Module 14
Free Predicate Logic

G. J. Mattey

November 29, 2012

Contents

1 Non-Designating Terms

One of the reasons to investigate modal logic is that we ordinarily talk about objects that could exist, although they do not. Philosophers call such would-be existents "possibilia." One way to talk about possibilia is to use definite descriptions that do not refer. We might use the expression "the flying horse" to describe a non-existent object in Greek mythology. Or, we might use the name "Pegasus" to that same end. If we think of constants of *PL* as surrogates for names in natural language, we seem to be faced with a problem.

In the standard semantical system *PI*, on a given interpretation, every constant refers to an object in that interpretation's domain. Therefore, if we assert a sentence containing the name, such as "Pegasus is a horse that flies," we would symbolize it as, perhaps '$H^1a \wedge F^1a$,' from which it can be inferred that '$(\exists x)(H^1x \wedge F^1x)$.' If the existential quantifier is to represent only those objects that exist in the actual world, then we will either have to give up the use of a constant to stand for the name 'Pegasus,' say that Pegasus is actual, or stop using the existential quantifier to represent what is actual.

One way to evade this trilemma is to modify underlying Predicate Logic so that we are "free" to use constants to refer to objects that are not actual. Free Predicate Logic allows the expression of true sentences about individuals that do not exist. The defining feature of Free Predicate Logic is that its constants need not designate an "existing" object. Thus we can make assertions about non-existent objects, such as Pegasus, without being committed to their existence merely by associating their name with a constant.

2 Syntax of Free Predicate Logic

The syntax of Free Predicate Logic is an extension of the syntax for Predicate Logic. A single one-place constant predicate letter is added. It will be written in non-italicized Roman type to distinguish it from predicates whose extension varies from interpretation to interpretation. Intuitively, this predicate applies only to what "exists." Precise meaning will be given to this notion when the semantics for Free Predicate Logic is developed.

Existence Predicate

'E^1' is a predicate letter of Free Predicate Logic

3 Semantics for Free Predicate Logic

Lablanc and Thomason have specified a semantics for non-modal Free Predicate Logic.[1] Here we adapt their semantics to mesh with the semantical system *PI* described in the previous module.

3.1 Inner and Outer Domains

The heart of the Leblanc-Thomason semantics is the bifurcation of the unitary domain of *PI* into two exclusive domains, one called "inner" and the other called "outer." The inner domain contains the objects which are taken to exist, while the outer domain contains those objects that are taken not to exist. We will assume here that the inner domain is not empty, though the outer domain may be empty.[2]

If $\mathbf{D}$ is the inner domain in an interpretation, we will call $\mathbf{D}'$ the outer domain. Thus a Free Predicate Logic interpretation is an ordered triple $\langle \mathbf{D}, \mathbf{D}', \mathbf{v} \rangle$ rather than an ordered pair consisting of a single domain and a valuation function.

3.2 The Interpretation of Expressions

Because there are two domains in the semantics for Free Predicate Logic, the rules for interpreting its expressions are more complex than those for the interpretation of expressions of standard Predicate Logic. In this section, we will discuss the interpretation of terms and predicates, of the special existence predicate, and of quantified sentences.

3.2.1 Terms and Predicates

In an interpretation of Free Predicate Logic, the extensions of predicates and the designations of terms (constants and parameters) are taken from the union $\mathbf{D} \cup \mathbf{D}'$ of the two domains. That is, they are taken from the combination of all the objects, both existent and non-existent. The extension of a predicate letter $\mathbf{P}^n$ is a set of ordered n-tuples drawn from the union of the two domains, that is, a subset of $(\mathbf{D} \cup \mathbf{D}')^n$. Because the objects in the extensions of predicates are not limited to the inner domain, there can be true predications

[1] Hughes Lablanc and Richmond H. Thomason, "Completeness Theorems for Some Presupposition-Free Logics", *Fundamenta Math.* 62, 125-164. See also Ermanno Bencivenga, "Free Logics", in Gabbay and Guenther (eds), *Handbook of Philosophical Logic*, Vol. III, pp. 373-426.

[2] Bencivenga notes that the inner domain may be empty on the Lablanc-Thomason semantics.

of objects in the outer domain—objects that do not exist. For example, 'Fa' can be true on an interpretation when 'a' designates an object in the outer domain and that object is in the extension of 'F,' even though it does not exist, which is to say that it is not a member of the inner domain.

