Logic, First Course, Winter 2020. Week 2, Practice Problems. Back to course website

Practice problems

The practice problems fall into four groups:

Translating and conditionals

The first five problems come from the Corpus of American Soap Operas.

Problem 1

> *i* = *I* am leaving
>
> *y* = *y*ou get out of my way

I am leaving only if you get out of my way.

Problem 2

> *i* = *I*'ll accept Bess
>
> *h* = *h*er middle name is Charlotte

I'll accept Bess only if her middle name is Charlotte

Problem 3

> *d* = a *d*ecision like that is supposed to be made
>
> *e* = all of us who love *E*ric Forrester are going to gather together
>
> *d* = all of us including *D*onna are going to gather together

A decision like that is supposed to be made if and only if all of us who love Eric Forrester are going to gather together and all of us including Donna are going to gather together.

Problem 4

i = *I* could find a way to generate income for you

y = *y*ou leave Chad Harris alone

h = you never bother *h*im

w = you never bother *W*hitney again

I could find a way to generate income for you if and only if you leave Chad Harris alone and you never bother him and you never bother Whitney again.

Problem 5

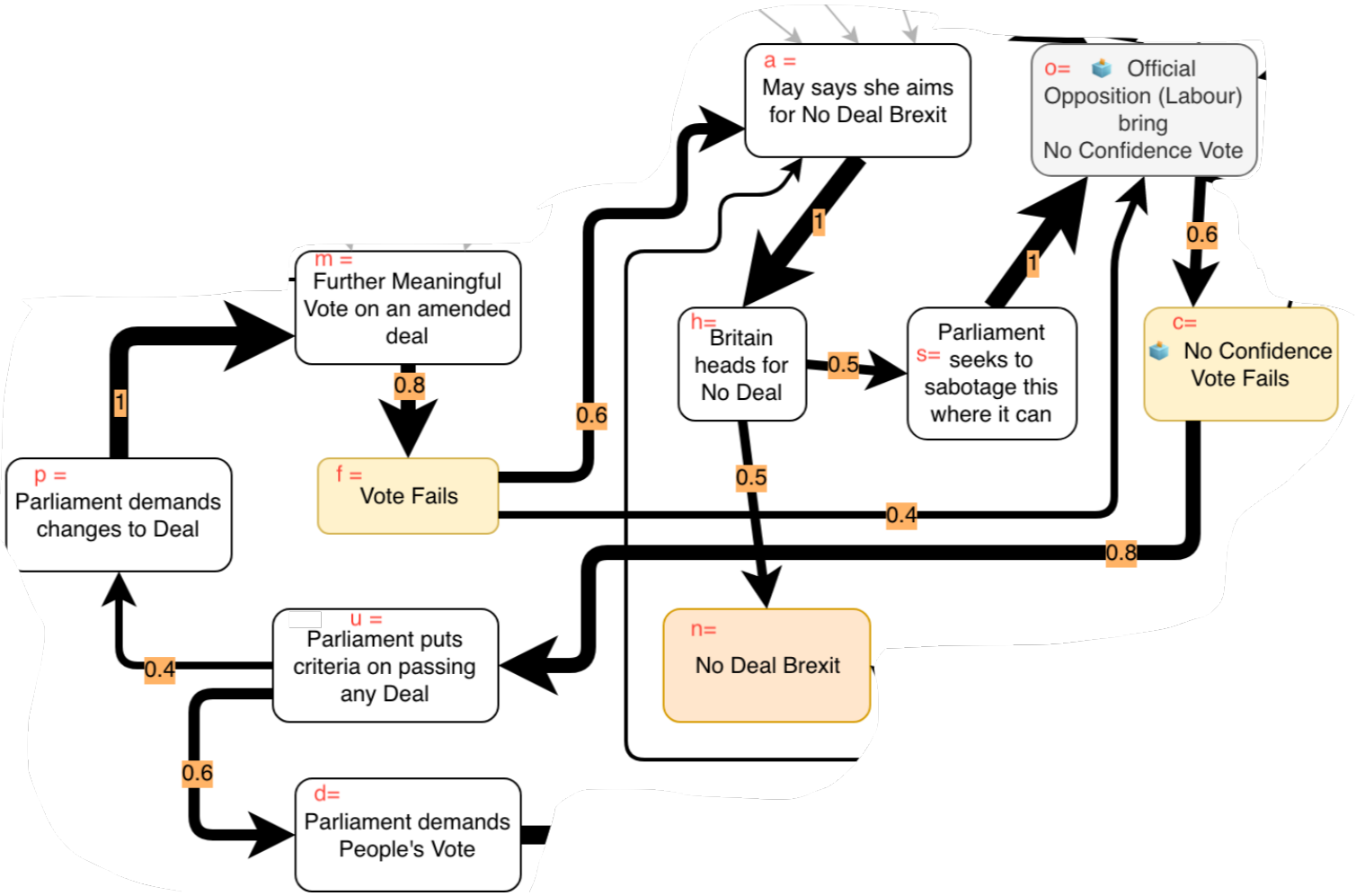
h = you try to *h*urt my mom

t = you try to *t*ake Johnny away from her

n = *n*o one is going to keep me quiet

If you try to hurt my mom or if you try to take Johnny away from her, no one is going to keep me quiet

The following five problems relate to the following Brexit flowchart,[1] which you can download and print/view here (to print/view, perhaps do a "save image as" and then print/view).



The following five problems use a single key, which is also marked on the diagram, and where the enumeration is alphabetical

a = May says she *a*ims for No Deal Brexit

c = The No *C*onfidence Vote Fails

d = Parliament *d*emands People's Vote

f = the Vote *f*ails

h = Britain *h*eads for No Deal

m = Further *M*eaningful Vote on an amended deal happens

n = *N*o Deal Brexit happens

o = *O*fficial Opposition (Labour) brings No Confidence Vote

p = *P*arliament demands changes to Deal

s = Parliament seeks to *s*abotage this where it can

u = Parliament p*u*ts criteria on passing any deal

You do not *need* to consult the diagram to solve these problems, although it *might* be helpful, since the diagram is where the sentences are coming from.

Problem 6

If Britain heads for No Deal then either Parliament seeks to sabotage this where it can or no Deal Brexit happens.

Problem 7

If Parliament demands changes to Deal, then if further Meaningful Vote on an amended deal happens then the Vote fails.

Problem 8

