

Mathematical Logic

Part Three

Recap from Last Time

What is First-Order Logic?

- ***First-order logic*** is a logical system for reasoning about properties of objects.
- Augments the logical connectives from propositional logic with
 - ***predicates*** that describe properties of objects,
 - ***functions*** that map objects to one another, and
 - ***quantifiers*** that allow us to reason about many objects at once.

“For any natural number n ,
 n is even iff n^2 is even”

$\forall n. (n \in \mathbb{N} \rightarrow (Even(n) \leftrightarrow Even(n^2)))$

\forall is the **universal quantifier**
and says “for any choice of n ,
the following is true.”

Some muggle is intelligent.

$\exists m. (Muggle(m) \wedge Intelligent(m))$

\exists is the **existential quantifier** and says "for some choice of m , the following is true."

“All A's are B's”

translates as

$\forall x. (A(x) \rightarrow B(x))$

Useful Intuition:

Universally-quantified statements are true unless there's a counterexample.

$$\forall x. (A(x) \rightarrow B(x))$$

If x is a counterexample, it must have property A but not have property B .

“Some A is a B ”

translates as

$\exists x. (A(x) \wedge B(x))$

Useful Intuition:

Existentially-quantified statements are false unless there's a positive example.

$$\exists x. (A(x) \wedge B(x))$$

If x is an example, it must have property A on top of property B .

The Aristotelian Forms

“All As are Bs”

“Some As are Bs”

$\forall x. (A(x) \rightarrow B(x))$

$\exists x. (A(x) \wedge B(x))$

“No As are Bs”

“Some As aren’t Bs”

$\forall x. (A(x) \rightarrow \neg B(x))$

$\exists x. (A(x) \wedge \neg B(x))$

It is worth committing these patterns to memory. We'll be using them throughout the day and they form the backbone of many first-order logic translations.

The Art of Translation

Using the predicates

- *Person*(p), which states that p is a person, and
- *Loves*(x, y), which states that x loves y ,

write a sentence in first-order logic that means “everybody loves someone else.”

$$\forall p. (Person(p) \rightarrow$$
$$\quad \exists q. (Person(q) \wedge p \neq q \wedge$$
$$\quad \quad Loves(p, q)$$
$$\quad)$$
$$)$$

Using the predicates

- $Person(p)$, which states that p is a person, and
- $Loves(x, y)$, which states that x loves y ,

write a sentence in first-order logic that means “there is a person that everyone else loves.”

$$\begin{aligned} & \exists p. (Person(p) \wedge \\ & \quad \forall q. (Person(q) \wedge p \neq q \rightarrow \\ & \quad \quad Loves(q, p) \\ & \quad) \\ &) \end{aligned}$$

Combining Quantifiers

- Most interesting statements in first-order logic require a combination of quantifiers.
- Example: “Everyone loves someone else.”

$$\forall p. (Person(p) \rightarrow \exists q. (Person(q) \wedge p \neq q \wedge Loves(p, q)))$$

For every person,

there is some person

who isn't them

that they love.

Combining Quantifiers

- Most interesting statements in first-order logic require a combination of quantifiers.
- Example: “There is someone everyone else loves.”

$\exists p. (Person(p) \wedge \forall q. (Person(q) \wedge p \neq q \rightarrow Loves(q, p)))$

There is some person

who everyone

who isn't them

loves.

For Comparison

$\forall p. (\text{Person}(p) \rightarrow \exists q. (\text{Person}(q) \wedge p \neq q \wedge \text{Loves}(p, q)))$

For every person,

there is some person

who isn't them

that they love.

$\exists p. (\text{Person}(p) \wedge \forall q. (\text{Person}(q) \wedge p \neq q \rightarrow \text{Loves}(q, p)))$