3.2.2 The Existence Predicate

To say, solely on the basis of Pegasus being a flying horse, that flying horses exist requires that Pegasus exist. In standard Predicate Logic, this might be symbolized by '$(\exists x)x = a$,' where 'a' designates Pegasus. In Free Predicate Logic, it would be symbolized using the special existence predicate as 'E^1a.'[3] The existence predicate is introduced specially because of the role it plays in the derivational system for *FPL* and because of the role it might play in more sophisticated systems of free modal logic.

The extension of the existence predicate in an interpretation is the set of (one-tuples of) the members of the inner domain of the frame on which the interpretation is based: $\mathbf{v_I}(\mathrm{E})=\{\langle\mathbf{u}\rangle\text{: } \mathbf{u} \in \mathbf{D}\}$. This set is the 1st Cartesian product of $\mathbf{D}$, which is $\mathbf{D}^1$.

3.2.3 Quantified Sentences

The interpretation of quantified sentences is, like the interpretation of the existence predicate, sensitive to the distinction to the inner and the outer domains. What makes a quantified sentence true depends entirely on which objects are in the inner domain. Thus, we say that a universally quantified sentence $(\forall\mathbf{x})\alpha(\mathbf{x}/\mathbf{u})$ is true just in case $\alpha(\mathbf{u})$ is true for all the objects in the inner domain which could be assigned to '$\mathbf{u}$' by the interpretation. An existentially quantified sentence $(\exists\mathbf{x})\alpha(\mathbf{x}/\mathbf{u})$ is true if and only if there is some object in the inner domain such that if '$\mathbf{u}$' designates it, the sentence $\alpha(\mathbf{u})$ is true.

In the case of constants, we lose our ability to generalize existentially. Consider an interpretation I on which $\mathrm{v_I}(a) \in \mathrm{D}'$, $\langle \mathrm{v_I}(a)\rangle \in \mathrm{v_I}(F)$, and there is no object d in D such that $\langle \mathrm{d}\rangle \in \mathrm{v_I}(F)$. That is, the value of 'a' is in the outer domain (and hence not in the inner domain) and the one-tuple consisting of that value is in the extension of 'F,' while no one-tuple consisting of an object from the inner domain is in the extension of 'F. As a result, the sentence 'Fa' is true, while '$(\exists x)Fx$' is false. The latter sentence is false because there is no variant of $\mathrm{v_I}$ which assigns a member d of the inner domain D to 'u' such that 'Fu' is true on that variant. In ordinary English, we might assert "Pegasus is a flying horse" without thereby asserting, "Flying horses exist."

The same considerations apply to existential generalizations of sentences containing parameters. On some interpretations, a parameter '$\mathbf{u}$' might be assigned an object in the outer domain. Thus the sentence '$\alpha(\mathbf{u})$' might be true, while the existential generalization of that sentence, $(\exists\mathbf{x})\alpha(\mathbf{x}/\mathbf{u})$, is false. The truth or falsehood of the latter sentence does not depend at all on what '$\mathbf{u}$' designates. All that matters is that there is some variant of the interpretation which assigns to '$\mathbf{u}$' a member of the inner domain such that $\alpha(\mathbf{u})$ is true.

Sentence with the existence predicate applied to a term are relevant to the truth-value of existential sentences. If I assert, "Lorena Ochoa is a dominant female professional golfer", and "Lorena Ochoa exists," then I am entitled to assert, "There exists a dominant female professional golfer." Symbolically, we can say that if 'Fa' and '$\mathrm{E}a$' are true on an interpretation I, then 'a' designates a member of the inner domain D, which allows existential generalization. If $\mathrm{v}_I(a)$=d, then there is a variant of v_I which assigns d to a parameter 'u,' so that 'Fu' is true, in which case '$(\exists x)Fx$' is true on that interpretation.

The existence predicate is needed to generate a sound version of another rule of inference, Universal Instantiation (when instantiation is made to constants). To say that everything is F, given the semantics, is only to say that every existing thing is F, so 'Fa' need not be true if 'a' does not designate an existing thing.

[3] We shall from this point forward drop the place-index '1.'

(On the other hand, 'Fa' may be true, given that the extensions of predicates can be drawn from the outer domain.)

