Exercises for CS2013

Propositional Logic

1. Translate into propositional logic (abbreviating ‘it has rained’ as r, ‘it’s been cold’ as c and ‘the plant is dead’ as d). Use truth tables to determine whether one of these is logically equivalent to the other.
   a. If it has rained and it’s been cold then the plant is dead.
   b. If it has rained then either it hasn’t been cold or the plant is dead.

Answers.

   a. If it has rained and it’s been cold then the plant is dead. $(r \wedge c) \rightarrow d$
   b. If it has rained then either it hasn’t been cold or the plant is dead. $r \rightarrow (\neg c \vee d)$

$(r\wedge c)\rightarrow d$ is indeed equivalent to $r\rightarrow(\neg c\vee d)$. Both are only F if r and c are true while d is F.

2. *Determine the status of each of these formulas (contingent, tautologous, or contradictory). Prove your case by using truth tables. ($\oplus$ is the exclusive disjunction.)*
   a. $p\oplus\neg p$
   b. $p \oplus p$
   c. $p\oplus(p\vee\neg p)$
   d. $p\oplus(p\wedge\neg p)$

Answers.

a is a tautology. b is a contradiction. c and d are contingent.

3. *Express each formula using only (at most) the connectives listed. In each case use a truth table to prove the equivalence. (Note:* $\oplus$ is exclusive \`or\`)
   a. Formula: $p\rightarrow q$. Connectives: $\{\neg,\vee\}$.
   b. Formula: $p\oplus q$. Connectives: $\{\neg,\vee,\wedge\}$.
   c. Formula: $p\leftrightarrow q$. Connectives: $\{\rightarrow, \wedge\}$.
   d. Formula: $(p\rightarrow q) \wedge ((\neg p)\rightarrow q)$. Conn: $\{\neg,\vee\}$.
   e. Formula: $\neg p$. Conn: { | } (the Sheffer stroke).

Answers. (Other answers possible)

   a. Formula $p\rightarrow q$. Connectives: $\{\neg,\vee\}$.
      - **Answer: $\neg p\vee q$**
   b. Formula: $p\oplus q$. Connectives: $\{\neg,\vee,\wedge\}$.
      - **Answer: $(p\wedge \neg q)\vee(q \wedge \neg p)$**
   c. Formula: $p\leftrightarrow q$. Connectives: $\{\rightarrow, \wedge\}$.
      - **Answer: $(p\rightarrow q)\wedge(q \rightarrow p)$**
   d. Formula: $(p\rightarrow q) \wedge ((\neg p)\rightarrow q)$. Conn: $\{\neg,\vee\}$.

- **Answer: q** (This was a trick question, since you don't need any connectives.)

e. Formula: ¬p. Conn: { | } (the Sheffer stroke). The truth table for | is:

p	**|**	**q**
T	F	T
T	T	F
F	T	T
F	T	F

- **Answer: p|p**

4. *Which of these are tautologies? Please prove your claims, using truth tables.*

a. $p \rightarrow (q \rightarrow p)$
b. $p \rightarrow (\neg p \rightarrow p)$
c. $(q \rightarrow p) \rightarrow (p \rightarrow q)$
d. $(q \rightarrow p) \vee (p \rightarrow q)$
e. $(p \rightarrow (q \rightarrow r)) \rightarrow (q \rightarrow (p \rightarrow r))$

Answers

a. $p \rightarrow (q \rightarrow p)$ *Tautologous*
b. $p \rightarrow (\neg p \rightarrow p)$ *Tautologous*
c. $(q \rightarrow p) \rightarrow (p \rightarrow q)$ *Contingent*
d. $(q \rightarrow p) \vee (p \rightarrow q)$ *Tautologous*
e. $(p \rightarrow (q \rightarrow r)) \rightarrow (q \rightarrow (p \rightarrow r))$ *Tautologous*

1,2,4,5 are known as "paradoxes of implication" because they contrast with implication in ordinary language.

5. *Reading formulas off truth tables. In class, a proof was sketched for the claim that every propositional logic formula can be expressed using the connectives {∧, ¬}. The proof proceeded essentially by "reading" the correct formula off the truth table Use this meticulous method to construct a formula equivalent to p⊕q.*

Answer.

