76

RECEIVED
10 NOV 2006

Automated Reasoning II

1. Suppose $a \succ b \succ c \succ d$. Determine how the following are ordered by the multi-set extension $\succ_{mul}$ of $\succ$. Justify your answers.

(a) $S_1=\{a, c\}$ and $S_2=\{a, c, d\}$

crossing out common occurrences of elements

$S_1'=\{\cancel{a}, \cancel{c}\}$ and $S_2'=\{\cancel{a}, \cancel{c}, d\}$

$S_1'=\{\}$ and $S_2'=\{d\}$

∴ $S_2' \succ_{mul} S_1'$

∴ $\{a, c, d\} \succ_{mul} \{a, c\}$ ✓

(b) $S_1=\{a, c\}$ and $S_2=\{a, a, c, d\}$

crossing out common occurrences of elements

$S_1'=\{\cancel{a}, \cancel{c}\}$ and $S_2'=\{\cancel{a}, a, \cancel{c}, d\}$

$S_1'=\{\}$ and $S_2'=\{a, d\}$

∴ $S_2' \succ_{mul} S_1'$

∴ $\{a, a, c, d\} \succ_{mul} \{a, c\}$ ✓

(c) $S_1=\{a, a, c\}$ and $S_2=\{a, a, c, d\}$

crossing out common occurrences of elements

$S_1'=\{\cancel{a}, \cancel{a}, \cancel{c}\}$ and $S_2'=\{\cancel{a}, \cancel{a}, \cancel{c}, d\}$

$S_1'=\{\}$ and $S_2'=\{d\}$

∴ $S_2' \succ_{mul} S_1'$

∴ $\{a, a, c, d\} \succ_{mul} \{a, a, c\}$ ✓

(d) $S_1=\{a, c, c\}$ and $S_2=\{a, a, c, d\}$

crossing out common occurrences of elements

$S_1'=\{\cancel{a}, \cancel{c}, c\}$ and $S_2'=\{\cancel{a}, a, \cancel{c}, d\}$

$S_1'=\{c\}$ and $S_2'=\{a,d\}$ why?

∴ $S_2' \succ_{mul} S_1'$

∴ $\{a, a, c, d\} \succ_{mul} \{a, c, c\}$ (✓).

(e) $S_1=\{a, c\}$ and $S_2=\{a, c, c, d\}$

crossing out common occurrences of elements

$S_1'=\{\cancel{a}, \cancel{c}\}$ and $S_2'=\{\cancel{a}, \cancel{c}, c, d\}$

$S_1'=\{\}$ and $S_2'=\{c, d\}$

∴ $S_2' \succ_{mul} S_1'$

∴ $\{a, c, c, d\} \succ_{mul} \{a, c\}$ ✓

4

2. Transform each of the following into a set of clauses by first using the optimized structural transformation which introduces a new symbol for each subformula which is disjunction.
(a) $\neg(p \vee q) \vee (\neg p \wedge r)$

Standard structural transformation: (applying rules of Property 4)

$$\begin{array}{llll} & Q_0 & & \\ \wedge & Q_0 & \leftrightarrow & \neg Q_1 \vee (\neg p \wedge r) \\ \wedge & Q_1 & \leftrightarrow & (p \vee q) \end{array}$$

Optimized structural transformation: (applying rules of Property 5)

$$\begin{array}{c} \neg\overset{-}{(p \vee q)} \overset{+}{\vee} \overset{+}{(\neg p \wedge r)} \\ Q_1 \quad Q_0 \end{array} \qquad \text{(polarity of the subformulae)}$$

$$\begin{array}{llll} & Q_0 & & \\ \wedge & Q_0 & \rightarrow & \neg Q_1 \vee (\neg p \wedge r) \\ \wedge & (p \vee q) & \rightarrow & Q_1 \quad ✓ \end{array}$$

so…

$$\begin{array}{lll} Q_0 & \wedge & (Q_0 \rightarrow \neg Q_1 \vee (\neg p \wedge r)) \wedge ((p \vee q) \rightarrow Q_1) \\ Q_0 & \wedge & (\neg Q_0 \vee \neg Q_1 \vee (\neg p \wedge r)) \wedge (\neg (p \vee q) \vee Q_1) \\ Q_0 & \wedge & (\neg Q_0 \vee ((\neg Q_1 \vee \neg p) \wedge (\neg Q_1 \vee r))) \wedge ((\neg p \wedge \neg q) \vee Q_1) \\ Q_0 & \wedge & ((\neg Q_0 \vee \neg Q_1 \vee \neg p)) \wedge (\neg Q_0 \vee \neg Q_1 \vee r)) \wedge ((\neg p \vee Q_1) \wedge (\neg q \vee Q_1)) \\ Q_0 & \wedge & (\neg Q_0 \vee \neg Q_1 \vee \neg p) \wedge (\neg Q_0 \vee \neg Q_1 \vee r) \wedge (\neg p \vee Q_1) \wedge (\neg q \vee Q_1) \end{array}$$

$\{[Q_0], [\neg Q_0; \neg Q_1; \neg p], [\neg Q_0; \neg Q_1; r], [\neg p; Q_1], [\neg q; Q_1]\}$ ✓

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

Standard structural transformation: (applying rules of Property 4)

$$\begin{array}{llll} & Q_0 & & \\ \wedge & Q_0 & \leftrightarrow & (\neg Q_1 \vee (\neg Q_3 \wedge r)) \\ \wedge & Q_1 & \leftrightarrow & (Q_2 \vee q) \\ \wedge & Q_2 & \leftrightarrow & (\neg p \vee s) \\ \wedge & Q_3 & \leftrightarrow & (\neg p \vee s) \end{array}$$

Optimized structural transformation: (applying rules of Property 5)

$$\begin{array}{c} \neg(\overset{-}{(\neg p \vee s)} \overset{-}{\vee} q) \overset{+}{\vee} (\neg\overset{-}{(\neg p \vee s)} \overset{+}{\wedge} r) \\ Q_2 \quad Q_1 \qquad\qquad Q_3 \end{array} \qquad \text{(polarity of the subformulae)}$$

$$\begin{array}{llll} & Q_0 & & \\ \wedge & Q_0 & \rightarrow & (\neg Q_1 \vee (\neg Q_3 \wedge r)) \quad ✓ \\ \wedge & (Q_2 \vee q) & \rightarrow & Q_1 \quad ✓ \end{array}$$

$\wedge \quad (\neg p \vee s) \quad \rightarrow \quad Q_2$ ✓
$\wedge \quad (\neg p \vee s) \quad \rightarrow \quad Q_3$ ✓

It is possible to further optimize, by noticing that Q2 and Q3 are equivalent

$$\begin{array}{llll} & Q_0 & & \\ \wedge & Q_0 & \rightarrow & (\neg Q_1 \vee (\neg Q_2 \wedge r)) \\ \wedge & (Q_2 \vee q) & \rightarrow & Q_1 \\ \wedge & (\neg p \vee s) & \rightarrow & Q_2 \end{array}$$

