Peter Smith, *Introduction to Formal Logic* (CUP, 2nd edition)

Exercises 10: PL semantics: evaluating wffs

(a) *Suppose we are working in a* PL *language where '*P*' means* Fred is a fool*; '*Q*' means* Fred knows some logic*; '*R*' means* Fred is a rocket scientist*. Translate the following sentences into this formal language as best you can. What do you think is lost in the translations, if you can only use the 'colourless' connectives '$\land$', '$\lor$' and '$\neg$'?*

(1) Even Fred is a rocket scientist. R

(2) Fred is a rocket scientist, but he knows no logic. $(\mathsf{R} \land \neg\mathsf{Q})$

(3) Fred is a rocket scientist, moreover he knows some logic. $(\mathsf{R} \land \mathsf{Q})$

(4) Fred's a fool, even though he knows some logic. $(\mathsf{P} \land \mathsf{Q})$

(5) Although Fred's a rocket scientist, he's a fool and even knows no logic. $(\mathsf{R} \land (\mathsf{P} \land \neg\mathsf{Q}))$

(6) Fred's a fool, yet he's a rocket scientist who knows some logic. $(\mathsf{P} \land (\mathsf{R} \land \mathsf{Q}))$

(7) Fred is a fool despite the fact that he knows some logic. $(\mathsf{P} \land \mathsf{Q})$

(8) Fred is not a rocket scientist who knows some logic. $\neg(\mathsf{R} \land \mathsf{Q})$

(9) Fred knows some logic unless he is a fool. $(\mathsf{Q} \lor \mathsf{P})$ (?)

Some comments:

(1) It does seem absurd to say that 'Even P' has a truth-relevant content different from P'. The use of 'even' signals some element of contrast or even unexpectedness given the context – but conveys no specific content. Think of it as like verbal highlighting!

(2–7) Very roughly speaking, in using 'A but B' rather than 'A and B' the speaker hints that she takes it that (or perhaps expects her audience to take it that) there is some kind of contrast between the truth or the current relevance of A and of B. But again the hint will usually convey no particular content. Similarly 'moreover', 'even though', 'yet', 'despite the fact that' give somewhat different colouring or tone to a conjunction, but no additional truth-relevant content.

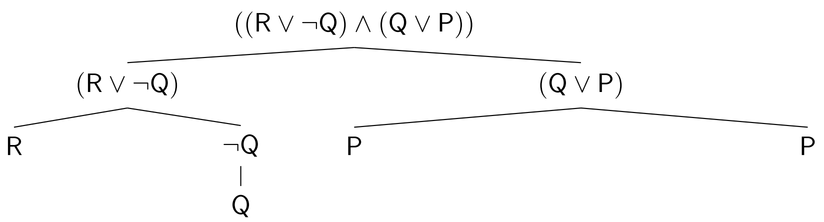
(9) 'Unless' is a more problematic case. If I say 'The match will go head (P) unless it rains (Q)', you will naturally expect the match to be off it does rain – so that might suggest my claim is equivalent to P *or* Q with *exclusive* or. On the other hand, suppose the doctor says 'You will not recover (R) unless you take the antibiotics (S)'. Is the doctor thereby promising that I *will* recover if I *do* take the medicine? Surely not: he is saying that my taking the antibiotics is needed for recovery, not that recovery is guaranteed. So in this case, the doctor's claim seems equivalent to R *or* S with *inclusive* 'or'.

In generally, a claim of the form *P, unless Q* can usually be coherently amplified *P, unless Q; and I'm not ruling out P*

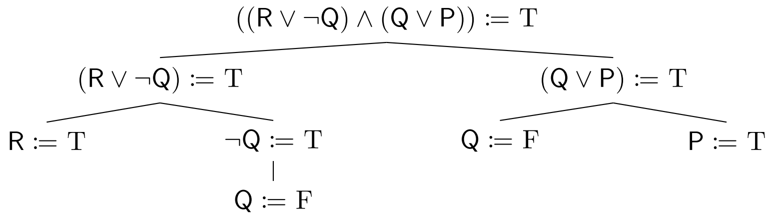
(b) *Confirm that the following strings are wffs by producing parse trees. Suppose that* $\mathsf{P} := T$, $\mathsf{Q} := F$, $\mathsf{R} := T$. *Evaluate the wffs first by chasing values up the trees. Then do the working again in the short form (i.e. as a mini-table, skipping redundant working when you can).*

