1993:3:3

1. $\{\neg P(x), Q(x)\}$
2. $\{\neg P(x), \neg Q(x), P(fx)\}$.
3. $\{P(b)\}$
4. $\{\neg P(f^4x)\}$.

   1 and 2 resolve to give
5. $\{\neg P(x), P(fx)\}$. First reletter this to get
6. $\{\neg P(w), P(fw)\}$

   Resolve 5 and 6 by unifying $w \to fx$, cut against $P(fx)$ to get
7. $\{\neg P(x), P(f^2x)\}$. First reletter this to get
8. $\{\neg P(w), P(f^2w)\}$.

   Resolve 7 and 8 by unifying $w \to f^2x$ cut against $P(f^2x)$ to get
9. $\{\neg P(x), P(f^4x)\}$. Resolve with 3 to get
10. $\{P(f^4b)\}$. Resolve with 4 to get the empty clause
11. $\{\}$.

That's the clever way to do it. I think what PROLOG does is something more like this. It cuts 4 against 2 to get $\{\neg P(f^3x), \neg Q(f^3x)\}$ and cuts against 1 to get $\{\neg P(f^3x)\}$.

Then repeat until you get $\{\neg P(x)\}$ which you can cut against 3. The point is that at each stage PROLOG only ever cuts the current goal clause against something it was given to start with. That way it has only a linear search for a cut at each stage instead of a quadratic one. I'm not sure what sort of relettering PROLOG does, and whether it can make copies of clauses, and reletter one to cut against the other as above. It certainly only ever does linear resolution.

(a) How long does it take?

(b)

(c)

(d)

$$\neg[(\forall y \exists x)\neg(p(x,y) \longleftrightarrow \neg(\exists z)(p(x,z) \wedge p(z,x)))]$$

Rewrite to get rid of the biconditional

$$\neg[(\forall y \exists x)\neg(p(x,y) \vee \neg(\exists z)(p(x,z) \wedge p(z,x)) \wedge p(x,y) \vee \neg(\exists z)(p(x,z) \wedge p(z,x))]$$

push in $\neg$

$$[(\exists y \forall x) \neg (p(x,y) \vee \neg (\exists z)(p(x,z) \wedge p(z,x)) \wedge p(x,y) \vee \neg (\exists z)(p(x,z) \wedge p(z,x))]$$

1996:5:10

$$(\forall z)(\exists x)(\forall y)((P(y) \to Q(z)) \to (P(x) \to Q(x)))$$

Given that the decision problem for first-order logic is undecidable, you haven't much chance of finding a proof of something or a convincing refutation of it unless you postpone work on it until you have a feel for what it is saying.

First we notice that as long as there is an x s.t. $Q(x)$ we can take that element to be a witness to the '$\exists x$' no matter what z is. This is because the truth of '$Q(x)$' ensures the truth of the whole conditional. On the other hand even if *nothing* is Q we are still OK as long as nothing is P—because the falsity of '$P(x)$' ensures the truth of the consequent of the main conditional. There remains the case where $(\forall x)(\neg Q(x))$ and $(\exists x)(P(x))$. But it's easy to check that in that case the whole conditional comes out true too.

So we can approach the search for a sequent calculus proof confident that there is one to be found.

Clearly the only thing we can do with

$$(\forall z)(\exists x)(\forall y)((P(y) \to Q(z)) \to (P(x) \to Q(x)))$$

is a $\forall$-R getting

$$\vdash (\exists x)(\forall y)((P(y) \to Q(a)) \to (P(x) \to Q(x)))$$

(I have relettered 'z' to 'a' for no particular reason). We could have got this by $\exists$-R by replacing 'a' by 'x' so that it came from

$$\vdash (\forall y)((P(y) \to Q(a)) \to (P(a) \to Q(a)))$$

but this doesn't appear to be valid. So we presumably have to keep an extra copy of '$\vdash (\exists x)(\forall y)((P(y) \to Q(a)) \to (P(x) \to Q(x)))$' and we got it from

$$\vdash (\exists x)(\forall y)((P(y) \to Q(a)) \to (P(x) \to Q(x))), \quad (\forall y)((P(y) \to Q(a)) \to (P(a) \to Q(a)))$$

which came by $\forall$-R from

$$\vdash (\exists x)(\forall y)((P(y) \to Q(a)) \to (P(x) \to Q(x))), \quad ((P(b) \to Q(a)) \to (P(a) \to Q(a)))$$

This obviously came from an $\exists$-R:

$$\vdash (\forall y)((P(y) \to Q(a)) \to (P(b) \to Q(b))), \quad ((P(b) \to Q(a)) \to (P(a) \to Q(a)))$$

…where i'm assuming the 'x' came from the 'b' we've already seen.

and this must've come from a $\forall$-R with a new variable:

$$\vdash (P(c) \to Q(a)) \to (P(b) \to Q(b))), \quad ((P(b) \to Q(a)) \to (P(a) \to Q(a)))$$

and now we've got all the quantifiers out of the way and have only the propositional rules to worry about: pretty straightforward from here. Four applications of $\to$-R take us to

$$P(c) \to Q(a),\ P(b) \to Q(a),\ P(b),\ P(a) \vdash Q(b),\ Q(a)$$

and if we break up the '$P(b) \to Q(a)$' on the left we get the two initial sequents:

$$P(c) \to Q(a),\ P(b),\ P(a),\ \underline{Q(a)} \vdash Q(b),\ Q(a)$$

and

$$P(c) \to Q(a),\ P(b),\ P(a) \vdash \underline{P(b)},\ Q(b),\ Q(a)$$

…where i have underlined the two formulæ that get glued together by the $\to$-L rule.

1996:6:10

Davis-Putnam: This procedure has three main steps:

1. Delete tautological clauses;
2. Delete unit clauses $\{A\}$ and remove $\neg A$ from all clauses. This is safe because a unit clause $\{A\}$ can be satisfied only if $A \mapsto$ **true** and once that is done A does not need to be considered further.
3. Delete any formula containing pure literals. (If a literal appears always positively or always negatively we can send it to **true** or to **false** without compromising later efforts to find an interpretation of the formula).

If a point is reached where none of the rules above can be applied, a variable is selected arbitrarily for a **case split** and the proof proceeds along both resulting clause sets. We will be happy if *either* resolves to the empty clause. This algorithm terminates because each case split removes a literal.

In this example, we have no tautological clauses or pure literals, so we start with a case split, arbitrarily selecting P to split. If P is true, our clauses are $\{R\}, \{\neg R\}$. We delete unit clause $\{R\}$, and then delete $\neg R$ from all clauses; we are left with the empty clause, which constitutes a refutation of the clause set (the empty disjunction), so the formula is valid. The P false case proceeds similarly, with Q for R. Resolution: There is only one rule of inference in resolution:

$$\frac{\{B, A\}\{\neg B, C\}}{\{A, C\}}$$

The algorithm terminates because as soon as a point is reached where the rule cannot be applied, the clause set is established as satisfiable. Repeatedly applying this rule to the given clause set:

$$\frac{\{\neg P, R\}\{P, \neg Q\}}{\{R, \neg Q\}}$$

$$\frac{\{\neg P, \neg R\}\{P, Q\}}{\{\neg R, Q\}}$$

$$\frac{\{R, \neg Q\}\{\neg R, \neg Q\}}{\Box}$$

The empty clause ($\Box$) is a contradiction: we have refuted the clause set and so proved the original formula.

