Announcements & Such

- *Shuggie Otis*.
- Administrative Stuff
  - **HW #5 first resubmission is due on Thursday.**
  - **My handout "Working with LMPL Interpretations" is posted (useful for part of HW #5). I will discuss this (again) today.**
  - **From now on, my office hours are: 4–6pm Tuesdays.**
- Today: Chapter 6 — LMPL Semantics
  - Validity and Invalidity in LMPL.
  - *Constructing* LMPL interpretations (to establish $\nvDash$ claims).
  - **Next**: Natural Deductions in LMPL (*i.e.*, rules for the quantifiers).

Constructing LMPL Interpretations to Prove $\nvDash$ Claims

- The notion of *semantic consequence* ($\vDash$) in LMPL is defined in the usual way. We say that $p_1, \ldots, p_n \vDash q$ in LMPL *iff* there is no LMPL interpretation on which all of $p_1, \ldots, p_n$ are true, but q is false.
- In HW #5, you are asked to prove that $p_1, \ldots, p_n \nvDash q$, for various p's and q's. This means you must *construct* (or, *find*) LMPL interpretations on which $p_1, \ldots, p_n$ are all true, but q is false.
- On page 2 of my "Working with LMPL Interpretations" handout, I have included two problems of this kind. There, I explain in detail *how I arrived at* my interpretations. This is a method you should emulate.
- On your HW's and exams, you will ***not*** need to explain *how you arrived at* your interpretations. But, you *will* need to *demonstrate* that your interpretations *really are counterexamples* (*i.e.*, that they *really are* interpretations on which $p_1, \ldots, p_n$ are all true, but q is false).

How Do We *Prove* $\vDash$ Claims in LMPL?

- In LSL, we had *systematic*, truth-table procedures for proving *both* negative ($\nvDash$) *and* affirmative ($\vDash$) semantical claims.
- The method of constructing LMPL interpretations *is* a general way to establish *negative* ($\nvDash$) LMPL-semantical claims.
- We will *not* be learning any systematic methods for (*directly*) establishing *affirmative* ($\vDash$) LMPL-semantical claims. There *are* such methods, but they are beyond the scope of this course.[a]
- In LMPL, we will rely on *natural deduction proofs* to give us an (*in*direct) method for demonstrating the *validity* of LMPL argument-forms. We'll talk about LMPL natural deductions soon.

[a] If an LMPL argument with k predicate letters is *in*valid, then there exists a *counterexample interpretation* $\mathcal{I}$ whose domain $\mathcal{D}$ has no more than 2^k elements. So, *exhaustive search* over *all* interpretations such that $|\mathcal{D}| \leq 2^k$ *is* a *decision procedure* for LMPL-validity. Note: this means checking $2^{2^k \cdot k}$ matrices. This is too many to check, even for small k. If $k = 2$, then $2^{2^k \cdot k} = 2^8 = 256$. For $k = 3$, this is 16777216! See pages 212–215 of Hunter's *Metalogic* (our 140A text). We discuss this in 140A.

Construction of LMPL Interpretations: Examples

- Here are six sample problems that require you to *construct* (or, *find*) LMPL interpretations that are *counterexamples* to $\vDash$ claims (the first two of these are solved on p. 2 of my handout on constructing LMPL interpretations):

(1) $(\forall x)(Fx \to Gx), (\forall x)(Fx \to Hx) \nvDash (\forall x)(Gx \to Hx)$

(2) $(\exists x)(Fx \,\&\, Gx), (\exists x)(Fx \,\&\, Hx), (\forall x)(Gx \to {\sim}Hx) \nvDash (\forall x)[Fx \leftrightarrow (Gx \vee Hx)]$

(3) $(\forall x)Fx \leftrightarrow (\forall x)Gx \nvDash (\exists x)(Fx \leftrightarrow Gx)$[a]

(4) $(\forall x)Fx \leftrightarrow A \nvDash (\forall x)(Fx \leftrightarrow A)$[b]

(5) $Fa \to (\exists x)Gx \nvDash (\exists x)Fx \to (\exists x)Gx$[c]

(6) $(\exists x)(\forall y)(Fx \to Gy) \nvDash (\exists y)(\forall x)(Fx \to Gy)$[d]

[a] One solution: $\mathcal{D} = \{a, b\}$, $\mathrm{Ext}(F) = \{a\}$, $\mathrm{Ext}(G) = \{b\}$.
[b] One solution: $\mathcal{D} = \{a, b\}$, 'A' is $\perp$, $\mathrm{Ext}(F) = \{a\}$.
[c] One solution: $\mathcal{D} = \{a, b\}$, $\mathrm{Ext}(F) = \{b\}$, $\mathrm{Ext}(G) = \varnothing$.
[d] One solution: $\mathcal{D} = \{a, b\}$, $\mathrm{Ext}(F) = \{a\}$, $\mathrm{Ext}(G) = \varnothing$.