1. Construct the truth table of p⊕q. This is the opposite of the biconditional.
2. Mark those two rows in the table that make p⊕q TRUE.
3. Corresponding with these two rows, construct a disjunction of two formulas, one of which is (p∧¬q), and the other (q∧¬p).
4. Use the De Morgan Laws to convert this disjunction (p∧¬q)∨(q∧¬p) into the equivalent formula ¬(¬(p∧¬q)∧¬ (q∧¬p))
5. Use truth tables again to check that these two formulas are indeed equivalent.

6. *Reading formulas off truth tables. As Exercise 3, but construct a formula equivalent to p|q. Does the method always produce the shortest answer (i.e. the*

shortest formula that is logically equivalent to the original while still only using negation and conjunction)?

Answer.

1. For p|q, we start with the disjunction:
   $(p\wedge\neg q) \vee (q\wedge\neg p) \vee (\neg p\wedge\neg q)$
2. Use the De Morgan Laws to covert this disjunction into the equivalent formula with conjunctions and negations.
   $\neg(\neg(p\wedge\neg q) \wedge \neg(q\wedge\neg p) \wedge \neg(\neg p\wedge\neg q))$
   This is much lengthier than the logically equivalent formula $\neg(p\wedge q)$.
3. The procedure in the proof does not always get you the shortest answer.

Exercises for CS2013

Predicate Logic Formal Semantic Evaluation

1. Suppose **M1** = (**D**,**I**) where **D**={bill, jill, mary} and
I(jill') = jill, **I**(mary') = mary, **I**(bill') = bill,
I(B)={bill, jill}
I(G)={mary}
I(A)={(bill, jill) , (mary,bill)}

Formally evaluate the following:
   a. Is ∃y B(y) T in M1?
   b. Is ∃y [B(y) ∧ G(y)] T in M1?
   c. Is ∃y [B(y) ∧ A(y,bill')] T in M1?

a. ∃y B(y) is T in M1 iff for some a, **I**(a) ∈ D, ∃y B(y)(a := y) is T in M1. Where ∃y B(y)(bill' := y), B(bill') is T iff I(bill') ∈ I(B), where bill ∈ {bill, jill}. We see it is T, so ∃y B(y) is T in M1.
b. ∃y [B(y) ∧ G(y)] is T in M1 iff for some a, **I**(a) ∈ D, ∃y [B(y) ∧ G(y)] (a := y) is T in M1. Where ∃y [B(y) ∧ G(y)] (bill' := y), [B(bill') ∧ G(bill')] is T iff I(bill') ∈ I(B) and I(bill') ∈ I(G), where bill ∈ {bill, jill} and bill ∈ {mary}. We see it is F, so ∃y [B(y) ∧ G(y)] (bill' := y) in M1. We would have to evaluate other values, e.g. ∃y [B(y) ∧ G(y)] (jill' := y) and ∃y [B(y) ∧ G(y)] (mary' := y). Doing this we will see that none of these alternative values for y results in ∃y [B(y) ∧ G(y)] being T in M1. Therefore, we conclude that ∃y [B(y) ∧ G(y)] is F in M1.
c. ∃y [B(y) ∧ A(y,bill')] is T in M1 iff for some a, **I**(a) ∈ D, ∃y [B(y) ∧ A(y,bill')] (a := y) is T in M1. Where ∃y [B(y) ∧ A(y,bill')] (bill' := y), [B(bill) ∧ A(bill',bill')] is T in M1 iff B(bill) is T in M1 and A(bill',bill') is T in M1. We see that B(bill) is T in M1 (as in example a). A(bill',bill') is T in M1 iff (I(bill'),I(bill')) ∈ I(A), where (bill,bill) ∈ {(bill, jill) , (mary,bill)}, which is F in M1. We would have to evaluate other values, e.g. ∃y [B(y) ∧ A(y,bill')] (jill' := y) and ∃y [B(y) ∧ A(y,bill')] (mary' := y). Doing this, we will see that none of these alternative values for y will make both conjuncts T in M1. Therefore, we conclude that ∃y [B(y) ∧ A(y,bill')] is F in M1.

2. Suppose **M2** = (**D**,**I**) where **D**={bill, jill, mary} and
I(jill') = jill, **I**(mary') = mary, **I**(bill') = bill,
I(B) = {bill, jill}
I(G) = {mary}
I(A) = {(bill, jill), (mary,bill), (jill, jill)}

Formally evaluate the following:
   a. Is ∀y B(y) T in M2?
   b. Is ∀y [B(y) ∨ G(y)] T in M2?
   c. Is ∀y [B(y) ∧ A(y,bill')] T in M2?
   d. Is ∀x [B(x) → A(x,jill)] T in M2?