Part of an answer to another question

Let A^* represent the formula A, converted into polynomial representation. First we note that in arithmetic mod 2, $x^2 \equiv x$, as $0^2 = 0 \equiv 0$ and $1^2 = 1 \equiv 1$, and all integers are congruent to 0 or 1 modulo 2. Now $(\neg A)^*$ is $1+A^*$, $(A \wedge B)^*$ is $A^* \cdot B^*$, $(A \vee B)^*$ becomes $A^*+B^*+A^*B^*$, $A \to B$ is $1+A^*+A^*B^*$, and $A \leftrightarrow B$ is $((1-A^*)+B^*) \cdot ((1-B^*)+A^*)$, which simplifies to $1+2A^*B^*-A^*-B^*$ and thence to $1+A^*+B^*$. Recursively applying these rules to any formula will convert it to equivalent polynomial form. $(A \wedge B) \leftrightarrow (B \wedge A)$ translates into $1+2(A^*B^*)^2 - A^*B^* - B^*A^* = 1$, hence the formula is a tautology. $A \leftrightarrow A$ translates into 1. $1 \leftrightarrow A$ translates into $1+2A^*-A^*-1 = A^*$. So if we adopt the notation $(A \leftrightarrow A)^n$, to represent formulae of the given type where $\leftrightarrow$ appears n times, we get: $(A \leftrightarrow A)^n = 1$ (n odd) or A (n even), $n \geq 0$. This works for $n = 0$, which is just the formula A.

1998:6:10

Clause 1 tells us that if x pees on itself it pees on a. Clause 2 tells us that if x does *not* pee on itself then it pees on fa. This drops a broad hint that perhaps a is $\{x : x \in x\}$ and fa is $\{x : x \notin x\}$. Clause 3 tells us that nothing pees on both a and fa which is starting to look good. Now ask whether or not $P(fa, fa)$? Well, $P(fa, fa) \to P(fa, a)$ by clause 1. Then use clause 3 to infer $\neg P(fa, fa)$ whence $\neg P(fa, fa)$. But then clause 2 tells us that $P(fa, fa)$ after all.

The final part.

Three clauses:

$$\{\neg P(x,x), P(x,a)\}, \{P(x,x), \neg P(x, f(a))\}\{\neg P(y, f(x)), \neg P(y,x)\}$$

I think this is Russell's paradox. $P(x,y)$ is $x \in y$; a is the complement of the Russell class, and f is complementation.

In the third clause make the substitutions a/x and $f(a)/y$ to get $\{\neg P(f(a), f(a)), \neg P(f(a), a)\}$

In the first clause make the substitution $f(a)/x$ to get $\{\neg P(f(a), f(a)), P(f(a), a)\}$ and resolve on $P(f(a), a)$ to get $\{\neg P(f(a), f(a))\}$.

In the second clause make the substitution $f(a)/x$ to get
$\{\neg P(f(a), f(a)), P(f(a), f(a))\}$ which is of course
$\{\neg P(f(a), f(a)), P(f(a), f(a))\}$
resolves with the current goal clause to get
damn

Question ????

0.0.1 Question ??????

The natural deduction proof i favour looks like this:

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{[p] \quad [\tau(x) \to \bot]}{p \wedge (\tau(x) \to \bot)}}{(\tau(x) \to \bot)}}{p \to (\tau(x) \to \bot)}}{Tp \to T(\tau(x) \to \bot)^a} \qquad \dfrac{}{Tp}}{T(\tau(x) \to \bot)}}{\tau(x)^m} \qquad \dfrac{}{[(\tau(x) \to \bot)]}}{\bot}}{(\tau(x) \to \bot) \to \bot}}{(\forall x)((\tau(x) \to \bot) \to \bot)}
$$

... where m superscript betrays a use of the mixed rule; a a use of the affixing rule. Cancelled assumptions are enclosed in [square brackets] as usual. (There may be other proofs as well.)

0.0.2 ??????

The shortest sequent calculus proof i can find is the following.

$$
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{p, \tau(x) \vdash \tau(x)}{p \vdash \tau(x), \neg\tau(x)}}{Tp, \vdash \tau(x), T(\neg\tau(x))}}{Tp, \vdash \tau(x), \tau(x)}}{Tp \vdash \tau(x)}}{Tp \vdash \forall x \tau(x)}
$$

0.0.3 ??????

Prove that a premiss of the form Tp really is needed.

Notice that since neither the affixing rule nor the mixed rule have anything like $T\phi$ as a conclusion, we can obtain models of this calculus in which Tp is always false (The modal logicians express this by saying that T is a *falsum* operator) and $\tau(x)$ is always false. Accordingly we cannot expect to be able to prove that even one thing is τ without some extra premisses.

????

Use resolution to prove that a graph and its complement cannot both be disconnected.

Robert Thatcher's model answer (edited by me)

Suppose G is a graph such that G and $\overline{G}$ are both disconnected. We will derive a contradiction by resolution.

If G is disconnected then there are vertices a and b which are not connected in G. If $\overline{G}$ is disconnected then there are vertices c and d which are disconnected in $\overline{G}$.

Let us have six propositional letters: ab, ac, ad, bc, bd, cd. The intended interpretation is that ab (for example) means that the edge ab belongs to the edge set of G.

First consider G. The first thing we know is that the edge ab is **not** in the edge set of G, hence we have the clause $\neg ab$. Since we know that a and b are disconnected in G, we cannot allow any indirect paths from a to b. There are two possible lengths of indirect path involving 1 or 2 indirect vertices. (It will turn out that we can get our desired contradiction without considering indirect paths that are longer, but we don't know that yet, and are just hoping for the best!) This tells us that $(\neg ac \vee \neg bc)$, or, in resolution jargon, $\{\neg ac, \neg bc\}$. Similarly we may add $\{\neg ad, \neg bd\}$. The paths involving 2 vertices are $acdb$ and $adcb$. We already know that the path cd must be present (since it cannot be in $\overline{G}$). Therefore we may add the clauses $\{\neg ac, \neg bd\}$ and $\{\neg ad, \neg cb\}$.

Now consider $\overline{G}$. This cannot have cd, so we can add $\{cd\}$ (this is not negated, since we are now considering edges that are not in $\overline{G}$, and hence must be in G). Similarly, we cannot have any indirect connections from c to d, so we cannot have the paths cad, cbd, $cabd$ and $cbad$. Since we know that ab cannot be in the graph, we can write these as the clauses: $\{ac, ad\}$, $\{bc, bd\}$, $\{ac, bd\}$ and $\{bc, ad\}$. Note that the last two do not contain ab since we know that we **must** have $\neg ab$ by choice of a and b.