Construction of LMPL Interpretations: Example #1

(1) $(\forall x)(Fx \to Gx), (\forall x)(Fx \to Hx) \nvDash (\forall x)(Gx \to Hx)$

- To prove (1), we need to construct (find) an interpretation $\mathcal{I}$ such that:
  (i) '$(\forall x)(Fx \to Gx)$' is true in $\mathcal{I}$.
  (ii) '$(\forall x)(Fx \to Hx)$' is true in $\mathcal{I}$.
  (iii) '$(\forall x)(Gx \to Hx)$' is false in $\mathcal{I}$.
- **Step 1**: We begin — *provisionally* — with the smallest domain $\mathcal{D} = \{a\}$.
- **Step 2**: We make sure that the object a is a *counterexample* to the conclusion '$(\forall x)(Gx \to Hx)$'. That is, we make sure that the *instance* '$Ga \to Ha$' of the conclusion is *false* on $\mathcal{I}$. So, we must have $a \in \text{Ext}(G)$, but $a \notin \text{Ext}(H)$. We can achieve this by: $\text{Ext}(G) = \{a\}$, and $\text{Ext}(H) = \varnothing$.
- **Step 3**: At the same time, we try to make *both* of the premises '$(\forall x)(Fx \to Gx)$' *and* '$(\forall x)(Fx \to Hx)$' *true* on $\mathcal{I}$.

- In this case, we can make both premises true simply by ensuring that $a \notin \text{Ext}(F)$. The simplest way to do this is to stipulate that $\text{Ext}(F) = \varnothing$ — which yields the following interpretation that does the trick:

$\mathcal{I}_{(1)}$:

	F	G	H
a	−	+	−

- We have discovered an interpretation $\mathcal{I}_{(1)}$ on which '$(\forall x)(Fx \to Gx)$' and '$(\forall x)(Fx \to Hx)$' are both true, but '$(\forall x)(Gx \to Hx)$' is false (*demonstrate this*!). Therefore, claim (1) is true.
- When you're asked to prove a claim like (1), you must do *2* things:
  - *Report* an interpretation (like $\mathcal{I}_2$) which serves as a counterexample to the validity of the LMPL argument-form, *and*
  - *Demonstrate* that your interpretation *really is* a counterexample — *i.e.*, *show* that your interpretation makes all the premises true and the conclusion false, using the methods above. You do ***not*** need to explain the process which led to the *discovery* of the interpretation.

Construction of LMPL Interpretations: Example #2

(2) $(\exists x)(Fx \,\&\, Gx), (\exists x)(Fx \,\&\, Hx), (\forall x)(Gx \to \sim Hx) \nvDash (\forall x)[Fx \leftrightarrow (Gx \vee Hx)]$

- We need an interpretation $\mathcal{I}$ on which '$(\exists x)(Fx \,\&\, Gx)$', '$(\exists x)(Fx \,\&\, Hx)$', and '$(\forall x)(Gx \to \sim Hx)$' are all $\top$, but '$(\forall x)[Fx \leftrightarrow (Gx \vee Hx)]$' is $\bot$.
- **Step 1**: We begin with the smallest possible domain $\mathcal{D} = \{a\}$.
- **Step 2**: We make sure that a is a *counterexample* to the conclusion '$(\forall x)[Fx \leftrightarrow (Gx \vee Hx)]$'. So, we make its *instance* '$Fa \leftrightarrow (Ga \vee Ha)$' $\bot$ on $\mathcal{I}$. One way to do this is: $a \in \text{Ext}(F)$, $a \notin \text{Ext}(G)$, and $a \notin \text{Ext}(H)$. So far, we have the following: $\text{Ext}(F) = \{a\}$, and $\text{Ext}(G) = \text{Ext}(H) = \varnothing$.
- **Step 3**: Now, we must make *all three* of the premises (*i*) '$(\exists x)(Fx \,\&\, Gx)$', (*ii*) '$(\exists x)(Fx \,\&\, Hx)$', and (*iii*) '$(\forall x)(Gx \to \sim Hx)$' $\top$ on $\mathcal{I}$. In order to make (*i*) $\top$ on $\mathcal{I}$, we must ensure that there is some object in the domain $\mathcal{D}$ which satisfies *both* 'F' *and* 'G'. But, since a must *not* satisfy both 'F' and 'G', this means we will need to *add another object b* to our domain $\mathcal{D}$.

