FIRST-ORDER LOGIC

Faculty of DS & AI
Autumn semester, 2026

Trong-Nghia Nguyen

Business AI Lab

2026-10

Content

- **Limitations of propositional logic**
- Syntax of FOL
- Quantifiers
- Representing knowledge in FOL
- Inference in FOL
- Examples and exercises
- Summary

Limitations of propositional logic

Why propositional logic is not enough

Example

1. Nam is a student of NEU.
2. All students of NEU study Data Science.
3. Hence, Nam studies Data Science.

Every human reader accepts (3) from (1) and (2).

In propositional logic each sentence is *one* symbol:

- P: "Nam is a student of NEU"
- Q: "All students of NEU study DS"
- R: "Nam studies DS"

$\{P, Q\} \not\models R$: the symbols hide the **objects** (Nam), the **properties** (is a student) and the word **"all"**.

Limitations of propositional logic

What first-order logic adds

- PL work-around: one symbol per student
  - $S_{Nam} \Rightarrow D_{Nam},\ S_{Mai} \Rightarrow D_{Mai},\ \ldots$ — one rule per student, a new rule for every new student
- First-order logic (FOL, predicate logic) describes the world with
  - **objects**: Nam, Mai, NEU, $\ldots$
  - **properties** and **relations**: $Student(x)$, $StudiesAt(x, y)$
  - **functions**: $father(x)$
  - **quantifiers**: $\forall$ (for all), $\exists$ (there exists)

The example in FOL

$$Student(Nam),\ \ \forall x\,(Student(x) \Rightarrow StudiesDS(x)) \ \models\ StudiesDS(Nam)$$

One general rule covers every student. FOL extends PL: every PL sentence is also an FOL sentence.

Content

- Limitations of propositional logic
- **Syntax of FOL**
- Quantifiers
- Representing knowledge in FOL
- Inference in FOL
- Examples and exercises
- Summary

Syntax of FOL

Basic elements

Element	Meaning	Examples
Constant	names one specific object	$Nam,\ Mai,\ NEU$
Variable	stands for any object	$x,\ y,\ z$
Function	maps objects to *one object*	$father(x)$: the father of x
Predicate	says something about objects: **true** / **false**	$Student(x),\ Friend(x, y)$
Connectives	as in propositional logic	$\neg,\ \wedge,\ \vee,\ \Rightarrow,\ \Leftrightarrow$
Quantifiers	for all / there exists	$\forall,\ \exists$

Naming convention in this course

Constants and predicates start with a **capital** letter (Nam, $Student$); variables and functions are **lowercase** (x, $father$).

Syntax of FOL

Terms and sentences

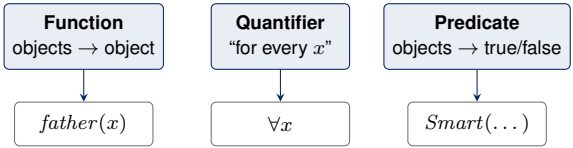
- **Term** = a **noun**: names an object, it is **not** true or false (“Nam”, “the father of Mai”)
- **Atomic sentence** = a **statement**: a predicate applied to terms, it **is** true or false (“Nam is a student”)

Expression	Kind	Read as	Result
Nam	term	“Nam”	a person
$father(Mai)$	term	“the father of Mai”	a person
$Student(Nam)$	sentence	“Nam is a student”	true / false
$Friend(Nam, father(Mai))$	sentence	“Nam is a friend of Mai’s father”	true / false
$father(Nam) = Hung$	sentence	“Nam’s father is Hung”	true / false

- To evaluate $Friend(Nam, father(Mai))$: first find *who* $father(Mai)$ is, then check the friendship
- “$=$” is a statement (“the same person”), **not** an assignment; $Hard(Student(Nam))$ **is not allowed**

Syntax of FOL

Reading an FOL sentence



$$\forall x\,(Student(x) \Rightarrow Smart(father(x)))$$

Read from the inside out:

1. $father(x)$ — the father of x (a person)
2. $Smart(father(x))$ — "x's father is smart"
3. $Student(x) \Rightarrow \dots$ — "if x is a student, $\dots$"
4. $\forall x$ — for **every** x

"For every x: if x is a student, then the father of x is smart."