Answers.

a. ∀y B(y) is T in M2 iff for every a, **I**(a) ∈ D, ∀y B(y)(a := y) is T in M2.

- where ∀y B(y)(bill' := y), B(bill') is T iff I(bill') ∈ I(B), where bill ∈ {bill, jill}, which is T.
- where ∀y B(y)(jill' := y), B(jill') is T iff I(jill') ∈ I(B), where jill ∈ {bill, jill}, which is T.
- where ∀y B(y)(mary' := y), B(mary') is T iff I(mary') ∈ I(B), where mary ∈ {bill, jill}, which is F.

∀y B(y) is F in M2 for not every a, **I**(a) ∈ D, is such that ∀y B(y)(a := y) is T in M2. Alternatively, we can say that ∀y B(y) is F in M2 for there is an a, **I**(a) ∈ D, such that ∀y B(y)(a := y) is F in M2. In general, it is easier to make demonstrations of falsity by showing the falsifying example, as we will do below.

b. ∀y [B(y) ∨ G(y)] is T in M2 iff for every a, **I**(a) ∈ D, ∀y [B(y) ∨ G(y)] (a := y) is T in M2.

- where ∀y [B(y) ∨ G(y)] (bill' := y), [B(bill') ∨ G(bill')] is T in M2 iff B(bill') is T in M2 or G(bill') is T in M2. B(bill') is T in M2 iff I(bill') ∈ I(B), and bill ∈ {bill, jill}, which is T. So, [B(bill') ∨ G(bill')] is T in M2.
- where ∀y [B(y) ∨ G(y)] (jill' := y), [B(jill') ∨ G(jill')] is T in M2 iff B(jill') is T in M2 or G(jill') is T in M2. B(jill') is T in M2 iff I(jill') ∈ I(B), and jill ∈ {bill, jill}, which is T. So, [B(jill') ∨ G(jill')] is T in M2.
- where ∀y [B(y) ∨ G(y)] (mary' := y), [B(mary') ∨ G(mary')] is T in M2 iff B(mary') is T in M2 or G(mary') is T in M2. G(mary') is T in M2 iff I(mary') ∈ I(B), and mary ∈ {mary }, which is T. So, [B(bill') ∨ G(bill')] is T in M2.

c. ∀y [B(y) ∧ A(y,bill')] is T in M2 iff for every a, **I**(a) ∈ D, ∀y [B(y) ∧ A(y,bill')] (a := y) is T in M2.

- where ∀y [B(y) ∧ A(y,bill')] (bill' := y), [B(mary') ∧ A(mary',bill')] is T in M2 iff B(mary') is T in M2 and A(mary',bill') is T in M2. B(mary') is T in M2 iff B(mary') is T in M2 iff I(mary') ∈ I(B), and mary ∈ {bill, jill}, which is F.

∀y [B(y) ∧ A(y,bill')] is F in M2 since there is an a, **I**(a) ∈ D, such that ∀y [B(y) ∧ A(y,bill')] (a := y) is F in M2.

d. ∀x [B(x) → A(x,jill')] is T in M2 iff for every a, **I**(a) ∈ D, ∀x [B(x) → A(x,jill')] (a := y) is T in M2.

- where ∀x [B(x) → A(x,jill')] (bill' := y) is T in M2, [B(bill') → A(bill',jill')] is T in M2 iff B(bill') is F in M2 or A(bill',jill')] is T in M2. We can pick on or the other to show. A(bill',jill')] is T in M2 iff (I(bill'),I(jill')) ∈ I(A), that is, (bill, jill) ∈ {(bill, jill), (mary,bill), (jill, jill)}, which it is, so A(bill',jill')] is T in M2.
- where ∀x [B(x) → A(x,jill')] (jill' := y) is T in M2, [B(jill') → A(jill',jill')] is T in M2 iff B(jill') is F in M2 or A(jill',jill')] is T in M2. We can pick on or the other to show. A(jill',jill')] is T in M2 iff (I(jill'),I(jill')) ∈ I(A), that is, (jill, jill) ∈ {(bill, jill), (mary,bill), (jill, jill)}, which it is, so A(jill',jill')] is T in M2.
- where ∀x [B(x) → A(x,jill')] (mary' := y) is T in M2, [B(mary') → A(mary',jill')] is T in M2 iff B(mary') is F in M2 or A(mary',jill')] is T in M2. We can pick on or the other to show. B(mary') is F in M2 iff I(mary') ∉ I(B), that is, mary ∉ {bill, jill}, which is T in M2.

