, Daria (dg4217)	01379079	t5	2020-11-03 18:36:46	**A* A B C D E F**

D. Galkina Intro to Symbolic AI. CW1. 03.11.20

1.i. p: Michel is either fulfilled or rich

q: Michel will live another 5 years

$$((\neg p) \rightarrow (\neg q))$$

ii. p: snowstorm arrives

q: Raheem will wear his boots

$$((p \rightarrow q) \land p)$$

iii. p: Arika is on set

q: Toshiro is on set

r: fuming will begin

s: caters cleared out

$$(((p \land q) \rightarrow r) \leftrightarrow s)$$

iv. p: Irad arrived

q: Sarah arrived

$$((p \lor (\neg q)) \land (\neg (p \land (\neg q))))$$

v. p: Herbert heard the performance

q: Anne-Sophèe heard the performance

r: Anne-Sophie answered her phone call

$$((\neg r) \rightarrow (\neg (p \land q)))$$

2. i. Satisfiability of a propositional formula A:

Propositional formula is satisfiable if there is some v, such that $h_v(A) = t$

where v is the atomic evaluation function &

h_v is the propositional evaluation function based on v.

ii ~~Where of~~ Logical Equivalence of two propositional formulas A & B:

Two propositional formulas are logically equivalent, if for every v, $h_v(A) = h_v(B)$, where v & h_v defined as above.

iii. $\neg A$ is satisfiable iff $\neg\neg A \not\equiv T$

so if $\neg\neg A \equiv T \therefore h_v(\neg\neg A) = t \rightarrow h_v(\neg A) = f$. & will always be false

for $\neg A$ to be satisfiable $h_v(\neg A)$ ~~it~~ needs to be true in some cases

however if $\neg\neg A \equiv T$, ~~then~~ $h_v(\neg A)$ will always be false.

$\therefore$ $\neg A$ is satisfiable iff $\neg\neg A \not\equiv T$

~~$\Rightarrow$~~ $h_v(\neg\neg A) = f \Rightarrow h_v(\neg A) = t$.

3. 2^3

p	q	r	(p	$\wedge$	$\neg q$	$\leftrightarrow$	$\neg$	($\neg r$	$\vee$	$\neg p$))	$\rightarrow$	($\neg\neg q$	$\rightarrow$	r)
t	t	t	t	f	f	f	t	f	f	f	t	t	t	t
t	t	f	t	f	f	t	f	t	t	f	f	t	f	f
t	f	t	t	t	t	t	t	f	f	f	t	f	t	t
t	f	f	t	t	t	f	f	t	t	f	t	f	t	f
f	t	t	f	f	f	t	f	f	t	t	t	t	t	t
f	t	f	f	f	f	t	f	t	t	t	f	t	f	f
f	f	t	f	f	t	t	f	f	t	t	t	f	t	t
f	f	f	f	f	t	t	f	t	t	t	t	f	t	f

As there are two cases when truth table found false it is not valid.,
(however it is satisfactory)

4. i a. CNF

b. CNF

c. CNF

d. CNF

e. DNF

f. neither. $\neg\neg p$ not literal (or DNF?)

g. CNF

h. DNF

ii. Property of Refutation - soundness & -completeness of resolution derivation
Let S be an CNF. $S \vdash_{res(PL)} \emptyset$ iff $S \models \bot$

It's important as based on that we can say S is satisfied iff $S \nvdash_{res(PL)} \emptyset$
meaning that if you can't derive $\emptyset$ from S by resolution derivation, then S is satisfiable ∴ the property helps define the satisfiability for S.

iii. a. $\{ \{p, s\}, \{q, r\}, \{\neg s, q\}, \{\neg p, \neg r, \neg s\} \}$

$\{ \{p, s\}, \{\neg p, \neg r, \neg s\} \}$ [q was pure]

$\{ \{p, s\} \}$ [$\neg r$ was pure]

(satisfiable)

b. $\{ \{\neg p, q, r\}, \{\neg q\}, \{p, r, q\}, \{\neg r, q\} \}$

$\{ \{\neg p, r\}, \{p, r\}, \{\neg r\} \}$ [unit propogation by unit clause $\{\neg q\}$]

$\{ \{\neg p\}, \{p\} \}$ [unit propogation by unit clause $\{\neg r\}$]

$\{ \{\} \}$ [unit propagation by unit clause $\{p\}$]

(unsatisfiable: $\emptyset$)

5. p: I'm going

q: You are going

r: Tara's going

$p \to \neg q$, $\neg q \to \neg r$, $r \vee \neg p$, $\neg p \to r$ therefore q.

CNF: $(p \to \neg q) \wedge (\neg q \to \neg r) \wedge (r \vee \neg p) \wedge (\neg p \to r) \wedge \{q\}$

Set $\{ \{\neg p, \neg q\}, \{q, \neg r\}, \{r, \neg p\}, \{p, r\}, \{q\} \}$.

