Exercise 2. Give an example of a sentence which is truth-functionally satisfiable but not first order satisfiable.

Exercise 3. [*Important!*] Show that a *quantifier-free* sentence (i.e. a sentence with no quantifiers) which is truth-functionally satisfiable must also be first order satisfiable.

Show therefore that if a *quantifier-free* sentence is valid, then it must be a tautology [this is a semantic version of Hilbert's first ϵ-theorem]. More generally, show that any S (finite or infinite) of *quantifier-free* sentences which is (simultaneously) truth-functionally satisfiable is also (simultaneously) first order satisfiable [Hint: Take any Boolean valuation of E which satisfies S. This Boolean valuation induces a certain interpretation I of the predicates of S (in a manner we have discussed earlier). Show that every element X of S is true under this interpretation I (use induction on the degree of X)].

Chapter V

First-Order Analytic Tableaux

§ 1. Extension of Our Unified Notation

We use "α", "β" exactly as we did for propositional logic (only now construing "formula" to mean "closed formula of quantification theory"). We now add two more categories γ and δ as follows.

For the moment let us work with unsigned formulas. Then "γ" shall denote any formula of one of the two forms $(\forall x)A$, $\sim(\exists x)A$, and for any parameter a, by $\gamma(a)$ we mean A^x_a, $\sim A^x_a$ respectively.

We use "δ" to denote any formula of one of the two forms $(\exists x)A$, $\sim(\forall x)A$, and by $\delta(a)$ we respectively mean A^x_a, $\sim(A^x_a)$. We refer to γ-formulas as of *universal* type, and δ-formulas as of *existential* type.

In working with signed formulas, γ shall be any signed formula of one of the forms $T(\forall x)A$, $F(\exists x)A$, and $\gamma(a)$ is respectively TA^x_a, FA^x_a. And δ shall be any signed formula of one of the forms $T(\exists x)A$, $F(\forall x)A$, and $\delta(a)$ is respectively TA^x_a, FA^x_a.

In considering sentences with constants in the universe U, we use γ, δ in the same manner and for any $k\in U$, we define $\gamma(k)$, $\delta(k)$ similarly.

Under any interpretation in a universe U, the following facts clearly hold:

F_1: α is true iff α_1, α_2 are both true.
F_2: β is true iff at least one of β_1, β_2 is true.
F_3: γ is true iff $\gamma(k)$ is true for *every* $k\in U$.
F_4: δ is true iff $\delta(k)$ is true for *at least one* $k\in U$.

As consequences of the above facts, we have the following laws concerning satisfiability, of which G_1, G_2, G_3 are immediate and G_4 (which the reader should look at most carefully) we will prove. In these laws, S is any set of formulas perhaps with parameters (but no other constants), and likewise with α, β, γ, δ. And "satisfiable" means *first order satisfiable*.

G_1: If S is satisfiable, and $\alpha\in S$, then $\{S,\alpha_1,\alpha_2\}$ is satisfiable.
G_2: If S is satisfiable and $\beta\in S$, then at least one of the two sets $\{S,\beta_1\}$, $\{S,\beta_2\}$ is satisfiable.
G_3: If S is satisfiable and $\gamma\in S$, then, for *every* parameter a, the set $\{S,\gamma(a)\}$ is satisfiable.
G_4: If S is satisfiable and $\delta\in S$, *and if a is any parameter which occurs in no element of* S, then $\{S,\delta(a)\}$ is satisfiable.

We leave the verification of G_1, G_2, G_3 to the reader; we shall now prove the very critical law G_4.

By hypothesis, there is an interpretation I of all predicates of S in some universe U and a mapping φ of all parameters of S into elements of U such that for every $A\in S$, the U-sentence A^φ is true under I. In particular, δ^φ is true under I. The sentence δ^φ is a sentence with no parameters but with constants in U and it is a sentence of existential type, call it δ_1. Since δ_1 is true under I, then (by F_4), there must be at least one element k of U such that $\delta_1(k)$ is true under I. Now φ is defined on all parameters of $\{S,\delta(a)\}$, except for the parameter a. We extend φ by defining $\varphi(a)=k$—call this extension φ^*. Then φ^* is defined on *all* parameters of $\{S,\delta(a)\}$. Clearly, for every $A\in S$, A^{φ^*} is the same expression as A^φ, so A^{φ^*} is true under I. And $[\delta(a)]^{\varphi^*}$ is the same sentence as $\delta_1(k)$, hence $[\delta(a)]^{\varphi^*}$ is true under I. Hence, for every $A\in\{S,\delta(a)\}$, A^{φ^*} is true under I. Thus $\{S,\delta(a)\}$ is satisfiable.