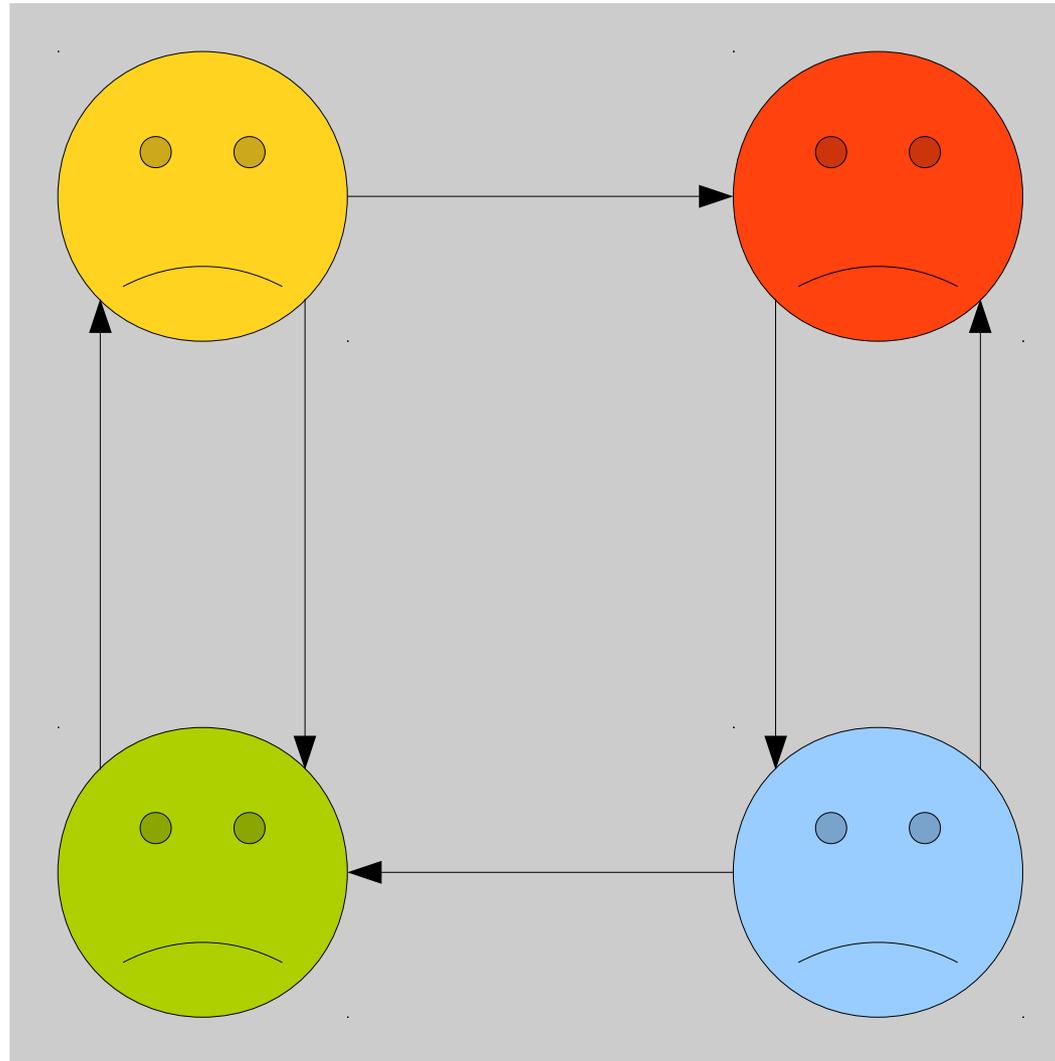
There is some person

who everyone

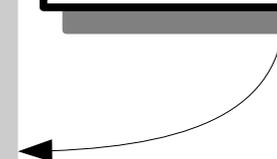
who isn't them

loves.