✓

so…

$Q_0 \wedge (Q_0 \rightarrow (\neg Q_1 \vee (\neg Q_2 \wedge r))) \wedge ((Q_2 \vee q) \rightarrow Q_1) \wedge ((\neg p \vee s) \rightarrow Q_2)$
$Q_0 \wedge (\neg Q_0 \vee (\neg Q_1 \vee (\neg Q_2 \wedge r))) \wedge (\neg (Q_2 \vee q) \vee Q_1) \wedge (\neg (\neg p \vee s) \vee Q_2)$
$Q_0 \wedge (\neg Q_0 \vee ((\neg Q_1 \vee \neg Q_2) \wedge (\neg Q_1 \vee r))) \wedge ((\neg Q_2 \wedge \neg q) \vee Q_1) \wedge ((\neg\neg p \wedge \neg s) \vee Q_2)$
$Q_0 \wedge (\neg Q_0 \vee ((\neg Q_1 \vee \neg Q_2) \wedge (\neg Q_1 \vee r))) \wedge ((\neg Q_2 \wedge \neg q) \vee Q_1) \wedge ((p \wedge \neg s) \vee Q_2)$
$Q_0 \wedge (\neg Q_0 \vee \neg Q_1 \vee \neg Q_2) \wedge (\neg Q_0 \vee \neg Q_1 \vee r) \wedge (\neg Q_2 \vee Q_1) \wedge (\neg q \vee Q_1) \wedge (p \vee Q_2) \wedge (\neg s \vee Q_2)$
$\{[Q_0] , [\neg Q_0 ; \neg Q_1 ; \neg Q_2] , [\neg Q_0 ; \neg Q_1 ; r] , [\neg Q_2 ; Q_1] , [\neg q ; Q_1] , [p ; Q_2] , [\neg s ; Q_2]\}$

✓

⑤

3. Suppose the language Σ includes the constants a, b, c and one binary function symbol f. Consider the following Herbrand interpretation.

$$I = \{p(\mathrm{a}), p(\mathrm{b}), q(f(\mathrm{a,b}))\}$$

I assume p( ), q( ), and r( ) are predicates, and are all valid symbols in language Σ.

The Herbrand universe is $T_\Sigma =$ {a, b, c,
$f(\mathrm{a,a}), f(\mathrm{a,b}), \ldots$
$f(\mathrm{a}, f(\mathrm{a,a})), \ldots$
$f(f(\mathrm{a,a}), f(\mathrm{a,a})), \ldots$
$f(f(f(\mathrm{a,a}),\mathrm{a}), f(\mathrm{a,a})), \ldots$
… }

(a) Determine if the following (ground) clauses hold in *I*. Explain.

i. $p(\mathrm{a}) \vee p(f(\mathrm{a,b}))$

Holds

$I \models p(\mathrm{a})$, and the connective is ∨ (so only one component of the clause needs to be true)
(note: hence it doesn't matter that $I \not\models p(f(\mathrm{a,b}))$. ✓

ii. $p(\mathrm{a}) \vee \neg p(f(\mathrm{a,b}))$

Holds

$I \models p(\mathrm{a})$ is true, and the connective is ∨. ✓
(note: also $I \models \neg p(f(\mathrm{a,b})$, since $I \not\models p(f(\mathrm{a,b})$,
but it is not necessary to consider this).

iii. $p(\mathrm{a}) \vee \neg q(f(\mathrm{a,b}))$

Holds

$I \models p(\mathrm{a})$ is true, and the connective is ∨. ✓
(note: also $I \models q(f(\mathrm{a,b}))$, so $I \not\models \neg q(f(\mathrm{a,b}))$).

iv. $\neg p(\mathrm{a}) \vee \neg q(f(\mathrm{a,b}))$

Does not hold

$I \not\models \neg p(\mathrm{a})$, since $I \models p(\mathrm{a})$
and
$I \not\models \neg q(f(\mathrm{a,b}))$, since $I \models q(f(\mathrm{a,b}))$ ✓

v. $\neg p(\mathrm{a}) \vee q(f(\mathrm{a,b}))$

Holds

$I \models q(f(\mathrm{a,b}))$, and the connective is ∨. ✓
(note: $I \not\models \neg p(\mathrm{a})$)

vi. $\neg p(\mathrm{c}) \vee q(f(\mathrm{a,a}))$

Holds

why?
$I \not\models p(\mathrm{c})$, gives $I \models \neg p(\mathrm{c})$, and the connective is ∨. ✓
(note: $I \not\models q(f(\mathrm{a,a}))$)

3/4

(b) Given $I = \{p(a), p(b), q(f(a,b))\}$. Determine if the following (non-ground) clauses hold in I. Explain.

I assume additionally, x and y are variables, and are both valid symbols in language Σ.

It is necessary to substitute each variable for each ground term in T_Σ, generally looking for a case that is a counter-example.

i. $p(x)$
Does not hold — since $p(c) \notin I$ ✓

ii. $p(x) \vee p(b)$
Holds — since $p(b) \in I$, and the connective is $\vee$
(note: $I \not\models p(x)$) ✓

iii. $\neg p(x)$
Does not hold — since $I \not\models p(x)$, then 'perhaps' $I \models \neg p(x)$,
but $I \not\models \neg p(a)$ since , $p(a) \in I$
and $I \not\models \neg p(b)$ since, $p(b) \in I$ ✓

iv. $p(x) \vee \neg p(b)$
Does not hold — since $I \not\models p(x)$, from (i)
and $I \not\models \neg p(b)$, since $p(b) \in I$ ✓

v. $p(x) \vee \neg q(y)$
Does not hold — since $I \not\models p(x)$, from (i)
and $I \not\models \neg q(y)$, since $q(f(a,b)) \in I$ ✓

vi. $\neg r(x, b)$
Holds — $I \not\models r(x, b)$ for all substitutions in x, then $I \models \neg r(x, b)$ ✓

why?

10

4. Find a 'total ordering' $\succ$ on the ground atoms A, B, C, D, E, such that the associated 'clause ordering' $\succ_C$ orders the following clauses like this (and justify):

$$B \vee C \succ_C A \vee A \vee \neg C \succ_C C \vee E \succ_C C \vee D \succ_C \neg A \vee D \succ_C \neg E$$

The relevant property is … clause ordering: property 9 (i) and (ii) summarized below:

(i) Given $C = \boxed{A} \vee C'$ and $D = \boxed{B} \vee D'$ with A maximal in C, and B maximal in D.
If $A \succ B$, then $C \succ D$

(ii)Given $C = \boxed{\neg A} \vee C'$ and $D = \boxed{A} \vee D'$ with A maximal in C, and $\neg A$ maximal in D.
then $C \succ D$

$C \vee D \succ_C \neg A \vee D$ … gives $C \succ \neg A$
… gives $C \succ A$ ✓

$C \vee E \succ_C C \vee D$ … *gives* $E \succ D$ ✓

$\neg A \vee D \succ_C \neg E$
and $E \succ D$ … gives $\neg A \succ \neg E$
… gives $A \succ E$ ✓

$B \vee C \succ_C C \vee E$ … *gives* $B \succ E$ (not needed)
$B \vee C \succ_C C \vee D$ … *gives* $B \succ D$ (not needed)

$B \vee C \succ_C A \vee A \vee \neg C$
and $C \succ A$
but $\neg C \succ C$ … *gives* $B \succ \neg C$
gives $B \succ C$ ✓

Hence,

$$B \succ C \succ A \succ E \succ D$$ ✓

Or more explicitly

$$\neg B \succ B \succ \neg C \succ C \succ \neg A \succ A \succ \neg E \succ E \succ \neg D \succ D$$

