Mathematical Logic

Part Two

Outline for Today

- **Recap from Last Time**
- **The Contrapositive**
- **Using Propositional Logic**
- **First-Order Logic**
- **First-Order Translations**

Recap from Last Time

Recap So Far

- A ***propositional variable*** is a variable that is either true or false.
- The ***propositional connectives*** are
  - Negation: $\neg p$
  - Conjunction: $p \land q$
  - Disjunction: $p \lor q$
  - Implication: $p \to q$
  - Biconditional: $p \leftrightarrow q$
  - True: $\top$
  - False: $\bot$

Logical Equivalence

- Two propositional formulas φ and ψ are called ***equivalent*** if they have the same truth tables.
- We denote this by writing $\varphi \equiv \psi$.
- Some examples:
  - $\neg(p \land q) \equiv \neg p \lor \neg q$
  - $\neg(p \lor q) \equiv \neg p \land \neg q$
  - $\neg p \lor q \equiv p \to q$
  - $p \land \neg q \equiv \neg(p \to q)$

One Last Equivalence

The Contrapositive

- The contrapositive of the statement

$$p \to q$$

  is the statement

$$\neg q \to \neg p$$

- These are logically equivalent, which is why proof by contradiction works:

$$\boldsymbol{p \to q \quad \equiv \quad \neg q \to \neg p}$$

Why All This Matters

- Suppose we want to prove the following statement:

“If $x + y = 16$, then $x \geq 8$ or $y \geq 8$”

$$x + y = 16 \to x \geq 8 \lor y \geq 8$$

Why All This Matters

- Suppose we want to prove the following statement:

“If $x + y = 16$, then $x \geq 8$ or $y \geq 8$”

$$x < 8 \land y < 8 \to x + y \neq 16$$

“If $x < 8$ and $y < 8$, then $x + y \neq 16$”

Theorem: If $x + y = 16$, then either $x \geq 8$ or $y \geq 8$.

Proof: By contrapositive. We prove that if $x < 8$ and $y < 8$, then $x + y \neq 16$. To see this, note that

$$\begin{aligned} x + y &< 8 + y \\ &< 8 + 8 \\ &= 16 \end{aligned}$$

So $x + y < 16$, so $x + y \neq 16$. ■

Why All This Matters

- Suppose we want to prove the following statement:

"If x + y = 16, then $x \geq 8$ or $y \geq 8$"

$$\neg(x + y = 16 \to x \geq 8 \lor y \geq 8)$$

Why All This Matters

- Suppose we want to prove the following statement:

“If $x + y = 16$, then $x \geq 8$ or $y \geq 8$”

$$x + y = 16 \land x < 8 \land y < 8$$

“$x + y = 16$, but $x < 8$ and $y < 8$.”

Theorem: If $x + y = 16$, then either $x \geq 8$ or $y \geq 8$.

Proof: Assume for the sake of contradiction that $x + y = 16$, but $x < 8$ and $y < 8$. Then

$$\begin{aligned} x + y &< 8 + y \\ &< 8 + 8 \\ &= 16 \end{aligned}$$

So $x + y < 16$, contradicting the fact that $x + y = 16$. We have reached a contradiction, so our assumption was wrong. Thus if $x + y = 16$, then $x \geq 8$ or $y \geq 8$. ■

Why This Matters

- Propositional logic is a tool for reasoning about how various statements affect one another.
- To better understand how to prove a result, it often helps to translate what you're trying to prove into propositional logic first.
- That said, propositional logic isn't expressive enough to capture all statements. For that, we need something more powerful.

