Mathematical Logic
Where We Are Now

• Our coverage of logic focused on these topics:
  • Propositional variables.
  • Propositional connectives.
  • Propositional equivalences.
  • Predicates, functions, and constant symbols.
  • Objects and propositions.
  • Quantifiers.
  • Evaluating first-order formulas relative to a world.
  • Translating into first-order logic.
  • Negating and simplifying first-order formulas.

• Your goal this week is to keep your proof skills sharp while mastering the ins and outs of first-order logic.
Things You Should Do Today

• Review the solutions for Problem Set One and make sure you *completely* and *unambiguously* understand the answers. Ask for help if this isn't the case!

• Read the “Guide to Negating Formulas” and “Guide to First-Order Translation” on the course website to get more exposure and practice with those skills.

• Continue working through PS2.
Things You Should Do Tomorrow

• Look over your feedback on PS1 and make sure you understand all the feedback you get *completely* and *unambiguously*. Ask the course staff for help, either on Piazza or in office hours, if you don't.

• Continue working on PS2.

• Start reviewing your partner's answers and compiling a single, definitive set of answers that you're going to turn in.

• Stop by office hours to get feedback on your proofs and take that feedback seriously.
Mechanics: *Negating Statements*
∀p. (Person(p) → 
  ∃q. (Person(q) ∧ p ≠ q ∧ Loves(p, q) 
    ) 
  )
\neg \forall p. (Person(p) \rightarrow \\
\exists q. (Person(q) \land p \neq q \land \\
\quad Loves(p, q)) \\
)
\neg \forall p. (\text{Person}(p) \rightarrow \\
\exists q. (\text{Person}(q) \land p \neq q \land \\
\text{Loves}(p, q) \\
) \\
) \\

Useful Resource:

Go to cs103.stanford.edu and read the Guide to Negating Formulas.
\neg \forall p. (Person(p) \to \\
\exists q. (Person(q) \land p \neq q \land \\
Loves(p, q) \\
) \\
)

\hline
\neg \forall x. A \\
\underline{\exists x. \neg A}
\(\neg \forall p. (\text{Person}(p) \rightarrow \exists q. (\text{Person}(q) \land p \neq q \land \text{Loves}(p, q))\)\)
\neg \forall p. (\text{Person}(p) \rightarrow \\
\exists q. (\text{Person}(q) \land p \neq q \land \\
\text{Loves}(p, q)) \\
) \\
)
\( \exists p. \neg (\text{Person}(p) \rightarrow \exists q. (\text{Person}(q) \land p \neq q \land \text{Loves}(p, q)) \) \)
\[\exists p. \neg (\text{Person}(p) \rightarrow \exists q. (\text{Person}(q) \land p \neq q \land \text{Loves}(p, q))\]
\[\exists p. \neg(Person(p) \rightarrow
\exists q. (Person(q) \land p \neq q \land
\text{Loves}(p, q))\]

\[\neg(A \rightarrow B) \quad \Rightarrow \quad A \land \neg B\]
∃p. ¬(Person(p) → 
    ∃q. (Person(q) ∧ p ≠ q ∧ 
        Loves(p, q) 
    )
)

\[\neg(A \rightarrow B)\]
\[\underline{A \land \neg B}\]
\[\exists p. \neg (\text{Person}(p) \to \exists q. (\text{Person}(q) \land p \neq q \land \text{Loves}(p, q)))\]
\( \exists p. \ ( \text{Person}(p) \land \neg \exists q. \ (\text{Person}(q) \land p \neq q \land \text{Loves}(p, q)) \) \)
\[ \exists p. \ (\text{Person}(p) \land \\
\quad \neg \exists q. \ (\text{Person}(q) \land p \neq q \land \text{Loves}(p, q)) \]
\[ \exists p. (\text{Person}(p) \land \\
\neg \exists q. (\text{Person}(q) \land p \neq q \land \\
\text{Loves}(p, q)) \\
\) \]

\[ \neg \exists x. A \\
\hline \\
\forall x. \neg A \]
\[ \exists p. \ (\text{Person}(p) \land \\
\neg \exists q. \ (\text{Person}(q) \land p \neq q \land \\
\text{Loves}(p, q) \\
) \)
\]

\[ \neg \exists x. \ A \]

\[ \Rightarrow \]

\[ \forall x. \ \neg A \]
\[ \exists p. \ (\text{Person}(p) \land \neg \exists q. \ (\text{Person}(q) \land p \neq q \land \text{Loves}(p, q))) \]
\( \exists p. \ (\text{Person}(p) \land \\
\forall q. \ \neg (\text{Person}(q) \land p \neq q \land \\
\text{Loves}(p, q)) \\
) \\
) \\
)

\[
\begin{array}{c}
\neg \exists x. A \\
\hline
\forall x. \neg A
\end{array}
\]
\[ \exists p. \ (\text{Person}(p) \land \\
\forall q. \ \neg(\text{Person}(q) \land p \neq q \land \\
\text{Loves}(p, q)) \) \]
$\exists p. \ (\text{Person}(p) \land$