So now we have a set of clauses representing the conditions that need to be satisfied if both G and $\overline{G}$ are to be disconnected. To recapitulate, these are:

$$\begin{gathered}\{\neg ab\}\ \{\neg ac, \neg bc\}\ \{\neg ad, \neg bd\}\ \{\neg ad, \neg bc\}\ \{\neg ac, \neg bd\}\\ \{cd\}\ \{ac, ad\}\ \{bc, bd\}\ \{ac, bd\}\ \{ad, bc\}\end{gathered}$$

Now we may combine these clauses (carefully) using resolution – the choice of clauses to resolve is crucial, since it is very easy to end up with many useless clauses of the form $\{A, \neg A\}$.

$$\dfrac{\dfrac{\{\neg ac, \neg bc\} \qquad \{bc, bd\}}{\{\neg ac, bd\}} \qquad \{\neg ac, \neg bd\}}{\{\neg ac\}} \tag{1}$$

$$\dfrac{\dfrac{\{\neg ad, \neg bd\} \qquad \{bd, bc\}}{\{\neg ad, bc\}} \qquad \{\neg ad, \neg bc\}}{\{\neg ad\}} \tag{2}$$

Now we have two literal clauses, we can use them to derive a contradiction:

$$\dfrac{\dfrac{\{ac, ad\} \qquad \{\neg ac\}}{\{ad\}} \qquad \{\neg ad\}}{\perp} \tag{3}$$

We have derived the empty clause.

Note that only six of the original 10 clauses were required for the resolution – partly because the two simple clauses are disjoint from the remaining eight (due to the way we initially wrote down the clauses – in a sense they have already been used in a resolution). The remaining two clauses are superfluous: they do not provide any information beyond the original eight. The clauses found when considering the 3-stage path in $\overline{G}$ are complementary to those found when considering similar paths in G, and hence if resolved in any way with them, give useless clauses of the form $\{A, \neg A\}$.

1 Answers to Larry's exercises

$P(a), P(b) \vdash P(a), P(b), P(a), P(b)$

$P(a), P(b) \vdash P(a) \wedge P(b), P(a) \wedge P(b)$

$\vdash P(a) \to P(a) \wedge P(b), P(b) \to P(a) \wedge P(b)$ ($\wedge$-R twice)

$\vdash (\exists z)(P(z) \to P(a) \wedge P(b)), (\exists z)(P(z) \to P(a) \wedge P(b))$ $\exists$-R twice

$\vdash (\exists z)(P(z) \to P(a) \wedge P(b))$ (contraction)

(This needs to be properly set using the stylefile bussproofs....

Most of these answer are by Dave Tonge. Not all, and some of his have been mutilated by me.

$\star$ **1** $\star$ *Is the formula $A \to \neg A$ satisfiable? Is it valid?*

The case where A is false satisfies. The case where A is true does not satisfy. Therefore the expression is satisfiable but not valid.

$\star$ **3** $\star$ *Work out the details above.*

Negate and convert $(A_1 \wedge \cdots \wedge A_k) \to B$ to CNF

Negate to give $\neg((A_1 \wedge \cdots \wedge A_k) \to B)$

Eliminate $\to$ to give $\neg(\neg(A_1 \wedge \cdots \wedge A_k) \vee B)$

Push in negations $(A_1 \wedge \cdots \wedge A_k) \wedge \neg B$

Remove parentheses to give $A_1 \wedge \cdots \wedge A_k \wedge \neg B$

Convert $M \to K \wedge P$ to clausal form.

Split into two formulae, $M \to K$ and $M \to P$.

Eliminate $\to$s to give $\neg M \vee K$ and $\neg M \vee P$.

Convert to clauses $\{\neg M, K\}$ and $\{\neg M, P\}$.

$\star$ **5** $\star$ *Write down a formula that is true in every domain that contains at least m elements. Write down a formula that is true in every domain that contains at most most m elements.*

At least m:

$$(\exists x_1 \ldots x_m)(\bigwedge_{k \neq j}(a_j \neq a_k)^1$$

An answer for the next is obviously obtainable by increasing m by one and negating!

At most m:

$$(\forall x_1 \ldots x_{m+1})(\bigvee_{i \neq j < m} x_j = x_i)$$

Many readers find the following more natural

$$(\exists x_1 \ldots x_m)(\forall y)(\bigvee_{1 \leq i \leq m} y = x_i)$$

This formula is logically more complicated (it has an alternation of quantifiers) but is shorter.

A brief question to ask yourself: how rapidly does the formula grow with n?

$\star$ **6** $\star$ *Verify these equivalences by appealing to the truth definition for first order logic.*

There are too many of these, so I'll just do the infinitary de Morgan law $\mathcal{M}_V \models \neg((\forall x)A) = ((\exists x)\neg A)$.

To show this we have to show that $\mathcal{M}_V \models \neg((\forall x)A)$ is equivalent to $\mathcal{M}_V \models ((\exists x)\neg A)$.

The first half becomes: for all $m \in M$ such that $\mathcal{M}_{V\{m/x\}} \models A$ does not hold. The second half becomes there exists an $m \in M$ for which $\mathcal{M}_{V\{m/x\}} \models A$ does not hold.

These two are plainly equivalent for if the first one does not hold then there there will not exist an m for which the second holds. Similarly, if the second is true then the first will not hold for all ms (for it won't hold for the m given by the first).

$\star$ **7** $\star$ *Explain why the following are not equivalences. Are they implications? In which direction?*

$((\forall x)A) \vee ((\forall x)B)? =?(\forall x)(A \vee B)$

$((\exists x)A) \wedge ((\exists x)B)? =?(\exists x)(A \wedge B)$

First one: The RHS could be true if A were true and B were false for a particular x. Thus, B would not be true for all x. There might be another x for which A were false and B true. Thus A is not true for all x. Thus the LHS can be false although the RHS is true. Because this case exists the two statements are not equivalent. However, there is a left-to-right implication.

[1] The temptation to write this as: $(\exists a_1 \ldots a_m)(\forall j, k < m)(k \neq j \rightarrow a_j \neq a_k)$ must be resisted. This is *not* correct, since the subscripts on the variables are not themselves variables and cannot be bound.

Second: The x for which A might not be the same as the x for which B. Therefore the RHS will could be false even if the LHS is true. The two statements are not equivalent. However, the RHS implies the LHS.

$\star$ **9** $\star$ *Verify that* $\circ$ *is associative and has* [] *for an identity.*

To show associativity we need to show that $(\phi \circ \theta) \circ \sigma = \phi \circ (\phi \circ \theta)$.

If we consider ϕ, θ and σ as functions $f(x)$, $g(x)$ and $h(x)$ which map literals to their substituted values then we get the composition

$$
\begin{aligned}
\lambda x.((f \circ g) \circ h) &= \lambda x.(f \circ (g \circ h)) \\
\lambda x.((\lambda y.f(g(x)))h) &= \lambda x.(f(g \circ h)) \\
\lambda x.(f(g(h(x)))) &= \lambda x.f(g(h(x)))p
\end{aligned}
$$

Which says that they are the same. This relies on our functions returning the literal given as an argument in cases where no substitution has been defined.