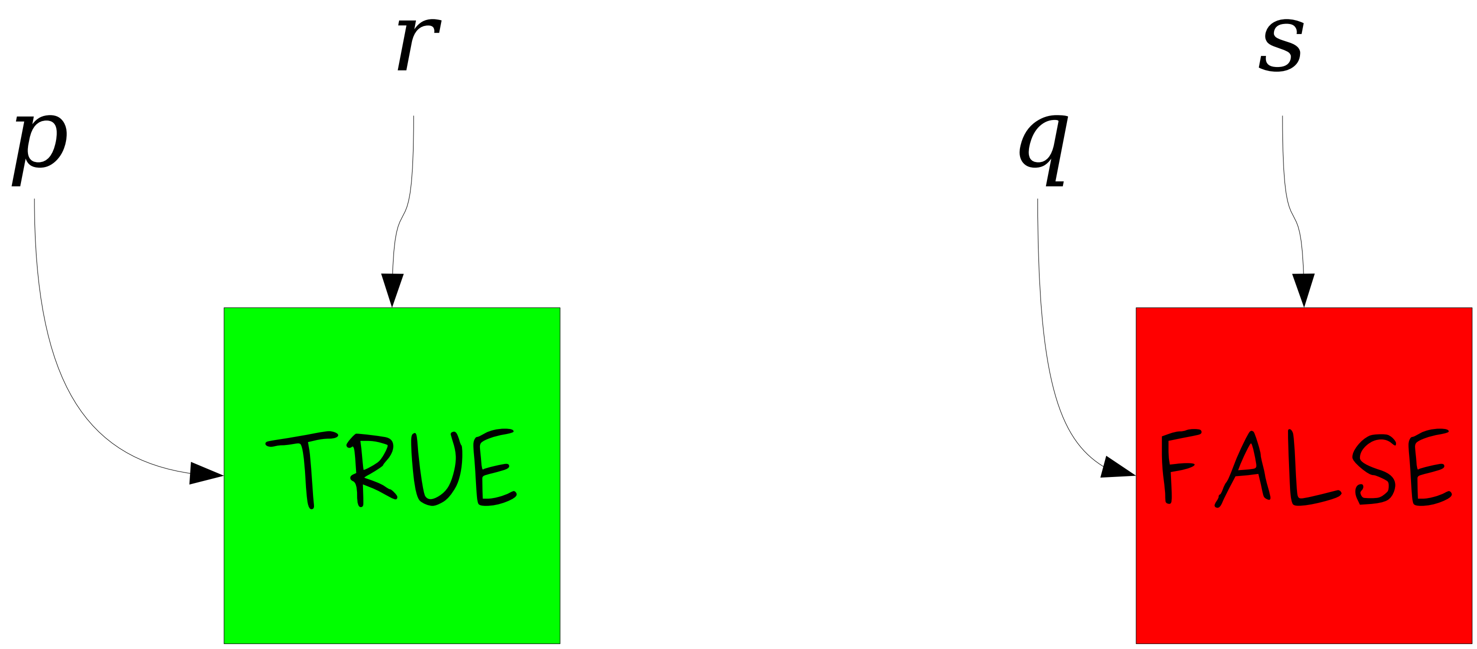
First-Order Logic

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, and
  - ***functions*** that map objects to one another,
  - ***quantifiers*** that allow us to reason about multiple objects simultaneously.

The Universe of Propositional Logic

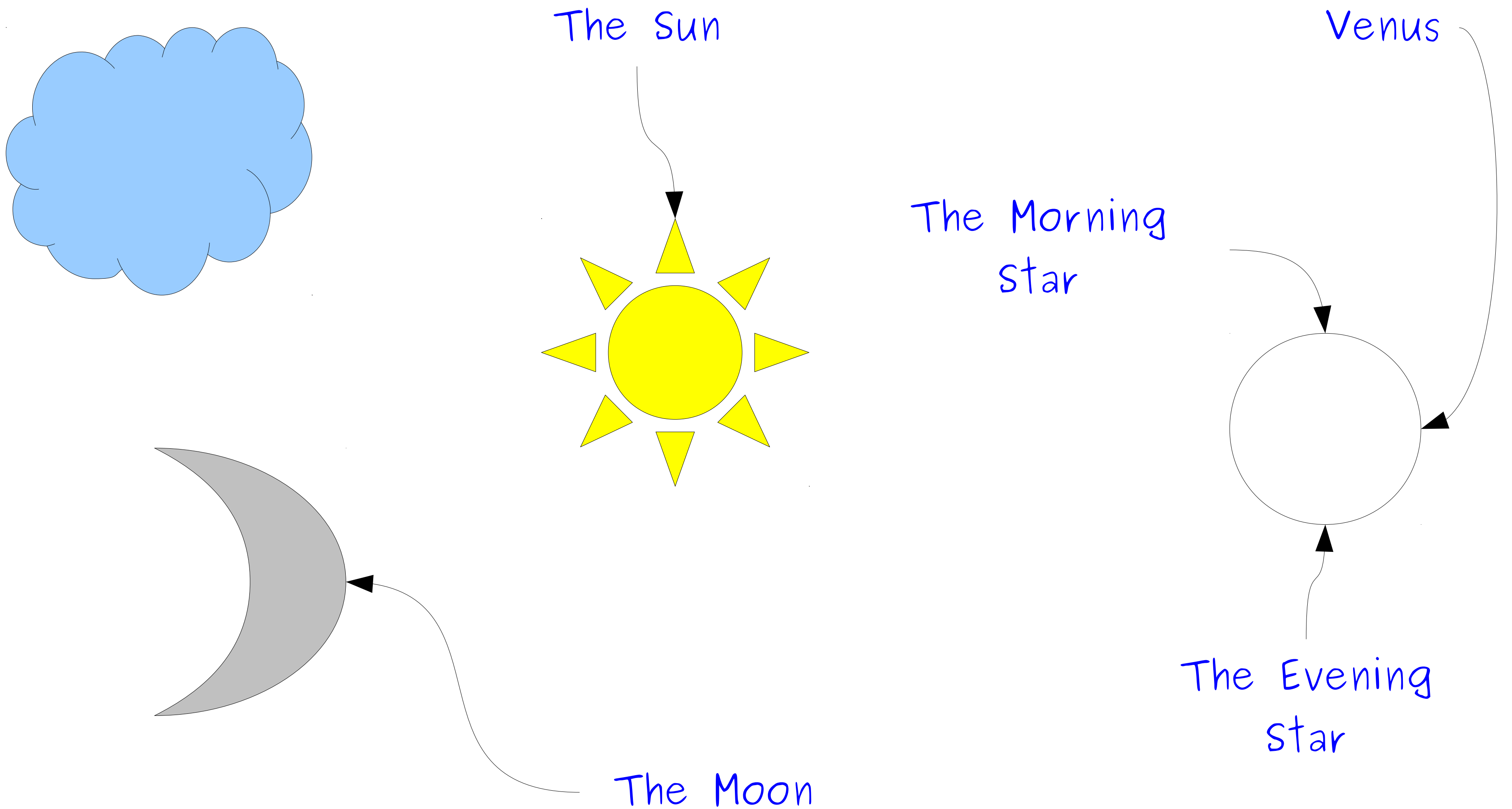
$$p \land q \to \neg r \lor \neg s$$



Propositional Logic

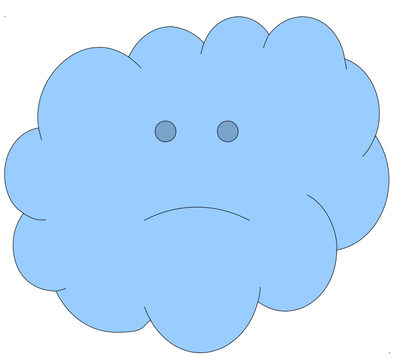
- In propositional logic, each variable represents a ***proposition***, which is either true or false.
- We can directly apply connectives to propositions:
  - $p \to q$
  - $\neg p \land q$
- The truth of a statement can be determined by plugging in the truth values for the input propositions and computing the result.
- We can see all possible truth values for a statement by checking all possible truth assignments to its variables.

