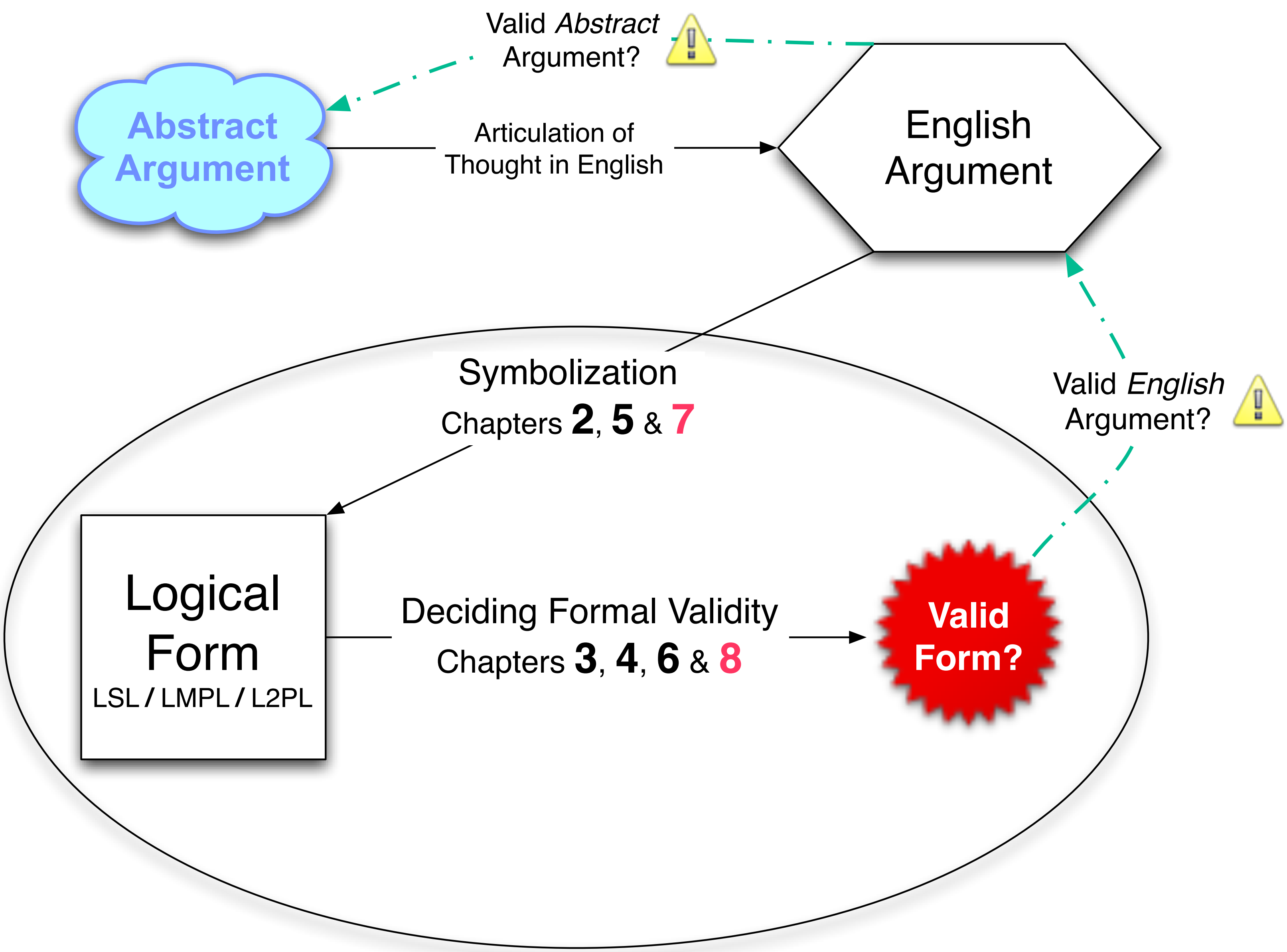
Announcements & Such

- Administrative Stuff
  - **HW #6 to be handed back today. Resubs due Thursday.**
  - **I will be posting both the sample in-class final and the take-home final on Thursday. [I'll discuss them Thursday.]**
  - **I'll have office hours on Thursday from 2–4. [Not today.]**
  - **Review session: Monday, May 10 @ 4–6pm @ Wheeler 213.**
  - **GSI Office Hours: Tamar (W: 10–12 & next W: 10–12), Julia (W: 2–4 & Tu: 3–5), David (F: tba)**
  - **In-class final: Thurs. May 13 @ 3–6pm *here* (A1 Hearst Annex).**
  - I've posted my solutions to HW #4 (I'll post others before final).
  - I've posted a handout with *all* natural deduction rules (for final).
- Today: Beyond Chapter 6 — "L2PL" — Binary Relations



Valid *Abstract* Argument?
Abstract Argument
Articulation of Thought in English
English Argument
Symbolization
Chapters 2, 5 & 7
Valid *English* Argument?
Logical Form
LSL / LMPL / L2PL
Deciding Formal Validity
Chapters 3, 4, 6 & 8
Valid Form?

Beyond LMPL: 2-Place Predicates (*a.k.a.*, Relations) II

- From the point of view of logic (as opposed to mathematics) what matters is *capturing validities*. And, LMPL captures more than LSL.
- But, LMPL also has its own *logical* limitations. The problem: we can't capture some of the intuitively valid arguments involving *relations*.
- Consider the following argument, which involves a 2-place predicate:

(1) Brutus killed Caesar.

(2) ∴ Brutus killed someone and someone killed Caesar.

- If we were to symbolize this argument using monadic predicates, we would end-up with something like the following LMPL reconstruction:

(1′) Kb.

(2′) $\therefore\ (\exists x)Bx \mathbin{\&} (\exists y)Ky$.

Where Kx: x killed Caesar, Bx: Brutus killed x, and b: Brutus.

- This argument is *not* valid in LMPL. But, the English argument *is* valid!

- The problem here is that "x killed y" is a *2-place* predicate (or *relation*).