(1) $((\mathsf{R} \lor \neg\mathsf{Q}) \land (\mathsf{Q} \lor \mathsf{P}))$

Here's the parse tree:



And now let's chase truth values up the parse tree:



Shortform working, in horrible detail:

P	Q	R	((R	∨	¬	Q)	∧	(Q	∨	P))
T	F	T	T	T	T	F	T	F	T	T
			1	3	2	1	5	1	4	1

where at step 1 we carry over the values of atoms, and then at succeeding numbered steps calculate the effect of applying the connective which has the number written under it.

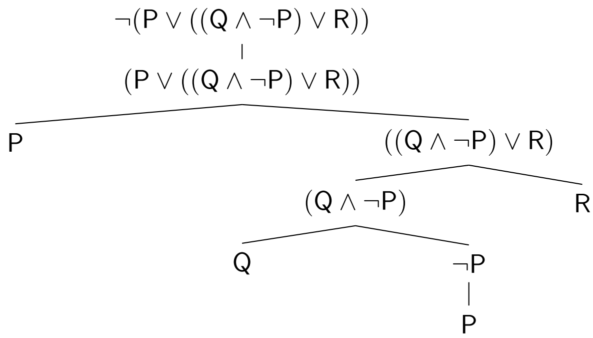
Skipping the first step, and eliminating unnecessary steps (remembering a disjunction is true if one disjunct is true), we can cut down the working to

P	Q	R	((R	∨	¬	Q)	∧	(Q	∨	P))
T	F	T		T			T		T	
				1			3		2	

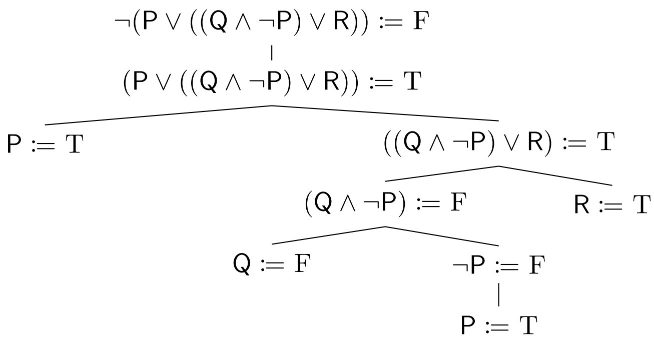
[The numbering of steps of the working here is just for guidance – you don't need to insert them into your answers!]

(2) $\neg(\mathsf{P} \vee ((\mathsf{Q} \wedge \neg\mathsf{P}) \vee \mathsf{R}))$

Here's the parse tree:



Now chasing the truth-values up the parse tree:



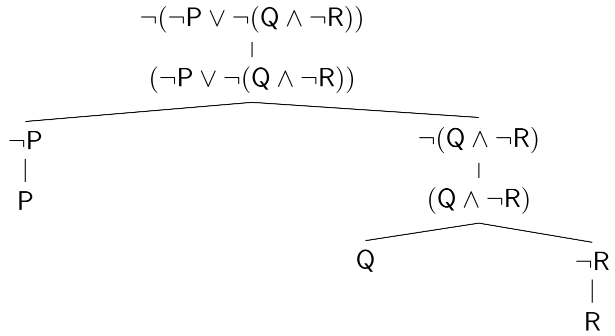
Shortform working:

P	Q	R	¬	(P ∨ ((Q ∧ ¬P) ∨ R))
T	F	T	F	T
			2	1

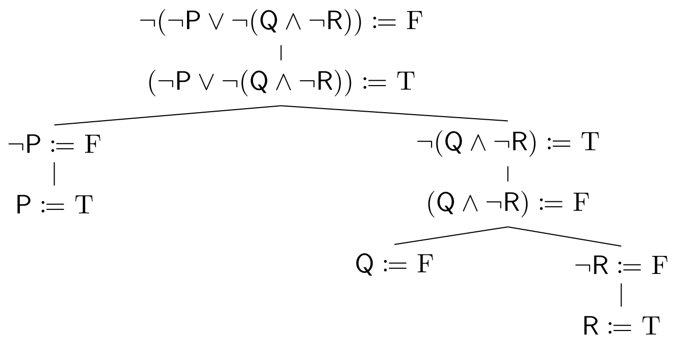
And yes, it is as simple as that! The subformula after the initial negation is a disjunction, and we are given that the first disjunct is true, so (1) the subformula is true, hence the whole formula is false. In effect, we are going up the left branch on the annotated parse tree.

(3) ¬(¬P ∨ ¬(Q ∧ ¬R))

Here's the parse tree:



Now chasing the truth-values up the parse tree:

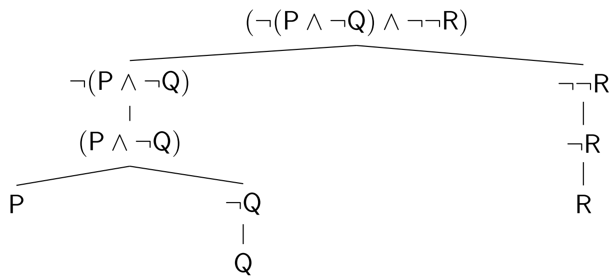


Shortform working:

P	Q	R	¬	(¬P	∨	¬(Q	∧	¬R))
T	F	T	F		T	T	F	
			4		3	2	1	

It wouldn't be wrong to put in more detail: and we could put in less (immediately jumping to step 2).

(4) (¬(P ∧ ¬Q) ∧ ¬¬R)



Now chasing the truth-values up the parse tree:

- $(\neg(\mathsf{P} \wedge \neg\mathsf{Q}) \wedge \neg\neg\mathsf{R}) := \mathrm{F}$
  - $\neg(\mathsf{P} \wedge \neg\mathsf{Q}) := \mathrm{F}$
    - $(\mathsf{P} \wedge \neg\mathsf{Q}) := \mathrm{T}$
      - $\mathsf{P} := \mathrm{T}$
      - $\neg\mathsf{Q} := \mathrm{T}$
        - $\mathsf{Q} := \mathrm{F}$
  - $\neg\neg\mathsf{R} := \mathrm{T}$
    - $\neg\mathsf{R} := \mathrm{F}$
      - $\mathsf{R} := \mathrm{T}$

Shortform working:

P	Q	R	$(\neg$	$(\mathsf{P} \wedge \neg\mathsf{Q})$	$\wedge$	$\neg\neg\mathsf{R})$
T	F	T	F	T	F	
			2	1	3	

You can do step 1 in your head!

(5) $(((\mathsf{P} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{R})) \vee \neg\neg(\mathsf{Q} \vee \neg\mathsf{R}))$

Here's the parse tree:

- $(((\mathsf{P} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{R})) \vee \neg\neg(\mathsf{Q} \vee \neg\mathsf{R}))$
  - $((\mathsf{P} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{R}))$
    - $(\mathsf{P} \vee \neg\mathsf{Q})$
      - P
      - $\neg\mathsf{Q}$
        - Q
    - $(\mathsf{Q} \vee \mathsf{R})$
      - Q
      - R
  - $\neg\neg(\mathsf{Q} \vee \neg\mathsf{R})$
    - $\neg(\mathsf{Q} \vee \neg\mathsf{R})$
      - $(\mathsf{Q} \vee \neg\mathsf{R})$
        - Q
        - $\neg\mathsf{R}$
          - R

Now chasing the truth-values up the parse tree:

- $(((\mathsf{P} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{R})) \vee \neg\neg(\mathsf{Q} \vee \neg\mathsf{R})) := \mathrm{T}$
  - $((\mathsf{P} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{R})) := \mathrm{T}$
    - $(\mathsf{P} \vee \neg\mathsf{Q}) := \mathrm{T}$
      - $\mathsf{P} := \mathrm{T}$
      - $\neg\mathsf{Q} := \mathrm{T}$
        - $\mathsf{Q} := \mathrm{F}$
    - $(\mathsf{Q} \vee \mathsf{R}) := \mathrm{T}$
      - $\mathsf{Q} := \mathrm{F}$
      - $\mathsf{R} := \mathrm{T}$
  - $\neg\neg(\mathsf{Q} \vee \neg\mathsf{R}) := \mathrm{F}$
    - $\neg(\mathsf{Q} \vee \neg\mathsf{R}) := \mathrm{T}$
      - $(\mathsf{Q} \vee \neg\mathsf{R}) := \mathrm{F}$
        - $\mathsf{Q} := \mathrm{F}$
        - $\neg\mathsf{R} := \mathrm{F}$
          - $\mathsf{R} := \mathrm{T}$

$(((\mathsf{P} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{R})) \vee \neg\neg(\mathsf{Q} \vee \neg\mathsf{R}))l$ Shortform working:

P	Q	R	$(((\mathsf{P} \vee \neg\mathsf{Q})$	$\wedge$	$(\mathsf{Q} \vee \mathsf{R}))$	$\vee$	$\neg\neg(\mathsf{Q} \vee \neg\mathsf{R}))$
T	F	T	T	T	T	T	
			1	3	2	4	

You can do step 1 in your head again!

Work out, in short form, the truth values of the following wffs on the assignment of values $\mathsf{P} := \mathrm{F}$, $\mathsf{Q} := \mathrm{F}$, $\mathsf{R} := \mathrm{T}$, $\mathsf{S} := \mathrm{F}$

(6) $\neg((\mathsf{P} \lor \mathsf{Q}) \land \neg(\neg\mathsf{Q} \lor \mathsf{R}))$

P	Q	R	S	$\neg$	$((\mathsf{P} \lor \mathsf{Q})$	$\land$	$\neg(\neg\mathsf{Q} \lor \mathsf{R}))$
F	F	T	F	T	F	F	
				3	1	2	

(7) $\neg\neg((\mathsf{P} \land \mathsf{Q}) \lor (\neg\mathsf{S} \lor \neg\mathsf{R}))$

P	Q	R	S	$\neg\neg$	$((\mathsf{P} \land \mathsf{Q})$	$\lor$	$(\neg\mathsf{S}$	$\lor$	$\neg\mathsf{R}))$
F	F	T	F	T		T		T	
				3		2		1	

(8) $(((\mathsf{S} \land \mathsf{Q}) \land \neg\neg\mathsf{R}) \lor \neg\mathsf{Q})$

P	Q	R	S	$(((\mathsf{S} \land \mathsf{Q}) \land \neg\neg\mathsf{R})$	$\lor$	$\neg\mathsf{Q})$
F	F	T	F		T	T
					2	1

(9) $(((\mathsf{P} \land (\mathsf{Q} \land \neg\mathsf{R})) \lor \neg\neg\neg(\mathsf{R} \land \mathsf{Q})) \lor (\mathsf{P} \land \mathsf{R}))$

P	Q	R	S	$(((\mathsf{P} \land (\mathsf{Q} \land \neg\mathsf{R}))$	$\lor$	$\neg$	$\neg\neg(\mathsf{R}$	$\land$	$\mathsf{Q}))$	$\lor$	$(\mathsf{P} \land \mathsf{R}))$
F	F	T	F		T	T		F		T	
					3	2		1		4	

(10) $(\neg((\mathsf{P} \lor (\mathsf{R} \land \neg\mathsf{S})) \lor \neg(\mathsf{Q} \land \neg\mathsf{P})) \land \neg(\mathsf{P} \lor \neg(\neg\mathsf{Q} \lor \mathsf{R})))$

It might be helpful to rebracket this to bring out its structure:

$$[\neg\{(\mathsf{P} \lor (\mathsf{R} \land \neg\mathsf{S})) \lor \neg(\mathsf{Q} \land \neg\mathsf{P})\} \land \neg\{\mathsf{P} \lor \neg(\neg\mathsf{Q} \lor \mathsf{R})\}]$$

And then a moment's reflection shows that we can in fact work out the value of this quite speedily!