- This new object b must be such that: $b \in \text{Ext}(F)$, and $b \in \text{Ext}(G)$. Now, we have $\text{Ext}(F) = \{a, b\}$, $\text{Ext}(G) = \{b\}$, and $\text{Ext}(H) = \varnothing$.
- All that remains is to ensure that premises (*ii*) and (*iii*) are also $\top$ on $\mathcal{I}$. In order to make (*ii*) $\top$ on $\mathcal{I}$, we'll need to make sure that there is some object in $\mathcal{D}$ which satisfies *both* 'F' *and* 'H'. We could *try* to make b satisfy *all three* 'F', 'G', *and* 'H'. But, if we were to do this, then premise (*iii*) would become *false* on $\mathcal{I}$, since its *instance* '$Gb \to \sim Hb$' would then be false on $\mathcal{I}$. Thus, we'll need to *add a third object c* to $\mathcal{D}$ such that: $c \in \text{Ext}(F)$, $c \notin \text{Ext}(G)$, and $c \in \text{Ext}(H)$ — and that does the trick:

$\mathcal{I}_{(2)}$:

	F	G	H
a	+	−	−
b	+	+	−
c	+	−	+

- We have discovered an interpretation $\mathcal{I}_{(2)}$ on which '$(\exists x)(Fx \,\&\, Gx)$', '$(\exists x)(Fx \,\&\, Hx)$', and '$(\forall x)(Gx \to \sim Hx)$' are all $\top$, but on which '$(\forall x)[Fx \leftrightarrow (Gx \vee Hx)]$' is false (*demonstrate this*!). $\therefore$ claim (2) is true.

Construction of LMPL Interpretations for $\nvDash$: Procedure

1. Begin with smallest domain possible $\mathcal{D} = \{\alpha\}$.
2. Make the conclusion of the $\nvDash$ claim false (for α).
   - That is, make the a-instance of the conclusion false.
3. Try to make all premises of the $\nvDash$ claim true (for α).
   - That is, make the a-instance of each of the premises true.
4. If you succeed, then you're done. Now *report and verify* your matrix.
5. If you fail, then add a new individual β to $\mathcal{D} = \{\alpha, \beta\}$, and continue.
6. Make the conclusion of the $\nvDash$ claim false.
   - If the conclusion is an $\forall$ claim, then it's already false.
   - If it's an $\exists$, then you must make sure its b-instance is also false.
7. Make the premises of the $\nvDash$ claim true.
   - If a premise is an $\forall$ claim, then *all* its instances must be true.
   - If it's an $\exists$ claim, only *one* of its instances needs to be true.
8. If you succeed, you're done. If not, add another (γ) to $\mathcal{D}$. Repeat . . .

Using Sentential Reasoning to "Verify" LMPL $\vDash$ Claims

$$(\forall x)(\exists y)(Fx \,\&\, Gy) \;\dashv\vDash\; (\exists y)(\forall x)(Fx \,\&\, Gy)$$

- To see why, think about the truth-conditions for each side:

$$(\forall x)(\exists y)(Fx \,\&\, Gy) \approx (\exists y)(Fa \,\&\, Gy) \,\&\, (\exists y)(Fb \,\&\, Gy) \,\&\, \cdots$$

$$\approx [(Fa \,\&\, Ga) \vee (Fa \,\&\, Gb) \vee \cdots] \,\&\, [(Fb \,\&\, Ga) \vee (Fb \,\&\, Gb) \vee \cdots] \,\&\, \cdots$$

$$\approx [Fa \,\&\, (Ga \vee Gb \vee \cdots)] \,\&\, [Fb \,\&\, (Ga \vee Gb \vee \cdots)] \,\&\, \cdots$$