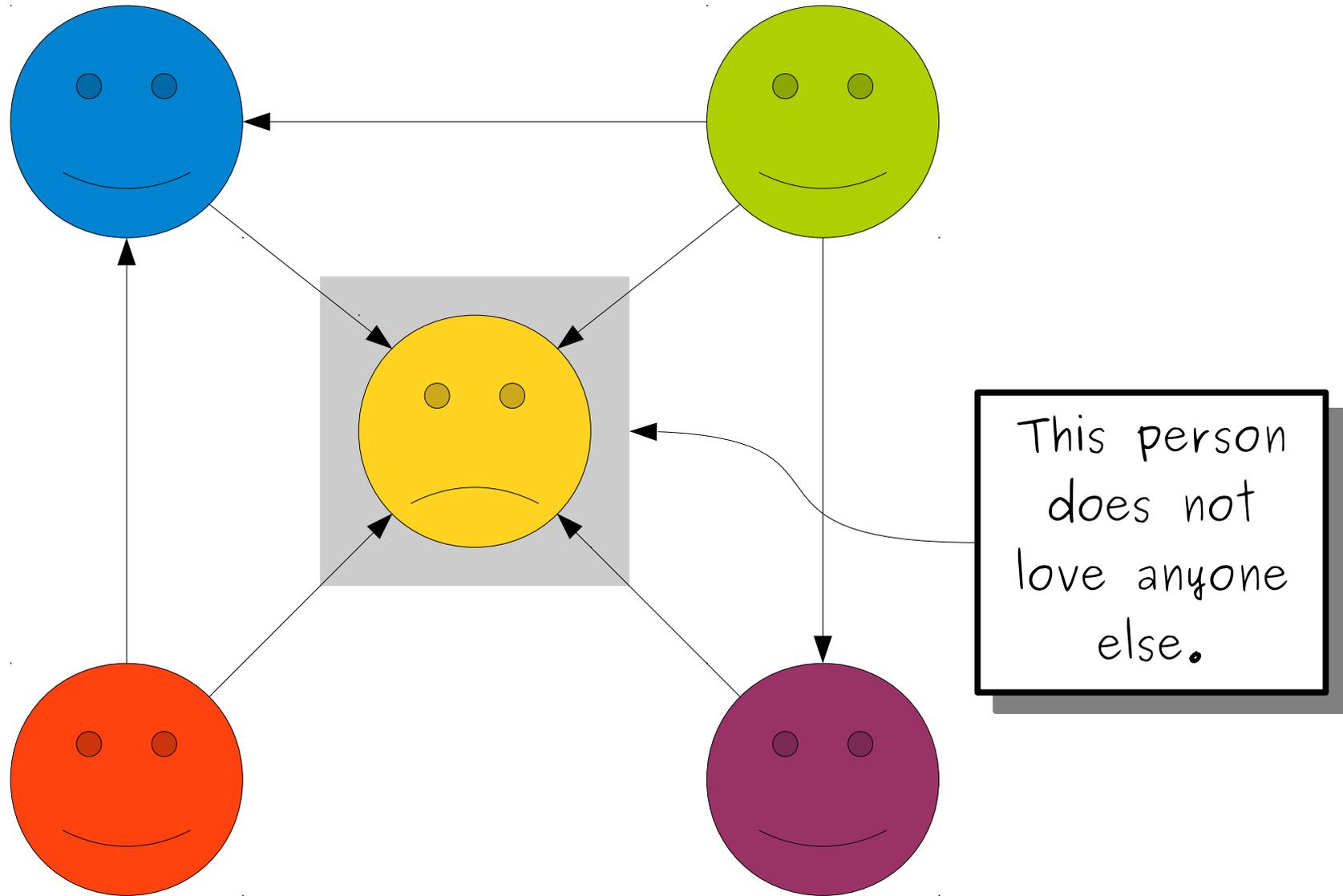
Everyone Loves Someone Else



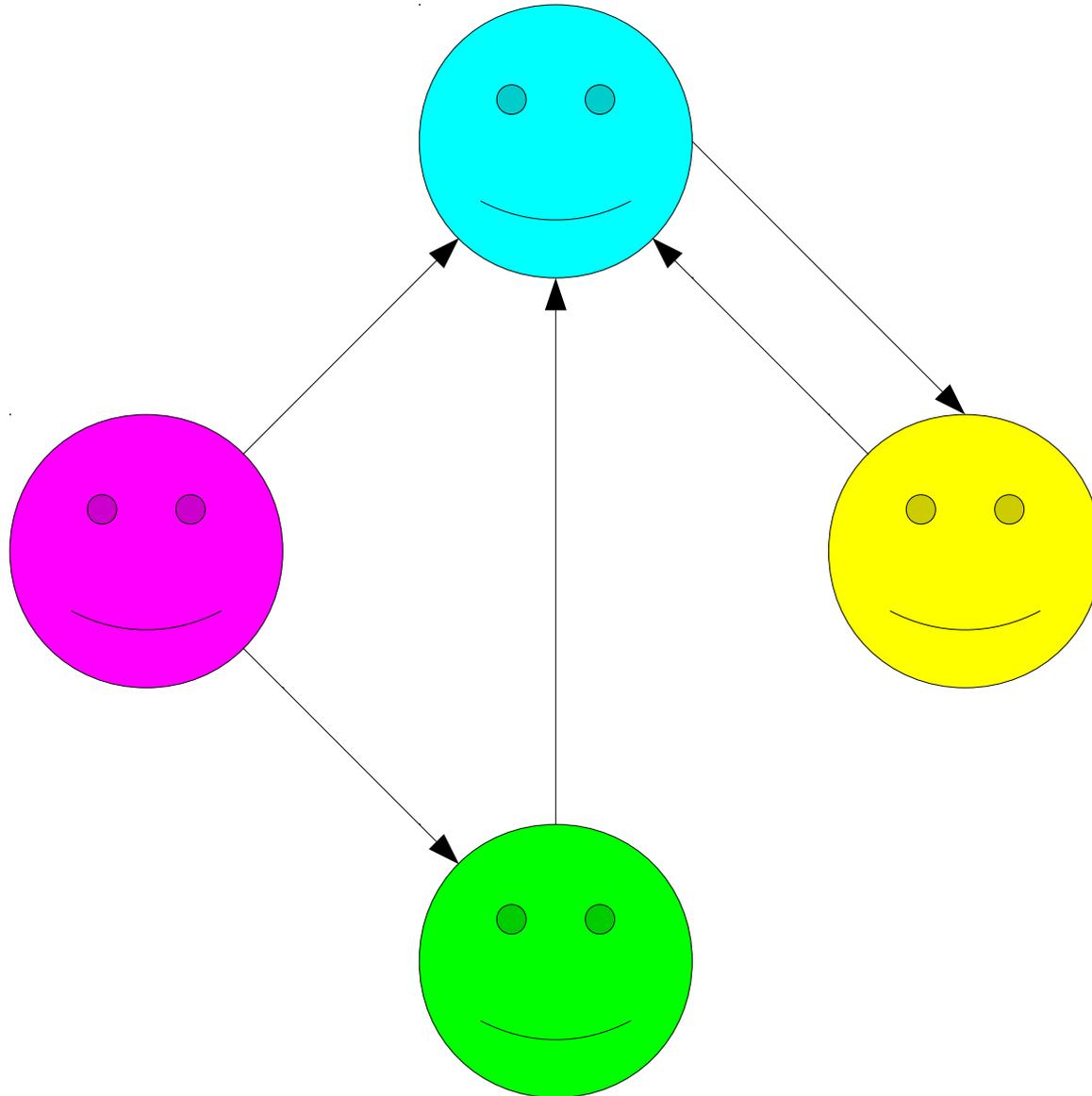
No one here
is universally
loved.



There is Someone Everyone Else Loves



Everyone Loves Someone Else *and*
There is Someone Everyone Else Loves



$$\forall p. (Person(p) \rightarrow \exists q. (Person(q) \wedge p \neq q \wedge Loves(p, q)))$$

For every person,

there is some person

who isn't them

that they love.

\wedge

$$\exists p. (Person(p) \wedge \forall q. (Person(q) \wedge p \neq q \rightarrow Loves(q, p)))$$

There is some person

who everyone

who isn't them

loves.

Quantifier Ordering

- The statement

$$\forall x. \exists y. P(x, y)$$

means “for any choice x , there's some y where $P(x, y)$ is true.”

- The choice of y can be different every time and can depend on x .

Quantifier Ordering

- The statement

$$\exists x. \forall y. P(x, y)$$

means “there is some x where for any choice of y , we get that $P(x, y)$ is true.”

- Since the inner part has to work for any choice of y , this places a lot of constraints on what x can be.

Order matters when mixing existential
and universal quantifiers!

Time-Out for Announcements!

Problem Set Two

- Problem Set One was due at 3:00PM today. You can submit late up until Monday at 3:00PM.
- Problem Set Two goes out now.
 - Checkpoint due Monday at 3:00PM.
 - Remaining problems due Friday.
- Play around with propositional and first-order logic and practice your proofwriting!
- As always, feel free to ask us questions in office hours or on Piazza!

University Townhall

- University President Marc Tessier-Lavigne and Provost Persis Drell are holding a town hall event on April 21st at noon in Cubberly Auditorium.
- Have input on the upcoming long-range planning process? Stop on by and let your voice be heard!