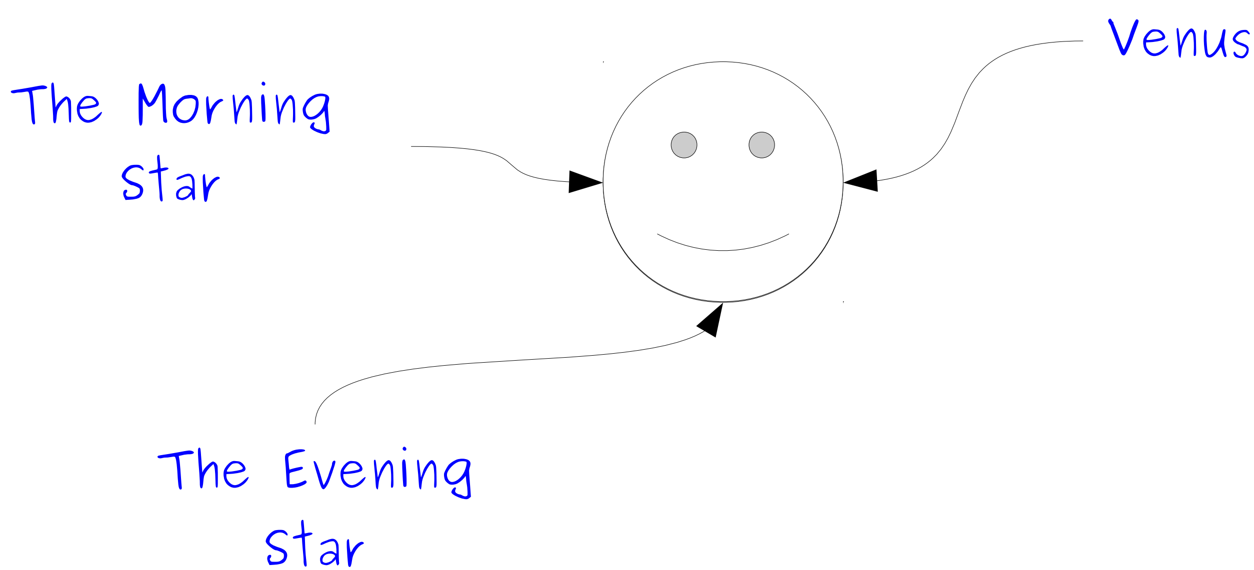
The Universe of First-Order Logic



First-Order Logic

- In first-order logic, each variable refers to some object in a set called the ***domain of discourse***.
- Some objects may have multiple names.
- Some objects may have no name at all.





Propositional vs. First-Order Logic

- Because propositional variables are either true or false, we can directly apply connectives to them.

$$p \to q \qquad\qquad \neg p \leftrightarrow q \land r$$

- Because first-order variables refer to arbitrary objects, it does not make sense to apply connectives to them.

$$\mathit{Venus} \to \mathit{Sun} \qquad\qquad 137 \leftrightarrow \neg 42$$

- ***This is not C!***

Reasoning about Objects

- To reason about objects, first-order logic uses ***predicates***.
- Examples:
  - *ExtremelyCute*(*Quokka*)
  - *DeadlockEachOther*(*House*, *Senate*)
- Predicates can take any number of arguments, but each predicate has a fixed number of arguments (called its ***arity***)
- Applying a predicate to arguments produces a proposition, which is either true or false.

First-Order Sentences

- Sentences in first-order logic can be constructed from predicates applied to objects:

$$LikesToEat(V, M) \land Near(V, M) \to WillEat(V, M)$$

$$Cute(t) \to Dikdik(t) \lor Kitty(t) \lor Puppy(t)$$

$$\boxed{x < 8 \to x < 137}$$

The notation $x < 8$ is just a shorthand for something like ***LessThan(x, 8)***. Binary predicates in math are often written like this, but symbols like $<$ are not a part of first-order logic.

Equality

- First-order logic is equipped with a special predicate **=** that says whether two objects are equal to one another.
- Equality is a part of first-order logic, just as → and ¬ are.
- Examples:

$$MorningStar = EveningStar$$

$$TomMarvoloRiddle = LordVoldemort$$

- Equality can only be applied to ***objects***; to see if ***propositions*** are equal, use ↔.

For notational simplicity, define $\neq$ as

$$x \neq y \quad \equiv \quad \neg(x = y)$$

Expanding First-Order Logic

$$(x < 8 \land y < 8) \;\rightarrow\; (\boxed{x + y} < 16)$$

Why is this allowed?

Functions

- First-order logic allows ***functions*** that return objects associated with other objects.
- Examples:

$$x + y$$

$$LengthOf(path)$$

$$MedianOf(x, y, z)$$

- As with predicates, functions can take in any number of arguments, but each function has a fixed arity.
- Functions evaluate to ***objects***, not ***propositions***.
- There is no syntactic way to distinguish functions and predicates; you'll have to look at how they're used.

How would we translate the statement

"For any natural number n,
n is even iff n^2 is even"

into first-order logic?