P	Q	R	S	$[\neg$	$\{(\mathsf{P} \lor (\mathsf{R} \land \neg\mathsf{S}))$	$\lor$	$\neg$	$(\mathsf{Q} \land$	$\neg\mathsf{P})\}$	$\land$	$\neg\{\mathsf{P} \lor \neg(\neg\mathsf{Q} \lor \mathsf{R})\}]$
F	F	T	F	F		T	T	F		F	
				4		3	2	1		5	

(c*) *In this book we have taken a maximalist line about the use of brackets in* PL *wffs. What conventions for dropping brackets could we have adopted (while still writing '$\land$' and '$\lor$' between the wffs they connect) in order to reduce the numbers of brackets in a typical wff while not reintroducing semantic ambiguities?*

Three suggestions:

(1) We could *drop outermost brackets*. So instead of our official '$(\mathsf{P} \land \mathsf{Q})$' we could write simply '$\mathsf{P} \land \mathsf{Q}$'. And instead of '$((\mathsf{P} \land \mathsf{Q}) \lor \neg\mathsf{R})$' we could write simply '$(\mathsf{P} \land \mathsf{Q}) \lor \neg\mathsf{R}$', again without any risk of ambiguity.

(2) '$((\mathsf{P} \land \mathsf{Q}) \land \mathsf{R})$' is obviously equivalent to '$(\mathsf{P} \land (\mathsf{Q} \land \mathsf{R}))$' (why?). And more generally, it doesn't matter how we bracket up a multi-conjunction $\alpha \land \beta \land \ldots \land \gamma$: it will be true in the same circumstances. So we could adopt a rule that says that *brackets can be omitted from multi-conjunctions* like these. And exactly similarly, we could adopt a rule that says that *brackets can be omitted from multi-disjunctions.*

(3) Suppose we adopt the first policy, and drop outmost brackets. Why would '$\neg\mathsf{P}\wedge\mathsf{Q}$' – even without the stylistically conventional but logically unnecessary spaces – still be unambiguous? Because ¬' glues to what immediately follows it. We have in effect adopted the policy that our connectives come in an order of precedence – negation binds more tightly than conjunction or disjunction. But so far, we treat conjunction and disjunction on a par.

Now, we could adopt a more fine-grained policy on precedence: for example, *negation binds more tightly than conjunction which binds more tightly than disjunction.* On this convention '$\mathsf{P} \wedge \mathsf{Q} \vee \mathsf{R}$' is unambiguous and is to be read as '$((\mathsf{P} \wedge \mathsf{Q}) \vee \mathsf{R})$'; and '$\neg\mathsf{P} \vee \neg\mathsf{Q} \wedge \mathsf{R}$' is unambiguous and is to be read as '$(\neg\mathsf{P} \vee (\neg\mathsf{Q} \wedge \mathsf{R}))$'. This still not a completely bracket-free notation: for example, if we want to negate the second example, we'd need to write '$\neg(\neg\mathsf{P} \vee \neg\mathsf{Q} \wedge \mathsf{R})$'. But we reduce the number of needed brackets.