- If we expand our language to include predicates that can take 2 arguments, then we can capture statements and arguments like these.
- In chapter 7, a more general language is introduced that allows n-place predicates, for any finite n. We will only discuss 2-place predicates.
- For instance, we can introduce the 2-place predicate Kxy: x killed y. With this relation in hand, we can express the above argument as:

(1*) Kbc.

(2*) $\therefore (\exists x)Kbx \,\&\, (\exists y)Kyc$.

- In 2-place predicate logic ("L2PL"), this argument *is* valid. So, this is a more accurate and faithful formalization of the English argument.
- We will (in chapter 8) discuss the semantics for 2-place predicate logic (L2PL). The natural deduction system for L2PL is *the same as* LMPL's!
- Before that, we will look at various complexities of L2PL *symbolization*.

Some Sample L2PL Symbolization Problems

1. Someone loves someone. [Lxy: x loves y]
   - First, work on the the quantifier with widest scope, then *work in*.
   - There exists an x such that x loves someone.

   (i) $(\exists x)$ x loves someone.

   - Now, work on expression within the scope of the quantifier in (i).

   (ii) x loves someone
   - there exists a y such that Lxy
   - $(\exists y)Lxy$

   - Plugging the symbolization of (ii) into (i) yields the **final product**:

$$(\exists x)(\exists y)Lxy$$

2. Everyone loves everyone.
   - For all x, x loves everyone.
   - $(\forall x)$ x loves everyone.
   - x loves everyone $\mapsto (\forall y)Lxy$
   - $(\forall x)(\forall y)Lxy$
3. Everyone loves someone.
   - For all x, x loves someone.
   - $(\forall x)$ x loves someone.
   - x loves someone $\mapsto (\exists y)Lxy$
   - $(\forall x)(\exists y)Lxy$
4. Someone loves everyone.
   - There exists an x such that x loves everyone.
   - $(\exists x)$ x loves everyone.
   - x loves everyone $\mapsto (\forall y)Lxy$
   - $(\exists x)(\forall y)Lxy$