3.2.4 Identity

The treatment of identity sentences in the semantics for Free Predicate Logic is the same as for the semantics for standard Predicate Logic. Identity in *PI* is treated semantically by the simple rule that an identity sentence is true on an interpretation just in case its two terms designate the same individual. As a result, we can validly assert the existence of everything to which a constant may refer. It is trivial that $\vDash_{PI}$ $\mathbf{a} = \mathbf{a}$ and that $\vDash_{PI}$ $\mathbf{u} = \mathbf{u}$.

Now consider the case of '$\mathbf{u} = \mathbf{a}$. On every *PI*-interpretation $\mathbf{I}$ there is a single domain $\mathbf{D}$ such that $\mathbf{v_I}(\mathbf{a}) \in \mathbf{D}$ and $\mathbf{v_I}(\mathbf{u}) \in \mathbf{D}$. No matter what value $\mathbf{v_I}$ assigns to '$\mathbf{u}$,' there is a variant of $\mathbf{v_I}$ which assigns to '$\mathbf{u}$' the same member $\mathbf{d}$ of the domain as $\mathbf{v_I}$ assigns to '$\mathbf{a}$.' Hence, $\mathbf{v_I}[\mathbf{d}/\mathbf{u}](\mathbf{a}) = \mathbf{v_I}(\mathbf{a})$, in which case $\mathbf{v_I}[\mathbf{d}/\mathbf{u}](\mathbf{u}) = \mathbf{v_I}[\mathbf{d}/\mathbf{u}](\mathbf{a})$. From this we can infer that $\mathbf{v_I}[\mathbf{d}/\mathbf{u}](\mathbf{u} = \mathbf{a}) = \mathbf{T}$, from which it follows that $\mathbf{v_I}(\exists\mathbf{x})(\mathbf{x} = \mathbf{a}) = \mathbf{T}$. Since the choice of interpretations is arbitrary, we have it that $\vDash_{PI}$ $(\exists\mathbf{x})\mathbf{x} = \mathbf{a}$.

The situation is different in Free Predicate Logic, where a constant can fail to designate an object in the inner domain. We may have a formula '$u = a$,' where on an interpretation $\mathbf{I}$, 'a' designates a member of the outer domain. In that case, there is no variant of a valuation function assigning a member $\mathbf{d}$ of the inner domain to 'u,' such that $\mathbf{v_I}[\mathbf{d}/u](u)$ is the same as $\mathbf{v_I}(a)$, so the identity formula is false on all those variants. In that case '$(\exists x)x = a$' is false. If '$(\exists x)x = a$' is false, then so is 'Ea', since 'a' fails to designate an object in the inner domain.

3.3 Formal Semantics for Free Predicate Logic

Below we give a formal treatment of the semantics for Free Predicate Logic. Not listed are semantical rules that carry over directly from the semantics for Predicate Logic, such as the rules for truth-functional compounds. Although the rules for the quantifiers are stated identically, it must be kept in mind that the domain $\mathbf{D}$ in the case of Free Predicate Logic is the inner domain, which shares no members with the outer domain.