HACK OVERFLOW

10AM - 10PM April 15th @ Y2E2

Come to the annual Women in Computer Science hackathon!

All Stanford students are welcome, and beginners encouraged!

- > Network with various tech companies
- > Learn from industry mentors and graduate students
- > Win amazing prizes

RSVP or Volunteer at our website:

<http://web.stanford.edu/group/wics/hackoverflow/spr2017/>



[/events/1591013420914468/](https://www.facebook.com/events/1591013420914468/)

Your Questions

“What was the biggest difficulty you had in your undergrad life and how did you overcome it?”

I'll talk about
this one in
class. 😊

“How much does GPA matter when applying to jobs? What about applying to coterms?”

Most places tend to use GPAs as a negative filter: a low GPA will raise some eyebrows, but unless you have a truly impressive (> 4.0) GPA anything that's "good enough" should be fine. Figure that a 3.5 or higher is probably good enough for most companies and a 3.6 or higher is good enough for the coterms.

Be very careful about making your GPA a goal in and of itself. It's easy to make yourself miserable doing that.

“What was your favorite class (other than 103) and least favorite class when you were at Stanford?”

There are a bunch of classes that I really, really liked. I had a knack for taking fun classes from professors who were about to retire, like Planetary Exploration, Set Theory, and A History of Russian Music. I also am hugely indebted to my IHUM and PWR classes for making me a better writer. E50 was also a ton of fun, as were CS154 and CS108.

I don't have a least favorite class - sorry!

Back to CS103!

Set Translations

Using the predicates

- $Set(S)$, which states that S is a set, and
- $x \in y$, which states that x is an element of y ,

write a sentence in first-order logic that means “the empty set exists.”

First-order logic doesn't have set operators or symbols “built in.” If we only have the predicates given above, how might we describe this?

$\exists S. (Set(S) \wedge \neg \exists x. x \in S)$

$\exists S. (Set(S) \wedge \forall x. \neg (x \in S))$

Both of these translations are correct. Just like in propositional logic, there are many different equivalent ways of expressing the same statement in first-order logic.

Using the predicates

- $Set(S)$, which states that S is a set, and
- $x \in y$, which states that x is an element of y ,

write a sentence in first-order logic that means “two sets are equal if and only if they contain the same elements.”

$$\forall S. (\text{Set}(S) \rightarrow$$
$$\quad \forall T. (\text{Set}(T) \rightarrow$$
$$\quad \quad (S = T \leftrightarrow \forall x. (x \in S \leftrightarrow x \in T))$$
$$\quad)$$
$$)$$

$$\forall S. (\text{Set}(S) \rightarrow$$
$$\quad \forall T. (\text{Set}(T) \rightarrow$$
$$\quad\quad (S = T \leftrightarrow \forall x. (x \in S \leftrightarrow x \in T)))$$
$$\quad)$$
$$)$$

You sometimes see the universal quantifier pair with the \leftrightarrow connective. This is especially common when talking about sets because two sets are equal when they have precisely the same elements.

Mechanics: Negating Statements

Negating Quantifiers

- We spent much of Monday's lecture discussing how to negate propositional constructs.
- How do we negate statements with quantifiers in them?

An Extremely Important Table

	When is this true?	When is this false?
$\forall x. P(x)$	For any choice of x , $P(x)$	$\exists x. \neg P(x)$
$\exists x. P(x)$	For some choice of x , $P(x)$	$\forall x. \neg P(x)$
$\forall x. \neg P(x)$	For any choice of x , $\neg P(x)$	$\exists x. P(x)$
$\exists x. \neg P(x)$	For some choice of x , $\neg P(x)$	$\forall x. P(x)$

Negating First-Order Statements

- Use the equivalences

$$\neg \forall x. A \equiv \exists x. \neg A$$

$$\neg \exists x. A \equiv \forall x. \neg A$$

to negate quantifiers.

- Mechanically:
 - Push the negation across the quantifier.
 - Change the quantifier from \forall to \exists or vice-versa.
- Use techniques from propositional logic to negate connectives.