Quantifiers

- The biggest change from propositional logic to first-order logic is the use of ***quantifiers***.
- A ***quantifier*** is a statement that expresses that some property is true for some or all choices that could be made.
- Useful for statements like "for every action, there is an equal and opposite reaction."

"For any natural number n,
n is even iff n^2 is even"

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

∀ is the **universal quantifier**
and says "for any choice of n,
the following is true."

The Universal Quantifier

- A statement of the form $\boldsymbol{\forall x.\ \psi}$ asserts that for ***every*** choice of x in our domain, ψ is true.
- Examples:

$$\forall v.\ (Puppy(v) \to Cute(v))$$

$$\forall n.\ (n \in \mathbb{N} \to (Even(n) \leftrightarrow \neg Odd(n)))$$

$$Tallest(SK) \to$$
$$\quad \forall x.\ (SK \neq x \to ShorterThan(x, SK))$$

Some muggles are intelligent.

$$\boxed{\exists m.}\ (Muggle(m) \land Intelligent(m))$$

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

The Existential Quantifier

- A statement of the form **$\exists x.\ \psi$** asserts that for ***some*** choice of x in our domain, ψ is true.
- Examples:

$$\exists x.\ (Even(x) \land Prime(x))$$

$$\exists x.\ (TallerThan(x, me) \land LighterThan(x, me))$$

$$(\exists x.\ Appreciates(x, me)) \rightarrow Happy(me)$$

Operator Precedence (Again)

- When writing out a formula in first-order logic, the quantifiers ∀ and ∃ have precedence just below ¬.
- Thus

$$\forall x.\ P(x) \lor R(x) \to Q(x)$$

is interpreted as the (malformed) statement

$$((\forall \boldsymbol{x}.\ P(\boldsymbol{x})) \lor R(\boldsymbol{x})) \to Q(\boldsymbol{x})$$

rather than the (intended, valid) statement

$$\forall x.\ (P(\boldsymbol{x}) \lor R(\boldsymbol{x}) \to Q(\boldsymbol{x}))$$

Time-Out for Announcements!

Problem Set Three

- Problem Set Two due at the start of today's lecture.
  - Due on Monday with a late period.
- Problem Set Three goes out now.
  - Checkpoint problem due on Monday at the start of class.
  - Remaining problems due next Friday at the start of class.
  - Explore graph theory and logic!
- **A note:** We may not cover everything necessary for the last two problems on this problem set until Monday.

Back to CS103!

Translating into First-Order Logic

Translating Into Logic

- First-order logic is an excellent tool for manipulating definitions and theorems to learn more about them.
- Applications:
  - Determining the negation of a complex statement.
  - Figuring out the contrapositive of a tricky implication.

Translating Into Logic

- ***Translating statements into first-order logic is a lot more difficult than it looks.***
- There are a lot of nuances that come up when translating into first-order logic.
- We'll cover examples of both good and bad translations into logic so that you can learn what to watch for.
- We'll also show lots of examples of translations so that you can see the process that goes into it.

Some Incorrect Translations

An Incorrect Translation

All puppies are cute!

$$\forall x.\ (Puppy(x) \land Cute(x))$$

This should work for <u>any</u> choice of x, including things that aren't puppies.

An Incorrect Translation

All puppies are cute!

$\cancel{\forall x. (Puppy(x) \land Cute(x))}$

Although the original statement is true, this logical statement is false. It's therefore not a correct translation.

A Better Translation

All puppies are cute!

$$\forall x.\ (Puppy(x) \to Cute(x))$$

This should work for <u>any</u> choice of x, including things that aren't puppies.

“All *P*'s are *Q*'s”

translates as

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

Another Bad Translation

Some blobfish is cute.

$$\exists x.\ (Blobfish(x) \to Cute(x))$$

What happens if

1. The above statement is false, but
2. x refers to a cute puppy?

Another Bad Translation

Some blobfish is cute.

$$\exists x.\ (Blobfish(x) \to Cute(x))$$

Although the original statement is false, this logical statement is true. It's therefore not a correct translation.

A Better Translation

Some blobfish is cute.

$$\exists x.\ (Blobfish(x) \land Cute(x))$$

What happens if

1. The above statement is false, but
2. x refers to a cute puppy?

“Some *P* is a *Q*”

translates as

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

Good Pairings

- The $\forall$ quantifier *usually* is paired with $\rightarrow$.
- The $\exists$ quantifier *usually* is paired with $\land$.
- In the case of $\forall$, the $\rightarrow$ connective prevents the statement from being *false* when speaking about some object you don't care about.
- In the case of $\exists$, the $\land$ connective prevents the statement from being *true* when speaking about some object you don't care about.

Checking a Translation

There's a tall tree that's a sequoia.

$$\exists t.\ (Tree(t) \land (Tall(t) \to Sequoia(t)))$$

This statement can be true even if no tall sequoias exist.

Checking a Translation

There's a tall tree that's a sequoia.

$$\exists t.\ (Tree(t) \land Tall(t) \land Sequoia(t))$$

Do you see why this statement doesn't have this problem?

Checking a Translation

Every tall tree is a sequoia.

$$\forall t.\ ((Tree(t) \land Tall(t)) \to Sequoia(t))$$

What do you think?
Is this a faithful
translation?

Translating into Logic

- We've just covered the biggest common pitfall: using the wrong connectives with $\forall$ and $\exists$.
- Now that we've covered that, let's go and see how to translate more complex statements into first-order logic.

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.”

Everybody loves someone else

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

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 someone that everyone else loves.”