To show that [] is the identity we need to consider it as a function g which maps all the argument literals to themselves, without substitution. ϕ remains the function f as before.

$$
\begin{aligned}
f \circ g &= f \\
g(f) &= f \\
f &= f
\end{aligned}
$$

$\star$ **11** $\star$ *Each of the following formulae is satisfiable but not valid. Exhibit a truth assignment that makes the formula true and another truth assignment that makes the formula false.*

$P \to Q$

True for $P = true$ and $Q = true$. False for $P = true$ and $Q = false$.

$P \vee Q \to P \wedge Q$

True for $P = Q = true$. False for $P = true$ and $Q = false$.

$\neg(P \vee Q \vee R)$

True for $P = Q = R = false$. False otherwise.

$\neg(P \wedge Q) \wedge \neg(Q \vee R) \wedge (P \vee R)$

True for $P = true$ and $Q = R = false$. False for $P = Q = R = true$.

$\star$ **12** $\star$ *Convert of the following propositional formulæ into Conjunctive Normal Form and also into Disjunctive Normal Form. For each formula, state whether it is valid, satisfiable, or unsatisfiable; justify each answer.*
$(P \to Q) \wedge (Q \to P)$

To obtain CNF we first eliminate $\to$ to get $(\neg P \vee Q) \wedge (\neg Q \vee P)$.

To obtain DNF we first eliminate $\to$ to get $(\neg P \vee Q) \wedge (\neg Q \vee P)$. Push in conjunctions to get $(\neg P \wedge (\neg Q \vee P)) \vee (Q \wedge (\neg Q \vee P))$. And again to get $(\neg P \wedge P) \vee (\neg P \wedge \neg Q) \vee (Q \wedge \neg Q) \vee (Q \wedge P)$. Remove those which are obviously false to get $(\neg P \wedge \neg Q) \vee (P \wedge Q)$.

This formula is satisfiable—it is satisfied when $P = Q$.

$$((P \wedge Q) \vee R) \wedge (\neg((P \vee R) \wedge (Q \vee R)))$$

Both CNF and DNF require one to push in negations to get

$$((P \wedge Q) \vee R) \wedge (\neg(P \vee R) \vee \neg(Q \vee R))$$

and then

$$((P \wedge Q) \vee R) \wedge ((\neg P \wedge \neg R) \vee (\neg Q \wedge \neg R))$$

.

TO get CNF push in disjunctions to get

$$(P \vee R) \wedge (Q \vee R) \wedge (\neg P \vee \neg Q) \wedge (\neg P \vee \neg R) \wedge (\neg R \vee \neg Q) \wedge (\neg R \vee \neg R)$$

which is

$$(P \vee R) \wedge (Q \vee R) \wedge (\neg P \vee \neg Q) \wedge (\neg R \vee \neg Q) \wedge \neg P \wedge \neg R$$

.

To get DNF push in conjunctions to get $(P \wedge Q \wedge \neg R \wedge \neg R) \vee (P \wedge Q \wedge \neg Q \wedge \neg R) \vee (R \wedge \neg P \wedge \neg R) \vee (R \wedge \neg Q \wedge \neg R)$.

The formula is unsatisfiable—if you look at it in DNF each conjunct has an atom in both negated and unnegated form so all conjuncts must be false so the whole disjunction is always false.

$\neg(P \vee Q \vee R) \vee ((P \wedge Q) \vee R)$

Both CNF and DNF require one to push in negations to get $(\neg P \wedge \neg Q \wedge \neg R) \vee ((P \wedge Q) \vee R)$.

To get CNF we need to push in disjunctions to get $(\neg P \wedge \neg Q \wedge \neg R) \vee ((P \vee R) \wedge (Q \vee R))$ then $((\neg P \wedge \neg Q \wedge \neg R) \vee (P \vee R)) \wedge ((\neg P \wedge \neg Q \wedge \neg R) \vee (Q \vee R))$ and then $(\neg P \vee P \vee R)) \wedge (\neg Q \vee P \vee R) \wedge (\neg R \vee P \vee R) \wedge (\neg P \vee Q \vee R) \wedge (\neg Q \vee Q \vee R) \wedge (\neg R \vee Q \vee R)$ which might as well be $(\neg Q \vee P \vee R) \wedge (\neg P \vee Q \vee R)$.

To get DNF we don't have to do much except expand brackets to $(\neg P \wedge \neg Q \wedge \neg R) \vee (P \wedge Q) \vee R$.

This is satisfiable—it is only false for $P = R = false$, $Q = true$ and $P = true$, $Q = R = false$.

$\neg(P \vee Q \to R) \wedge (P \to R) \wedge (Q \to R)$

Both CNF and DNF need one to get rid of $\to$s to give $\neg(\neg(P \vee Q) \vee R) \wedge (\neg P \vee R) \wedge (\neg Q \vee R)$. Push in negations to get $((P \vee Q) \wedge \neg R) \wedge (\neg P \vee R) \wedge (\neg Q \vee R)$.

We would appear to have the CNF already—$(P \vee Q \vee \neg R) \wedge (\neg P \vee R) \wedge (\neg Q \vee R)$.

To get DNF we ned to push in conjunctions to get $(P \vee Q \vee \neg R) \wedge (\neg P \vee \neg Q) \wedge (\neg P \vee R) \wedge (R \vee \neg Q) \wedge R$. Again to give $(P \vee Q \vee R) \wedge (P \vee R) \wedge (\neg P \vee Q \vee R) \wedge (Q \vee R) \wedge (\neg P \vee \neg Q \vee \neg R)$.

This is satisfiable - for example in the case $P = Q = false, R = true$ but it can be false as it is when $P = Q = R = false$.

$\star$ **14** $\star$ *Prove* $(P \wedge Q \to R) \wedge (P \vee Q \vee R) \to ((P \leftrightarrow Q) \to R)$ *and* $((P \to Q) \to P) \to P$ *by resolution. Show the steps of converting the formula into clauses.*

We have to negate and remove $\to$s first.

$$\neg((P \wedge Q \to R) \wedge (P \vee Q \vee R) \to ((P \leftrightarrow Q) \to R))$$
$$\neg(\neg((\neg(P \wedge Q) \vee R) \wedge (P \vee Q \vee R)) \vee (\neg((\neg P \vee Q) \wedge (\neg Q \vee P)) \vee R))$$
$$((\neg P \vee \neg Q \vee R) \wedge (P \vee Q \vee R)) \wedge \neg((P \wedge \neg Q) \vee (\neg P \wedge Q) \vee R)$$
$$(\neg P \vee \neg Q \vee R) \wedge (P \vee Q \vee R) \wedge (\neg(P \wedge \neg Q) \wedge \neg(\neg P \wedge Q) \wedge \neg R)$$
$$(\neg P \vee \neg Q \vee R) \wedge (P \vee Q \vee R) \wedge (\neg P \vee Q) \wedge (P \vee \neg Q) \wedge \neg R$$

This gives us the clauses $\{\neg P, \neg Q, R\}$, $\{P, Q, R\}$, $\{\neg P, Q\}$, $\{\neg Q, P\}$, $\{\neg R\}$.