- Precedence as in PL: $\neg > \wedge > \vee > \Rightarrow > \Leftrightarrow$
- **Convention**: always write parentheses after a quantifier — they show what it covers: $\forall x\,(\dots)$

Content

- Limitations of propositional logic
- Syntax of FOL
- **Quantifiers**
- Representing knowledge in FOL
- Inference in FOL
- Examples and exercises
- Summary

Quantifiers

Universal quantifier $\forall$ (“for all”)

- $\forall x\ P$ is true **iff** P is true for **every** object
- In a small world, $\forall$ is a big **AND**:

$$\begin{aligned} & \forall x\,(Student(x) \Rightarrow Hard(x)) \\ \equiv\ & (Student(An) \Rightarrow Hard(An)) \\ \wedge\ & (Student(Binh) \Rightarrow Hard(Binh)) \\ \wedge\ & (Student(Chi) \Rightarrow Hard(Chi)) \end{aligned}$$

Our small world

Person	Student	Hard
An	✓	✓
Binh	✓	×
Chi	×	×

- One **counter-example** (Binh) is enough to make a $\forall$-sentence false
- “All FDA students are hard-working”: $\forall x\,(FDAStudent(x) \Rightarrow Hard(x))$

Quantifiers

Universal quantifier $\forall$ ("for all")

- $\forall x\, P$ is true **iff** P is true for **every** object
- In a small world, $\forall$ is a big **AND**:

$$\begin{aligned} & \forall x\,(Student(x) \Rightarrow Hard(x)) \\ \equiv\ & (Student(An) \Rightarrow Hard(An)) \\ \wedge\ & (Student(Binh) \Rightarrow Hard(Binh)) \\ \wedge\ & (Student(Chi) \Rightarrow Hard(Chi)) \end{aligned}$$

$$= \textbf{true} \wedge \textbf{false} \wedge \textbf{true} = \textbf{false}$$ (because of Binh)

Our small world

Person	Student	Hard
An	✓	✓
Binh	✓	×
Chi	×	×

- One **counter-example** (Binh) is enough to make a $\forall$-sentence false
- "All FDA students are hard-working": $\forall x\,(FDAStudent(x) \Rightarrow Hard(x))$

Quantifiers

Existential quantifier $\exists$ ("there exists")

- $\exists x\ P$ is true **iff** P is true for **at least one** object
- In a small world, $\exists$ is a big **OR**:

$$\begin{aligned} & \exists x\,(Student(x) \wedge Hard(x)) \\ \equiv\ & (Student(An) \wedge Hard(An)) \\ \vee\ & (Student(Binh) \wedge Hard(Binh)) \\ \vee\ & (Student(Chi) \wedge Hard(Chi)) \end{aligned}$$

Our small world

Person	Student	Hard
An	✓	✓
Binh	✓	×
Chi	×	×

- One **witness** (An) is enough to make an $\exists$-sentence true
- "An has a friend": $\exists x\ Friend(An, x)$

Quantifiers

Existential quantifier $\exists$ ("there exists")

- $\exists x\ P$ is true **iff** P is true for **at least one** object
- In a small world, $\exists$ is a big **OR**:

$$\begin{aligned} & \exists x\,(Student(x) \wedge Hard(x)) \\ \equiv\ & (Student(An) \wedge Hard(An)) \\ \vee\ & (Student(Binh) \wedge Hard(Binh)) \\ \vee\ & (Student(Chi) \wedge Hard(Chi)) \end{aligned}$$

$$= \textbf{true} \vee \textbf{false} \vee \textbf{false} = \textbf{true} \quad \text{(because of An)}$$

- One **witness** (An) is enough to make an $\exists$-sentence true
- "An has a friend": $\exists x\ Friend(An, x)$

Our small world

Person	Student	Hard
An	✓	✓
Binh	✓	×
Chi	×	×

Quantifiers

$\forall$ goes with $\Rightarrow$, $\exists$ goes with $\wedge$

Sentence	What it really says	Value
$\forall x\,(Student(x) \Rightarrow Hard(x))$	every student is hard-working ✓	**false** (Binh)

Small world: Student = {An, Binh}, Hard = {An}.