Interpretations in Free Predicate Logic

$\mathbf{I} = \{\mathbf{D}, \mathbf{D}', \mathbf{v}\}$

$\mathbf{D} \neq \varnothing$

$\mathbf{D} \cap \mathbf{D}' = \varnothing$

Special Semantical Rules for Free Predicate Logic

1. $\mathbf{v_I}(\mathbf{a}) \in \mathbf{D} \cup \mathbf{D}'$
2. $\mathbf{v_I}(\mathbf{u}) \in \mathbf{D} \cup \mathbf{D}'$
3. $\mathbf{v}(\mathbf{P}^n) \subseteq (\mathbf{D} \cup \mathbf{D}')^n$
4. $\mathbf{v_I}(\mathrm{E}) = \mathbf{D}^1$
5. $\mathbf{v_I}((\forall\mathbf{x})(\alpha(\mathbf{x}/\mathbf{u})) = \mathbf{T}$ (where $\mathbf{u}$ is free for $\mathbf{x}$ in $\alpha(\mathbf{x})$) if and only if for all $\mathbf{d} \in \mathbf{D}$, $\mathbf{v}_{\mathrm{I}}[\mathbf{d}/\mathbf{u}](\alpha(\mathbf{u})) = \mathbf{T}$; $\mathbf{v_I}((\forall\mathbf{x})(\alpha(\mathbf{x}/\mathbf{u})) = \mathbf{F}$ if and only if for some $\mathbf{d} \in \mathbf{D}$, $\mathbf{v_I}[\mathbf{d}/\mathbf{u}](\alpha(\mathbf{u})) = \mathbf{F}$.
6. $\mathbf{v_I}((\exists\mathbf{x})(\alpha(\mathbf{x}/\mathbf{u})) = \mathbf{T}$ (where $\mathbf{u}$ is free for $\mathbf{x}$ in $\alpha(\mathbf{x})$) if and only if for some $\mathbf{d} \in \mathbf{D}$, $\mathbf{v}_{\mathrm{I}}[\mathbf{d}/\mathbf{u}](\alpha(\mathbf{u})) = \mathbf{T}$; $\mathbf{v_I}((\exists\mathbf{x})(\alpha(\mathbf{x}/\mathbf{u})) = \mathbf{F}$ if and only if for all $\mathbf{d} \in \mathbf{D}$, $\mathbf{v_I}[\mathbf{d}/\mathbf{u}](\alpha(\mathbf{u})) = \mathbf{F}$.

7. $\mathbf{v_I}(\mathbf{t}_i\mathbf{=}\mathbf{t}_j)\mathbf{=T}$ if and only if $\mathbf{v_I}(\mathbf{t}_i)\mathbf{=}\mathbf{v_I}(\mathbf{t}_j)$;
   $\mathbf{v_I}(\mathbf{t}_i\mathbf{=}\mathbf{t}_j)\mathbf{=F}$ if and only if $\mathbf{v_I}(\mathbf{t}_i)\neq\mathbf{v_I}(\mathbf{t}_j)$;

Here it should be noted that '**D**' in the rules for the quantifiers denotes the inner domain only, so that the quantifier rules stated here are substantively different from those for standard Predicate Logic.

We will carry the notions of semantical entailment, semantical equivalence, validity, and semantical consistency directly over from *PI*. There is nothing in *FPI* that requires a modification in any of them. They may be referred to as *Free Quantificational Semantical Entailment*, etc.

As was intimated in the earlier discussion, Free Predicate Logic is weaker than Predicate Logic. In general: $\{(\forall)\mathbf{x})\alpha(\mathbf{x/u})\} \nvDash_{FPL} \alpha(\mathbf{a/x})$ and $\{\alpha(\mathbf{a/x}) \nvDash_{FPL} (\exists\mathbf{x})\alpha(\mathbf{x/u})\}$. In both cases, '**a**' might designate in an interpretation a member of the outer domain rather than the inner domain. We do have the weaker results that: $\{(\forall\mathbf{x})\alpha(\mathbf{x/u}), \mathrm{E}\mathbf{a}\} \vDash_{FPL} \alpha(\mathbf{a/x})$ and $\{\alpha(\mathbf{a/x}), \mathrm{E}\mathbf{a}\} \vDash_{FPL} (\exists\mathbf{x})\alpha(\mathbf{x/u})\}$.

Though proofs will not be given here, it seems clear from the semantical rules just given that the properties of *Free Quantificational Bivalence* and *Free Quantificational Truth-Functionality* hold for *FPI*.

4 Derivations in Free Predicate Logic

A derivational system for Free Predicate Logic, *FPD*, can be built by modifying the elimination rule for the universal quantifier and the introduction rule for the existential quantifier.

We begin with Universal Elimination. In the derivational system *PD*, we were able to instantiate a universal sentence to any term whatsoever. But in the semantical sysem for Free Predicate Logic, this procedure is not valid. On an interpretation where a term '**t**' is not assigned to a member of the inner domain, the universal sentence '$(\forall\mathbf{x})\alpha(\mathbf{x/u}))$' may be true while the instantiation '$\alpha(\mathbf{t/u}))$' is false. This result applies to all terms, both constants and parameters.

To prevent such a result, we will restrict the use of Universal Elimination in a manner similar to the restriction on Universal Introduction in *PD*.[4] We shall allow instantiation only to a parameter that flanks a restricted scope line in which the sentence to be instantiated occurs. We can replace the first, invalid, inference, with one that conforms to the semantics for Free Predicate Logic.