$$\approx (Fa \,\&\, Fb \,\&\, Fc \,\&\, \cdots) \,\&\, (Ga \vee Gb \vee Gc \vee \cdots)$$

$$(\exists y)(\forall x)(Fx \,\&\, Gy) \approx (\forall x)(Fx \,\&\, Ga) \vee (\forall x)(Fx \,\&\, Gb) \vee \cdots$$

$$\approx [(Fa \,\&\, Ga) \,\&\, (Fb \,\&\, Ga) \,\&\, \cdots] \vee [(Fa \,\&\, Gb) \,\&\, (Fb \,\&\, Gb) \,\&\, \cdots] \vee \cdots$$

$$\approx [Ga \,\&\, (Fa \,\&\, Fb \,\&\, \cdots)] \vee [Gb \,\&\, (Fa \,\&\, Fb \,\&\, \cdots)] \vee \cdots$$

$$\approx (Ga \vee Gb \vee Gc \vee \cdots) \,\&\, (Fa \,\&\, Fb \,\&\, Fc \,\&\, \cdots)$$

- $\therefore$ These two formulas are *equivalent*, since the two red formulas are $(Ga \vee Gb \vee \cdots) \,\&\, (Fa \,\&\, Fb \,\&\, \cdots) \approx (Fa \,\&\, Fb \,\&\, \cdots) \,\&\, (Ga \vee Gb \vee \cdots)$

Natural Deduction Proofs in LMPL

- The natural deduction rules for LMPL will *include* the rules for LSL that we already know (*viz.*, Ass., &E, &I, $\rightarrow$E, $\rightarrow$I, $\sim$E, $\sim$I, DN, $\vee$E, $\vee$I, *Df*.).
- Plus, we will be *adding* 4 new rules. We will need both introduction and elimination rules for each of the two quantifiers ($\exists$I, $\exists$E, $\forall$I, $\forall$E).
- As in LSL, the system will be *sound and complete* (140A!). That is, $\vdash$ will apply to the same sequents that $\vDash$ does in our semantics for LMPL.
- We begin with the simplest: the introduction rule for $\exists$ ($\exists$I). Intuitively, if we have proved $\phi\tau$ for some individual constant τ, then we may infer that ϕ is true of *something* (*e.g.*, that $(\exists x)\phi x$).
- *E.g.*, if we've proved '$Pa \,\&\, Qa$', we may validly infer '$(\exists x)(Px \,\&\, Qx)$'.
- We may also infer '$(\exists x)(Pa \,\&\, Qx)$' and '$(\exists x)(Px \,\&\, Qa)$' from '$Pa \,\&\, Qa$'.
- These (and similar) considerations lead us to the $\exists$I rule . . .

The Rule of $\exists$-Introduction

Rule of $\exists$-Introduction: For any sentence $\phi\tau$, if $\phi\tau$ has been inferred at line j in a proof, then at line k we may infer $\ulcorner(\exists v)\phi v\urcorner$, labeling the line 'j $\exists$I' and writing on its left the numbers that occur on the left of j.

$$\begin{array}{llll} a_1,\ldots,a_n & (\mathrm{j}) & \phi\tau & \\ & \vdots & & \\ a_1,\ldots,a_n & (\mathrm{k}) & (\exists v)\phi v & \mathrm{j}\ \exists\mathrm{I} \end{array}$$

Where $\ulcorner(\exists v)\phi v\urcorner$ is obtained syntactically from $\phi\tau$ by:

- Replacing ***one or more occurrences*** of τ in $\phi\tau$ by a *single* variable v.
- Note: the variable v ***must not already occur in*** the expression $\phi\tau$. [This prevents *double-binding*, *e.g.*, '$(\exists x)(\exists x)(Fx \,\&\, Gx)$'.]
- And, finally, prefixing the quantifier $\ulcorner(\exists v)\urcorner$ in front of the resulting expression (which may now have both $\ulcorner v\urcorner$'s and $\ulcorner\tau\urcorner$'s occurring in it).