Quantifiers

$\forall$ goes with $\Rightarrow$, $\exists$ goes with $\wedge$

Sentence	What it really says	Value
$\forall x\,(Student(x) \Rightarrow Hard(x))$	every student is hard-working ✓	**false** (Binh)
$\forall x\,(Student(x) \wedge Hard(x))$	**everyone** is a hard-working student — too strong	**false** (Chi)

Small world: Student = {An, Binh}, Hard = {An}.

Quantifiers

$\forall$ goes with $\Rightarrow$, $\exists$ goes with $\wedge$

Sentence	What it really says	Value
$\forall x\,(Student(x) \Rightarrow Hard(x))$	every student is hard-working ✓	**false** (Binh)
$\forall x\,(Student(x) \wedge Hard(x))$	**everyone** is a hard-working student — too strong	**false** (Chi)
$\exists x\,(Student(x) \wedge Hard(x))$	some student is hard-working ✓	**true** (An)

Small world: Student = {An, Binh}, Hard = {An}.

Quantifiers

$\forall$ goes with $\Rightarrow$, $\exists$ goes with $\wedge$

Sentence	What it really says	Value
$\forall x\,(Student(x) \Rightarrow Hard(x))$	every student is hard-working ✓	**false** (Binh)
$\forall x\,(Student(x) \wedge Hard(x))$	**everyone** is a hard-working student — too strong	**false** (Chi)
$\exists x\,(Student(x) \wedge Hard(x))$	some student is hard-working ✓	**true** (An)
$\exists x\,(Student(x) \Rightarrow Hard(x))$	someone is **not a student** or is hard-working — too weak	**true** (Chi)

Small world: Student = {An, Binh}, Hard = {An}.

Why $\exists x\,(A(x) \Rightarrow B(x))$ is almost always wrong

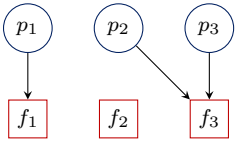
It is true as soon as **one** object is not an A: Chi is not a student, so the sentence is true even if nobody were hard-working.

Quantifiers

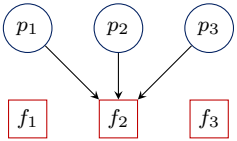
Order of quantifiers

- Same quantifiers can be swapped: $\forall x \forall y \equiv \forall y \forall x, \quad \exists x \exists y \equiv \exists y \exists x$
- **Mixed quantifiers cannot**. $Knows(x, y)$: person x knows field y

$\forall x \exists y \, Knows(x, y)$
every person knows **some** field
(possibly a different one)



$\exists y \forall x \, Knows(x, y)$
there is **one** field
that every person knows



- The right picture also satisfies the left sentence, but **not** the other way round

Content

- Limitations of propositional logic
- Syntax of FOL
- Quantifiers
- **Representing knowledge in FOL**
- Inference in FOL
- Examples and exercises
- Summary

Representing knowledge in FOL

Translation patterns

English	FOL	Equivalent form
All A are B	$\forall x\,(A(x) \Rightarrow B(x))$	
Some A are B	$\exists x\,(A(x) \wedge B(x))$	
No A is B	$\forall x\,(A(x) \Rightarrow \neg B(x))$	$\neg\exists x\,(A(x) \wedge B(x))$
Not all A are B	$\neg\forall x\,(A(x) \Rightarrow B(x))$	$\exists x\,(A(x) \wedge \neg B(x))$
Only A are B	$\forall x\,(B(x) \Rightarrow A(x))$	

- A, B stand for predicates, e.g. $A = Student$, $B = Hard$
- "Only students are hard-working" = "if someone is hard-working, they are a student" (arrow reversed)

Representing knowledge in FOL

Translating sentences

- **Method**: (1) choose the symbols (2) pick the pattern for each quantifier (3) read the formula back

1. “All pet owners love animals.”
2. “There is a student who studies **all** AI subjects.”
3. “Every student has a hard-working friend.”
4. “Not every student is hard-working.”