1	$(\forall x)Fx$	Assumption
2	Fu	1 ∀ E (*PD*)

1	u	$(\forall x)Fx$	Assumption
2		Fu	1 ∀ E (*FPD*)

In effect, this isolation of the instantiation reflects the semantical reasoning that the derivation is supposed to capture. We suppose for an interpretation that '$(\forall x)Fx$' is true. Then we assume that some object d is in the inner domain, and we let 'u' stand for that member. Then 'Fu' is true. But this is only under the assumption that the parameter 'u' denotes an existing thing. We state the rule as follows.

[4]The following rules of inference are adaptations of the rules given in Karel Lambert and Bas van Fraassen, *Derivation and Counterexample: An Introduction to Philosophical Logic*, 1972.

Universal Elimination for Parameters (*FPL*)

$$
\begin{array}{l|l|ll}
 & \mathbf{v} & (\forall \mathbf{x})\alpha(\mathbf{x}/\mathbf{u}) & \text{Already Derived} \\
 & & \vdots & \\
 & & \alpha(\mathbf{v}/\mathbf{u}) & \forall\text{ E (Parameters)}
\end{array}
$$

In the system *PD*, there is nothing be to be done with the result in our example, except to generalize universally once again.

Proof that: $(\forall x)Fx \vdash_{FPL} (\forall y)Fy$

1	$(\forall x)Fx$	Assumption
2	u | $(\forall x)Fx$	1 Reiteration
3	| Fu	2 $\forall$ E (Parameters)
4	$(\forall y)Fy$	2-3 $\forall$ I (*PD*)

To instantiate to constants, we must use a different technique, since constants may not flag restricted sub-derivations. Here, we will use the existence predicate. So we will say that if some condition holds for everything in the inner domain, and 'a' designates something in that domain (and hence 'Ea' is true), then that condition holds for what 'a' designates. So, we will require that a universal sentence be instantiated when for constant **a**, the sentence 'E**a**' occurs at the same scope line.

Universal Elimination for Constants (*FPL*)

$$
\begin{array}{|ll}
(\forall \mathbf{x})\alpha(\mathbf{x}/\mathbf{u}) & \text{Already derived} \\
\vdots & \\
\mathrm{E}\mathbf{a} & \text{Already derived} \\
\vdots & \\
\alpha(\mathbf{a}/\mathbf{u}) & \forall\text{ E (Constants)}
\end{array}
$$

We must similarly restrict Existential Generalization. From a semantical point of view, we want to be able to generalize only when a term is assigned to a member of the inner domain. As we saw with Universal Generalization, this can be represented by flagged restricted sub-derivations. A sentence containing a parameter may be generalized upon only when it occurs within a restricted scope line with that parameter as its flag.

Existential Introduction for Parameters

$$
\begin{array}{l|l|ll}
 & \mathbf{v} & \alpha(\mathbf{v}/\mathbf{u}) & \text{Already Derived} \\
 & & \vdots & \\
 & & (\exists \mathbf{x})\alpha(\mathbf{x}/\mathbf{u}) & \exists\text{ I (Parameters)}
\end{array}
$$

The effect of this rule as stated may be stronger than we would like. There are only two rules which allow us to end a quantificational restricted sub-derivation: Universal Introduction and Existential Elimination. The former rule requires that the flagging parameter be replaced with a variable, and an appropriate quantifier added. But the rule ∃ I (Parameters) itself carries out this replacement. So the only way the result of Existential Introduction can be brought out is through the use of Existential Elimination, as with the following derivation.

Proof that: $(\exists x)Fx \vdash_{FPL} (\exists y)Fy$

1		$(\exists x)Fx$	Assumption
2	u	Fu	Assumption
3		$(\exists y)Fy$	2 ∃ I (Parameters)
4		$(\exists y)Fy$	1 2-3 ∃ E (*PD*)

We have no way, however, to generate the following derivation, since there is no way to remove the barrier.[5]

Attempt to prove: $(\forall x)Fx \vdash_{FPL} (\exists x)Fx$

1		$(\forall x)Fx$	Assumption
2	u	$(\forall x)Fx$	1 Reiteration
3		Fu	2 ∀ E (Parameters)
4		$(\exists x)Fx$	3 ∃ I (Parameters)
5		$(\exists x)Fx$	???