Checking that this is valid …

Highlighting the maximal atom according to the ordering of ground atoms above

1. $\boxed{B} \vee C$
2. $A \vee A \vee \boxed{\neg C}$
3. $\boxed{C} \vee E$
4. $\boxed{C} \vee D$
5. $\boxed{\neg A} \vee D$
6. $\boxed{\neg E}$

so, ordering the clauses according to the ordering of ground atoms above,

$$1 \succ_C 2 \succ_C 3 \succ_C 4 \succ_C 5 \succ_C$$

which, is as required

10

5. Explain the importance of Property 11(v), in the proof of the model existence theorem (Property 12).

The context of properties 11 and 12 is as follows:

- N is a set of ground clauses input into a model construction process.
- $\succ$ is an ordering on the ground atoms, which is extended to ordering of the clauses.
- $I_N^{\succ}$ is the final candidate model for N, with respect to $\succ$, created by model construction

Property 11(v) states: If in the clause $C = C' \vee A$ produces A during model construction, then $\boldsymbol{I_N^{\succ}} \nvDash \boldsymbol{C'}$ (otherwise the clause $C' \vee A$ would be true in I and non-productive)

so, A is maximal in the clause C

A is productive during model construction, $I_N \vDash A$

and... C' will remain false in I_N for all clauses

Property 12 is the **model existence theorem**.

Let $\succ$ be a clause ordering, and

let N be saturated with respect to Res, and suppose $\bot \notin N$

then $I_N^{\succ} \vDash N$, so, the Interpretation is a model of the set (with no counterexamples)

Proof property 12:

Given $\bot \notin N$, suppose the model existence theorem is wrong, so ... $I_N^{\succ} \nvDash N^{(1)}$

Let C be *minimal* in $\succ$ such that $I_N^{\succ} \nvDash C$, $^{(2)}$ (where $C \in N$)

then, since C is false in $I_N^{\succ}$, C is not productive$^{(3)}$

and, as $C \neq \bot$, there exists a maximal atom A in C, i.e. $C = C' \vee \boxed{A} \vee \boxed{A}$ or $C = C' \vee \boxed{\neg A}^{(4)}$

so proving by contradiction and by cases

Case 1: $C = C' \vee \boxed{\neg A}^{(5)}$

$\Rightarrow I_N^{\succ} \nvDash \neg A$ and $I_N^{\succ} \nvDash C'$ $^{(6)}$ $\Rightarrow I_N^{\succ} \vDash A$ and $I_N^{\succ} \nvDash C'$ $^{(7)}$

$\Rightarrow$ some $D = D' \vee \boxed{A}$ produces A (where $D \in N$)$^{(8)}$

now, $Res(D' \vee \boxed{A}, C' \vee \boxed{\neg A}) = D' \vee C'^{(9)}$

so, $D' \vee C' \in N$, and $C \succ D' \vee C'^{(10)}$

But this means that there must be a clause smaller than C, which contradicts $^{(2)}$

Case 2:

$C = C' \vee \boxed{A} \vee \boxed{A}^{(11)}$ $\Rightarrow I_N^{\succ} \nvDash A$ and $I_N^{\succ} \nvDash C'$ $^{(12)}$

then, $Factoring(C' \vee \boxed{A} \vee \boxed{A}) = C' \vee \boxed{A}$ yields a smaller counter-example$^{(13)}$

$C' \vee \boxed{A} \in N^{(14)}$

But this means that there must be a clause smaller than C, which contradicts $^{(2)}$

Now to the question .....

In case 1 we are considering a valid model $I_N^{\succ} \vDash N$ where ...

$D' \vee C'$

$D' \vee \boxed{A}$

$C' \vee \boxed{\neg A}$

If property 11(v) were not true, then $D' \vDash N$ is possible. In this event, the restriction on A to be maximal in clauses 2 and 3 is lifted. We are hence free to re-order these clauses so that $C' \vee \boxed{\neg A}$ *is* indeed minimal, and the proof by contradiction in case 1 fails, and the model existence theorem is not true.

0

6. Let $\succ$ be a 'total and well founded ordering' on ground atoms such that, if the atom A contains more symbols than B (not counting brackets), then $A \succ B$.
Let N be the following set of clauses:

1. $\neg q(z, z)$
2. $\neg q(f(x), y) \vee q(f(f(x)), y) \vee p(x)$
3. $\neg p(a) \vee \neg p(f(a)) \vee q(f(a), f(f(a)))$
4. $p(f(x)) \vee p(g(y))$
5. $\neg p(g(a)) \vee p(f(f(a)))$

? ?

I assume: p(_) and q(_) are valid predicates, f(_) and g(_) are valid functions, x, y and z are valid variables, a is a valid constant.

$T_\Sigma =$ {a, f(a,a), g(a,a), …
f(a, f(a,a)), …
f(f(a,a), f(a,a)), …
f(f(f(a,a),a), f(a,a)), …
… }

(a) Which literals are strictly maximal in the clauses of N? Justify.

1. $\neg q(z, z)$
There is only one literal in the clause, so for all substitutions of z for ground terms, $\neg q(z, z)$ is strictly maximal. ✓

2. $\neg q(f(x), y) \vee q(f(f(x)), y) \vee p(x)$
In these terms, the first has 3 symbols, the second has 4 symbols and the third has 1 symbol *inside* the predicate brackets. Hence, $q(f(f(x)), y)$ appears to be strictly maximal under the stated ordering.
However, there are two variables here (x and y), which could be substituted for ground atoms in different ways. Since, the term-3 only contains x, and both term-1 and term-2 contain both x and y, it is possible to substitute y by a ground atom with more brackets than x, and outsize term-3. The competition is hence between term-1 and term-2.
Any substitution of x and y must be consistent across term-1 and term-2. The structure of these terms means that $q(f(f(x)), y)$ will always be larger. Hence the first impression that $q(f(f(x)), y)$ is strictly maximal under the stated ordering was correct ✓

3. $\neg p(a) \vee \neg p(f(a)) \vee q(f(a), f(f(a)))$
There are no substitutions possible, since these are already ground terms. In these terms, the first has 1 symbol, the second has 2 symbols and the third has 5 symbols inside the predicate brackets. Hence, $q(f(a), f(f(a)))$ is strictly maximal under the stated ordering. ✓

4. $p(f(x)) \vee p(g(y))$
both terms have 2 symbols inside the predicate brackets, and so there is no order defined. Under non-strict ordering both would have been maximal, if the *same*

substitution was made for x and y, but under strict ordering there would be no maximal literal in this clause. It is of course possible to substitute x and y for ground atoms in different ways, so that either term-1 or term-2 would be maximal. This indicates that there is no maximal literal in this clause. ✗

5. $\neg p(g(a)) \vee p(f(f(a)))$
There are no substitutions possible, since these are already ground terms. In these terms, the first has 2 symbols, the second has 3 symbols inside the predicate brackets. Hence, $p(f(f(a)))$ is strictly maximal under the stated ordering. ✓

So, strictly maximal literals are boxed below

1. $\boxed{\neg q(z, z)}$
2. $\neg q(f(x), y) \vee \boxed{q(f(f(x)), y)} \vee p(x)$
3. $\neg p(a) \vee \neg p(f(a)) \vee \boxed{q(f(a), f(f(a)))}$
4. $p(f(x)) \vee p(g(y))$
5. $\neg p(g(a)) \vee \boxed{p(f(f(a)))}$

(b) Define a selection function S such that N is saturated under $Res^{\succ}_{S}$. Explain.