Let's check if the clases where therefore $\neg q$ is satisfiable $\Rightarrow$

$\{ \{\neg p, \neg q\}, \{q, \neg r\}, \{r, \neg p\}, \{p, r\}, \{\neg q\} \}$

$\{ \{r, \neg q\}, \{q, \neg r\}, \{r\}, \{\neg q\} \}$ [resolve in all ways on p]

$\{ \{q\}, \{\neg q\} \}$ [unit propagation by unit clause $\{r\}$]

$\{ \{\} \}$ [unit propagation by unit clause $\{q\}$]

$\emptyset$ in set $\therefore$ unsatisfiable suggesting the original argument is Valid

extra

Let's see: $\{ \{\neg p, \neg q\}, \{q, \neg r\}, \{r, \neg p\}, \{p, r\}, \{q\} \}$

$\{ \{r, \neg q\}, \{q, \neg r\}, \{r\} \{q\} \}$

$\{ \{q\}, \{q\} \}$

$\therefore$ Satisfachable . [illegible]

6. i. L_1 - signature:

$C = \{andrea\}$

$P_1 = \{auntaunt\}$ auntaunt (x) : aunt's aunt

$P_2 = \{cupcake\}$ cupcake (x, y) : cupcake given by x to y

$\forall X (auntaunt(X) \rightarrow \exists Z(cupcake(X, Z) \wedge \neg(Z = andrea)))$

ii L_2 - signature

$P_1 = \{computer\}$ computer (x) : x is a computer

$P_2 = \{connected\}$ connected(x,y) : $\wedge \neg (X = Y)$ connect x to y

$\exists X \forall Y (computer(X) \wedge computer(Y) \wedge \neg(connected(Y, Y)) \rightarrow connected(X, Y))$

If computer is known:

$C = \{computer\}$

$P_1 = \{computers\}$

$P_2 = \{connected\}$

$\forall X (computers(X) \wedge \neg(X = computer) \wedge \neg(connected(X, X)) \rightarrow connected(computer, X))$

iii. L_3 - signature

$P_1 = \{paint\_K, paint\_PL, room\}$:

$P_2 = \{hang\}$ hang (x, y) : hang painting y in room z.

$\exists Z (\forall Y ((paint\_K(Y) \wedge room(Z) \wedge hang(Z, Y))) \rightarrow \forall X (paint\_PL(X) \rightarrow hang(Z, X)))$

If room is known then $C = \{room\}$

$P_1 = \{paint\_K, paint\_PL\}$

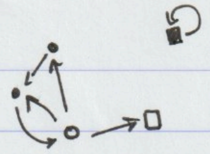
$P_2 = \{hang\}$

$\forall Y (paint\_K(Y) \wedge hang(room, Y)) \rightarrow \forall X (paint\_PL(X) \rightarrow hang(room, X))$.

IV. L_4 - signature:

$P_2 = \{\text{loves}\}$ $\text{loves}(x, y)$: x loves y.

$$\exists X \exists Y \neg(\text{loves}(X, Y)) \rightarrow \neg(\forall X \exists Y (\text{loves}(X, Y)))$$

7. 

i. $\forall X(a(k,X) \rightarrow \neg(X=j))$

for any X that has a directed arrow pointing at it ~~from~~ from k, X won't be j.

<u>false</u> : k only directs to j

ii. $c(l) \rightarrow \exists X(b(X) \land c(X) \land a(l,X))$

If l is a circular object than there is at least one X that's: circular, filled & l directs to it.

<u>true</u>: j & k are both circular & filled & k points.

iii. $\exists X \neg \exists Y(\neg(X=Y) \land a(X,Y))$

for any X, there is no Y ~~thach~~ that X & Y are not one object and X points to Y.

<u>false</u>: there are objects that point at other but themselves.

iv. $\forall X(\neg s(X) \rightarrow \exists Y(c(Y) \land b(Y) \land a(X,Y)))$

If X is not square then it connects to some object that's circular & filled.

<u>false</u>: j connects only to l (& it isn't filled)

v. $\forall X(\exists Y(\neg(X=Y) \land a(X,Y)) \rightarrow \exists Y(a(X,Y) \lor a(Y,X)))$

~~For all X, there is Y that's not X and X connects, if so,~~

for X there is some Y that X points to & it's not itself, and for those cases there is some y that either points to X, or X points to it & it's true for all X.

~~false: not the case for every case~~

<u>true</u>: if X connects to Y than it will connect to Y.

vi. $\forall X \forall Y(a(X,j) \land a(Y,j) \rightarrow (a(X,Y) \lor a(Y,X)))$

For all X & Y that connect to j, they have connetion between each other.

<u>true</u>: l & k connect to j & l connects to k.