Representing knowledge in FOL

Translating sentences

- **Method**: (1) choose the symbols (2) pick the pattern for each quantifier (3) read the formula back

1. "All pet owners love animals."
   $\forall x \, \forall y \, ((Owns(x, y) \land Animal(y)) \Rightarrow AnimalLover(x))$
2. "There is a student who studies **all** AI subjects."
3. "Every student has a hard-working friend."
4. "Not every student is hard-working."

Representing knowledge in FOL

Translating sentences

- **Method**: (1) choose the symbols (2) pick the pattern for each quantifier (3) read the formula back

1. "All pet owners love animals."
   $\forall x \, \forall y \, ((Owns(x, y) \land Animal(y)) \Rightarrow AnimalLover(x))$
2. "There is a student who studies **all** AI subjects."
   $\exists x \, (Student(x) \land \forall y \, (AISubject(y) \Rightarrow Studies(x, y)))$
3. "Every student has a hard-working friend."

4. "Not every student is hard-working."

Each quantifier follows its own pattern, also when it is nested: "all AI subjects" $\rightarrow \forall y \, (\ldots \Rightarrow \ldots)$, "a hard-working friend" $\rightarrow \exists y \, (\cdots \land \ldots)$.

Representing knowledge in FOL

Translating sentences

- **Method**: (1) choose the symbols (2) pick the pattern for each quantifier (3) read the formula back

1. "All pet owners love animals."
   $\forall x \, \forall y \, ((Owns(x, y) \land Animal(y)) \Rightarrow AnimalLover(x))$
2. "There is a student who studies **all** AI subjects."
   $\exists x \, (Student(x) \land \forall y \, (AISubject(y) \Rightarrow Studies(x, y)))$
3. "Every student has a hard-working friend."
   $\forall x \, (Student(x) \Rightarrow \exists y \, (Friend(x, y) \land Hard(y)))$
4. "Not every student is hard-working."

Each quantifier follows its own pattern, also when it is nested: "all AI subjects" $\rightarrow \forall y \, (\ldots \Rightarrow \ldots)$, "a hard-working friend" $\rightarrow \exists y \, (\cdots \land \ldots)$.

Representing knowledge in FOL

Translating sentences

- **Method**: (1) choose the symbols (2) pick the pattern for each quantifier (3) read the formula back

1. "All pet owners love animals."
   $\forall x \, \forall y \, ((Owns(x, y) \land Animal(y)) \Rightarrow AnimalLover(x))$
2. "There is a student who studies **all** AI subjects."
   $\exists x \, (Student(x) \land \forall y \, (AISubject(y) \Rightarrow Studies(x, y)))$
3. "Every student has a hard-working friend."
   $\forall x \, (Student(x) \Rightarrow \exists y \, (Friend(x, y) \land Hard(y)))$
4. "Not every student is hard-working."
   $\neg \forall x \, (Student(x) \Rightarrow Hard(x)) \;\equiv\; \exists x \, (Student(x) \land \neg Hard(x))$

Each quantifier follows its own pattern, also when it is nested: "all AI subjects" $\rightarrow \forall y \, (\ldots \Rightarrow \ldots)$, "a hard-working friend" $\rightarrow \exists y \, (\cdots \land \ldots)$.

Representing knowledge in FOL

Common mistakes

English	Wrong	Right
Some FDA student is hard-working	$\exists x\,(FDA(x) \Rightarrow Hard(x))$ true if anyone is not an FDA student	$\exists x\,(FDA(x) \wedge Hard(x))$
All FDA students are hard-working	$\forall x\,(FDA(x) \wedge Hard(x))$ says everyone is an FDA student	$\forall x\,(FDA(x) \Rightarrow Hard(x))$
A student who studies all AI subjects	$\exists x\,\forall y\,(Student(x) \wedge Studies(x,y) \wedge AISubject(y))$ says every object is an AI subject	$\exists x\,(Student(x) \wedge \forall y\,(AISubject(y) \Rightarrow Studies(x,y)))$
If it rains, every road is wet	$\forall x\,(Rain(x) \Rightarrow Wet(x))$ rain is not a property of a road	$Rain \Rightarrow \forall x\,(Road(x) \Rightarrow Wet(x))$

Self-check: read your formula back word by word — if it sounds strange, it is wrong.