Four Important Properties of Binary Relations

- **Reflexivity**. A binary relation R is said to be *reflexive* iff $(\forall x)Rxx$.
- **Symmetry**. R is *symmetric* iff $(\forall x)(\forall y)(Rxy \to Ryx)$.
- **Transitivity**. R is *transitive* iff $(\forall x)(\forall y)(\forall z)[(Rxy \,\&\, Ryz) \to Rxz]$.
- If R has *all three* of these properties, then R is an *equivalence relation*.
- **Fact.** If R is Euclidean and reflexive, then R is an equivalence relation.

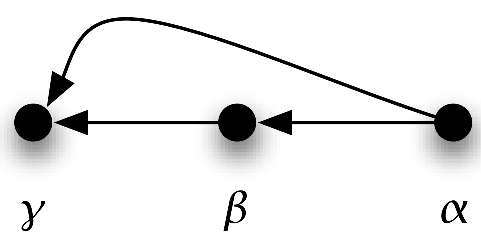
Relation	Reflexive?	Symmetric?	Transitive?	Euclidean?
$x > y$	No	No	Yes	No
$x \vDash y$	Yes	No	Yes	No
x is a sibling of y	No	Yes	No	No
$x \approx y$	Yes	Yes	No	No
x respects y	No	No	No	No
$x = y$	Yes	Yes	Yes	Yes

L2PL Interpretations I

- Here's an example L2PL interpretation. Oxy: x was older than y, $\mathcal{D}$: The Three Stooges, $\text{Ref}(a) = $ Curly, $\text{Ref}(b) = $ Larry, and $\text{Ref}(c) = $ Moe.
- The matrix representation of $\text{Ext}(O)$ for this interpretation is:

O	α	β	γ
α	$-$	$+$	$+$
β	$-$	$-$	$+$
γ	$-$	$-$	$-$

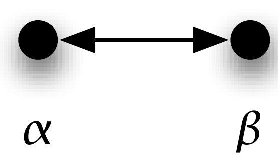
- The pictorial or diagrammatic representation of $\text{Ext}(O)$ is:



L2PL Interpretations III

($\mathcal{I}_1$) Let $\mathcal{D}$ be the set consisting of George W. Bush (α) and Jeb Bush (β). And, let Bxy: x is a brother of y. Determine $\mathcal{I}_1$-truth-values for:

1. $(\forall x)(\exists y)Bxy$
2. $(\exists y)(\forall x)Bxy$



- (1) is $\top$ on $\mathcal{I}_1$, since *both* of its $\mathcal{D}$-instances are $\top$ on $\mathcal{I}_1$.
  - ‘$(\exists y)Bay$’ is $\top$ on $\mathcal{I}_1$ because its instance ‘Bab’ is $\top$ on $\mathcal{I}_1$.
    - That is, $\langle \alpha, \beta \rangle \in \text{Ext}(B)$. Note: $\text{Ext}(B) = \{\langle \alpha, \beta \rangle, \langle \beta, \alpha \rangle\}$.
  - ‘$(\exists y)Bby$’ is $\top$ on $\mathcal{I}_1$ because its instance ‘Bba’ is $\top$ on $\mathcal{I}_1$.
- (2) is $\bot$ on $\mathcal{I}_1$, since *both* of its $\mathcal{D}$-instances are $\bot$ on $\mathcal{I}_1$.
  - ‘$(\forall x)Bxa$’ is $\bot$ on $\mathcal{I}_1$ because its instance ‘Baa’ is $\bot$ on $\mathcal{I}_1$.
    - That is, $\langle \alpha, \alpha \rangle \notin \text{Ext}(B)$.
  - ‘$(\forall x)Bxb$’ is $\bot$ on $\mathcal{I}_1$ because its instance ‘Bbb’ is $\bot$ on $\mathcal{I}_1$.

