VALDETTARO, Filippo (fv516)

Imperial College London

Department of Computing

Academic Year **2020-2021**

Page created Tue Nov 3 23:15:01 GMT 2020

CATE

70051 rac101 2

t5 fv516 v1

0012000014276

Electronic submission

Fri - 30 Oct 2020 14:17:07

fv516

Exercise Information

Module:	70051 Introduction to Symbolic Artificial Intelligence (MSc AI)	**Issued:**	Tue - 20 Oct 2020
Exercise:	2 (CW)	**Due:**	Tue - 03 Nov 2020
Title:	Logic	**Assessment:**	Individual
FAO:	Craven, Robert (rac101)	**Submission:**	Electronic

Student Declaration - Version 1

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

Signed: (electronic signature) Date: 2020-10-30 14:15:49

For Markers only: (circle appropriate grade)

VALDETTARO, Filippo (fv516)	01202766	t5	2020-10-30 14:15:49	**A* A B C D E F**

Coursework 1: Logic

Filippo Valdettaro

October 30, 2020

Question 1

(i) p: Michael is fulfiled
q: Michael is rich
r: Michael will live another five years
$((\neg\,(p \vee q)) \to (\neg r))$

(ii) p: The snowstorm arrives
q: Robin will wear his boots
r: I'm sure the snowstorm will arrive
$(((\neg p) \vee q) \wedge r)$

(iii) p: Akria is on set
q: Toshiro is on set
r: Filming will begin
s: Caterers have cleared out
$((p \wedge q) \to (r \leftrightarrow s))$

(iv) Slighlty ambiguous phrasing, assuming the "both" refers to Irad arriving and Sarah not arriving because there would be more natural ways to say that they can't both have arrived than the one given. Moreover, the punctuation suggests this is the correct interpretation.
p: Irad arrived
q: Sarah arrived
$((p \vee (\neg q)) \wedge (\neg(p \wedge (\neg q))))$

(v) p: Anne-Sophie answered her phone calls
q: Herbert heard the performances
r: Anne-Sophie heard the performances
$((\neg p) \to (\neg(q \wedge r)))$

Question 2

(i) A propositional formula A is defined to be satisfiable if and only if there is some atomic evaluation function v from which it is possible to construct a propositional evaluation function h_v that satisfies $h_v(A) = \mathbf{t}$.

(ii) Two propositional formulas A and B are logically equivalent if and only if for any atomic evaluation function v it is the case that the propositional evaluation function constructed from it, h_v, satisfies $h_v(A) = h_v(B)$. This relationship is denoted as $A \equiv B$.

(iii) Assume that $\neg\neg A \not\equiv \top$. It follows that there exists some propositional evaluation function h_v such that $h_v(\top) \neq h_v(\neg\neg A)$ and therefore $h_v(\neg\neg A) \neq \mathbf{t}$. A propositional evaluation function maps formulas to either $\mathbf{t}$ or $\mathbf{f}$ so it follows that $h_v(\neg\neg A) = \mathbf{f}$. Further, we use the property that for any formula B $h_v(\neg B) = \mathbf{t}$ iff $h_v(B) = \mathbf{f}$ (and therefore $h_v(\neg B) = \mathbf{f}$ iff $h_v(B) = \mathbf{t}$) in the case $B = \neg A$ to conclude that for such a h_v, $h_v(\neg A) = \mathbf{t}$ and that therefore $\neg A$ is satisfiable.
Assume next that $\neg A$ is satisfiable. It follows that for some h_v, $h_v(\neg A) = \mathbf{t}$ and, applying the property used earlier that for any formula B, $h_v(\neg B) = \mathbf{f}$ iff $h_v(B) = \mathbf{t}$ in the case $B = \neg A$, one concludes that $h_v(\neg\neg A) = \mathbf{f}$. However, for any h_v we must have $h_v(\top) = \mathbf{t}$ so it follows that for such a h_v, $h_v(\neg\neg A) \neq h_v(\top)$ and by the definition of logical equivalence we must have $\neg\neg A \not\equiv \top$.
Therefore, $\neg\neg A \not\equiv \top$ if and only if $\neg A$ is satisfiable.