If the Vote fails or if Parliament seeks to sabotage this where it can, official Opposition (Labour) brings No Confidence Vote.

Problem 9

If the Vote fails and May says she aims for No Deal Brexit, Britain heads for No Deal.

Problem 10

if the No confidence Vote Fails and if Parliament puts criteria on passing any deal, then parliament demands changes to Deal or Parliament demands People's Vote.

Necessary and sufficient conditions

The following examples come from the Corpus of US Supreme Court Opinions.

Problem 11

i = the tax *i*s imposed

l = the occurrence of "a *l*ocal activity, delivery of goods within the State upon their purchase for consumption."

The occurrence of "a local activity, delivery of goods within the State upon their purchase for consumption" was a necessary condition for the tax being imposed.

Problem 12

i = the tax *i*s imposed

l = the occurrence of "a *l*ocal activity, delivery of goods within the State upon their purchase for consumption."

A necessary condition for the tax being imposed was the occurrence of "a local activity, delivery of goods within the State upon their purchase for consumption."

Note that the way that "is" statements sometimes allow for symmetry ends up making "*q* is a necessary condition for *p*" and "a necessary condition for *p* is *q*." We translate *both* of these by *p* – > *q*. It is similar to the symmetry between "History major Brianna is an honors student" and "An honors student is history major Brianna." One might feel a little more natural to you, but both have the same truth-conditions. This symmetry was *not* mentioned in the initial presentation in Translating conditionals, but it is something useful to be aware of.

Problem 13

b = the vote of the *b*oard occurs

t = the *t*ransaction is valid

The vote of the board occurring is a necessary condition for the transaction being valid.

Problem 14

b = the vote of the *b*oard occurs

t = the *t*ransaction is valid

a necessary condition for the transaction being valid is the vote of the board occurring.

Problem 15

i = the defendent is cognitively incapacitated

e = a defense of insanity is established

the defendant being cognitively incapacitated is a sufficient condition for a defense of insanity to be established.

Problem 16

i = the defendent is cognitively incapacitated

e = a defense of insanity is established

A sufficient condition for a defense of insanity to be established is for the defendant to be cognitively incapacitated.

Problem 17

b = to *b*e a federal officer

e = the exercise of federal judicial power

To be a federal officer is a necessary but not sufficient condition for the exercise of federal judicial power.