$\forall q. \ \neg(\text{Person}(q) \land p \neq q \land$

$\text{Loves}(p, q)$

$)$

$)$

$\neg(A \land B)$

$\frac{\neg(A \land B)}{A \rightarrow \neg B}$
\[ \exists p. \ (\text{Person}(p) \land \forall q. \neg(\text{Person}(q) \land p \neq q \land \text{Loves}(p, q)) \]
\[\exists p. \ (\text{Person}(p) \land \forall q. \ \neg(\text{Person}(q) \land p \neq q \land \text{Loves}(p, q))\)\]

\[\neg(A \land B) \quad \frac{\neg(A \land B)}{A \rightarrow \neg B}\]
\[ \exists p. (\text{Person}(p) \land \\
\forall q. (\text{Person}(q) \land p \neq q \rightarrow \\
\neg \text{Loves}(p, q)) \) \]
\[ \exists p. \, (\text{Person}(p) \land \\
\forall q. \, (\text{Person}(q) \land p \neq q \rightarrow \\
\neg \text{Loves}(p, q)) \) \]
∀p. (Person(p) →
    ∃q. (Person(q) ∧ p ≠ q ∧
        Loves(p, q))
  )
)
)

∃p. (Person(p) ∧
    ∀q. (Person(q) ∧ p ≠ q →
        ¬Loves(p, q))
  )
)
Your turn!

Try negating this formula with the other folks at your table.

See what you come up with!
¬∃p. (Person(p) \land \\
\forall q. (Person(q) \land p \neq q \rightarrow \\
\quad Loves(q, p))
\)

)
∀p. ¬(Person(p) ∧
    ∀q. (Person(q) ∧ p ≠ q → Loves(q, p)
    )
  )
∀p. (Person(p) →
    −∀q. (Person(q) ∧ p ≠ q →
        Loves(q, p))
)
)
∀p. (Person(p) → 
   ∃q. ¬(Person(q) ∧ p ≠ q → Loves(q, p)) 
  )
∀p. (Person(p) →
   ∃q. (Person(q) ∧ p ≠ q ∧
         ¬Loves(q, p)
   )
)
)
\[
\exists p. \ (Person(p) \land \\
\forall q. \ (Person(q) \land p \neq q \rightarrow \\
\quad Loves(q, p))
\]

\[
\forall p. \ (Person(p) \rightarrow \\
\exists q. \ (Person(q) \land p \neq q \land \\
\quad \neg Loves(q, p))
\]
Techniques: *Translating Statements*
Common Patterns

• A statement of the form

$$\forall x. \ (P(x) \rightarrow Q(x))$$

can be read as “all P's are Q's.”

• A statement of the form

$$\exists x. \ (P(x) \land Q(x))$$

can be read as “there is a P that is also a Q” or “some P's are Q's.”

• **Remember:** If you see $\exists$ paired with $\rightarrow$ or $\forall$ paired with $\land$, the statement is probably incorrect!
Given the predicates

- \textit{Person}(p), which states that p is a person, and
- \textit{CanLearnFrom}(x, y), which says that x can learn from y,

write a statement in first-order logic that says “everyone has someone they can learn from.”
Everyone has someone they can learn from
Every person $p$ has someone they can learn from
Every person $p$ has someone they can learn from

“All As are Bs.”

$\forall x. (A(x) \rightarrow B(x))$
∀p. (Person(p) →
  p has someone they can learn from)
∀p. (\text{Person}(p) \rightarrow \text{there is a person } q \text{ that } p \text{ can learn from})
∀p. (Person(p) →
   there is a person q that p can learn from
)

“Some As are Bs.”
∃x. (A(x) ∧ B(x))
∀p. (Person(p) →
   ∃q. (Person(q) ∧
       p can learn from q
     )
   )
)
\( \forall p. \ (Person(p) \rightarrow \exists q. \ (Person(q) \land CanLearnFrom(p, q)) \) \)
Consider this statement:

“If someone is happy, then everyone is happy.”

What is the contrapositive of this statement?
If someone is happy, then everyone is happy
someone is happy $\rightarrow$ everyone is happy
someone is happy → (∀x. Happy(x))
(∃x. Happy(x)) → (∀x. Happy(x))
(∃x. Happy(x)) → (∀x. Happy(x))
$(\exists x. \text{Happy}(x)) \rightarrow (\forall x. \text{Happy}(x))$
\neg (\forall x. \ Happy(x)) \to \neg (\exists x. \ Happy(x))
\[(\exists x. \neg \text{Happy}(x)) \rightarrow \neg (\exists x. \text{Happy}(x))\]
(∃x. ¬Happy(x)) → (∀x. ¬Happy(x))
(∃x. ¬Happy(x)) → (∀x. ¬Happy(x))

“If someone is not happy, then everyone is not happy.”
Consider this statement:

“If someone is happy, then everyone is happy.”

What is the negation of this statement?
\((\exists x. \text{Happy}(x)) \rightarrow (\forall x. \text{Happy}(x))\)
(∃x. Happy(x)) → (∀x. Happy(x))
¬((∃x. Happy(x)) → (∀x. Happy(x)))
(∃x. Happy(x)) → (∀x. Happy(x))

(∃x. Happy(x)) ∧ ¬(∀x. Happy(x))
(∃x. Happy(x)) → (∀x. Happy(x))

(∃x. Happy(x)) ∧ (∃x. ¬Happy(x))
(∃x. Happy(x)) → (∀x. Happy(x))

(∃x. Happy(x)) ∧ (∃x. ¬Happy(x))

“Someone is happy and someone is not happy.”