If we resolve the last one with the first two we get $\{\neg P, \neg Q\}$ and $\{P, Q\}$. This gives us all four possible clauses starring P and Q so we will certainly get the empty clause. ($\{\}$). We have assumed the negation of the theorem and derived a contradiction. This proves the theorem.

We have to negate and eliminate $\to$s first.

$$\neg(\neg(\neg(\neg P \vee Q) \vee P) \vee P)$$
$$\neg(\neg((P \wedge \neg Q) \vee P) \vee P)$$
$$\neg((\neg(P \wedge \neg Q) \wedge \neg P) \vee P)$$
$$\neg(((\neg P \vee Q) \wedge \neg P) \vee P)$$
$$(\neg((\neg P \vee Q) \wedge \neg P) \wedge \neg P)$$
$$((\neg(\neg P \vee Q) \wedge P) \wedge \neg P)$$
$$(((P \wedge \neg Q) \wedge P) \wedge \neg P)$$
$$P \wedge P \wedge \neg P \wedge \neg Q$$
$$P \wedge \neg P \wedge \neg Q$$

This gives us the clauses $\{P\}$, $\{\neg P\}$ and $\{\neg Q\}$. Resolving $\{P\}$ with $\{\neg P\}$ gives a contradiction ($\{\}$). We have assumed the negation of Peirce's law and derived a contradiction, thus proving the law.

$\star$ **15** $\star$ *Using linear resolution, prove that* $(P \wedge Q) \to (R \wedge S)$ *follows from* $(P \to R) \wedge (Q \to S)$ *and* $R \wedge P \to S$.

The two assumed formulæ and the negated conclusion give us the clauses $\{\neg P, R\}$, $\{\neg Q, S\}$ and $\{\neg R, \neg P, S\}$. We need to resolve these with the clauses given by the negation of the formula we are trying to prove. These clauses are $\{P\}$, $\{Q\}$ and $\{\neg R, \neg S\}$.

Take $\{\neg R, \neg S\}$ and resolve with $\{\neg R, \neg P, S\}$ to get $\{\neg R, \neg P\}$.

Resolve the result with $\{\neg P, R\}$ to get $\{\neg P\}$.

Resolve the result with $\{P\}$ to get a contradiction ($\{\}$). This proves the formula.

⋆ **16** ⋆ *Convert these axioms to clauses, showing all steps. Then prove* $Winterstorm \rightarrow Miserable$ *by resolution:* $Rain \wedge (Windy \vee \neg Umbrella) \rightarrow Wet$*,* $Winterstorm \rightarrow Storm \wedge Cold$*,* $Wet \wedge Cold \rightarrow Miserable$ *and* $Storm \rightarrow Rain \wedge Windy$*.*

First we need to construct all our definite clauses.

$$\begin{aligned}
& (Rain \wedge (Windy \vee \neg Umbrella) \wedge Wet \\
expand\wedge \quad & ((Rain \wedge Windy) \vee (Rain \wedge \neg Umbrella)) \rightarrow Wet \\
remove \rightarrow \quad & \neg((Rain \wedge Windy) \vee (Rain \wedge \neg Umbrella)) \vee Wet \\
& (\neg(Rain \wedge Windy) \wedge \neg(Rain \wedge \neg Umbrella)) \vee Wet \\
& ((\neg Rain \vee \neg Windy) \wedge (\neg Rain \vee \neg Umbrella)) \vee Wet \\
& (\neg Rain \vee \neg Windy \vee Wet) \wedge (\neg Rain \vee \neg Umbrella \vee Wet) \\
clauses\ are \quad & \{\neg Rain, \neg Windy, Wet\} \text{ and } \{\neg Rain, \neg Umbrella, Wet\}
\end{aligned}$$

$$\begin{aligned}
& Winterstorm \rightarrow Storm \wedge Cold \\
remove \rightarrow \quad & \neg Winterstorm \vee (Storm \wedge Cold) \\
expand\wedge \quad & (\neg Winterstorm \wedge Storm) \wedge (\neg Winterstorm \wedge Cold) \\
clauses\ are \quad & \{\neg Winterstorm, Storm\} and \{\neg Winterstorm, Cold\}
\end{aligned}$$

$$\begin{aligned}
& Wet \wedge Cold \rightarrow Miserable \\
remove \rightarrow \quad & \neg(Wet \wedge Cold) \vee Miserable \\
push\ in\ \neg \quad & \neg Wet \vee \neg Cold \vee Miserable \\
clauses\ are \quad & \{\neg Wet, \neg Cold, Miserable\}
\end{aligned}$$

$$\begin{aligned}
& Storm \rightarrow Rain \wedge Windy \\
remove \rightarrow \quad & \neg Storm \vee (Rain \wedge Windy) \\
expand\wedge \quad & (\neg Storm \vee Rain) \wedge (\neg Storm \vee Windy) \\
clauses\ are \quad & \{\neg Storm, Rain\} and \{\neg Storm, Windy\}
\end{aligned}$$

In order to prove $Winterstorm \rightarrow Miserable$ we have to assume its negation and derive a contradiction. So, let's find out the clauses that would give us.

$$\begin{aligned}
& Winterstorm \rightarrow Miserable \\
negate \quad & \neg(Winterstorm \rightarrow Miserable) \\
remove \rightarrow \quad & \neg(\neg Winterstorm \vee Miserable) \\
push\ in \neg \quad & Winterstorm \wedge \neg Miserable \\
clauses\ are \quad & \{Winterstorm\} and \{\neg Miserable\}
\end{aligned}$$

Now, using all the clauses we have gathered we need to use resolution to get a contradiction.

Resolve $\{Winterstorm\}$ with $\{\neg Winterstorm, Storm\}$ to give $\{Storm\}$.

Resolve $\{Winterstorm\}$ with $\{\neg Winterstorm, Cold\}$ to give $\{Cold\}$.

Resolve $\{Storm\}$ with $\{\neg Storm, Windy\}$ to give $\{Windy\}$.

Resolve $\{Storm\}$ with $\{\neg Storm, Rain\}$ to give $\{Rain\}$.

Resolve $\{Rain\}$ with $\{\neg Rain, \neg Windy, Wet\}$ to give $\{\neg Windy, Wet\}$.

Resolve $\{Windy\}$ with $\{\neg Windy, Wet\}$ to give $\{Wet\}$.

Resolve $\{Wet\}$ with $\{\neg Wet, \neg Cold, Miserable\}$ to give $\{\neg Cold, Miserable\}$.

Resolve $\{Cold\}$ with $\{\neg Cold, Miserable\}$ to give $\{Miserable\}$.

Resolve $\{Miserable\}$ with $\{\neg Miserable\}$ to give a contradiction ($\{\}$). This proves the theorem by contradiction of the negated theorem.