1. $\underline{\neg q(z, z)}$
2. $\neg q(f(x), y) \vee \underline{q(f(f(x)), y)} \vee p(x)$
3. $\neg p(a) \vee \neg p(f(a)) \vee \underline{q(f(a), f(f(a)))}$
4. $p(f(x)) \vee p(g(y))$
5. $\neg p(g(a)) \vee \underline{p(f(f(a)))}$

The strictly maximal literals are underlined and in gray. There is no strictly maximal literal in clause 4.

The rules for resolution with ordered selection ($Res^{\succ}_{S}$) are as follows. If all the rules do not apply, then resolution need not be preformed.
If the complimentary clauses have the form $\dfrac{C \vee A \qquad \neg B \vee D}{(C \vee D)\sigma}$ where σ is the mgu(A,B)

(i) $A\sigma$ strictly maximal wrt. $C\sigma$
(ii) nothing is selected in C by S
(iii) either $\neg B$ is selected, or else, nothing is selected in $\neg B \vee D$ and $\neg B\sigma$ is maximal wrt. $D\sigma$

If the chosen selection function is S, and the clause is C, then for S : C ↦ *any negated predicate*.
Selected terms are boxed.

1. $\boxed{\underline{\neg q(z, z)}}$
2. $\boxed{\neg q(f(x), y)} \vee \underline{q(f(f(x)), y)} \vee p(x)$
3. $\boxed{\neg p(a)} \vee \boxed{\neg p(f(a))} \vee \underline{q(f(a), f(f(a)))}$
4. $p(f(x)) \vee p(g(y))$
5. $\boxed{\neg p(g(a))} \vee \underline{p(f(f(a)))}$

If we just consider whether it is necessary to apply resolution to these clauses (not whether it would be successful, or what the mgu is).

There are no clauses which can be equivalent to '$C \vee A$'.... since a negative literal is selected in each clause except 4 (thus excluding 1,2,3,5), and neither literal is *strictly* maximal in 4.
Hence, there can be no resolution performed. (Note, clauses 1, 2, 3, 5 could all be equivalent to '$C \vee A$').

There are similar rules for ordered positive factoring with selection
If the clauses have the form

$$\frac{C \vee A \quad \vee B}{(C \vee A)\sigma} \qquad \text{where } \sigma \text{ is the mgu}(A,B)$$

(i) $A\sigma$ strictly maximal wrt. $C\sigma$
(ii) nothing is selected in C

Considering each clause in turn,
For clause 1, no factoring is possible.
For clause 2, no factoring is possible, and also fails for (ii)
For clause 3, no positive factoring is possible, and also fails for (ii)
For clause 4, not possible to factor, but fails for (i)
For clause 5, no factoring is possible
No new terms can be created by resolution where factoring may be an issue, since no resolution has been performed.

Hence, no factoring need be performed.

Hence, no inferences need to be performed on these clauses, and so this selection function gives *N* that is saturated under $Res^{\succ}_{S}$.

✓

~~8~~ 7

> **7.** Let $\succ$ be a total and well founded ordering on ground atoms such that, if the atom A contains more symbols than B, then $A \succ B$.
> Let N be the following set of clauses:
>
> $$p(x) \vee p(f(x))$$
> $$p(y) \vee \neg p(f(f(y)))$$
>
> Use resolution and the clause ordering based on $\succ$ to derive $\perp$ from N. Justify each step.

Assume p is a valid predicate, f is a valid unary function, and x and y are valid variables in the language Σ. There must be at least one constant term. Let that constant term be a.
Hence, $T_\Sigma = \{$ p(a), p(f(a), p(f(f(a))), … $\}$

The strictly maximal literal in these terms under the stated ordering (for all possible substitutions) is show by the boxes …

1. $p(x) \vee \boxed{p(f(x))}$
2. $p(y) \vee \boxed{\neg p(f(f(y)))}$

This ordering of terms is determined, since only variable x appears in clause 1, and only variable y appears in clause 2, and hence any substitution must maintain the discrepancy in the number of symbols in terms 1 and 2 of these clauses.

Applying the substitution $\sigma = \{ x/f(y)) \}$, the mgu between the maximal terms in clauses 1 and 2

3. $p(f(y))) \vee \boxed{p(f(f(y))))}$
4. $p(y) \vee \boxed{\neg p(f(f(y)))}$

As a result of the model existence theorem, it is only necessary to resolve and factor on maximal literals. Hence…

5. $[\boxed{p(f(y)))} \vee p(y)]$ Res(3,4) ✓

This cannot be unified with 2. No other inferences are possible, other than between 1 and 2, and in every case, all other possible substitutions would lead to a similar situation.

No other viable inferences are hence possible.
The proof hence terminates. (Completeness of resolution)
And the proof fails.

So $\perp$ or the empty clause has been *not* derived from N.

(Incidentally, this means that the original formula that gave rise to the above clauses is satisfiable).

③

8. Let the ordering be defined by

$$q \succ p(a_4) \succ p(a_3) \succ p(a_2) \succ p(a_1) \succ p(a_0).$$

Let *N* be the following set of clauses:

1. $q \vee q \vee p(a_1)$
2. $q \vee p(a_1)$
3. $p(a_3) \vee p(a_2) \vee p(a_0)$
4. $p(a_3) \vee p(a_1) \vee p(a_0)$
5. $p(a_2) \vee \neg p(a_1)$
6. $\neg p(a_1) \vee p(a_1)$
7. $p(a_1) \vee p(a_0)$

Which of the clauses in *N* are redundant with respect to *N*? Justify.

Consider the ordering $q \succ p(a_4) \succ p(a_3) \succ p(a_2) \succ p(a_1) \succ p(a_0)$.
The maximal clauses are boxed below.

1. $\boxed{q} \vee \boxed{q} \vee p(a_1)$
2. $\boxed{q} \vee p(a_1)$
3. $\boxed{p(a_3)} \vee p(a_2) \vee p(a_0)$
4. $\boxed{p(a_3)} \vee p(a_1) \vee p(a_0)$
5. $\boxed{p(a_2)} \vee \neg p(a_1)$
6. $\boxed{\neg p(a_1)} \vee p(a_1)$
7. $\boxed{p(a_1)} \vee p(a_0)$

so the clausal ordering that results from the ordering of literals that was specified is …

$$1 \succ 2 \succ 3 \succ 4 \succ 5 \succ 6 \succ 7$$

Now considering redundancy…

1. ~~$q \vee q \vee p(a_1)$~~
2. $q \vee p(a_1)$
3. ~~$p(a_3) \vee p(a_2) \vee p(a_0)$~~
4. ~~$p(a3) \vee p(a1) \vee p(a0)$~~
5. $p(a_2) \vee \neg p(a_1)$
6. ~~$\neg p(a_1) \vee p(a_1)$~~
7. $p(a_1) \vee p(a_0)$

The relevant sections of the notes are ….