There is a person that everyone else loves

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

Combining Quantifiers

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

$$\forall p.\ (Person(p) \to \exists q.\ (Person(q) \land p \neq q \land Loves(p, q)))$$

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."

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

There is some person

who everyone

who isn't them

loves.

For Comparison

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

For every person,

there is some person

who isn't them

that they love.

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

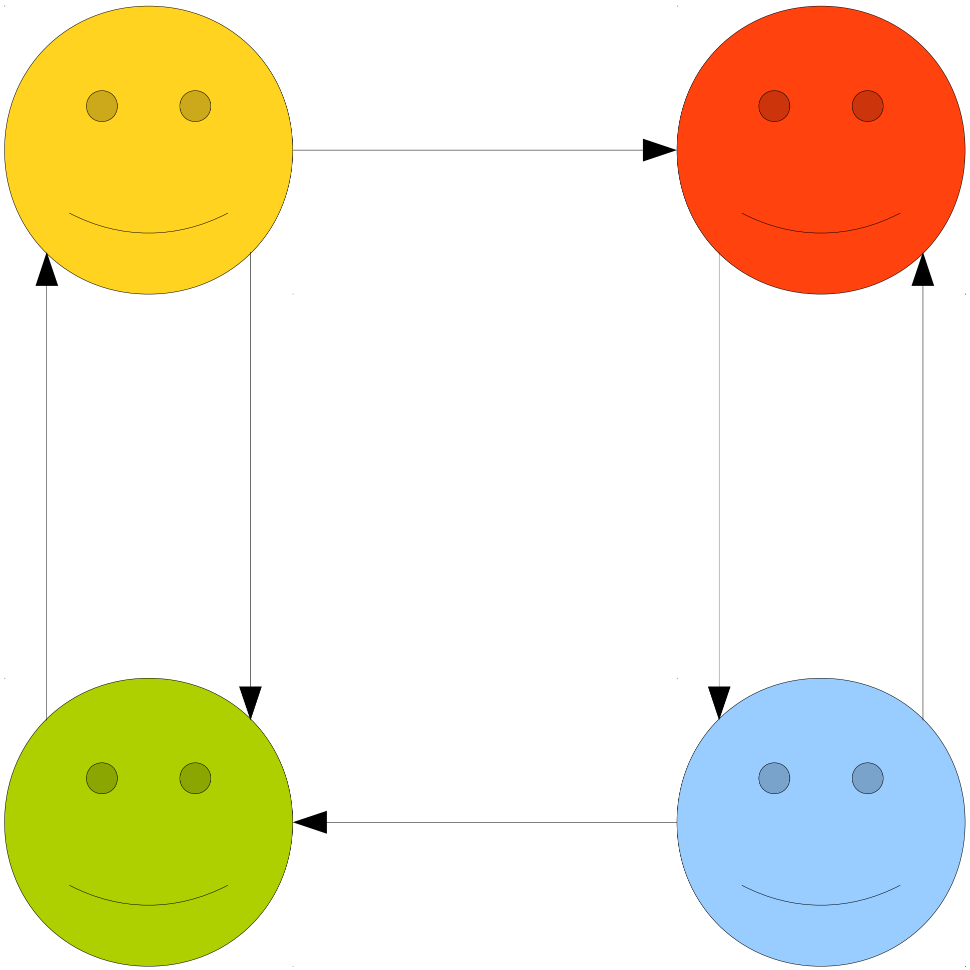
There is some person

who everyone

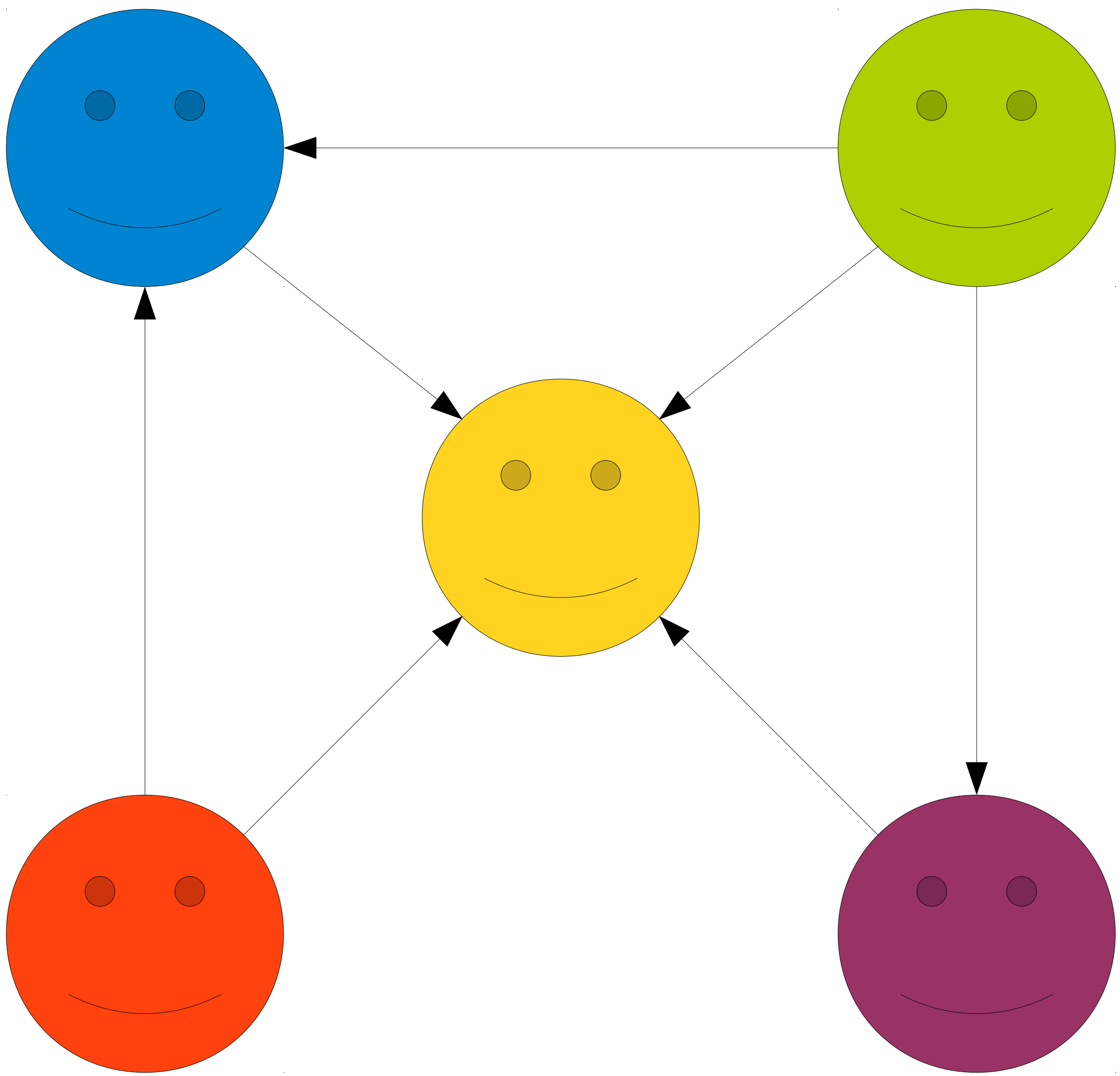
who isn't them

loves.