$\star$ **17** $\star$ *Let $\sim$ be a 2-place predicate symbol, which we write using infix notation: for instance, $x \sim y$ rather than $\sim (x, y)$. Consider the following axioms:*

$$(\forall x) \qquad x \sim x \tag{4}$$

$$(\forall xy) \qquad (x \sim y \rightarrow y \sim x) \tag{5}$$

$$(\forall xyz) \qquad (x \sim y \wedge y \sim z \rightarrow x \sim z) \tag{6}$$

Let the universe be the set of natural numbers, $\mathbb{N} = \{0, 1, 2, \cdots\}$. Which axioms hold if the interpretation of $\sim$ is...

1. *the empty relation, ϕ?*

   (1) does not hold. (2), (3) hold.

   Notice that the empty relation on an empty set **is** *reflexive!!*

   the universal relation, $\{(x, y)|x, y \in \mathbb{N}\}$?

   (1), (2), (3) all hold.

2. *the relation $\{(x, x)|x \in \mathbb{N}\}$?*

   (1), (2), (3) all hold.

3. *the relation $\{(x, y)|x, y \in \mathbb{N} \wedge x + y \text{ is even}\}$?*

   (1), (2), (3) all hold.

4. *the relation $\{(x, y)|x, y \in N \wedge x + y = 100\}$?*

   (1), (3) do not hold. (2) holds.

5. *the relation $\{(x, y)|x, y \in N \wedge x = y\ (mod\ 16)\}$?*

   (1), (2), (3) all hold.

$\star$ **18** $\star$ *Taking $\sim$ and R as 2-place relation symbols, consider the following axioms:*

$$
\begin{array}{rl}
(\forall x) & \neg R(x,x) \\
(\forall xy) & \neg(R(x,y) \wedge R(y,x)) \\
(\forall xyz) & (R(x,y) \wedge R(y,z) \rightarrow R(x,z)) \\
(\forall xy) & (R(x,y) \vee x = y \vee R(y,x)) \\
(\forall xz) & (R(x,z) \rightarrow (\exists y)(R(x,y) \wedge R(y,z)))
\end{array}
$$

Exhibit two interpretations that satisfy axioms 1-3 and falsify axioms 4 and 5. Exhibit two interpretations that satisfy axioms 1-4 and falsify axiom 5. Exhibit two interpretations that satisfy axioms 1-5. Consider only interpretations that make = denote the equality relation.

1-3 true, 4 and 5 false.

Domain is sets of natural numbers. $\hat{R}$ is $\subseteq$ or $\subset$ (strict subset) or $\supseteq$ or $\supset$ (strict superset).

Another cute example (due to Loretta He) is: Domain is $\mathbb{N}$, and the relation is $\{\langle x,y\rangle : x+1 < y\}$. The converse of this relation will do too, of course.

1-4 true, 5 false.

Domain is natural numbers. $\hat{R}$ is $<$ or $>$ or $\leq$ or $\geq$.

1-5 true.

Domain is real numbers. $\hat{R}$ is $<$ or $>$ or $\leq$ or $\geq$.

$\star$ **19** $\star$ *Consider a first-order language with 0 and 1 as constant symbols, with - as a 1-place function symbol and + as a 2-place function symbol, and with = as a 2 place predicate symbol.*

(a) Describe the Herbrand Universe for this language.

$$
\begin{array}{rcl}
\mathcal{C} & = & \{0,1\} \\
\mathcal{F}_1 & = & \{-\} \\
\mathcal{F}_2 & = & \{+\} \\
\mathcal{F}_n(n>2) & = & \phi \\
\mathcal{P}_1 & = & \phi \\
\mathcal{P}_2 & = & \{=\} \\
\mathcal{P}_n(n>2) & = & \phi \\
H_0 & = & \{0,1\} \\
H_1 & = & \{0,1,-(0),-(1)\} \\
H & = & \{0,1,-(0),-(1),-(-(0)),-(-(1)),+(0,0),+(0,1),+(1,0), \\
 & & +(1,1),+(0,-(0)),+(0,-(1)),+(-(0),0),+(-(1),1)\cdots\}
\end{array}
$$

$$HB = \{=(0,0), =(0,1), =(1,0), =(1,1), =(-(0),-(0)), \\ =(-(1),-(0)), =(+(0,1),+(-(1),-(0))), \cdots\}$$

(b) The language can be interpreted by taking the integers for the universe and giving 0, 1, -, + and = their usual meanings over the integers. What do those symbols denote in the corresponding Herbrand interpretation?

For extra brownie points write a Context-Free grammar that generates this set ...

In the interpretation = is interpreted by the set of all ordered pairs formed from two expressions α and β such that the result of putting an equals sign between α and β is a theorem of the theory we have in mind. + similarly is the set of all ordered triples of expressions $\langle\alpha, \beta, \gamma\rangle$ such that the result of putting a '+' sign and an '=' sign between them in the obvious way gives an expression that is a theorem of, again, whatever the theory is that we have in mind. Interpretations for the others are defined similarly.

$\star$ **20** $\star$ *For each of the following pairs of terms, give a most general unifier or explain why none exists.*

$f(g(x), z)$ *and* $f(y, h(y))$

$f(g(x), h(g(x)))$ is the most general unifier.

$j(x, y, z)$ *and* $j(f(y, y), f(z, z), f(a, a))$

$j(f(f(f(a, a), f(a, a)), f((a, a), f(a, a))), f(f(a, a), f(a, a)), f(a, a))$ is the most general unification.

$j(x, z, x)$ *and* $j(y, f(y), z)$

Any unification requires that $x = y = z$ and that $z = f(y)$ also. Therefore the terms cannot be unified without allowing $f(f(f(\cdots)))$.

$j(f(x), y, a)$ *and* $j(y, z, z)$

This cannot be unified because it required that $y = z = a$ and also that $y = f(x)$. This will only work if $f(x) = a$ for all x.

$j(g(x), a, y)$ *and* $j(z, x, f(z, z))$

$j(g(a), a, f(g(a), g(a)))$ is the most general unification.