Definition of redundancy

Let *N* be a set of ground clauses, and *C* a ground clause (not necessarily in *N*)
C_R is redundant with respect to N, if
there exists $C_1 \ldots C_n \in N$, ($n >= 0$) such that
$C_R \succ$ all Ci,
And, $C_1 \ldots C_n \models C$

Property 18:

(i) If C is a tautology ($\models C$), then C is redundant wrt. any set of clauses N
(ii) $C\sigma \subset D$, then D is redundant wrt. $N \cup \{C\}$
(iii) $C\sigma \subseteq D$, then $D \vee \bar{L}\sigma$ is redundant wrt. $N \cup \{C \vee L, D\}$ (where $\bar{L}$ denotes the complement of L)

- clause 6 is a tautology of the form $A \vee \neg A$ which is always true. Hence clause 6 is redundant with respect to N. (property 18)

- clause 2 is a subset of clause 1, that is
  $q \vee p(a_1) \ \subset\ q \vee \underline{q \vee p(a_1)}$ ✓
  so clause 1 is redundant (property 18)
- in the same way clause 7 is a subset of clause 4,
  $p(a_1) \vee p(a_0) \ \subset p(a_3) \vee \underline{p(a_1) \vee p(a_0)}$ ✓
  so clause 4 is redundant (property 18)
- in a similar way, $\mathrm{Res}(7,5) = [p(a_0)\ ;\ p(a_2)]$
  and this is a subset of clause 3, and so clause 3 is also redundant ✓
  (justified by definition of redundancy)

Are there any more redundant clauses now considering the ordering of the literals as specified?

Building a model for these clauses, using the ordering determined above

Clauses in set N	Candidate model, I_C	Δ_C	Comment
7 $\underline{p(a_1)} \vee p(a_0)$	$\varnothing$	$\{p(a_1)\}$	Productive
6 $\underline{\neg p(a_1)} \vee p(a_1)$	$\{p(a_1)\}$	$\varnothing$	true in I_C
5 $\underline{p(a_2)} \vee \neg p(a_1)$	$\{p(a_1)\}$	$\{p(a_2)\}$	Productive
4 $\underline{p(a_3)} \vee p(a_1) \vee p(a_0)$	$\{p(a_1), p(a_2)\}$	$\varnothing$	true in I_C
3 $\underline{p(a_3)} \vee p(a_2) \vee p(a_0)$	$\{p(a_1), p(a_2)\}$	$\varnothing$	true in I_C
2 $\underline{q} \vee p(a_1)$	$\{p(a_1), p(a_2)\}$	$\varnothing$	true in I_C
1 $q \vee q \vee p(a_1)$	$\{p(a_1), p(a_2)\}$	$\varnothing$	no strictly maximal

The strictly maximal literals are underlined and in gray. There are no counter-examples.
So the final model is $I_N = \{\, p(a_1), p(a_2) \,\}$
Since non-redundant clauses are either (i) productive or (ii) minimal counter-examples, as expected, clauses 1, 2, 3, 4 and 6 are confirmed to be *possibly* redundant, and there are two certainly non-redundant clauses, 5 and 7 (no need to examine these further!)

5. $p(a_2) \vee \neg p(a_1)$
7. $p(a_1) \vee p(a_0)$ (the smallest clause)

So, we now need to determine whether clauses 2 and 3 are redundant in in ordered…

Can 2. $q \vee p(a_1)$ be shown to be true from clauses 3 to 7?
No. Hence 2 is non-redundant.
(Likewise, it's worth checking that clause 5 cannot be show to be true from clauses 6 and 7, which is the case).

Hence the final conclusion is that … clauses 1, 3, 4 and 6 are redundant.

✓

10

9. Redundant clauses remain redundant, if the theorem prover derives new clauses and adds them to the current set of clauses. Prove:

If N and M are sets of clauses and $N \subseteq M$, then Red(N) $\subseteq$ Red(M).

Considering the definition of redundancy,
Let N be a set of ground clauses, and C a ground clause (not necessarily in N)
C_R is redundant with respect to N, if
there exists $C_1 \ldots C_n \in$ N, (n >= 0) such that
$C_R \succ$ all Ci,
And, $C_1 \ldots C_n \models C_R$

Intuitively this is correct. If there are 'essential clauses' and redundant clauses (R) in N, then these same clauses are present in M. The relationships that made R redundant in N, still hold in M, because there are no less elements in M, and the same subset of smaller clauses that made R redundant is present in M. ✓

Consider a set of clauses N, in which one is redundant (C_{R1}), and the clauses which cause this redundancy are $C_1 \ldots C_i$.
$N = \{C_1 \ldots C_i, C_{R1}\}$
Hence $C_{R1} \succ$ all $C_1 \ldots C_i$, and $C_1 \ldots C_n \models C_R$

Now let the theorem prover derive new clauses, which may or may not be redundant, and add them to the clauses in N, such that M is a superset of N ($N \subseteq M$)
$M = \{ C_1 \ldots C_i, D_1 \ldots D_i, C_{R1} \ldots C_{Ri}, \}$
Now assume C_{R1} is no longer redundant in M
so either $C_{R1} \succ$ all $C_1 \ldots C_i$ no longer holds
or $C_1 \ldots C_n \models C_R$ is no longer true
neither of which is possible
so, C_{R1} is redundant in both N and M

Having shown that clauses redundant in N are also redundant in M, then whether or not new redundant clauses are added in M, Red(N) $\subseteq$ Red(M)

non-ground case?

3

10. Consider the following.

1. $(A \wedge B \wedge C) \rightarrow D$
2. $(\neg A \wedge M) \rightarrow L$
3. $(F \wedge E) \rightarrow \neg D$
4. $(G \wedge M) \rightarrow C$
5. $(\neg B \wedge F) \rightarrow \neg H$
6. $(\neg D \wedge B \wedge E) \rightarrow G$
7. $(M \wedge \neg I) \rightarrow J$
8. $(H \wedge M) \rightarrow K$
9. $(K \wedge J \wedge \neg L) \rightarrow E$
10. $(\neg H \wedge F) \rightarrow L$
11. $(M \wedge L) \rightarrow \neg F$
12. $(K \wedge I \wedge A) \rightarrow E$
13. Therefore $M \rightarrow \neg F$

(a) Use the standard transformation to convert the problem in clausal form.

This statement can be re-written in logic format as …

$$\begin{array}{llllll}
((A \wedge B \wedge C) \rightarrow D) & \wedge & ((\neg A \wedge M) \rightarrow L) & \wedge & ((F \wedge E) \rightarrow \neg D) \wedge \\
((G \wedge M) \rightarrow C) & \wedge & ((\neg B \wedge F) \rightarrow \neg H) & \wedge & ((\neg D \wedge B \wedge E) \rightarrow G) \wedge \\
((M \wedge \neg I) \rightarrow J) & \wedge & ((H \wedge M) \rightarrow K) & \wedge & ((K \wedge J \wedge \neg L) \rightarrow E) \wedge \\
((\neg H \wedge F) \rightarrow L) & \wedge & ((M \wedge L) \rightarrow \neg F) & \wedge & ((K \wedge I \wedge A) \rightarrow E) \\
 & & & & \models (M \rightarrow \neg F)
\end{array}$$