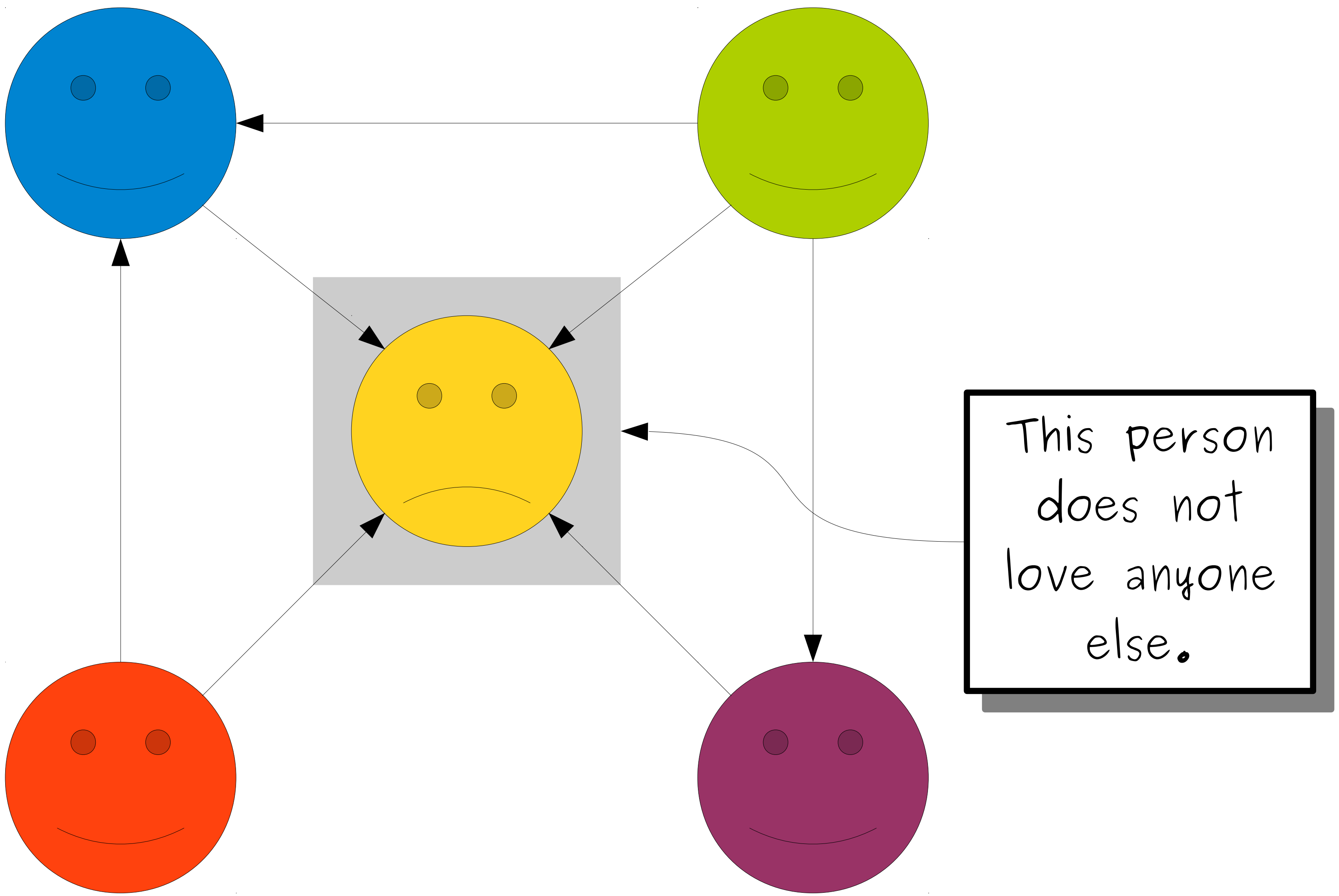
Everyone Loves Someone Else



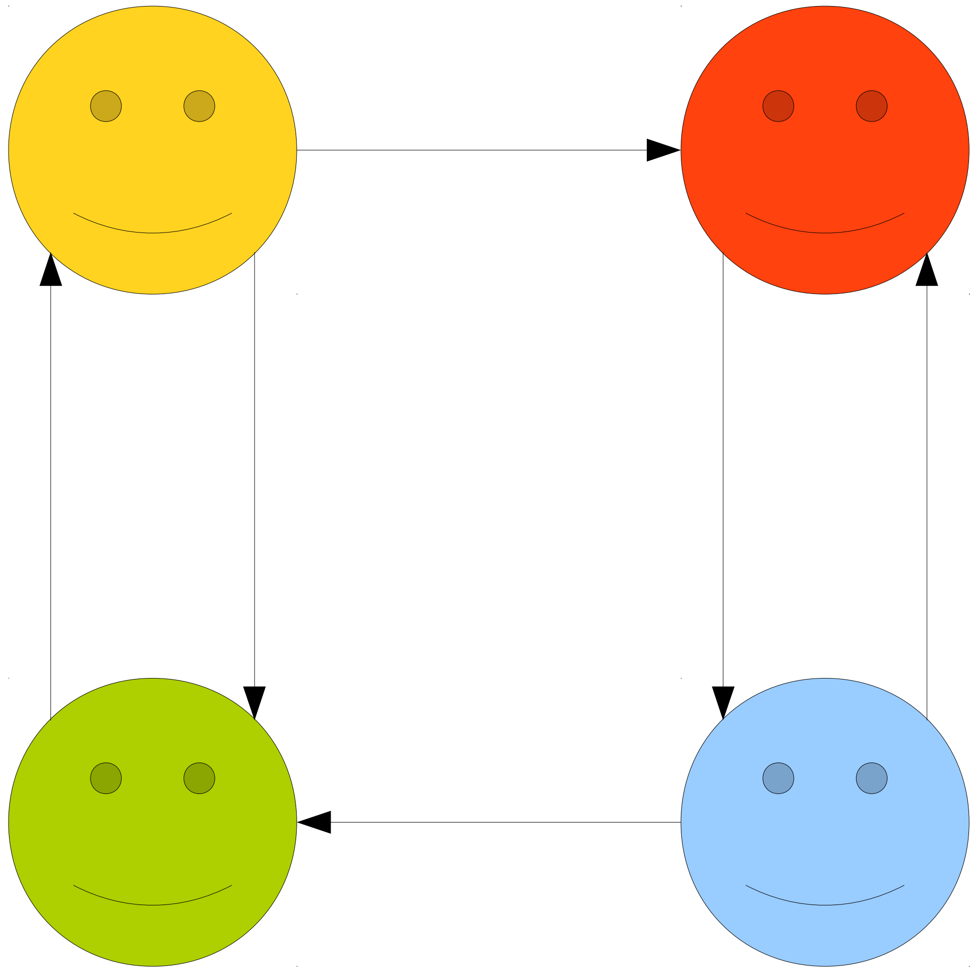
There is Someone Everyone Else Loves



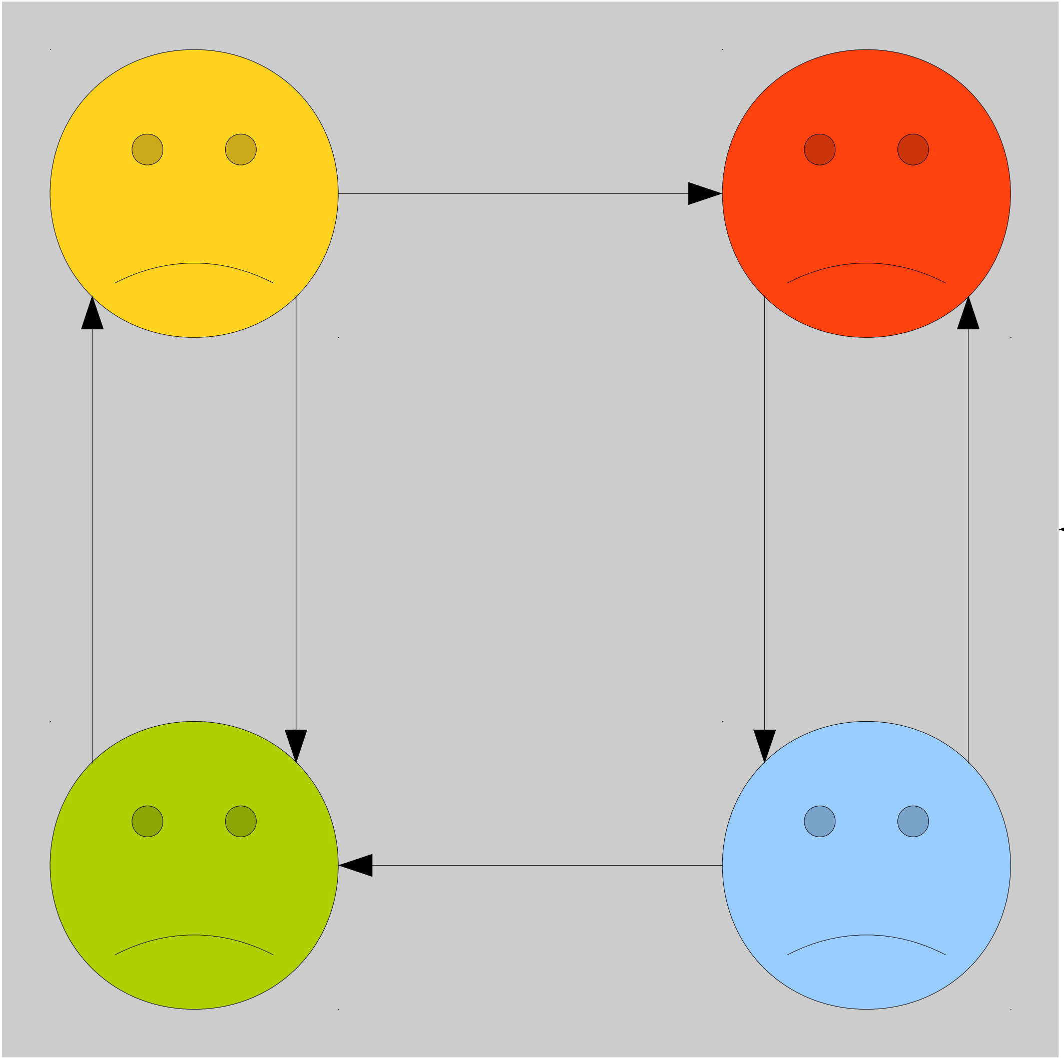
There is Someone Everyone Else Loves



Everyone Loves Someone Else