The Rule of ∀-Elimination

Rule of ∀-Elimination: For any sentence $\ulcorner(\forall\nu)\phi\nu\urcorner$ and constant τ, if $\ulcorner(\forall\nu)\phi\nu\urcorner$ has been inferred at a line j, then at line k we may infer $\phi\tau$, labeling the line 'j ∀E' and writing on its left the numbers that appear on the left of j.

$$\begin{array}{llll} a_1,\ldots,a_n & (j) & (\forall\nu)\phi\nu & \\ & \vdots & & \\ a_1,\ldots,a_n & (k) & \phi\tau & j\ \forall E \end{array}$$

Where $\phi\tau$ is obtained syntactically from $\ulcorner(\forall\nu)\phi\nu\urcorner$ by:

- Deleting the quantifier prefix $\ulcorner(\forall\nu)\urcorner$.
- Replacing ***every occurrence*** of ν in the open sentence $\phi\nu$ by ***one and the same*** constant τ. [This prevents *fallacies*, *e.g.*, $(\forall x)(Fx \,\&\, Gx)\ \ Fa \,\&\, Gb$.]
- Note: since '∀' means *everything*, there are *no* restrictions on *which* individual constant may be used in an application of ∀E.

An Example Proof Involving Both ∃I and ∀E

Let's prove that $(\forall x)(Fx \to Gx), Fa \vdash (\exists x)(\sim Gx \to Hx)$.

1	(1)	(∀x)(Fx→Gx)	Premise
2	(2)	Fa	Premise
3	(3)	~Ga	Assumption
4	(4)	~Ha	Assumption
1	(5)	Fa→Ga	1 ∀E
1,2	(6)	Ga	5,2 →E
1,2,3	(7)	⋀	3,6 ~E
1,2,3	(8)	~~Ha	4,7 ~I
1,2,3	(9)	Ha	8 DN
1,2	(10)	~Ga→Ha	3,9 →I
1,2	(11)	(∃x)(~Gx→Hx)	10 ∃I

- This example illustrates a typical pattern in quantificational proofs: quantifiers are removed from the premises using elimination rules, sentential (*viz.*, LSL) rules are applied, and then quantifiers are reintroduced using introduction rules to obtain the conclusion.

The Rule of ∀-Introduction: Some Background

- It is useful to think of a universal claim $\ulcorner(\forall\nu)\phi\nu\urcorner$ as a *conjunction* which asserts that the predicate expression ϕ is satisfied by *all objects* in the domain of discourse (*i.e.*, the conjunction $\ulcorner\phi a \,\&\, (\phi b \,\&\, (\phi c \,\&\, \ldots))\urcorner$ is true).
- So, in order to be able to *introduce* the universal quantifier (*i.e.*, to *legitimately infer* $\ulcorner(\forall\nu)\phi\nu\urcorner$ in a proof), we must be in a position to prove $\phi\tau$, for *any* individual constant τ. This is called *generalizable reasoning*.
- Consider the following *legitimate* introduction of a universal claim:

Problem is: (∀x)(Fx→Gx), (∀x)Fx ⊢ (∀x)Gx

1	(1)	(∀x)(Fx→Gx)	Premise
2	(2)	(∀x)Fx	Premise
1	(3)	Fa→Ga	1 ∀E
2	(4)	Fa	2 ∀E
1,2	(5)	Ga	3,4 →E
1,2	(6)	(∀x)Gx	5 ∀I

The Rule of ∀-Introduction: II

- We can legitimately infer '$(\forall x)Gx$' at line 6 of this proof, because our inference to 'Gb' is *generalizable* — *i.e.*, we could have deduced $\ulcorner G\tau\urcorner$, for *any* individual constant τ — using *exactly parallel* reasoning.
- However, consider the following *il*legitimate "∀-Introduction" step:

1	(1)	(∀x)(Fx→Gx)	Premise	
2	(2)	Fb	Premise	
1	(3)	Fb→Gb	1 ∀E	
1,2	(4)	Gb	2,3 →E	
1,2	(5)	(∀x)Gx	4 ∀I	NO!!

- This is *not* a valid inference, since $(\forall x)(Fx \to Gx), Fb \nvDash (\forall x)Gx$!
- So, what went wrong? The problem is that the inference to 'Gb' at (4) is *not* generalizable. We can *not* deduce $\ulcorner G\tau\urcorner$ — for *any* τ — from the premises '$(\forall x)(Fx \to Gx)$' and 'Fb'. We can *only* infer '$G\boldsymbol{b}$'.