Hint: on problems 17-19, remember from Asserting necessary but not sufficient that the main connectives of these are conjunctions, and that one of the conjuncts is a negated conditional while the other is just a plain conditional.

Problem 18

p = there is proof of prejudice

d = a person submits a due process claim

there being proof of prejudice is generally a necessary but not sufficient condition for a person submitting a due process claim

Problem 19

e = an aid program being run evenhandedly

s = it satisfies constitutional scrutiny

an aid program being run evenhandedly is a necessary but not a sufficient condition for it to satisfy constitutional scrutiny.

Problem 20

c = the compulsory commitment provision is brought into play

d = the defendant is found not guilty of the crime with which he is charged because of insanity "at the time of its commission."

A necessary and sufficient condition for bringing the compulsory commitment provision into play is that the defendant be found not guilty of the crime with which he is charged because of insanity " at the time of its commission."

Hint: this one is asserting both a necessary and sufficient condition. Hence, it is asserting two things. What connective would you use to handle that?

Tautologies

In the following problems, you are to determine whether the formula is a tautology or not. If it is a tautology, fill out the truth-table correctly so that its main connective column is all true, check your answer and then the answer will turn green. If it is not a tautology, then find a row where the formula is false, click the 'Non-tautology' button, and enter in the truth-values of the basic propositional letters on that row, and then the answer will turn green.

Note 1: we will go over these problems in the second lecture of Week 2.

Note 2: some of the formulas here are more complicated than we have seen previously. Hence, prior to each tautology problem, we give a problem like the ones from Week 1 which help you parse the formulas. It is structured like the Practice finding the main connectives problems.

Note 3: even when the formula is not a tautology, it is possible to correctly solve the problem simply by filling out the entire table-- this is because in the event that it is not a tautology you will have found one place in the main connective column where it is false by filling out the entire truth-table. But often in practice one can show something is not a tautology without having to fill out the entire table. In exam situations, you will additionally be asked explicitly after this kind of problem whether the formula is a tautology or not.

Problem 21

Practice parsing the first formula. Hint: don't do the last connective (the bottom-most right connective), so that your tree continues to display. Having the tree in front of you may be helpful when doing the truth-table, since one does the truth-table following the tree from the bottom up. (This hint applies to all the remaining problems in this section, but will not be repeated). Also, ignore the "Success! You may now submit your solution" since this is just practice.

((p ∨ q) → (p ∧ q))

Then determine whether the formula is a tautology:

((p ∨ q) → (p ∧ q))

p	q	((	p	∨	q	)	→	(	p	∧	q	))
T	T		T		T				T		T	
T	F		T		F				T		F	
F	T		F		T				F		T	
F	F		F		F				F		F	

Non-Tautology | Check

Problem 22

Practice parsing the formula:

((p ∧ q) → (p ∨ q))

Then determine whether the formula is a tautology:

((p ∧ q) → (p ∨ q))

p	q	((	p	∧	q	)	→	(	p	∨	q	))
T	T		T		T				T		T	
T	F		T		F				T		F	
F	T		F		T				F		T	
F	F		F		F				F		F	

Non-Tautology | Check

Problem 23

Practice parsing the formula:

((p → q) → (q → p))

Then determine whether the formula is a tautology:

((p → q) → (q → p))

p	q	((	p	→	q	)	→	(	q	→	p	))
T	T	T		T			T		T		T	
T	F	T		F			F		F		T	
F	T	F		T			T		T		F	
F	F	F		F			F		F		F	

Non-Tautology Check

Problem 24

Practice parsing the formula:

((p ↔ q) → (q → p))

Then determine whether the formula is a tautology:

((p ↔ q) → (q → p))

p	q	((	p	↔	q	)	→	(	q	→	p	))
T	T	T		T			T		T		T	
T	F	T		F			F		F		T	
F	T	F		T			T		T		F	
F	F	F		F			F		F		F	

Non-Tautology Check

Problem 25

Practice parsing the formula:

(((¬(p ∧ (q ∧ r)) ∧ p) ∧ q) → ¬r)

Then determine whether the formula is a tautology:

(((¬(p ∧ (q ∧ r)) ∧ p) ∧ q) → ¬r)

p	q	r	(	(	(	¬	(	p	∧	(	q	∧	r	)	)	∧	p	)	∧	q	)	→	¬	r	)
T	T	T		T		T		T		T		T		T		T		T		T				T
T	T	F		T		T		F		T		T		T		T		T		F				F
T	F	T		T		F		T		T		F		T		T		F		T				T
T	F	F		T		F		F		T		F		T		F		F		F				F
F	T	T		F		T		T		F		T		T		F		T		T				T
F	T	F		F		T		F		F		T		T		F		T		F				F
F	F	T		F		F		T		F		F		T		F		F		T				T
F	F	F		F		F		F		F		F		T		F		F		F				F

Non-Tautology | Check

Problem 26

Practice parsing the formula:

((p → q) ∨ (q → p))

Then determine whether the formula is a tautology:

((p → q) ∨ (q → p))

p	q	((	p	→	q	)	∨	(	q	→	p	))
T	T	T		T		T		T		T		T
T	F	T		F		F		F		T		T
F	T	F		T		T		T		F		F
F	F	F		F		F		F		F		F

Non-Tautology | Check

Problem 27

Practice parsing the formula:

((p ↔ q) ∨ (q ↔ p))

Then determine whether the formula is a tautology:

((p ↔ q) ∨ (q ↔ p))

p	q	((	p	↔	q	)	∨	(	q	↔	p	))
T	T	T		T		T		T		T		T
T	F	T		F		F		F		T		T
F	T	F		T		T		T		F		F
F	F	F		F		F		F		F		F

Non-Tautology | Check

Problem 28

Practice parsing the formula:

(p → (q → p))

Then determine whether the formula is a tautology:

(p → (q → p))

p	q	(	p	→	(	q	→	p	))
T	T	T			T			T	
T	F	T			F			T	
F	T	F			T			F	
F	F	F			F			F	

Non-Tautology | Check

Problem 29

Practice parsing the formula:

(p → (q → (p ∨ q)))

Then determine whether the formula is a tautology:

(p → (q → (p ∨ q)))

p	q	(	p	→	(	q	→	(	p	∨	q	)))
T	T	T			T			T			T	
T	F	T			F			T			F	
F	T	F			T			F			T	
F	F	F			F			F			F	

Non-Tautology | Check

Problem 30

Practice parsing the formula:

(p → (q → (p ∨ q)))

Then determine whether the formula is a tautology:

((p ∧ (q ∧ r)) ∨ (¬p ∧ (¬q ∧ ¬r)))

p	q	r	((	p	∧	(	q	∧	r	))	∨	(	¬	p	∧	(	¬	q	∧	¬	r	)))
T	T	T	T		T		T		T			T		T		T		T			T	
T	T	F	T		T		F		F			T		T		T		F			F	
T	F	T	T		F		T		T			T		F		F		T			T	
T	F	F	T		F		F		F			T		F		F		F			F	
F	T	T	F		T		T		T			F		T		T		T			T	
F	T	F	F		T		F		F			F		T		T		F			F	
F	F	T	F		F		T		T			F		F		F		T			T	
F	F	F	F		F		F		F			F		F		F		F			F	

Non-Tautology | Check

Equivalences

In the following problems, you are to determine whether the two formulas which are separated by a comma are equivalent or inequivalent. If they are equivalent, fill out the truth-table correctly, check your answer and then your answer will turn green. If they are inequivalent, then find a row where one is true and the other is false, click the 'Inequivalent' button, and enter in the truth-values of the basic propositional letters on that row, and then your answer will turn green.

Note 1: we will go over these problems in the second lecture of Week 2.

Note 2: some of the formulas here are more complicated than we have seen previously. Hence, prior to each tautology problem, we give a problem like the ones from Week 1 which help you parse the formulas. It is structured like the Practice finding the main connectives problems.

Note 3: even when the formula is not an equivalence, it is possible to correctly solve the problem simply by filling out the entire table-- this is because in the event that it is not a equivalence you will have found one place in the main connective columns where the truth-values differ by filling out the entire truth-table. But often in practice one can show two things are inequivalent without having to fill out the entire table. In exam situations, you will additionally be asked explicitly after this kind of problem whether the two formulas are equivalent.

Problem 31

Practice parsing the first formula. Hint: don't do the last connective (the bottom-most right connective), so that your tree continues to display. Having the tree in front of you may be helpful when doing the truth-table, since one does the truth-table following the tree from the bottom up. (This hint applies to all the remaining problems in this section, but will not be repeated). Also, ignore the "Success! You may now submit your solution" since this is just practice.