$\star$ **34** $\star$ *Convert these formulæ into clauses, showing each step: negating the formula, eliminating $\rightarrow$ and $\leftrightarrow$, moving the quantifiers, Skolemizing, dropping the universal quantifiers and converting the matrix into CNF.*

$$\begin{array}{rl}
 & (\forall x)(\exists y)R(x,y)) \rightarrow ((\exists y)(\forall x)R(x,y)) \\
negate\,and\,remove\ \rightarrow & ((\forall x)(\exists y)R(x,y)) \wedge \neg((\exists y)(\forall x)R(x,y)) \\
move\,quantifiers & ((\forall x)(\exists y)R(x,y)) \wedge (\forall y)(\neg(\forall x)R(x,y)) \\
 & ((\forall x)(\exists y)R(x,y)) \wedge ((\forall y)(\exists x)\neg R(x,y)) \\
skolemise\,and\,clause & \{R(x,f(x))\},\ \{\neg R(g(x),x)\}
\end{array}$$

$$
\begin{array}{rl}
 & ((\exists y)(\forall x)R(x,y)) \to ((\forall x)(\exists y)R(x,y)) \\
\textit{negate and remove } \to & ((\exists y)(\forall x)R(x,y)) \wedge \neg((\forall x)(\exists y)R(x,y)) \\
 & ((\exists y)(\forall x)R(x,y)) \wedge ((\exists x)(\forall y)\neg R(x,y)) \\
\textit{skolemise and clause} & \{R(x,a)\},\ \{\neg R(b,x)\}
\end{array}
$$

$$
\begin{array}{rl}
 & (\exists x)(\forall yz)((P(y) \to Q(z)) \to (P(x) \to Q(x))) \\
\textit{negate and remove } \to & \neg(\exists x)(\forall yz)((P(y) \wedge \neg Q(z)) \vee \neg P(x) \vee Q(x)) \\
 & (\forall x)(\exists yz)\neg((P(y) \wedge \neg Q(z)) \vee \neg P(x) \vee Q(x)) \\
 & (\forall x)(\exists yz)((\neg P(y) \vee Q(z)) \wedge P(x) \wedge \neg Q(x)) \\
\textit{skolemise and clause} & \{\neg P(f(x)), Q(g(x))\},\ \{P(x)\},\ \{\neg Q(x)\}
\end{array}
$$

$\star$ **22** $\star$ *Find a refutation from the following set of clauses using linear resolution*

$\{P(f(x,y)), \neg Q(x), \neg R(y)\}$, $\{\neg P(v)\}$, $\{\neg R(z), Q(g(z))\}$ and $\{R(a)\}$.

Unify 'x' $\mapsto$ '$g(z)$' and cut $\{P(f(x,y)), \neg Q(x), \neg R(y)\}$ against $\{\neg R(z), Q(g(z))\}$ with cut formula $Q(g(z))$ to give $\{P(f(g(z),z)), \neg R(z)\}$. Unify '$z$' $\mapsto$ 'a' and cut $\{R(a)\}$ against $\{P(f(g(z),z)), \neg R(z)\}$ to give $\{P(f(g(a),a))\}$. Unify 'v' $\mapsto$ '$f(g(a),a)$' and cut the result against $\{\neg P(v)\}$ to give a refutation ($\{\}$).

$\star$ **37** $\star$ *Find a refutation from the following set of clauses using resolution with factoring.*

$\{\neg P(x,a), \neg P(x,y), \neg P(y,x)\}$, $\{P(x,f(x)), P(x,a)\}$ and $\{P(f(x),x), P(x,a)\}$.

Binding both 'y' and 'x' to 'a' in clause 1 we get (4) $\{\neg P(a,a)\}$ (with factoring). We can resolve (4) with both clauses 2 and 3 (seperately) to get (5) $\{P(a,f(a))\}$ and (6) $\{P(f(a),a)\}$. 5 resolves with 1 (bind 'y' to 'a' and 'x' to '$f(a)$') to give (7) $\{\neg P(f(a),a), \neg P(f(a),a)\}$ which reduces by factoring to $\{\neg P(f(a),a)\}$ and this resolves with (6) to the empty clause.

$\star$ **24** $\star$ *Prove the following formulae by resolution, showing all steps of the conversion into clauses. Remember to negate first!*

$(\forall x)(P \vee Q(x)) \to (P \vee (\forall x)Q(x))$

$$
\begin{array}{rl}
 & (\forall x)(P \vee Q(x)) \to (P \vee (\forall x)Q(x)) \\
\textit{negate and remove } \to & \neg(\neg((\forall x)(P \vee Q(x))) \vee (P \vee (\forall x)Q(x))) \\
\textit{move quantifiers} & ((\forall x)(P \vee Q(x))) \wedge \neg((\forall x)(P \vee Q(x))) \\
 & ((\forall x)(P \vee Q(x))) \wedge ((\exists x)\neg(P \vee Q(x))) \\
 & ((\forall x)(P \vee Q(x))) \wedge ((\exists x)(\neg P \wedge \neg Q(x))) \\
\textit{skolemise and clause} & \{P, Q(x)\},\ \{\neg P\}\ \{\neg Q(a)\}
\end{array}
$$

Resolving the three clauses together gives a contradiction. Therefore the negation of the formula is inconsistent. Therefore the formula is proven.

$(\exists xy)(P(x, y) \to (\forall vw)P(v, w))$

$$\begin{array}{rl} & (\exists xy)(P(x,y) \to (\forall vw)P(v,w)) \\ negate\, and\, remove\ \to & \neg((\exists xy)(\neg P(x,y) \vee (\forall vw)P(v,w)) \\ move\, quantifiers & (\forall xy)(\exists vw)(P(x,y) \wedge \neg P(v,w)) \\ skolemise\, and\, clause & \{P(x,y)\},\ \{\neg P(f(x,y), g(x,y))\} \end{array}$$

These two clauses resolve to the empty clause when x takes on the value of $f(x, y)$ and y the value of $g(x, y)$. Thus the formula is proven.

$\neg(\exists x)(\forall y)(R(y, x) \leftrightarrow \neg R(y, y))$

$$\begin{array}{rl} & \neg(\exists x)(\forall y)(R(y,x) \leftrightarrow \neg R(y,y)) \\ negate\, and\, remove\ \leftrightarrow & (\exists x)(\forall y)((\neg R(y,x) \vee \neg R(y,y)) \wedge (R(y,y) \vee R(y,x))) \\ skolemise\, and\, clause & \{\neg R(x,a), \neg R(x,x)\},\ \{R(x,x), R(x,a)\} \end{array}$$

If we resolve these two clauses together we get a contradiction. Thus formula is proven.

$\star$ **25** $\star$ *Dual Skolemisation*

Let $\mathcal{L}$ be a language, and let $\Psi : \mathcal{L} \to \mathcal{L}$ be a map such that, for all formulæ ϕ, $\Psi(\phi)$ is satisfiable iff ϕ is. (Skolemisation is an example). We will now show that the map $\lambda\phi.\neg(\Psi(\neg\phi))$ preserves validity.

ϕ is valid iff

$\neg\phi$ is not satisfiable iff

$\Psi(\neg\phi)$ is not satisfiable iff

$\neg(\Psi(\neg\phi))$ is valid.

Now all we have to check is that if Ψ is skolemisation then $\lambda\phi.\neg(\Psi(\neg\phi))$ is dual skolemisation à *la Larry Paulson*. Take a formula in prenex normal form, as it might be

$$\forall x \exists y \forall z \phi$$

where ϕ is anything without quantifiers. Negate it to get

$$\exists x \forall y \exists z \neg\phi$$

and skolemize to get

$$\neg\phi[(f(y))/z, a/x]$$

and negate again to get

$$\phi[(f(y))/z, a/x]$$

...which is exactly what you would have got if you dual skolemised the formula we started with.