§ 2. Analytic Tableaux for Quantification Theory

Whether we work with signed formulas or not, our tableaux rules for first order logic are the following four:

Rule A: $$\frac{\alpha}{\begin{matrix}\alpha_1\\ \alpha_2\end{matrix}}$$

Rule B: $$\frac{\beta}{\beta_1 \mid \beta_2}$$

Rule C: $$\frac{\gamma}{\gamma(a)},$$ where a is *any* parameter.

Rule D: $$\frac{\delta}{\delta(a)},$$ where a is a *new* parameter.

Rules A, B are the same as in propositional logic. The new rules C, D (for eliminating quantifiers), are *direct* rules; the only one of the four rules which is a branching rule is Rule B. In *signed* notation, Rules C, D are as follows:

Rule C: $\dfrac{T(\forall x)A}{TA^x_a} \qquad \dfrac{F(\exists x)A}{FA^x_a}$

Rule D: $\dfrac{T(\exists x)A}{TA^x_a}$, with proviso (that a is new)

$\dfrac{F(\forall x)A}{FA^x_a}$, with same proviso

Using *unsigned* formulas, our quantificational rules are:

Rule C: $\dfrac{(\forall x)A}{A^x_a} \qquad \dfrac{\sim(\exists x)A}{\sim A^x_a}$

Rule D: $\dfrac{(\exists x)A}{A^x_a}$, with proviso

$\dfrac{\sim(\forall x)A}{\sim A^x_a}$, with proviso

Discussion concerning Rule D. This rule is a formalization of the following informal argument used constantly in mathematics. Suppose in the course of an argument we have proved that there exists an element x having a certain property P—i.e. we have proved the statement $(\exists x)Px$. We then say, "let a be such an x" and we write Pa. Of course, we are not asserting that P holds for *every* a, but just for at least one. If we subsequently show that for another property Q, there exists an x such that Qx, we cannot legitimately say "let a be such an x", because we have already committed the symbol "a" to being the name of some x such that Px and we do not know that there is any single x having both the properties P and Q. Thus we take a *new* parameter b, and say "let b be such an x", and we write Qb. This is the reason for the proviso in Rule D.

Actually we can liberalize Rule D by replacing the clause "providing a is new", by "providing a is new, or else a has not been previously introduced by Rule D, and does not occur in δ, and no parameter of δ has been previously introduced by Rule D". Under this liberalization, proofs can sometimes be shortened (cf. Example 2 below.)

The idea behind this liberalization is this. Suppose in the course of an argument we prove a sentence $(\forall x)Px$ (which is of type γ). Then we conclude Pa. We have not really committed "a" to being the name of any particular individual; Pa holds for *every* value of a. So if we subsequently prove a sentence $(\exists x)Qx$, we can legitimately say, "let a be such an x", and for the same value of a, Pa will also hold.

The above discussion is but an informal foreshadowing of a precise argument showing the *consistency* of the tableau method for first order logic. Actually, if we stick to the *strict* version of Rule D, the consistency is almost immediate from the conditions G_1, G_2, G_3, G_4 of satisfiability which we stated in § 1.

For suppose θ is a branch of a tableau and that θ is satisfiable. If we extend θ by rule A, C or D then the resulting extension is again satisfiable (by G_1, G_3, G_4, respectively). If we simultaneously extend θ to 2 branches θ_1, θ_2 by one application of rule B, then at least one of θ_1, θ_2 is again satisfiable (by G_2). Thus any immediate extension of a tableau which is satisfiable (in the sense that at least one of its branches is satisfiable) is again satisfiable. Therefore (by induction) if the origin of a tableau is satisfiable, then at least one branch of the tableau is satisfiable and hence open. Therefore if a tableau closes, then the origin is indeed unsatisfiable—stated otherwise, every provable sentence is valid.

The precise justification of the *liberalized* version of Rule D is a bit more delicate; we shall return to this later. Meanwhile we wish to get on with some concrete working familarity with First Order Tableaux.

Example 1. The following tableau is a proof of the sentence

$$(\forall x)[Px\supset Qx]\supset[(\forall x)Px\supset(\forall x)Qx]$$

(1)	$\sim[(\forall x)[Px\supset Qx]\supset[(\forall x)Px\supset(\forall x)Qx]]$				
(2)	$(\forall x)(Px\supset Qx)$	(1)			
(3)	$\sim[(\forall x)Px\supset(\forall x)Qx]$	(1)			
(4)	$(\forall x)Px$	(3)			
(5)	$\sim(\forall x)Qx$	(3)			
(6)	$\sim Qa$	(5)			
(7)	Pa	(4)			
(8)	$Pa\supset Qa$	(2)			
(9)	$\sim Pa$	(8)	(10)	Qa	(8)
	x			x	

Example 2. We wish to give 2 different proofs of the sentence $(\exists y)[(\exists x)Px\supset Py]$. The first proof uses the *strict* form of Rule D, and the second (which is shorter) uses the *liberalized* version of Rule D:

Proof 1.