We have shown that ∀x [B(x) → A(x,jill')] is T in M2 iff for every a, **I**(a) ∈ D, ∀x [B(x) → A(x,jill')] (a := y) is T. Notice that the truth conditions for conditionals rely on either a false antecedent or a true consequent.

3. Suppose **M3** = (**D**,**I**), where D = {jill, bill, phil, will, mary} and
I(is_happy') = {jill, bill, phil}
I(is_hungry') = {jill, bill, phil, will, mary}
I(jill') = jill (and so on).

Formally show with the model and analysis the following:
   a. ∀*x is_hungry'(x)* and ∀*z is_hungry'(z)* are both T in M3.
   b. ∃*x is_happy'(x)* and ∃*z is_happy'(z)* are both T in M3.
   c. ∀*x is_hungry'(x)* entails ∃*z is_hungry'(z)* in M3.

Answers.
The answers are easy to calculate following the previous examples.

What do your observations about these examples lead you to understand about quantifiers and variables?

Answer.
The choice of variable in the same formula (meaning x or z or another variable) does not change the truth value of the expression. If a universally quantified expression is T, then the existentially quantified expression (same predicates and logical structure) is T.

4. Determine the free and bound variables.

   a. $\exists x \neg P(x)$
   b. $\neg \forall x \neg P(x)$
   c. $\exists y\, Q(x)$
   d. $\forall x\, P(b)$
   e. $\forall x(\exists y\, R(x,y))$
   f. $\forall x(\exists y\, R(x,z))$
   g. $\forall x\, \exists x \neg P(x)$
   h. $\forall x\, (P(x)) \wedge Q(x)$
   i. $\exists y\, Q(y) \wedge \forall x\, Q(x)$

Answers.
a. x is bound. There are no free variables.
b. x is bound. There are no free variables.
c. x is a free variable.
d. There are no free or bound variables.
e. x and y are both bound variables. There are no free variables.
f. x is a bound variable and z is a free variable.
g. x is a bound variable. Note that x is only bound to the first quantifier; the second quantifier is vacuous.
h. The variable x in P(x) is bound, while the variable x in Q(x) is free.

i. y and x are both bound variables. There are no free variables.

5. Formally calculate the Truth values of the following expressions in the model:

M4: a model where D = {a, b}, I(Q) = {a}, I(P) = {a, b}.

a. $\exists x\ (Q(x) \wedge P(x))$
b. $\forall x\ (Q(x) \wedge P(x))$
c. $\forall x\ (Q(x) \rightarrow P(x))$
d. $\exists x\ (Q(x) \rightarrow P(x))$

Answers.
The answers to a, b, and c are found (or similar to) examples above. The answer to d is similar to c, but only one value needs to be found for the variable in order for the expression to be true.

Predicate Logic Formal Semantics

A model **M** is an ordered pair <**D,I**>, where **D** is a set of entities, and **I** is an *interpretation function*. **M** defines gives a specific 'meaning' to the non-logical symbols (logical symbols are $\neg \vee \wedge \rightarrow \forall \exists$).

If a is an individual constant then **I**(a) $\in$ **D**. Assume for every entity in D there is a constant.

If P is a 1-place predicate then **I**(P) $\subseteq$ **D**

If R is a 2-place predicate, then **I**(R) $\subseteq \{(\alpha,\beta): \alpha \in \mathbf{D} \text{ and } \beta \in \mathbf{D}\}$

So on for 3-place predicates etc.

A formula of the form P(a) is T with respect to M iff I(a) $\in$ I(P)

A formula of the form R(a,b) is T with respect to M iff **(**I(a),I(b)**)** $\in$ I(R).

A formula of the form $\neg\varphi$ is T wrt M iff φ is F wrt M.

A formula of the form $\varphi \vee \psi$ is T wrt M iff φ is T wrt M or ψ is T wrt M or both.

A formula of the form $\varphi \wedge \psi$ is T wrt M iff φ is T wrt M and ψ is T wrt M.

A formula of the form $\varphi \rightarrow \psi$ is T wrt M iff φ is F wrt M or ψ is T wrt M.

A formula of the form $\exists x\varphi$ is T wrt M iff there is an a, where I(a) $\in$ D, such that $\varphi(x{:=}a)$ is T wrt M.

A formula of the form $\forall x\varphi$ is T wrt M iff for every a, where I(a) $\in$ D, such that $\varphi(x{:=}a)$ is true wrt M.