¬(a ∨ (b ∨ c))

Practice parsing the second formula:

(¬a ∧ (¬b ∨ ¬c))

Then determine whether the two formulas are equivalent

¬(a ∨ (b ∨ c)), (¬a ∧ (¬b ∨ ¬c))

a	b	c	¬	(	a	∨	(	b	∨	c	)),	(	¬	a	∧	(	¬	b	∨	¬	c	))
T	T	T		T			T			T			T			T				T		
T	T	F		T			T			F			T			T				F		
T	F	T		T			F			T			T			F				T		
T	F	F		T			F			F			T			F				F		
F	T	T		F			T			T			F			T				T		
F	T	F		F			T			F			F			T				F		
F	F	T		F			F			T			F			F				T		
F	F	F		F			F			F			F			F				F		

Inequivalent | Check

Problem 32

Practice parsing the first formula:

¬(a ∨ (b ∨ c))

Practice parsing the second formula:

(¬a ∧ (¬b ∧ ¬c))

Then determine whether the two formulas are equivalent

¬(a ∨ (b ∨ c)), (¬a ∧ (¬b ∧ ¬c))

a	b	c	¬	(	a	∨	(	b	∨	c	)),	(	¬	a	∧	(	¬	b	∧	¬	c	))
T	T	T		T		T		T		T			T			T				T		
T	T	F		T		T		F		F			T			T				F		
T	F	T		T		F		T		T			T			F				T		
T	F	F		T		F		F		F			T			F				F		
F	T	T		F		T		T		T			F			T				T		
F	T	F		F		T		F		F			F			T				F		
F	F	T		F		F		T		T			F			F				T		
F	F	F		F		F		F		F			F			F				F		

Inequivalent | Check

Problem 33

Practice parsing the first formula:

¬(d → (e → f))

Practice parsing the second formula:

(d ∧ (e ∧ ¬f))

Then determine whether the two formulas are equivalent

¬(d → (e → f)), (d ∧ (e ∧ ¬f))

d	e	f	¬	(	d	→	(	e	→	f	)),	(	d	∧	(	e	∧	¬	f	))
T	T	T		T		T		T		T		T		T		T			T	
T	T	F		T		T		T		F		T		T		F			F	
T	F	T		T		T		F		T		T		F		T			T	
T	F	F		T		T		F		F		T		F		F			F	
F	T	T		F		T		T		T		F		T		T			T	
F	T	F		F		T		T		F		F		T		F			F	
F	F	T		F		F		F		T		F		F		T			T	
F	F	F		F		F		F		F		F		F		F			F	

Inequivalent | Check

Problem 34

Practice parsing the first formula:

¬(d → (f → e))

Practice parsing the second formula:

(d ∧ (¬f ∧ e))

Then determine whether the two formulas are equivalent

¬(d → (f → e)), (d ∧ (¬f ∧ e))

d	e	f	¬	(	d	→	(	f	→	e	)),	(	d	∧	(	¬	f	∧	e	))
T	T	T		T		T		T		T		T			T			T		T
T	T	F		T		F		T		T		T			F			T		T
T	F	T		T		T		F		T		T			T			F		F
T	F	F		T		F		F		F		T			F			F		F
F	T	T		F		T		T		T		F			T			T		T
F	T	F		F		F		T		F		F			F			T		T
F	F	T		F		T		F		F		F			T			F		F
F	F	F		F		F		F		F		F			F			F		F

Inequivalent | Check

Problem 35

Practice parsing the first formula:

(l → (m → (n → o)))

Practice parsing the second formula:

(l → (m → (o → n)))

Then determine whether the two formulas are equivalent

(l → (m → (n → o))), (l → (m → (o → n)))

l	m	n	o	(	l	→	(	m	→	(	n	→	o	)	)	)	,	(	l	→	(	m	→	(	o	→	n	)	)	)
T	T	T	T	T			T			T			T					T			T			T			T			
T	T	T	F	T			T			T			F					T			T			F			T			
T	T	F	T	T			T			F			T					T			T			T			F			
T	T	F	F	T			T			F			F					T			T			F			F			
T	F	T	T	T			F			T			T					T			F			T			T			
T	F	T	F	T			F			T			F					T			F			F			T			
T	F	F	T	T			F			F			T					T			F			T			F			
T	F	F	F	T			F			F			F					T			F			F			F			
F	T	T	T	F			T			T			T					F			T			T			T			
F	T	T	F	F			T			T			F					F			T			F			T			
F	T	F	T	F			T			F			T					F			T			T			F			
F	T	F	F	F			T			F			F					F			T			F			F			
F	F	T	T	F			F			T			T					F			F			T			T			
F	F	T	F	F			F			T			F					F			F			F			T			
F	F	F	T	F			F			F			T					F			F			T			F			
F	F	F	F	F			F			F			F					F			F			F			F			