(d*) Polish notation *for the propositional calculus – introduced by Jan Łukasiewicz in the 1920s – is a bracket-free notation in which connectives are written* before *the wffs they connect.*

Traditionally, for the Negation of α we write $N\alpha$; for the Konjunction of α and β we write $K\alpha\beta$; for the disjunction of the Alternatives α and β we write $A\alpha\beta$. Since capital letters are used for connectives, it is customary in Polish notation to use lower case letters for propositional atoms. Hence '$(\neg\mathsf{P} \wedge \mathsf{Q})$' *becomes* 'KNpq', '$\neg(\mathsf{P} \wedge \mathsf{Q})$' *becomes* 'NKpq', '$\neg((\mathsf{P} \wedge \neg\mathsf{Q}) \vee \mathsf{R})$' *becomes* 'NAKpNqr', *etc.*

(1) *Rewrite the syntactic rules of §9.1(c) for a language using Polish notation.*

The *wffs* of a particular PL language in Polish style are determined as follows. Having explicitly specified its atomic wffs (say, in a particular four-atom case, p, q, r, s), then

(P1) Any atomic wff of the language counts as a wff.
(P2) If α and β are wffs, so is $K\alpha\beta$.
(P3) If α and β are wffs, so is $A\alpha\beta$.
(P4) If α is a wff, so is $N\alpha$.
(P5) Nothing else is a wff.

(2) *Render the Polish wffs* KNpNq, KNNpq, NKpNq, AKpqr, ApKqr, AANpNqNr, AKNpqKpNq, ANKKpqKqrNArs *into our notation.*

i. *KNpNq*: $(\neg\mathsf{P} \wedge \neg\mathsf{Q})$
ii. *KNNpq*: $(\neg\neg\mathsf{P} \wedge \mathsf{Q})$
iii. *NKpNq*: $\neg(\mathsf{P} \wedge \neg\mathsf{Q})$
iv. *AKpqr*: $((\mathsf{P} \wedge \mathsf{Q}) \vee \mathsf{R})$
v. *ApKqr*: $(\mathsf{P} \vee (\mathsf{Q} \wedge \mathsf{R}))$
vi. *AANpNqNr*: $((\neg\mathsf{P} \vee \neg\mathsf{Q}) \vee \neg\mathsf{R})$
vii. *AKNpqKpNq*: $((\neg\mathsf{P} \wedge \mathsf{Q}) \vee (\mathsf{P} \wedge \neg\mathsf{Q}))$
viii. *ANKKpqKqrNArs*: $(\neg((\mathsf{P} \wedge \mathsf{Q}) \wedge (\mathsf{Q} \wedge \mathsf{R})) \vee \neg(\mathsf{R} \vee \mathsf{S}))$

(3) *Render the wffs (1) to (5) from (b) into Polish notation.*

i. $((\mathsf{R} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{P}))$: *KArNqAqp*
ii. $\neg(\mathsf{P} \vee ((\mathsf{Q} \wedge \neg\mathsf{P}) \vee \mathsf{R}))$: *NApAKqNpr*
iii. $\neg(\neg\mathsf{P} \vee \neg(\mathsf{Q} \wedge \neg\mathsf{R}))$: *NANpNKqNr*
iv. $(\neg(\mathsf{P} \wedge \neg\mathsf{Q}) \wedge \neg\neg\mathsf{R})$: *KNKpNqNNr*
v. $(((\mathsf{P} \vee \neg\mathsf{Q}) \wedge (\mathsf{Q} \vee \mathsf{R})) \vee \neg\neg(\mathsf{Q} \vee \neg\mathsf{R}))$: *AKApNqAqrNNAqNr*

Bonus (but easy) question: how do we find the main connective of a wff in Polish form?

(4) *(Difficult!) Show that Polish notation, although bracket-free, introduces no semantic ambiguities (every Polish wff can be parsed in only one way).*

From our wff-building rules, we know that a Polish wff has to be of the form $N\alpha$, $K\alpha\beta$, $A\alpha\beta$.

Obviously given a wff $N\alpha$, there's only one way to parse it – it's built up by applying N to α!

But what if we are give a wff starting K? Could the same wff be parsed as both $K\alpha\beta$ and $K\alpha'\beta'$ where $\alpha \neq \alpha'$, $\beta \neq \beta'$? (Similarly could a wff starting A be parsed in two ways?) If that were possible, then we could have a case where α and α' are both wffs and one is the initial segment of the other.

But that's impossible. You'll find the outline of that sort of result as the proof for Theorem 8.5(iii) here – but you'll need to understand the idea of a proof by induction.