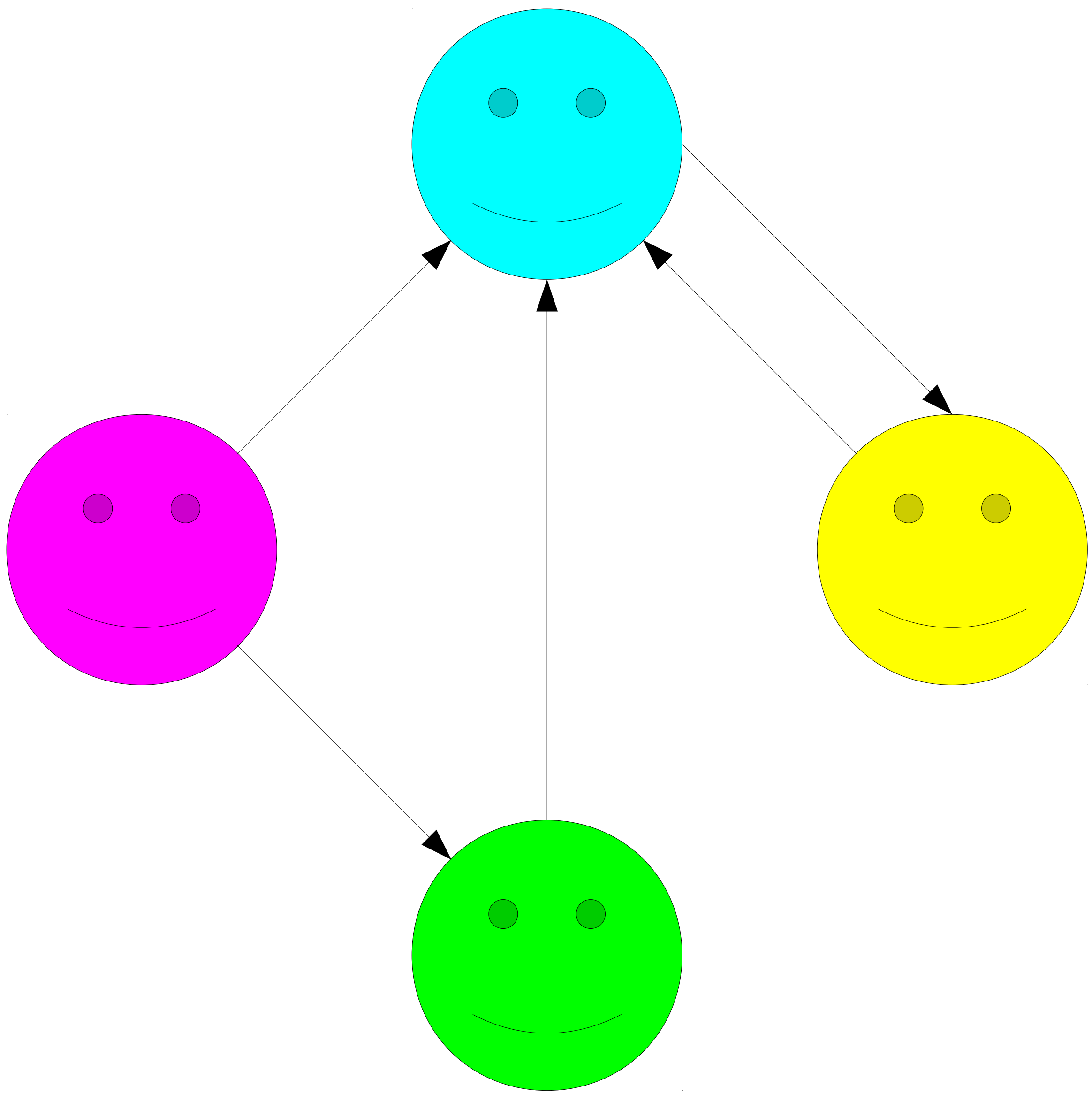


Everyone Loves Someone Else



No one here is universally loved.

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



$$\forall p.\ (Person(p) \to \exists q.\ (Person(q) \land p \neq q \land Loves(p, q)))$$

For every person,

there is some person

who isn't them

that they love.

$$\land$$

$$\exists p.\ (Person(p) \land \forall q.\ (Person(q) \land p \neq q \to Loves(q, p)))$$

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!