Taking a Negation

$\forall x. \exists y. \text{Loves}(x, y)$
(*“Everyone loves someone.”*)

$\neg \forall x. \exists y. \text{Loves}(x, y)$

$\exists x. \neg \exists y. \text{Loves}(x, y)$

$\exists x. \forall y. \neg \text{Loves}(x, y)$

(*“There's someone who doesn't love anyone.”*)

Two Useful Equivalences

- The following equivalences are useful when negating statements in first-order logic:

$$\neg(p \wedge q) \equiv p \rightarrow \neg q$$

$$\neg(p \rightarrow q) \equiv p \wedge \neg q$$

- These identities are useful when negating statements involving quantifiers.
 - \wedge is used in existentially-quantified statements.
 - \rightarrow is used in universally-quantified statements.
- When pushing negations across quantifiers, we *strongly recommend* using the above equivalences to keep \rightarrow with \forall and \wedge with \exists .

Negating Quantifiers

- What is the negation of the following statement, which says “there is a cute puppy”?

$$\exists x. (\mathit{Puppy}(x) \wedge \mathit{Cute}(x))$$

- We can obtain it as follows:

$$\neg \exists x. (\mathit{Puppy}(x) \wedge \mathit{Cute}(x))$$

$$\forall x. \neg (\mathit{Puppy}(x) \wedge \mathit{Cute}(x))$$

$$\forall x. (\mathit{Puppy}(x) \rightarrow \neg \mathit{Cute}(x))$$

- This says “no puppy is cute.”
- Do you see why this is the negation of the original statement from both an intuitive and formal perspective?

$$\exists S. (Set(S) \wedge \forall x. \neg(x \in S))$$

(“There is a set that doesn't contain anything”)

$$\neg \exists S. (Set(S) \wedge \forall x. \neg(x \in S))$$

$$\forall S. \neg(Set(S) \wedge \forall x. \neg(x \in S))$$

$$\forall S. (Set(S) \rightarrow \neg \forall x. \neg(x \in S))$$

$$\forall S. (Set(S) \rightarrow \exists x. \neg \neg(x \in S))$$

$$\forall S. (Set(S) \rightarrow \exists x. x \in S)$$

(“Every set contains at least one element”)

These two statements are *not* negations of one another. Can you explain why?

$$\exists S. (Set(S) \wedge \forall x. \neg(x \in S))$$

(“There is a set that doesn't contain anything”)

$$\forall S. (Set(S) \wedge \exists x. (x \in S))$$

(“Everything is a set that contains something”)

Remember: \forall usually goes with \rightarrow , not \wedge

Restricted Quantifiers

Quantifying Over Sets

- The notation

$$\forall x \in S. P(x)$$

means “for any element x of set S , $P(x)$ holds.” (It’s vacuously true if S is empty.)

- The notation

$$\exists x \in S. P(x)$$

means “there is an element x of set S where $P(x)$ holds.” (It’s false if S is empty.)

Quantifying Over Sets

- The syntax

$$\forall x \in S. \varphi$$

$$\exists x \in S. \varphi$$

is allowed for quantifying over sets.

- In CS103, feel free to use these restricted quantifiers, but please do not use variants of this syntax.
- For example, don't do things like this:



$$\forall x \text{ with } P(x). Q(x)$$



$$\forall y \text{ such that } P(y) \wedge Q(y). R(y).$$



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



Expressing Uniqueness

Using the predicate

- *Level(l)*, which states that *l* is a level,

write a sentence in first-order logic that means “there is only one level.”

A fun diversion:

http://www.onemorelevel.com/game/there_is_only_one_level

$$\exists l. (Level(l) \wedge \forall x. (x \neq l \rightarrow \neg Level(x)))$$

$\exists l. (Level(l) \wedge$
 $\forall x. (Level(x) \rightarrow x = l)$
)

Expressing Uniqueness

- To express the idea that there is exactly one object with some property, we write that
 - there exists at least one object with that property, and that
 - there are no other objects with that property.
- You sometimes see a special “uniqueness quantifier” used to express this:

$$\exists!x. P(x)$$

- For the purposes of CS103, please do not use this quantifier. We want to give you more practice using the regular \forall and \exists quantifiers.

Next Time

- ***Binary Relations***
 - How do we model connections between objects?
- ***Equivalence Relations***
 - How do we model the idea that objects can be grouped into clusters?
- ***First-Order Definitions***
 - Where does first-order logic come into all of this?
- ***Proofs with Definitions***
 - How does first-order logic interact with proofs?