Converting into clausal form…

Applying standard resolution to the left-hand side

$$\begin{array}{lllll}
(\neg(A \wedge B \wedge C) \vee D) & \wedge & (\neg(\neg A \wedge M) \vee L) & \wedge & (\neg(F \wedge E) \vee \neg D) \quad \wedge \\
(\neg(G \wedge M) \vee C) & \wedge & (\neg(\neg B \wedge F) \vee \neg H) & \wedge & (\neg(\neg D \wedge B \wedge E) \vee G) \wedge \\
(\neg(M \wedge \neg I) \vee J) & \wedge & (\neg(H \wedge M) \vee K) & \wedge & (\neg(K \wedge J \wedge \neg L) \vee E) \wedge \\
(\neg(\neg H \wedge F) \vee L) & \wedge & (\neg(M \wedge L) \vee \neg F) & \wedge & (\neg(K \wedge I \wedge A) \vee E)
\end{array}$$

(applying definition of →)

Now only ∧, ∨ , ¬ connectives remain

$$\begin{array}{lllll}
((\neg A \vee \neg B \vee \neg C) \vee D) & \wedge & ((\neg\neg A \vee \neg M) \vee L) & \wedge & ((\neg F \vee \neg E) \vee \neg D) \wedge \\
((\neg G \vee \neg M) \vee C) & \wedge & ((\neg\neg B \vee \neg F) \vee \neg H) \wedge & & ((\neg\neg D \vee \neg B \vee \neg E) \vee G) \wedge \\
((\neg M \vee \neg\neg I) \vee J) & \wedge & ((\neg H \vee \neg M) \vee K) & \wedge & ((\neg K \vee \neg J \vee \neg\neg L) \vee E) \wedge \\
((\neg\neg H \vee \neg F) \vee L) & \wedge & ((\neg M \vee \neg L) \vee \neg F) & \wedge & ((\neg K \vee \neg I \vee \neg A) \vee E)
\end{array}$$

(applying deMorgan's Laws, once or twice)

$$\begin{array}{llllll}
((\neg A \vee \neg B \vee \neg C) \vee D) & \wedge & ((A \vee \neg M) \vee L) & \wedge & ((\neg F \vee \neg E) \vee \neg D) \wedge \\
((\neg G \vee \neg M) \vee C) & \wedge & ((B \vee \neg F) \vee \neg H) & \wedge & ((D \vee \neg B \vee \neg E) \vee G) \wedge \\
((\neg M \vee I) \vee J) & \wedge & ((\neg H \vee \neg M) \vee K) & \wedge & ((\neg K \vee \neg J \vee L) \vee E) \wedge \\
((H \vee \neg F) \vee L) & \wedge & ((\neg M \vee \neg L) \vee \neg F) & \wedge & ((\neg K \vee \neg I \vee vA) \vee E)
\end{array}$$

(¬¬Z = Z)

$$\begin{array}{llllll}
(\neg A \vee \neg B \vee \neg C \vee D) & \wedge & (A \vee \neg M \vee L) & \wedge & (\neg F \vee \neg E \vee \neg D) \wedge \\
(\neg G \vee \neg M \vee C) & \wedge & (B \vee \neg F \vee \neg H) & \wedge & (D \vee \neg B \vee \neg E \vee G) \wedge \\
(\neg M \vee I \vee J) \wedge & & (\neg H \vee \neg M \vee K) & \wedge & (\neg K \vee \neg J \vee L \vee E) \wedge \\
(H \vee \neg F \vee L) \wedge & & (\neg M \vee \neg L \vee \neg F) & \wedge & (\neg K \vee \neg I \vee \neg A \vee E)
\end{array}$$

(removing additional brackets)

This is now in NNF,
and CNF

Re-writing directly in clausal form gives …
$\{ [\neg A ; \neg B ; \neg C ; D],\ [A ; \neg M ; L],\ [\neg F ; \neg E ; \neg D],\ [\neg G ; \neg M ; C],\ [B ; \neg F ; \neg H],$
$[D ; \neg B ; \neg E ; G],\ [\neg M ; I ; J],\ [\neg H ; \neg M ; K],\ [\neg K ; \neg J ; L ; E],\ [H ; \neg F ; L],$
$[\neg M ; \neg L ; \neg F],\ [\neg K ; \neg I ; \neg A ; E] \}$

The original above formula has the form X ⊨ Y
To decide whether this statement is satisfiable we need to show that …

X ∪ {¬Y} is unsatifiable

$$\begin{array}{llll}
\neg Y & = & \neg (M \rightarrow \neg F) & \\
& = & \neg (\neg M \vee \neg F) & \text{(definition of} \rightarrow) \\
& = & \neg\neg M \wedge \neg\neg F & \text{(deMorgan's law)} \\
& = & M \wedge F & (\neg\neg Z = Z)
\end{array}$$

so, the full clausal form to enter resolution for the original expression, is
$\{ [\neg A ; \neg B ; \neg C ; D],\ [A ; \neg M ; L],\ [\neg F ; \neg E ; \neg D],\ [\neg G ; \neg M ; C],\ [B ; \neg F ; \neg H],$
$[D ; \neg B ; \neg E ; G],\ [\neg M ; I ; J],\ [\neg H ; \neg M ; K],\ [\neg K ; \neg J ; L ; E],\ [H ; \neg F ; L],$
$[\neg M ; \neg L ; \neg F],\ [\neg K ; \neg I ; \neg A ; E],$
$[M],\ [F] \}$

✓

(b) Using an ordering or a selection function, or both, describe a strategy for minimizing the number of inferences necessary to find a contradiction. How many inferences did you need to prove the problem?

{
[¬A ; ¬B ; ¬C ; D], (1)
[A ; ¬M ; L], (2)
[¬F ; ¬E ; ¬D], (3)
[¬G ; ¬M ; C], (4)
[B ; ¬F ; ¬H], (5)
[D ; ¬B ; ¬E ; G], (6)
[¬M ; I ; J], (7)
[¬H ; ¬M ; K], (8)
[¬K ; ¬J ; L ; E], (9)
[H ; ¬F ; L], (10)
[¬M ; ¬L ; ¬F], (11)
[¬K ; ¬I ; ¬A ; E], (12)
[M], (13)
[F] (14)
}

Choosing to use only an ordering, and using the strategy that negative literals are maximal in any clause as infrequently as possible.
Going through the clauses gives
D ≻ A ; B ; C (1)
A ; L ≻ M (2)
C ≻ G ; M (4)
B ≻ F ; H] (5)
D ; G ≻ B ; E (6)
I ; J ≻ M (7)
K ≻ H ; M (8)
L : E ≻ K ; J (9)
L; H ≻ F (10)
E ≻ K ; I ; A (12)

Combining these requirements gives … (one of many possible combinations)
D ≻ C ≻ G ≻ E ≻ A ≻ L ≻ B ≻ K ≻ H ≻ J ≻ I ≻ F ≻ M

So, marking the strictly maximal clauses {
[¬A ; ¬B ; ¬C ; $\underline{D}$], (1)
[$\underline{A}$; ¬M ; L], (2)
[¬F ; ¬E ; $\underline{\neg D}$], (3)
[¬G ; ¬M ; $\underline{C}$], (4)
[$\underline{B}$; ¬F ; ¬H], (5)
[$\underline{D}$; ¬B ; ¬E ; G], (6)
[¬M ; I ; $\underline{J}$], (7)
[¬H ; ¬M ; $\underline{K}$], (8)
[¬K ; ¬J ; L ; $\underline{E}$], (9)
[H ; ¬F ; $\underline{L}$], (10)