The Rule of ∀-Introduction: III

Rule of ∀-Introduction: For any sentence $\phi\tau$, if $\phi\tau$ has been inferred at a line j, then *provided that τ does not occur in any premise or assumption whose line number is on the left at line* j, we may infer $\ulcorner(\forall\nu)\phi\nu\urcorner$ at line k, labeling the line 'j ∀I' and writing on its left the same numbers as occur on the left at line j.

$a_1,\ldots,a_n$	(j)	$\phi\tau$	
	⋮		
$a_1,\ldots,a_n$	(k)	$(\forall\nu)\phi\nu$	j ∀I

Where $\ulcorner(\forall\nu)\phi\nu\urcorner$ is obtained by:

- Replacing ***every*** occurrence of τ in $\phi\tau$ with ν and prefixing $\ulcorner(\forall\nu)\urcorner$.
  [Again, 'every' prevents *fallacies*, *e.g.*, $(\forall x)(Fx \to Gx)$ $(\forall x)(\forall y)(Fx \to Gy)$.]
- τ ***does not occur in*** any of the formulae $a_1,\ldots,a_n$. [ensures *generalizability*]
- ν ***does not occur in*** $\phi\tau$. [prevents *double-binding*]

The Rule of ∀-Introduction: Four Examples

- Here are four examples of LMPL sequents involving the three quantifier rules we've learned so far (∃I, ∀E, and ∀I).

(1) $(\forall x)(Fx \to Gx) \vdash (\forall x)Fx \to (\forall x)Gx$

(2) $\sim(\exists x)(Fx \,\&\, Gx) \vdash (\forall x)(Fx \to \sim Gx)$

(3) $\sim(\forall x)Fx \vdash (\exists x)\sim Fx$

(4) $(\forall x)[Fx \to (\forall y)Gy] \vdash (\forall x)(\forall y)(Fx \to Gy)$

Proof of (1)

Problem is: (∀x)(Fx→Gx) ⊢ (∀x)Fx→(∀x)Gx

1	(1)	(∀x)(Fx→Gx)	Premise
2	(2)	(∀x)Fx	Assumption
1	(3)	Fa→Ga	1 ∀E
2	(4)	Fa	2 ∀E
1,2	(5)	Ga	3,4 →E
1,2	(6)	(∀x)Gx	5 ∀I
1	(7)	(∀x)Fx→(∀x)Gx	2,6 →I

Proof of (2)

Problem is: ~(∃x)(Fx&Gx) ⊢ (∀x)(Fx→~Gx)

1	(1)	~(∃x)(Fx&Gx)	Premise
2	(2)	Fa	Assumption
3	(3)	Ga	Assumption
2,3	(4)	Fa&Ga	2,3 &I
2,3	(5)	(∃x)(Fx&Gx)	4 ∃I
1,2,3	(6)	Λ	1,5 ~E
1,2	(7)	~Ga	3,6 ~I
1	(8)	Fa→~Ga	2,7 →I
1	(9)	(∀x)(Fx→~Gx)	8 ∀I

Proof of (3)

Problem is: ~(∀x)Fx ⊦ (∃x)~Fx

1	(1)	~(∀x)Fx	Premise
2	(2)	~(∃x)~Fx	Assumption
3	(3)	~Fa	Assumption
3	(4)	(∃x)~Fx	3 ∃I
2,3	(5)	Λ	2,4 ~E
2	(6)	~~Fa	3,5 ~I
2	(7)	Fa	6 DN
2	(8)	(∀x)Fx	7 ∀I
1,2	(9)	Λ	1,8 ~E
1	(10)	~~(∃x)~Fx	2,9 ~I
1	(11)	(∃x)~Fx	10 DN

Proof of (4)

Problem is: (∀x)(Fx→(∀y)Gy) ⊦ (∀x)(∀y)(Fx→Gy)

1	(1)	(∀x)(Fx→(∀y)Gy)	Premise
2	(2)	Fa	Assumption
1	(3)	Fa→(∀y)Gy	1 ∀E
1,2	(4)	(∀y)Gy	3,2 →E
1,2	(5)	Gb	4 ∀E
1	(6)	Fa→Gb	2,5 →I
1	(7)	(∀y)(Fa→Gy)	6 ∀I
1	(8)	(∀x)(∀y)(Fx→Gy)	7 ∀I