Inequivalent | Check

Problem 36

Practice parsing the first formula:

¬¬¬(p ∧ q)

Practice parsing the second formula:

¬(p ∧ q)

Then determine whether the two formulas are equivalent

¬¬¬(p ∧ q), ¬(p ∧ q)

p	q	¬	¬	¬	(	p	∧	q	),	¬	(	p	∧	q	)
T	T				T		T			T			T		
T	F				T		F			T			F		
F	T				F		T			F			T		
F	F				F		F			F			F		

Inequivalent | Check

Problem 37

Practice parsing the first formula:

(p ∨ (¬p ∨ q))

Practice parsing the second formula:

(r ∨ (¬r ∨ q))

Then determine whether the two formulas are equivalent

(p ∨ (¬p ∨ q)), (r ∨ (¬r ∨ q))

p	q	r	(	p	∨	(	¬	p	∨	q	)),	(	r	∨	(	¬	r	∨	q	))
T	T	T	T					T		T		T				T			T	
T	T	F	T					T		T		F				F			T	
T	F	T	T					T		F		T				T			F	
T	F	F	T					T		F		F				F			F	
F	T	T	F					F		T		T				T			T	
F	T	F	F					F		T		F				F			T	
F	F	T	F					F		F		T				T			F	
F	F	F	F					F		F		F				F			F	

Inequivalent | Check

Problem 38

Practice parsing the first formula:

(a → (b ∨ c))

Practice parsing the second formula:

((a → b) ∨ (a → c))

Then determine whether the two formulas are equivalent

(a → (b ∨ c)), ((a → b) ∨ (a → c))

a	b	c	(	a	→	(	b	∨	c	)), ((	a	→	b	)	∨	(	a	→	c	))
T	T	T	T			T			T		T			T			T			T
T	T	F	T			T			F		T			T			T			F
T	F	T	T			F			T		T			F			T			T
T	F	F	T			F			F		T			F			T			F
F	T	T	F			T			T		F			T			F			T
F	T	F	F			T			F		F			T			F			F
F	F	T	F			F			T		F			F			F			T
F	F	F	F			F			F		F			F			F			F

Inequivalent | Check

Problem 39

Practice parsing the first formula:

(a → (b ∧ c))

Practice parsing the second formula:

((a → b) ∧ (a → c))

Then determine whether the two formulas are equivalent

(a → (b ∧ c)), ((a → b) ∧ (a → c))

a	b	c	(	a	→	(	b	∧	c	)), ((	a	→	b	)	∧	(	a	→	c	))
T	T	T	T			T			T		T			T			T			T
T	T	F	T			T			F		T			T			T			F
T	F	T	T			F			T		T			F			T			T
T	F	F	T			F			F		T			F			T			F
F	T	T	F			T			T		F			T			F			T
F	T	F	F			T			F		F			T			F			F
F	F	T	F			F			T		F			F			F			T
F	F	F	F			F			F		F			F			F			F

Inequivalent | Check

Problem 40

Practice parsing the first formula:

(g ↔ (h ↔ i))

Practice parsing the second formula:

(i ↔ (h ↔ g))

Then determine whether the two formulas are equivalent

(g ↔ (h ↔ i)), (i ↔ (h ↔ g))

g	h	i	(	g	↔	(	h	↔	i	)), (	i	↔	(	h	↔	g	))
T	T	T	T			T			T		T			T			T
T	T	F	T			T			F		F			T			T
T	F	T	T			F			T		T			F			T
T	F	F	T			F			F		F			F			T
F	T	T	F			T			T		T			T			F
F	T	F	F			T			F		F			T			F
F	F	T	F			F			T		T			F			F
F	F	F	F			F			F		F			F			F

Inequivalent | Check

These are practice problems for this course. It is run on the Carnap software, which is an:

1. This is a selection from the lower-left hand corner of Jon Worth's December 1, 2019 diagram, featured on his site, which was discussed in this New York times article.↩