Prenex normal form with prefix and matrix refuted as not bivalent

We evaluate the prenex normal form using equations from en.wikipedia.org/wiki/Prenex_normal_form. We assume the Meth8/VŁ4 apparatus and method.

LET: p q r s t u ϕ phi, ψ psi, ρ rho, x, y, z; (p=p) true;
necessity, all; % possibility, some or one; & And; + Or; > Imply; = Equivalent.

The designated *proof* value is T for tautology. Truth tables with 16-values and 128-values are row-major. Non-repeating truth table rows are row-major and presented horizontally.

Every formula in classical logic is equivalent to a formula in prenex normal form. For example, if ϕ(y) , ψ(z) , and ρ(x) are quantifier-free formulas with the free variables shown then

Prenex normal form:
∀x ∃y ∀z(ϕ(y) ∨(ψ(z) → ρ(x))) (1.0.1.1)
((#s&(%t&#u))&((p&t)+((q&u)>(r&s)))) = (p=p) ; (1.0.1.2)
FFFF FFFF FFFF FFFF, ... NNFN NNNN

Prefix:
∀x ∃y ∀z (1.0.1.1.1)
(#s&(%t&#u)) (1.0.1.1.2)

Matrix:
ϕ(y) ∨(ψ(z) → ρ(x)) , (1.0.2.1)
((p&t)+((q&u)>(r&s))) (1.0.2.2)

Not prenex normal form:
∀x((∃y ϕ(y)) ∨((∃z ψ(z)) → ρ(x))) (1.0.3.1)
(#s&((%t&(p&t))+((%u&(q&u))>(r&s)))) = (p=p) ; (1.0.3.2)
FFFF FFFF NNNN NNNN, ... NNFN NNNN, ... NNFF NNNN

The prenex and not prenex forms are supposed to be logically equivalent.

((#s&(%t&#u))&((p&t)+((q&u)>(r&s)))) =
(#s&((%t&(p&t))+((%u&(q&u))>(r&s)))) ; TTTT TTTT CCTT CCCC (1.0.4.2)

Eq. 1.0.4.2 is *not* tautologous. From the text example, prenex is supposed to be equivalent to a not-prenex rendition, but the prenex model fails at this point.

LET: p q r x, ϕ, ψ

The rules for conjunction and disjunction say that

(∀x ϕ) ∧ψ is equivalent to ∀x(ϕ ∧ψ) (1.1.1)
((#p&q)+r)=(#p&(q+r)) ; TTTT FNFN TTTT FNFN (1.1.2)

$(\forall x\ \phi) \lor \psi$ is equivalent to $\forall x(\phi \lor \psi)$ (1.2.1)
`((#p&q)&r)=(#p&(q&r))      ; TTTT TTTT TTTT TTTT` (1.2.2)

and

$(\exists x\ \phi) \land \psi$ is equivalent to $\exists x(\phi \land \psi)$ (2.1.1)
`((%p&q)+r)=(%p&(q+r))      ; TTTT CTCT TTTT CTCT` (2.1.2)

$(\exists x\ \phi) \lor \psi$ is equivalent to $\exists x(\phi \lor \psi)$ (2.2.1)
`((%p&q)&r)=(%p&(q&r))      ; TTTT TTTT TTTT TTTT` (2.2.2)

The equivalences are valid when x does not appear as a free variable of ψ.

Negation

The rules for negation say that

$\neg\exists x\phi$ is equivalent to $\forall x\neg\phi$ (3.1.1)
`~%p&q)=(#p&~q)             ; TCCT TCCT TCCT TCCT` (3.1.2)

$\neg\forall x\phi$ is equivalent to $\exists x\neg\phi$ (3.2.1)
`(~#p&q)=(%p&~q)            ; NFFN NFFN NFFN NFFN` (3.2.2)

Implication

There are four rules for implication: two that remove quantifiers from the antecedent and two that remove quantifiers from the consequent. These rules can be derived by rewriting the implication $\phi \to \psi$ as $\neg \phi \lor \psi$ and applying the rules for disjunction above. As with the rules for disjunction, these rules require that the variable quantified in one subformula does not appear free in the other subformula.

The rules for removing quantifiers from the antecedent are:

$(\forall x\ \phi) \to \psi$ is equivalent to $\exists x(\phi \to \psi)$ (4.1.1)
`((#p&q)>r)=(%p&(q>r))      ; CTFN CTCT CTFN CTCT` (4.1.2)

$(\exists x\ \phi) \to \psi$ is equivalent to $\forall x(\phi \to \psi)$ (4.2.1)
`((%p&q)>r)=(#p&(q>r))      ; FNCT FNFN FNCT FNFN` (4.2.2)

The rules for removing quantifiers from the consequent are:

$\phi \to (\exists x\ \psi)$ is equivalent to $\exists x(\phi \to \psi)$ (5.1.1)
`(q>(%p&r))=(%p&(q>r))      ; CTTT CTTT CTTT CTTT` (5.1.2)

$\phi \to (\forall x\ \psi)$ is equivalent to $\forall x(\phi \to \psi)$ (5.2.1)
`(q>(#p&r))=(#p&(q>r))      ; FNTT FNTT FNTT FNTT` (5.2.2)

The unnumbered examples in the text are *not* tautologous.

The intutionistic logic equations listed in the text are supposed to fail. We found the first one was tautologous.

$\forall x\ (\ \phi \lor \psi\ )$ implies $(\ \forall x\ \phi\ ) \lor \psi$ (6.1.1)

(#p&(q+r))>((#p&q)+r) ; TTTT TTTT TTTT TTTT (6.1.2)

Two Eqs. 1.2.2 and 2.2.2 as rendered were tautologous for the rules to map conjunction as quantified. This suggests that if all the connective rules are derived from the And connective, then there could be a better chance for success. However, that exercise pales in light of rules for negation and implication as found *not* tautologous. Hence, the prenex model was *not* tautologous. What follows is that the prenex model is not bivalent.

> Remark: Since about 1933 when Kurt Gödel reduced his quantified equations to prenex normal form, the format was adopted by many for exposition. We previously showed that one explanation for why the incompleteness theorems are not tautologous is because the Gödel's misuse of bivalent logic via the the prenex format. That finding is further supported by this instant analysis of the format.

What further follows is that many theorems produced with prenex for computer science, mathematics, and physics are now suspicious. A notable example is the satisfiability algorithms produced by Martin Davis and Hilary Putnam which are now mistaken.