I'm not entirely sure what use this is. I suppose one might use it in the following circumstances. You want to prove ϕ from Γ. You (i) **dual**-skolemise ϕ and convert to **disjunctive** normal form; (ii) negate Γ and convert to **disjunctive** normal form. Then you write clauses which are of course conjunctions not disjunctions and resolving to the empty clause means you have established the truth of $\neg\Gamma \vee \phi$. Two things to think about: (i) is there a problem about relettering clauses? (ii) Did we really need to dual skolemise?

The way to understand what skolemisation is doing is something like this: skolemisation is supposed to preserve satisfiability not truth. Truth is a property of a formula in an environment, but satisfiability is a property of the formula. Skolemisation is something you do to a formula not to a formula-in-an-environment. (Any logical manipulation on a formula of course, like conversion to CNF is also, strictly something you do to a formula not a formula-in-an-environment, but with CNF you can carry the environment along with you). Corresponding to skolemisation is a function on environments. I suppose AC is an assertion that this function is uniform in some sense or functorial or something. For example, there is a way of turning any interpretation of '$(\exists x)(F(x))$' into an interpretation of '$F(a)$': one adds structure. To go the other way one throws structure away. The forgetful functor?

Postscript: both skolemisation and dual skolemisation are maps from a language into itself that preserve something. In one case satisfiability and in the other case validity. In this context we can anticipate a map used in the complexity theory course. Take a formula in conjunctive normal form (so it's a conjunction of disjunctions) In general the individual conjuncts may have lots of literals in them, and be something like $(p \vee q \vee \neg r \vee s)$. This conjunct has four literals in it. Now consider the result of replacing this conjunction by the (conjunction of the) two conjuncts $(p \vee q \vee t) \wedge (\neg t \vee \neg r \vee s)$, where '$t$' is a new variable not present in the original formula. The new formula is satisfiable iff the original one was. Also (although the new formula has more conjuncts) we have replaced longer conjuncts by shorter conjuncts. You will see later why this is a useful trick.

2002 p6 q11

(a) A is consistent.

(b) We have derived a contradiction from something that has been negated. So the thing that had been negated is valid. That thing is the skolemised version of A. So the skolemised version of A is valid. Doesn't seem to tell us anything. Skolemisation preserves satisfiability.

(c) If $\neg A$ is refutable, then it shouldn't matter which variable you choose for a case split: you should get the empty clause every time. OTOH if he means by the question that if you split on p say, and the clauses arising from p resolve to the empty clause but the clauses arising from $\neg p$ don't, then the formula is consistent.

$\star$ **34** $\star$ *Definite clauses*

If we resolve two nonempty definite clauses we get a nonempty definite clause. So, by induction, the only things we can deduce from nonempty definite clauses are other nonempty definite clauses. Since no definite clause is empty, we cannot deduce the empty disjunction, which is to say we cannot deduce the false!

One of Larry's questions

Are $\{P(x,b), P(a,y)\}$ and $\{P(a,b)\}$ equivalent?

$\{P(x,b), P(a,y)\}$ is $(\forall xy)(P(x,b) \vee P(a,y))$. Instantiating '$x$' to '$a$' and '$y$' to '$b$' we infer $P(a,b)$. The converse is obviously not going to be provable: take a universe just containing a and b and decide that $\neg P(b,b)$ and $\neg P(a,a)$. (don't worry about the truth-values of the other three atomics—they don't matter).

Are $\{P(y,y), P(y,a)\}$ and $\{P(y,a)\}$ equivalent?

This one looks uncannily like a tarted up version of Russell's paradox. Perhaps i just have a nasty suspicious mind. Let's rewrite P as $\in$ (and as infix) to get

Are $\{y \in y, y \in a\}$ and $\{y \in a\}$ equivalent?

which are $(\forall y)(y \notin y \to y \in a)$ and $(\forall y)(y \in a)$ and it's already much clearer what is going on. They are obviously not equivalent, but it might be an idea to cook up a small finite countermodel. One like this, perhaps:

$a \notin a$, $a \in b$, $c \in c$, $c \notin b$, $b \in b$.

b contains all things that are not members of themselves (namely a), but it doesn't contain everything (it's missing c).

2005:5:9

1.0.4 part (a)

In order to prove the following formula by resolution, what set of clauses should be submitted to the prover? Justify your answer briefly

$$\forall x[(P(x) \vee Q) \to \neg R(x)] \wedge \forall x[(Q \to \neg S(x)) \to P(x) \wedge R(x)] \to \forall x S(x)$$

(Just to remind myself what I'm doing . . .) If ϕ is satisfiable, so is $sk(\phi)$, the skolemisation of ϕ. So if we want to prove ϕ we investigate $\neg\phi$ and hope that it isn't satisfiable. However if it *is* satisfiable, so is $sk(\neg\phi)$. So we put $sk(\neg\phi)$ into clause form and hope to find a contradiction.

The negation of ϕ is the conjunction of

1. $\forall x[(P(x) \vee Q) \to \neg R(x)]$
2. $\forall x[(Q \to \neg S(x)) \to P(x) \wedge R(x)]$
3. $\exists x \neg S(x)$

These give us

1. $(P(x) \vee Q) \to \neg R(x)$
2. $(Q \to \neg S(y)) \to P(y) \wedge R(y)$
3. $\neg S(a)$

(i) gives us the two clauses $\{\neg P(x), \neg R(x)\}$ and $\{\neg Q, \neg R(x)\}$;
(iii) obviously gives us the clause $\{\neg S(a)\}$
(ii) gives us the two expressions $(Q \to \neg S(y)) \to P(y)$ and $(Q \to \neg S(y)) \to P(y)R(y)$. The first becomes
$\neg(Q \to \neg S(y)) \vee P(y)$ which becomes
$(Q \wedge S(y)) \vee P(y)$ which becomes
$(Q \vee P(y)) \wedge (S(y) \vee P(y))$ which gives us the two clauses
$\{Q, P(y))\}$ and $\{S(y), P(y)\}$.
The second differs from the first only in having 'R' instead of 'S' and so we get the two clauses
$\{Q, R(y))\}$ and $\{S(y), R(y)\}$.

1.0.5 part (b)

The third clause says that P is transitive, and the first clause says that P is irreflexive. P is starting tolook like $<$ on the naturals. It looks even more the that when you reflect that caluse two makes sense if you think of f as successor.

1.0.6 part (c)

Resolve $\{Q(a)\}$ with $\{\neg Q(a), P(a), \neg R(y), \neg Q(y)\}$ to obtain
$\{P(a), \neg R(y), \neg Q(y)\}$. Next resolve this with $\{\neg P(a)\}$ to obtain
$\{\neg R(y), \neg Q(y)\}$. Next resolve this with $\{R(b)\}$ to obtain
$\{\neg Q(b)\}$. Next resolve this with $\{\neg S(b), \neg R(b), Q(b)\}$ to obtain

$\{\neg R(b), \neg S(b)\}$. Next resolve this with $\{S(b)\}$ to obtain $\{\neg R(b)\}$. Resolve this with $\{R(b)\}$ to obtain the empty clause. And it was linear resolution all the way!!