L2PL Interpretations IV

- Just as with LMPL, L2PL interpretations can be used as counterexamples to validity claims. Establishing $\nvDash$ claims works just as you'd expect.
- We have just seen an L2PL interpretation that shows the following:

$$(\forall x)(\exists y)Rxy \nvDash (\exists x)(\forall y)Rxy$$

- Interpretation $\mathcal{I}_1$ on the previous slide is a counterexample. Why?
  - $(\forall x)(\exists y)Bxy$ is $\top$ on $\mathcal{I}_1$, since both of its instances are $\top$ on $\mathcal{I}_1$.
  - $(\exists x)(\forall y)Rxy$ is $\bot$ on $\mathcal{I}_1$, since both of its instances are $\bot$ on $\mathcal{I}_1$.
- Here is a *very important* L2PL invalidity:

(†) $(\forall x)(\exists y)Rxy, (\forall x)(\forall y)(\forall z)[(Rxy \,\&\, Ryz) \to Rxz] \nvDash (\exists x)Rxx$

- (†) reveals a surprising difference between LMPL (and LSL) and L2PL — **sometimes *infinite* interpretations are needed to prove $\nvDash$ in L2PL!**

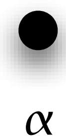
Why (†) is So Important — L2PL *vs* LMPL: Infinite Domains

- In LMPL, if p is true on any interpretation $\mathcal{I}$, then it is true on a *finite* interpretation. Indeed, p will be true on an interpretation of size no greater than 2^k, where k is the # of monadic predicate letters in p.
- In L2PL, some statements are true *only* on *infinite* interpretations. It is for this reason that there is no general decision procedure for validity (or logical truth) in L2PL. (†) on the last slide is a good example of this.

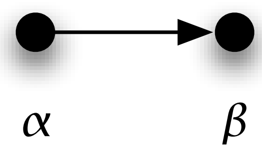
(†) $(\forall x)(\exists y)Rxy, (\forall x)(\forall y)(\forall z)[(Rxy \,\&\, Ryz) \to Rxz] \nvDash (\exists x)Rxx$

- **Fact**. *Only infinite interpretations $\mathcal{I}$ can be counterexamples to the validity in* (†). To see why, try to *construct* such an interpretation.
- We start by showing that no interpretation $\mathcal{I}_1$ with a 1-element domain can be an interpretation on which the premises of (†) are $\top$ and its conclusion is $\bot$. Then, we will repeat this argument for I_2 and $\mathcal{I}_3$.
- This reasoning can, in fact, be shown correct for *all* (finite) n. So, only $\mathcal{I}$'s with infinite domains will work [*e.g.*, $\mathcal{D} = \mathbb{N}$, Rxy: $x < y$].
- Begin with a 1-element domain $\{\alpha\}$. For the conclusion of (4) to be $\bot$, no

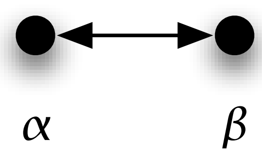
object can be related to itself: $(\forall x){\sim}Rxx$. Thus, we must have ${\sim}Raa$:



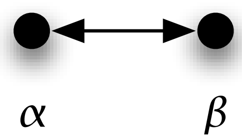
- But, to make the first premise $\top$, we need there to be *some* y such that Ray is $\top$. That means we need *another object* β to allow Rab. Thus:

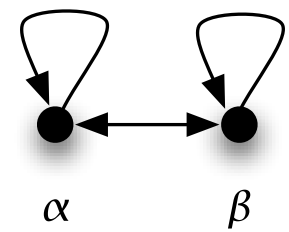


- Now, because we need the conclusion to remain $\bot$, we must have ${\sim}Rbb$. And, because we need the first premise to remain $\top$, we need there to be *some* y such that Rby is $\top$. We could *try* to make Rba $\top$, as follows:

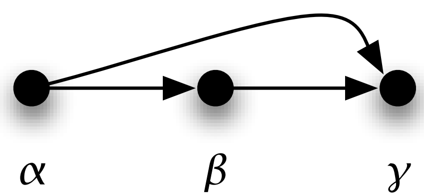


- But, this picture is not consistent with the second premise being $\top$ and (at the same time) the conclusion being $\bot$. If R is transitive, then $Rab \,\&\, Rba$ (as pictured) entails Raa, which makes the conclusion $\top$.

Transitivity of R and 

entails: 

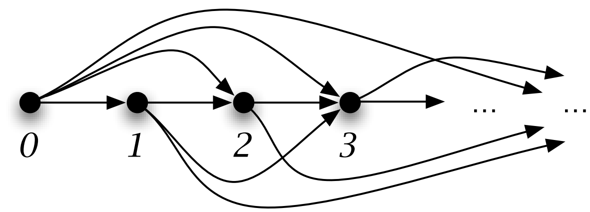
- Thus, the only way to consistently ensure that there is some y such that Rby is to introduce *yet another object* γ (such that Rbc), which yields:



- Again, in order to make the conclusion $\bot$, we must have $\sim Rcc$, and in order to make the first premise $\top$, there must be some y such that Rcy.

- We could *try* to make either Rca or Rcb true. But, both of these choices will end-up with the same sort of inconsistency we just saw with β.

- In other words, *no finite interpretation* will give us what we want here.
- However, if we let $\mathcal{D} = \mathbb{N}$ and Rxy: $x < y$, then we get what we want.



- That is, the relation Rxy: $x < y$ *on the natural numbers* $\mathbb{N}$ is such that:
  - For all x, there exists a y such that $x < y$. [seriality]
  - For all x, y, z, if $x < y$ and $y < z$, then $x < z$. [transitivity]
  - For all x, $x \nless x$. [irreflexivity]
- It is crucial that the set $\mathbb{N}$ of *all* natural numbers is *infinite*. The relation $<$ cannot satisfy all three of these properties on *any finite* domain.
- *I.e.*, no finite subset of $\mathbb{N}$ will suffice to show that the invalidity in (4) holds. Equivalently, the following sentence of L2PL is $\perp$ on *all finite* $\mathcal{I}$'s:

  $$p \overset{\text{def}}{=} (\forall x)(\exists y)Rxy \,\&\, (\forall x)(\forall y)(\forall z)[(Rxy \,\&\, Ryz) \rightarrow Rxz] \,\&\, (\forall x){\sim}Rxx$$
- This sort of thing *cannot happen* in LMPL. In this sense, the introduction of a single 2-place predicate involves a *quantum leap* in complexity.

Some Further Remarks on Validity in L2PL

- As I just explained, there is no general decision procedure for $\vDash$ claims in L2PL. This is because we can't always establish $\nvDash$ claims in finite time.
- However, there is a method for proving $\vDash$ claims — *natural deduction*. And, L2PL's natural deduction system *is exactly the same as LMPL's*!
- Before we get to proofs, however, I want to look at the alternating quantifier example that I said separates LMPL and L2PL.
- As we have seen, $(\forall x)(\exists y)Rxy \nvDash (\exists y)(\forall x)Rxy$. But, the converse entailment *does* hold. That is, $(\exists y)(\forall x)Rxy \vDash (\forall x)(\exists y)Rxy$.
- We will *prove — i.e., deduce* — $(\exists y)(\forall x)Rxy \vdash (\forall x)(\exists y)Rxy$ shortly.
- Before we do that, let's think about $(\exists y)(\forall x)Rxy \vDash (\forall x)(\exists y)Rxy$ using our definitions, and our informal method of thinking of $\forall$ as & and $\exists$ as $\vee$. This is interesting for both directions of the entailment.
- But, we need to be much more careful here than with LMPL!