Representing knowledge in FOL

Negating quantifiers

- When $\neg$ passes a quantifier, the quantifier **flips**:

$$\neg\forall x\, P \;\equiv\; \exists x\, \neg P \qquad\qquad \neg\exists x\, P \;\equiv\; \forall x\, \neg P$$

- Examples
  - $\neg\forall x\, Friend(An, x) \equiv \exists x\, \neg Friend(An, x)$
    "An is not a friend of everyone" = "there is someone An is not friends with"
  - $\neg\exists x\, Friend(An, x) \equiv \forall x\, \neg Friend(An, x)$
    "An has no friend" = "nobody is An's friend"
- Negating a pattern (used in the next section):

$$\neg\forall x\, (A(x) \Rightarrow B(x)) \;\equiv\; \exists x\, \neg(\neg A(x) \vee B(x)) \;\equiv\; \exists x\, (A(x) \wedge \neg B(x))$$

Content

- Limitations of propositional logic
- Syntax of FOL
- Quantifiers
- Representing knowledge in FOL
- **Inference in FOL**
- Examples and exercises
- Summary

Inference in FOL

Resolution in FOL

Same idea as Robinson's method in PL

$$KB \models \alpha \text{ iff } KB \wedge \neg\alpha \text{ is unsatisfiable}$$

1. Add $\neg\alpha$ to the KB
2. Convert every sentence to CNF; split into clauses
3. Resolve pairs of clauses until $\perp$

Choose clauses with **unit preference** (short clauses first) and **set of support** (always use $\neg\alpha$ or a clause derived from it).

Only two new things

1. **CNF must remove the quantifiers**
2. **Unification**: $Student(x)$ and $\neg Student(Nam)$ become complementary after replacing x by Nam

Truth tables cannot be used: the world may contain infinitely many objects.

Inference in FOL

Converting FOL to CNF

Step	Action	Rule
1	Eliminate $\Leftrightarrow$	$\alpha \Leftrightarrow \beta \equiv (\alpha \Rightarrow \beta) \wedge (\beta \Rightarrow \alpha)$
2	Eliminate $\Rightarrow$	$\alpha \Rightarrow \beta \equiv \neg\alpha \vee \beta$
3	Move $\neg$ inwards	De Morgan; $\neg\neg\alpha \equiv \alpha$; $\neg\forall \rightarrow \exists\neg$, $\neg\exists \rightarrow \forall\neg$
4	**Remove the quantifiers**	$\exists$: give the object a **new name**; then drop every $\forall$
5	Distribute $\vee$ over $\wedge$	$\alpha \vee (\beta \wedge \gamma) \equiv (\alpha \vee \beta) \wedge (\alpha \vee \gamma)$; split into clauses

- Steps 1, 2, 3, 5 are the **four PL steps**; only step 4 is new
- Do step 3 **before** step 4: a $\neg$ in front of a quantifier changes $\forall$ into $\exists$ and vice versa

Inference in FOL

Step 4: removing the quantifiers

- $\exists$ **— give the unknown object a new name** (a *Skolem constant*)
  - $\exists x$ says "someone exists" without saying who — like the police writing "a suspect, call him Mr. X"
  - $\exists x\,(Student(x) \wedge Hard(x)) \;\rightsquigarrow\; Student(A) \wedge Hard(A)$
  - The name must be **new**: writing Nam instead of A would claim that *Nam* is that student
- $\forall$ **— simply drop it**: every variable left in a clause means "for all"
  - $\forall x\,(\neg Deceiver(x) \vee Cheater(x)) \;\rightsquigarrow\; \neg Deceiver(x) \vee Cheater(x)$

Scope of this course

We only meet $\exists$ that is **not inside** a $\forall$, so a new constant is enough. ("Everyone has a mother", $\forall y\,\exists x\,Mother(x, y)$, would need a name that depends on y — not covered.)