(1)	$F(\exists y)[(\exists x)Px \supset Py]$	
(2)	$F(\exists x)Px \supset Pa$	(1)
(3)	$T(\exists x)Px$	(2)
(4)	FPa	(2)
(5)	TPb	(3)
(6)	$F(\exists x)Px \supset Pb$	(1)
(7)	FPb	
	$\times$	

Proof 2.

(1)	$F(\exists y)[(\exists x)Px \supset Py]$	
(2)	$F(\exists x)Px \supset Pa$	(1)
(3)	$T(\exists x)Px$	(2)
(4)	FPa	(2)
(5)	TPa	(3)
	$\times$	

Exercises. Prove the following formulas:

$(\forall y)[(\forall x)Px \supset Py]$

$(\forall x)Px \supset (\exists x)Px$

$(\exists y)[Py \supset (\forall x)Px]$

$\sim(\exists y)Py \supset [(\forall y)((\exists x)Px \supset Py)]$

$(\exists x)Px \supset (\exists y)Py$

$(\forall x)[Px \wedge Qx] \equiv (\forall x)Px \wedge (\forall x)Qx]$

$[(\forall x)Px \vee (\forall x)Qx] \supset (\forall x)[Px \vee Qx]$

(the converse is not valid!)

$(\exists x)(Px \vee Qx) \equiv ((\exists x)Px \vee (\exists x)Qx)$

$(\exists x)(Px \wedge Qx) \supset ((\exists x)Px \wedge (\exists x)Qx)$

(the converse is not valid).

In the next group, C is any *closed* formula—or at least the variable x does not occur free in it:

$(\forall x)[Px \vee C] \equiv [(\forall x)Px \vee C]$

$(\exists x)[Px \wedge C] \equiv [(\exists x)Px \wedge C]$

$(\exists x)C \equiv C$

$(\forall x)C \equiv C$

$(\exists x)[C \supset Px] \equiv [C \supset (\exists x)Px]$

$(\exists x)[Px \supset C] \equiv [(\forall x)Px \supset C]$

$(\forall x)[C \supset Px] \equiv [C \supset (\forall x)Px]$

$(\forall x)[Px \supset C] \equiv [(\exists x)Px \supset C]$

Show $(H \wedge K) \supset L$, where

$H = (\forall x)(\forall y)[Rxy \supset Ryx]$ (*R* is *symmetric*)

$K = (\forall x)(\forall y)(\forall z)[(Rxy \wedge Ryz) \supset Rxz]$ (*R* is *transitive*)

$L = (\forall x)(\forall y)[Rxy \supset Rxx]$ (*R* is *reflexive* on its domain of definition).

For a *hard* one, try the following exercise (taken from Quine [1]): Show $(A \wedge B) \supset C$, where

$A = (\forall x)[(Fx \wedge Gx) \supset Hx] \supset (\exists x)[Fx \wedge \sim Gx]$

$B = (\forall x)\ [Fx \supset Gx] \vee (\forall x)[Fx \supset Hx]$

$C = (\forall x)\ [(Fx \wedge Hx) \supset Gx] \supset (\exists x)[Fx \wedge Gx \wedge \sim Hx]$

§ 3. The Completeness Theorem

Now we turn to the proof of one of the major results in quantification theory: Every valid sentence is provable by the tableau method.

This is a form of Gödel's famous completeness theorem. Actually Gödel proved the completeness of a different formalization of Quantification Theory, but we shall later show how the completeness of the tableau method implies the completeness of the more conventional formalizations. The completeness proof we now give is along the lines of Beth [1] or Hintikka [1]—and also Anderson and Belnap [1].

Let us first briefly review our completeness proof for tableaux in *propositional* logic, and then see what modifications will suggest themselves. In the case of propositional logic, we reach a completed tableau after finitely many stages. Upon completion, every open branch is a Hintikka set. And by Hintikka's lemma, every Hintikka set is truth-functionally satisfiable.

Our first task is to give an appropriate definition of "Hintikka set" for first order logic in which we specify conditions not only on the α's and β's but also on the γ's and δ's as well. We shall define Hintikka sets for arbitrary universes U of constants.

Definition. By a Hintikka set (for a universe U) we mean a set S (of U-formulas) such that the following conditions hold for every α, β, γ, δ in E^U:

H_0: No *atomic* element of E^U and its negation (or conjugate, if we are working with signed formulas) are both in S.

H_1: If $\alpha \in S$, then α_1, α_2 are both in S.

H_2: If $\beta \in S$, then $\beta_1 \in S$ or $\beta_2 \in S$.

H_3: If $\gamma \in S$, then for *every* $k \in U$, $\gamma(k) \in S$.

H_4: If $\delta \in S$, then for at least one element $k \in U$, $\delta(k) \in S$.

Now we show