[¬M ; ¬L ; ¬F], (11)
[¬K ; ¬I ; ¬A ; E], (12)
[M], (13)
[F] (14)
}

Now resolution is deterministic.(performing factoring when required)

15. Res(1,3)=[¬A ; ¬B ; ¬C ; ¬F ; ¬E]
16. Res(3,6)=[¬F ; ¬B ; ¬E ; G]
17. Res(10,11)=[H ; ¬M ; ¬F]
18. Res(4,15)=[¬G ; ¬M ; ¬A ; ¬B ; ¬F ; ¬E]
19. Res(16,18)=[¬M ; ¬A ; ¬B ; ¬F ; ¬E]
20. Res(9,19)=[¬F ; ¬M ; ¬A ; ¬B ; ¬K ; ¬J ; L]
21. Res(12,19)=[¬K ; ¬I ; ¬M ; ¬A ; ¬B ; ¬F]
22. Res(2,20)=[¬F ; ¬M ; ¬B ; ¬K ; ¬J ; L]
23. Res(2,21)=[L ; ¬K ; ¬I ; ¬M ; ¬B ; ¬F]
24. Res(11,22)=[¬M ; ¬F ; ¬B ; ¬K ; ¬J]
25. Res(11,23)=[¬K ; ¬I ; ¬M ; ¬B ; ¬F]
26. Res(5,24)=[¬H ; ¬M ; ¬F ; ¬K ; ¬J]
27. Res(5,25)=[¬H ; ¬K ; ¬I ; ¬M ; ¬F]
28. Res(8,26)=[¬H ; ¬M ; ¬F ; ¬J]
29. Res(8,27)=[¬H ; ¬I ; ¬M ; ¬F]
30. Res(17,28)=[¬M ; ¬F ; ¬J]
31. Res(17,29)=[¬I ; ¬M ; ¬F]
32. Res(7,30)=[¬M ; I ; ¬F]
33. Res(31,32)=[¬M ; ¬F]
34. Res(14,33)=[¬M]
35. Res(34,13)=[]

21 inference steps (neglecting negative factoring)

Consider a scheme that makes negative literals maximal as often as possible in the original. Then run the derivation until a problem occurs, and swap the predicates that cause the problem in the ordering list, and reassign and restart. The ordering given by this procedure is … ?

M ≻ F ≻ L ≻ H ≻ B ≻ K ≻ A ≻ I ≻ J ≻ E ≻ D ≻ C ≻ G

This approach takes on-board the lessons from the scheme above, where many negative factoring are hidden in the inference count.

{
[¬A ; ¬B ; ¬C ; D], (1)
[A ; ¬M ; L], (2)
[¬F ; ¬E ; ¬D], (3)
[¬G ; ¬M ; C], (4)
[B ; ¬F ; ¬H], (5)
[D ; ¬B ; ¬E ; G], (6)
[¬M ; I ; J], (7)
[¬H ; ¬M ; K], (8)
[¬K ; ¬J ; L ; E], (9)
[H ; ¬F ; L], (10)
[¬M ; ¬L ; ¬F], (11)
[¬K ; ¬I ; ¬A ; E], (12)*
[M], (13)
[F] (14)
}

Now resolution is deterministic

15 Res(13,2)=[A ; L] =[M; A ; ¬M ; L]
16 Res(13,4)=[¬G ; C] =[M; ¬G ; ¬M ; C]

17	Res(13,7)=[I ; J]	=[M; ¬M ; I ; J]
18	Res(13,8)=[¬H ; K]	=[M; ¬H ; ¬M ; K]
19	Res(13,11)=[¬L ; ¬F]	=[M; ¬M ; ¬L ; ¬F]
20	Res(14,3)=[¬E ; ¬D]	=[F; ¬F ; ¬E ; ¬D]
21	Res(14,5)=[B ; ¬H]	=[F; B ; ¬F ; ¬H]
22	Res(14,10)=[H ; L]	=[F; H ; ¬F ; L]
23	Res(19,14)=[¬L]	= [¬L ; ¬F][F]
24	Res(23,9)=[¬K ; ¬J ; E]	=[¬L] [¬K ; ¬J ; L ; E]
25	Res(23,15)=[A]	=[¬L] [A ; L]
26	Res(23,22)=[H]	=[¬L] [H ; L]
27	Res(18,26)=[K]	=[¬H ; K][H]
28	Res(21,26)=[B]	=[B ; ¬H][H]
29	Res(12,27)=[¬I ; ¬A ; E]	=[¬K ; ¬I ; ¬A ; E][K]
30	Res(1,28,)=[¬A ; ¬C ; D]	=[¬A ; ¬B ; ¬C ; D][B]
31	Res(6,28)=[D ; ¬E ; G]	=[D ; ¬B ; ¬E ; G][B]
32	Res(12,27)=[¬I ; ¬A ; E]	=[¬K ; ¬I ; ¬A ; E][K]
33	Res(24,27)=[¬J ; E]	=[¬K ; ¬J ; E][K]
34	Res(25,30)=[¬C ; D]	=[¬A ; ¬C ; D][A]
35	Res(25,29)=[¬I ; E]	=[¬I ; ¬A ; E][A]
36	Res(17,35)=[J, E]	=[I ; J] [¬I ; E]
37	Res(33,36)=[E, E]	=[J; E][¬J ; E]
38	Fac(37)=[E]	=[E ; E]
39	Res(20,38)=[¬D]	=[¬E ; ¬D][E]
40	Res(34,39)=[¬C]	=[¬C ; D] [¬D]
41	Res(16,40)=[¬G]	=[¬G ; C] [¬C]
42	Res(31,38)=[D ; G]	=[D ; ¬E ; G] [E]
43	Res(42,39)=[G]	=[D ; G] [¬D]
44	Res(41,43)=[]	=[¬G] [G]

30 inference steps

In this scheme the number of inferences does include any 'hidden' negative factoring, and so is much better than the first scheme.

Several selection schemes were considered, but none gave anything as efficient as the second scheme above, including when combined with ordering.

(c) Use MSPASS to find a proof of the minimal length and submit the input file together with the proof by email. Specify which options were used to obtain the proof.