If two quantifiers use the same variable name, rename one of them before step 4.

Inference in FOL

Unification and the resolution rule

- **Unify** two literals = find a substitution θ that makes them identical; $\{x/Mai\}$ means "replace x by Mai"

p	q	θ
$Knows(Nam, x)$	$Knows(Nam, Mai)$	$\{x/Mai\}$
$Knows(Nam, x)$	$Knows(y, Lan)$	$\{x/Lan,\ y/Nam\}$
$Knows(Nam, Mai)$	$Knows(Lan, x)$	**fail**: constants $Nam \neq Lan$ cannot be changed

Resolution rule in FOL

Two clauses with literals that become complementary after θ: remove the pair, join the rest, and **apply** θ **to everything that is left**.

$$\frac{\neg Student(x) \vee Smart(father(x)) \qquad Student(Nam)}{Smart(father(Nam))} \quad \theta = \{x/Nam\}$$

Content

- Limitations of propositional logic
- Syntax of FOL
- Quantifiers
- Representing knowledge in FOL
- Inference in FOL
- **Examples and exercises**
- Summary

Examples and exercises

Proving in FOL

Example 1 — problem and representation

Knowledge base

1. Anyone who deceives others is a cheater.
2. Anyone who agrees with a cheater is also a cheater.
3. Some timid person agrees with someone who deceives others.

Conclusion: not every cheater is non-timid.

Predicates: $Deceiver(x)$, $Cheater(x)$, $Timid(x)$, $Agree(x, y)$: x agrees with y.

In FOL

1. $\forall x\,(Deceiver(x) \Rightarrow Cheater(x))$
2. $\forall x\,\forall y\,((Agree(x, y) \wedge Cheater(y)) \Rightarrow Cheater(x))$
3. $\exists x\,(Timid(x) \wedge \exists y\,(Agree(x, y) \wedge Deceiver(y)))$

α: $\neg\forall x\,(Cheater(x) \Rightarrow \neg Timid(x))$

($\equiv \exists x\,(Cheater(x) \wedge Timid(x))$: some cheater is timid)

Examples and exercises

Proving in FOL

Example 1 — clauses

C_1: $\neg Deceiver(x) \vee Cheater(x)$

C_2: $\neg Agree(x, y) \vee \neg Cheater(y) \vee Cheater(x)$

C_3: $Timid(A)$

C_4: $Agree(A, B)$

C_5: $Deceiver(B)$

C_6: $\neg Cheater(x) \vee \neg Timid(x)$ $(\neg\alpha)$

- C_2: $(P \wedge Q) \Rightarrow R \equiv \neg P \vee \neg Q \vee R$
- C_3–C_5: two unknown people $\Rightarrow$ new names: the timid person A, the deceiver B; the $\wedge$ gives 3 clauses
- C_6: $\neg\alpha = \neg\neg\forall x\,(\dots) = \forall x\,(Cheater(x) \Rightarrow \neg Timid(x))$
- Each clause has its own variables: the x of C_1 and the x of C_6 are unrelated

Examples and exercises

Proving in FOL

Example 1 — resolution

Step	Clauses	θ	Resolvent
1	C_5 and C_1	$\{x/B\}$	$C_7 : \ Cheater(B)$

Examples and exercises

Proving in FOL

Example 1 — resolution

Step	Clauses	θ	Resolvent
1	C_5 and C_1	$\{x/B\}$	$C_7: \ Cheater(B)$
2	C_7 and C_2	$\{y/B\}$	$C_8: \ \neg Agree(x, B) \vee Cheater(x)$

Examples and exercises

Proving in FOL

Example 1 — resolution

Step	Clauses	θ	Resolvent
1	C_5 and C_1	$\{x/B\}$	$C_7: \ Cheater(B)$
2	C_7 and C_2	$\{y/B\}$	$C_8: \ \neg Agree(x, B) \vee Cheater(x)$
3	C_8 and C_4	$\{x/A\}$	$C_9: \ Cheater(A)$