Question 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	t	f	t	f	t	f	f	t	$\mathbf{t}$	t	f	t	t	t
t	t	f	t	f	f	t	t	f	t	f	t	f	t	$\mathbf{f}$	t	f	t	f	f
t	f	t	t	t	t	f	t	t	f	t	f	f	t	$\mathbf{t}$	f	t	f	t	f
t	f	f	t	t	t	f	f	f	t	f	t	f	t	$\mathbf{t}$	f	t	f	t	f
f	t	t	f	f	f	t	t	f	f	t	t	t	f	$\mathbf{t}$	t	f	t	t	t
f	t	f	f	f	f	t	t	f	t	f	t	t	f	$\mathbf{f}$	t	f	t	f	f
f	f	t	f	f	t	f	t	f	f	t	t	t	f	$\mathbf{t}$	f	t	f	t	t
f	f	f	f	f	t	f	t	f	t	f	t	t	f	$\mathbf{t}$	f	t	f	t	f

There are atomic evaluation formulas which cause the truth value of the whole formula to be false, namely when once applied to p, q, r they take the values $\mathbf{t}, \mathbf{t}, \mathbf{f}$ or $\mathbf{f}, \mathbf{t}, \mathbf{f}$, so the formula isn't valid.

Question 4

(i) (a) CNF
(b) Both CNF and DNF
(c) Neither CNF nor DNF
(d) Both CNF and DNF

(e) CNF

(f) Neither CNF nor DNF

(g) Both CNF and DNF

(h) Both CNF and DNF

(ii) The refutation-soundeness and -completeness property of a resolution derivation is that for any propositional formula in CNF, one can always carry out a resolution derivation that will enable one to determine whether the formula is satisfiable. In particular, the property of refutation-soundness and -completeness can be stated as $S \vdash_{res(PL)} \emptyset$ iff $S \models \bot$ for any S in CNF. Here, $\emptyset$ denotes the empty cause, representing $\bot$ in set-theoretic notation, and the theorem can be interpreted as claiming that for any such S, there is always some resolution derivation (denoted by $\vdash_{res(PL)}$) that will eventually result in an empty clause iff S is unsatisfiable (denoted by $S \models \bot$).

This is an important property because it allows one to systematically test for satisfiability of any propositional formula, as any propositional formula can be expressed in CNF and any CNF can be tackled systematically using resolution derivations. This property guarantees that such an approach will always eventually yield a conclusive and useful result on the satisfiability of any such formula.

(iii) (a) $\{\{p, s\}\{q, r\}, \{\neg s, q\}, \{\neg p, \neg r, \neg s\}\}$
$\implies \{\{p, s\}, \{\neg p, \neg r, \neg s\}\}$ [pure rule on q]
$\implies \{\{p, s\}\}$ [pure rule on $\neg r$]
$\implies \{\}$ [pure rule on s (or p)]

(b) $\{\{\neg p, q, r\}, \{\neg q\}, \{p, r, q\}, \{\neg r, q\}\}$
$\implies \{\{\neg p, r\}, \{p, r\}, \{\neg r\}\}$ [unit propagation by $\{\neg q\}$]
$\implies \{\{\neg p\}, \{p\}\}$ [unit propagation by $\{\neg r\}$]
$\implies \{\{\}\}$ [unit propagation by $\{p\}$]

Question 5

Start by translating the argument into propositional logic. Assigning propositions as follows:
p: I'm going
q: You're going
r: Tara's going
results in the argument being translated into: $(p \to \neg q), (\neg q \to \neg r), (r \vee \neg p), (r \vee p) \models q$ In terms of satisfiability, this can be reinterpreted as the following formula being unsatisfiable: $((p \to \neg q) \wedge (\neg q \to \neg r) \wedge (r \vee \neg p) \wedge (r \vee p)) \wedge \neg q$. Next, we can transform this expression into CNF by converting every $A \to B$ into the logically equivalent $\neg A \vee B$ resulting in the CNF formula:

$$(\neg p \vee \neg q) \wedge (q \vee \neg r) \wedge (r \vee \neg p) \wedge (r \vee p) \wedge \neg q$$

which using the set theoretic notation is $\{\{\neg p, \neg q\}, \{q, \neg r\}, \{r, \neg p\}, \{r, p\}, \{\neg q\}\}$.