This question has been **withdrawn**, but I include my preliminary MSPASS files and output

not received

8

11. Use the expansion rules from Lecture II.8 to compute <u>maximal</u>, <u>strict</u> semantic tableaux for the following sets of formulae and explain whether the branches/tableaux are closed. Justify steps in the derivation. (presumably this really means lecture 9)

(a) $\{\neg((\neg p \rightarrow q) \rightarrow (p \vee q))\}$

1: $\neg((\neg p \rightarrow q) \rightarrow (p \vee q))$

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

3: $\neg(p \vee q)$

4: $\neg p$

5: $\neg q$

6: $\neg\neg p$ 7: q

8: p

!

lines 2 & 3. applying α-rule on 1: $\boxed{\alpha \text{ is } \neg(F \rightarrow G) \equiv (\alpha_1 \text{ is } F) \text{ and } (\alpha_2 \text{ is } \neg G)}$

lines 4 & 5. applying α-rule on 3: $\boxed{\alpha \text{ is } \neg(F \vee G) \equiv (\alpha_1 \text{ is } \neg F) \text{ and } (\alpha_2 \text{ is } \neg G)}$

there are no more rules to apply to lines 4 and 5

lines 6 & 7. applying β-rule on 2: $\boxed{\beta \text{ is } (F \rightarrow G) \equiv (\beta_1 \text{ is } \neg F) \text{ or } (\beta_2 \text{ is } G)}$

there are no more rules to apply to line 7

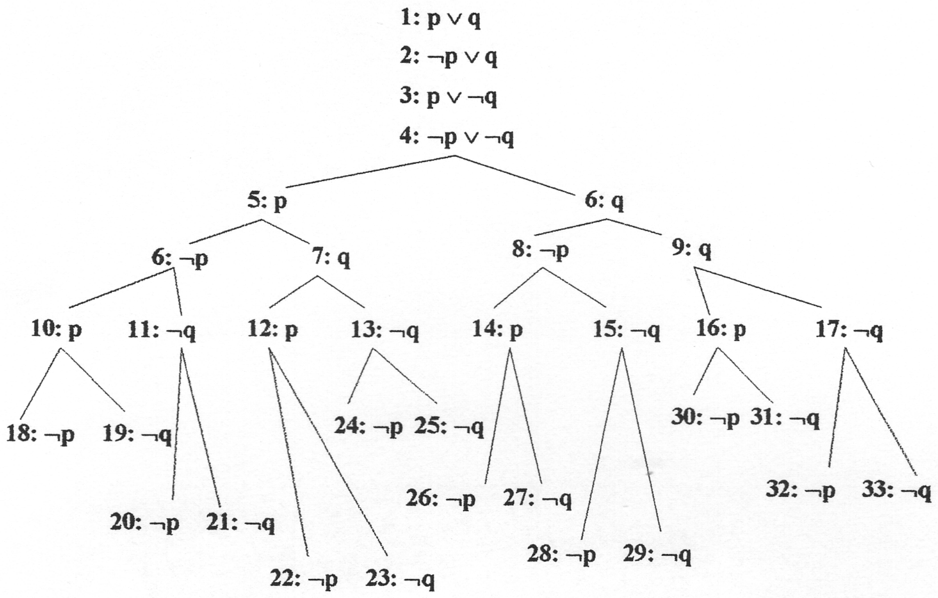
line 8 $\neg\neg p = p$ (by negation elimination) on line 6

there are no more rules to apply to line 8

The left-hand branch is closed because lines 4 & 8 are complimentary ($\neg p$ and p).
The right-hand branch is closed because line 5 and 7 are complimentary ($\neg q$ and q).
This accounts for all the branches.
Hence the tableau is closed.
(Hence the original formula from, which the clauses above are derived, is satisfiable).

This tableau is maximal because all the α- and β-rules have been applied, and strict because they have only be applied at most once.

(b) $\{ p \vee q, \neg p \vee q, p \vee \neg q, \neg p \vee \neg q \}$



All of lines 1 to 4 are β formulae of the type $\boxed{\beta \text{ is } (F \vee G) \text{ gives } \beta_1 \text{ is } (F) \text{ or } \beta_2 \text{ is } (G)}$.
Lines 5 and 6 arise from applying the rule to line 1
Lines 6 & 7, 8 & 9 arise from applying the rule to line 2.
Lines 10,11,12 & 13, 14,15,16 & 17 arise from applying the rule to line 3.
Lines 18,19,20,21,22,23,24 &25, 26,27,28,29,30,31,32 &33 arise from applying the rule to line 4.

-The branch leading to 18, and to 19, and to 20, and to 21 is closed by lines 5 and 6 (p ,¬p)
-The branch leading to 22 is closed by lines 5 and 22 (p ,¬p)
-The branch leading to 23 is closed by lines 7 and 23 (q ,¬q)
-The branch leading to 24, and to 25 is closed by lines 7 and 13 (q ,¬q)
-The branch leading to 26, and to 27 is closed by lines 8 and 14 (p ,¬p)
-The branch leading to 28, and to 29 is closed by lines 6 and 15 (q ,¬q)
-The branch leading to 30 is closed by lines 16 and 30 (p ,¬p)
-The branch leading to 31 is closed by lines 6 and 31 (q ,¬q)
-The branch leading to 32, and to 33 is closed by line 6 and 17 (q ,¬q)

This accounts for all the branches. Hence the tableau is closed. ✓
As before, the derivation is strict and maximal.

⑨

12. Prove that the following set of formulae is unsatisfiable by using first-order semantic tableaux. Justify each step.

$$\{\neg[(\forall x\, p(x) \wedge \exists y\, q(y)) \rightarrow (p(f(u)) \wedge \exists z\, q(z))]\}$$

Usually the first step in a proof is to negate the original formula, but here the negated formula has already been given

1: $\{\neg[(\forall x\, p(x) \wedge \exists y\, q(y)) \rightarrow (p(f(u)) \wedge \exists z\, q(z))]\}$

2: $(\forall x\, p(x) \wedge \exists y\, q(y))$

3: $\neg(p(f(u)) \wedge \exists z\, q(z))$

4: $\forall x\, p(x)$

5: $\exists y\, q(y)$

6: $p(\underline{f(u)})$ X WRONG.

7: $q(v_1)$

8: $\neg p(f(u))$ | 9: $\neg\exists z\, q(z)$

10: $\neg \underline{q(v_1)}$ X

lines 2 & 3. applying α-rule on 1: $\boxed{\alpha \text{ is } \neg(F \rightarrow G) \equiv (\alpha_1 \text{ is } F) \text{ and } (\alpha_2 \text{ is } \neg G)}$

lines 4 & 5. applying α-rule on 2: $\boxed{\alpha \text{ is } (F \wedge G) \equiv (\alpha_1 \text{ is } F) \text{ and } (\alpha_2 \text{ is } G)}$

line 6 applying γ–rule on 4: $\boxed{\gamma \text{ is } \forall x F \equiv \gamma(t) \text{ is } F[x/t]}$ X
substitution used $\{x/f(u)\}$ X
for γ–rules, we can make *any* substitution, and it is clear that this will be useful later in closing a branch

mix-up.

line 7 applying δ–rule on line 5: $\boxed{\delta \text{ is } \exists x F \equiv \delta(t) \text{ is } F[x/t]}$
substitution used $\{y/v_1\}$ X
for δ–rules, we can make *only* a substitution to that has not previously been introduced. The choice is in how to name this unique variable, here v_1

lines 8 & 9. applying β-rule on 3: $\boxed{\beta \text{ is } \neg(F \wedge G) \equiv (\beta_1 \text{ is } \neg F) \text{ or } (\beta_2 \text{ is } \neg G)}$

This *closes* the left-hand branch since lines 6 and 8 are complimentary ($\neg p(f(u)), p(f(u))$) be more precise.

line 11 applying γ–rule on line 9: $\boxed{\gamma \text{ is } \neg\exists x F \equiv \gamma(t) \text{ is } \neg F[x/t]}$
substitution used $\{x/v_1\}$ X in general, not a good strategy.
for γ–rules, we can make *any* substitution, and it is clear that this will be in closing a branch …

This *closes* the right-hand branch since lines 10 and 11 (complimentary $\neg q(v_1), q(v_1)$)

The tableau is *closed*, and hence, the formula is *unsatisfiable*.

⑦