Examples and exercises

Proving in FOL

Example 1 — resolution

Step	Clauses	θ	Resolvent
1	C_5 and C_1	$\{x/B\}$	$C_7: \ Cheater(B)$
2	C_7 and C_2	$\{y/B\}$	$C_8: \ \neg Agree(x, B) \vee Cheater(x)$
3	C_8 and C_4	$\{x/A\}$	$C_9: \ Cheater(A)$
4	C_9 and C_6	$\{x/A\}$	$C_{10}: \ \neg Timid(A)$

Examples and exercises

Proving in FOL

Example 1 — resolution

Step	Clauses	θ	Resolvent
1	C_5 and C_1	$\{x/B\}$	$C_7: \ Cheater(B)$
2	C_7 and C_2	$\{y/B\}$	$C_8: \ \neg Agree(x, B) \vee Cheater(x)$
3	C_8 and C_4	$\{x/A\}$	$C_9: \ Cheater(A)$
4	C_9 and C_6	$\{x/A\}$	$C_{10}: \ \neg Timid(A)$
5	C_{10} and C_3	$\{\}$	$\perp$

Conclusion

$KB \wedge \neg\alpha$ is unsatisfiable, so $KB \models \alpha$. In words: B deceives, so B is a cheater; A agrees with B, so A is a cheater — and A is timid.

Examples and exercises

Exercises

Exercise 1

Knowledge base

1. Hung likes all kinds of food.
2. Apples are food.
3. Chicken is food.
4. Anything that anyone eats and is not killed by is food.
5. Phong eats peanuts and is still alive.
6. Lan eats everything that Phong eats.
7. Anyone who is alive has not been killed.

Prove: (a) Hung likes peanuts; (b) Lan eats peanuts.

Predicates

- $Food(x)$, $Likes(x, y)$
- $Eats(x, y)$: x eats y
- $Killed(x)$, $Alive(x)$

Think about sentence 7

Why is it needed? Try the proof without it.

Examples and exercises

Exercises

Exercise 2 and Exercise 3

Exercise 2

1. All dogs bark at night.
2. Anyone who has a cat has no mice in their house.
3. Anyone who has trouble sleeping does not keep any animal that barks at night.
4. Mrs. Binh has either a cat or a dog.

Prove: if Mrs. Binh has trouble sleeping, then there are no mice in her house.

Predicates: $Dog(x)$, $Cat(x)$, $BarksAtNight(x)$, $Has(x, y)$, $HasMice(x)$, $LightSleeper(x)$.

Exercise 3

1. Minh owns a dog.
2. Everyone who owns a dog is an animal lover.
3. Animal lovers do not kill animals.

Prove: Minh does not kill any animal.

Predicates: $Dog(x)$, $Owns(x, y)$, $AnimalLover(x)$, $Animal(x)$, $Kills(x, y)$.

Hint: negate the goal — which quantifier do you get?

Tasks for every exercise: (a) translate into FOL; (b) convert to clauses; (c) prove by resolution.

Content

- Limitations of propositional logic
- Syntax of FOL
- Quantifiers
- Representing knowledge in FOL
- Inference in FOL
- Examples and exercises
- **Summary**

Summary

PL vs. FOL and the key ideas

	Propositional logic	First-order logic
The world contains	facts (true / false)	objects, their properties and relations
Basic unit	symbol P	atomic sentence $Student(Nam)$
Quantifiers	no	$\forall$ (with $\Rightarrow$), $\exists$ (with $\wedge$)
CNF	4 steps	4 PL steps + remove the quantifiers
Resolution	complementary literals	complementary *after unification*

- **Terms** name objects, **sentences** are true or false
- $\forall$ **with** $\Rightarrow$, $\exists$ **with** $\wedge$; $\neg\forall \equiv \exists\neg$, $\neg\exists \equiv \forall\neg$
- Resolution in FOL = Robinson's method + **new names for** $\exists$ + **unification**

Thank you!

You’re now ready to explore the exciting world of AI!