- First, consider what $(\exists y)(\forall x)Rxy$ says on a domain of size n:
$$(\exists y)(\forall x)Rxy \approx_n (\forall x)Rxa \vee (\forall x)Rxb \vee \cdots \vee (\forall x)Rxn$$
$$\approx_n (Raa \,\&\cdots\&\, Rna) \vee (Rab \,\&\cdots\&\, Rnb) \vee \cdots \vee (Ran \,\&\cdots\&\, Rnn)$$
- Next, consider what $(\forall x)(\exists y)Rxy$ says on a domain of size n:
$$(\forall x)(\exists y)Rxy \approx_n (\exists y)Ray \,\&\, (\exists y)Rby \,\&\cdots\&\, (\exists y)Rny$$
$$\approx_n (Raa \vee \cdots \vee Ran) \,\&\, (Rba \vee \cdots \vee Rbn) \,\&\cdots\&\, (Rna \vee \cdots \vee Rnn)$$
- Then, we notice that these two sentential forms are intimately related. Specifically, we note that $(\exists y)(\forall x)Rxy$ has the following n-form:
$\mathcal{X}_n = (p_1 \,\&\, p_2 \,\&\cdots\&\, p_n) \vee (q_1 \,\&\, q_2 \,\&\cdots\&\, q_n) \vee \cdots \vee (r_1 \,\&\, r_2 \,\&\cdots\&\, r_n)$
- And, we notice that $(\forall x)(\exists y)Rxy$ has the following n-form:
$\mathcal{Y}_n = (p_1 \vee q_1 \vee \cdots \vee r_1) \,\&\, (p_2 \vee q_2 \vee \cdots \vee r_2) \,\&\cdots\&\, (p_n \vee q_n \vee \cdots \vee r_n)$
- **Fact.** $\mathcal{X}_n \vDash \mathcal{Y}_n$, *for any* n. Each disjunct of $\mathcal{X}_n$ entails every conjunct of $\mathcal{Y}_n$. **Caution!** This ***doesn't*** show that $(\exists y)(\forall x)Rxy \vDash (\forall x)(\exists y)Rxy$!
- **Fact.** $\mathcal{Y}_n \nvDash \mathcal{X}_n$, for all $n > 1$. This can be shown (next slide) using only LSL reasoning. This ***does*** show that $(\forall x)(\exists y)Rxy \nvDash (\exists y)(\forall x)Rxy$.
- The moral is that our "informal" semantical approach to the quantifiers works for LMPL, since no infinite domains are required for $\nvDash$ in LMPL.

- However, our "informal" semantical approach breaks down for L2PL, since we sometimes need an infinite domain to establish $\nvDash$ in L2PL.
- In L2PL, if the "informal" method above reveals $p_n \nvDash q_n$ for *some* finite n, then it *does* follow that $p \nvDash q$. For instance, $\mathcal{Y}_2 \nvDash \mathcal{X}_2$ on the last slide:
  - $(Raa \vee Rab) \,\&\, (Rba \vee Rbb) \nvDash (Raa \,\&\, Rba) \vee (Rab \,\&\, Rbb)$
  - This is just an LSL problem with 4-atoms [$A = Raa$, $B = Rab$, $C = Rba$, $D = Rbb$]. Truth-tables will generate a counterexample.
- On the other hand, if (in L2PL) our "informal" method indicates (as above) that $p_n \vDash q_n$ for *all* finite n, this does *not* guarantee $p \vDash q$. *E.g.*:
  - $p = (\forall x)(\exists y)Rxy \,\&\, (\forall x)(\forall y)(\forall z)[(Rxy \,\&\, Ryz) \to Rxz]$.
  - $q = (\exists x)Rxx$.
- We showed above (informally) that $p_n \vDash q_n$ for *all* finite n. But, we also saw that there are infinite interpretations on which p is $\top$ but q is $\bot$.