To this end, we need a new rule which allows us to end the barrier and bring out a sentence which does not contain the parameter which flags the barrier (so long as the sentence is not in the scope of any assumption). This makes sense semantically, because the barrier is supposed to reflect an assumption about what the flagging parameter refers to. That assumption is moot if the sentence does not contain the parameter.

Barrier Removal (*FPL*)

$$\begin{array}{l|l|ll} & \mathbf{u} & \vdots & \\ & & \alpha & \text{BR} \\ & \alpha & & \end{array}$$

Provided that: u does not occur in α,
α does not lie in the scope of any undischarged assumption.

Use of this rule allows us to get the desired result.

[5] This is an intended feature of the derivational system of Lambert and van Fraassen, since they want their system to accomodate a semantics that allows for the domain to be empty. In our semantics, the domain may not be empty, so the above inference is valid and should be permitted by the rules.

Proof that: $(\forall x)Fx \vdash_{FPL} (\exists x)Fx$

1		$(\forall x)Fx$	Assumption
2	u	$(\forall x)Fx$	1 Reiteration
3		Fu	2 $\forall$ E (Parameters)
4		$(\exists x)Fx$	3 $\exists$ I (Parameters)
5		$(\exists x)Fx$	4 BR

To generalize on a constant a, we require the occurrence of the sentence 'E**a**' occurs at the same scope line.

Existential Introduction for Constants

$\alpha(\mathbf{a}/\mathbf{u})$	Already Derived
$\vdots$	
E**a**	
$\vdots$	
$(\exists \mathbf{x})\alpha(\mathbf{x}/\mathbf{u})$	$\exists$ I (Constants)

Here are some examples of derivations that hold in *PD* but are not permitted by the rules of *FPD*. These are derivational correlates of examples discussed in the semantical section above. In each case, the problem lies in the attempt to use constants as the basis of the application of quantifier rules. We begin with Existential Generalization.

Attempt to prove: $\{Fa\} \vdash_{FPD} (\exists x)Fx$

1	Fa	Assumption
2	$(\exists x)Fx$	1 $\exists$ I (*PD*)

Step 2 is not permitted because the 'a' is a constant and not a parameter. The following attempt to generalize from a parameter fails as well, since the parameter does not fall within a restricted scope line containing 'u.'

Attempt to prove: $\{Fu\} \vdash_{FPD} (\exists x)Fx$

1	Fu	Assumption
2	$(\exists x)Fx$	1 $\exists$ I (*PD*)

If we do enclose it in the needed scope line, the existential sentence lies in the scope of an undischarged assumption.

Attempt to prove: $\{Fu\} \vdash_{FPD} (\exists x)Fx$

1	u	Fu	Assumption
2		$(\exists x)Fx$	1 $\exists$ I (*FPD*)

A special case of a failure of Existential Generalization involves the identity predicate.

Attempt to prove: $\vdash_{FPD} (\exists x)x = a$

1	$a = a$	$=$ I
2	$(\exists x)x = a$	1 $\exists$ I (*PD*)

Next we examine two attempted uses of Universal Instantiation, each of which fails because it does not meet one of the conditions of the rule in Free Predicate Logic.

Attempt to prove: $\{(\forall x)Fx\} \vdash_{FPD} Fa$

1	$(\forall x)Fx$	Assumption
2	Fa	2 $\forall$ E (*PD*)

Attempt to prove: $\{(\forall x)Fx\} \vdash_{FPD} Fu$

1	$(\forall x)Fx$	Assumption
2	Fu	2 $\forall$ E (*PD*)

This derivation fails because the quantifier rule operates on a sentence by instantiation to a sentence containing a constant, which is not permitted without '*Ea*.'

Attempt to prove: $\{(\forall x)Fx\} \vdash_{FPD} Fa$

1	$(\forall x)Fx$	Assumption
2	Fa	2 $\forall$ E (*PD*)

Finally, we can get the desired results for objects which are taken as existing.

Proof that: $\{Fa, \mathrm{E}a\} \vdash_{FPD} (\exists x)Fx$

1	Fa	Assumption
2	$\mathrm{E}a$	Assumption
3	$(\exists x)Fx$	1 2 $\exists$ E (Parameters)

Proof that: $\{(\forall x)Fx, \mathrm{E}a\} \vdash_{FPD} Fa$

1	$(\forall x)Fx$	Assumption
2	$\mathrm{E}a$	Assumption
3	Fa	1 2 $\forall$ E (Constants)