Now we can apply DP:

$\{\{\neg p, \neg q\}, \{q, \neg r\}, \{r, \neg p\}, \{r, p\}, \{\neg q\}\}$
$\implies$ $\{\{\neg p\}, \{r, \neg p\}, \{r, p\}\}$ [unit propagation by $\{\neg q\}$]
$\implies$ $\{\{r\}\}$ [unit propagation by $\{\neg p\}$]
$\implies$ $\{\}$ [unit propagation by $\{r\}$]
and therefore the expression is satisfiable. This means that the argument is not valid.

Question 6

(i) Employing the signature $\mathcal{L} = (\mathcal{C}, \{\mathcal{P}_2\}, \{\})$ with $\mathcal{C} = \{andrea\}$, $\mathcal{P}_2 = \{aunt, cupcake\}$ ($aunt(A, B)$ being true iff A is B's aunt and $cupcake(A, B)$ being true iff A gave B a cupcake), the sentence becomes:

$$\forall X(aunt(andrea, X) \to (\exists Y(cupcake(X, Y) \land (Y \neq andrea))))$$

(ii) Employing the signature $\mathcal{L} = (\mathcal{C}, \{\mathcal{P}_1, \mathcal{P}_2\}, \{\})$ with $\mathcal{C} = \{\}$, $\mathcal{P}_1 = \{computer\}$, $\mathcal{P}_2 = \{connected\}$ ($computer(A)$ being true if A is a computer and $connected(A, B)$ being true if A is connected to B), the sentence becomes:

$$\forall X((\neg connected(X, X) \land computer(X)) \to \exists Y(connected(Y, X) \land computer(Y)))$$

(iii) We employ the signature $\mathcal{L} = (\mathcal{C}, \{\mathcal{P}_1\}, \{\mathcal{F}_1\})$ with $\mathcal{C} = british$, $\mathcal{P}_1 = \{kandinsky, klee\}$, $\mathcal{F}_1 = \{gallery, room\}$. Interpret $british$ to represent the British Gallery, $kandisky(A)$, $klee(A)$ to be true iff A is a painting by Kandinsky or Klee respectively. The function-symbol $gallery(A)$ maps the painting A to the gallery it's hanging in and $room(A)$ maps A to the room it's in. Then, the sentence becomes:

$$\begin{aligned}&\forall X((klee(X) \land (gallery(X) = british)) \to \\ &\quad \forall Y(kandinsky(Y) \land (gallery(X) = gallery(Y))) \to (room(X) = room(Y)))\end{aligned}$$

(iv) Employing the signature $\mathcal{L} = (\mathcal{C}, \{\mathcal{P}_2\}, \{\})$ with $\mathcal{C} = \{\}$, $\mathcal{P}_2 = \{loves\}$ (interpreted as $loves(X, Y)$ being true iff X loves Y), then the sentence can be expressed as:

$$(\exists X \forall Y \neg loves(X, Y)) \to \neg(\forall X \exists Y loves(X, Y))$$

Question 7

(i) **False**. Consider the assignment that maps X to j. Then $a(k, X)$ is true but $\neg X = j$ is false so the statement is false.

(ii) **True**. $c(l)$ is immediately true, so we need to show the consequent is also true. We can show it to be true by assigning X to j and the consequent must be true since $\varphi(j)$ is a black circle that is being pointed to by $\varphi(l)$ and therefore this assignment satisfies the conditions to make all conjuncts true.

(iii) **True**. Consider the assignment that maps X to the black square that points to and is pointed itself only. Then, there is no such Y that can make $(\neg(X = Y) \wedge a(X, Y))$ and therefore the negation of this statement is true for this particular X and any Y. Thus, the overall sentence is also true.

(iv) **False**. Consider the assignment that maps X to j. Then, the antecedent is true but for the consequent to be true, we need a black circle to be pointed to by j. j does not point to any such circle so the sentence must be false.

(v) **False**. Consider the assingment mapping X to k. The consequent is false because no object points at and is pointed to by k. The antecedent is true, however, because k has an outgoing arrow pointing to j. Therefore the sentence is false.

(vi) **False**. Two assignment make the statement false, the one mapping both X and Y to k and the one that maps both X and Y to l. In both cases, the antecedent is true (both k and l point to j) but the consequent is false, since neither k nor l have an arrow pointing to themselves.