Predicate Logic

Zhaoguo Wang

Adapted From:

教材《数理逻辑与集合论》第4章、第5章

Predicate Logic（谓词逻辑）

1.1 Propositional Logic

Step 0. Proof.

Step 1. Convert program into mathematical formula.

Step 2. Ask the computer to solve the formula.

1.2 First Order Logic

Step 1. Convert it into first order logic formula.

Step 2. Ask the computer to solve the formula.

1.3 Auto-active Proof

Step 1. Axiom system

Step 2. Ask the computer to check the invariants

Propositional Logic

```
void f(bool a, bool b)
{
    unsigned x, y;
    if (a)
        x = 1;
    else
        x = 0;
    if (b)
        y = 1;
    else
        y = 0;
    assert(x + y > 0);
}
```

This is complicated.

$$formula\{X1 = [00...01]\}\wedge$$
$$formula\{X2 = [00...00]\}\wedge$$
$$formula\{Y4 = [00...01]\}\wedge$$
$$formula\{Y5 = [00...00]\}\wedge$$
$$\wedge_{i=0,\ldots,31}\ (X3_i \leftrightarrow (A \wedge X1_i) \vee (\neg A \wedge X2_i))\wedge$$
$$\wedge_{i=0,\ldots,31}\ (Y6_i \leftrightarrow (B \wedge Y4_i) \vee (\neg B \wedge Y5_i)) \rightarrow$$
$$formula\{X3 + Y6 > 0\}$$

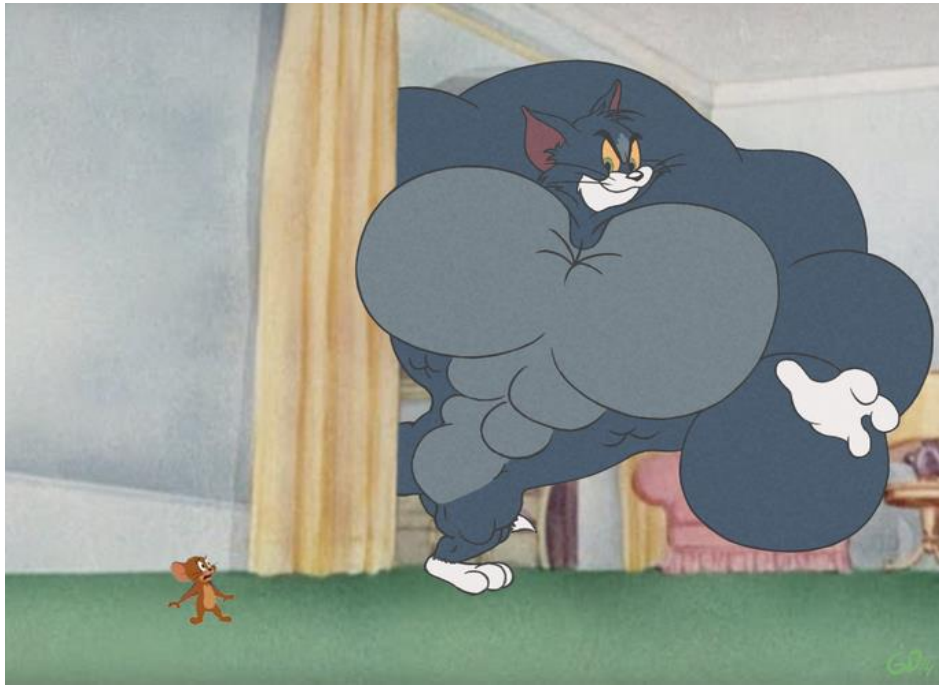
How to simplify it?

Predicate（谓词）

DEFINITION

Predicates describe properties of individuals（个体词）.

The range of individuals is domain of discourse（论域）.



individuals

$$STRONGER(Tom, Jerry)$$

We use upper-case words to represent predicates.

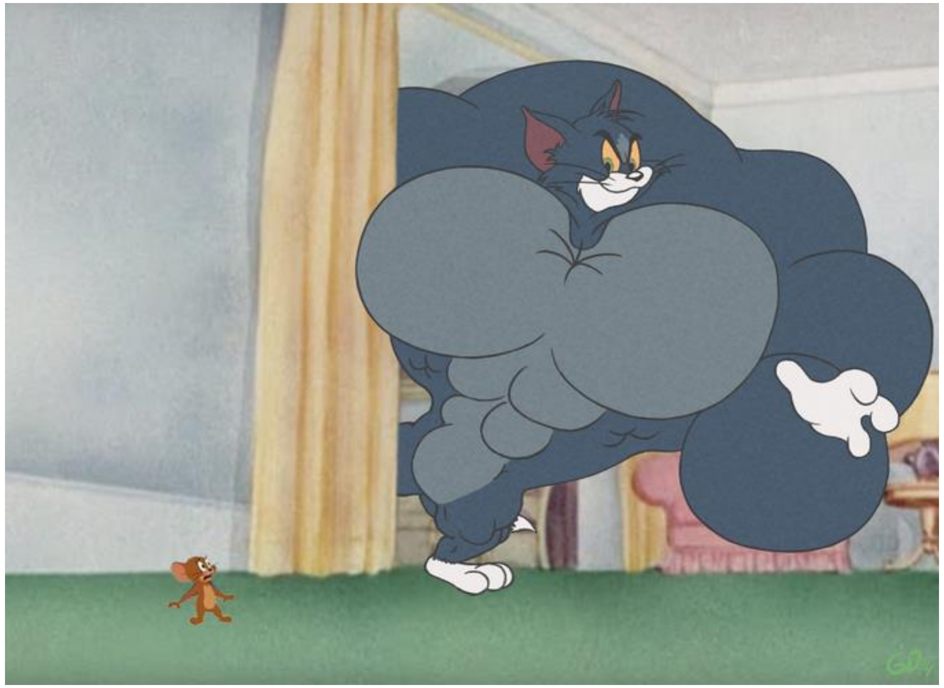
Predicate（谓词）

DEFINITION

Predicates describe properties of individuals（个体词）.

The range of individuals is domain of discours

Tom and Jerry are individual constants.（个体常项）



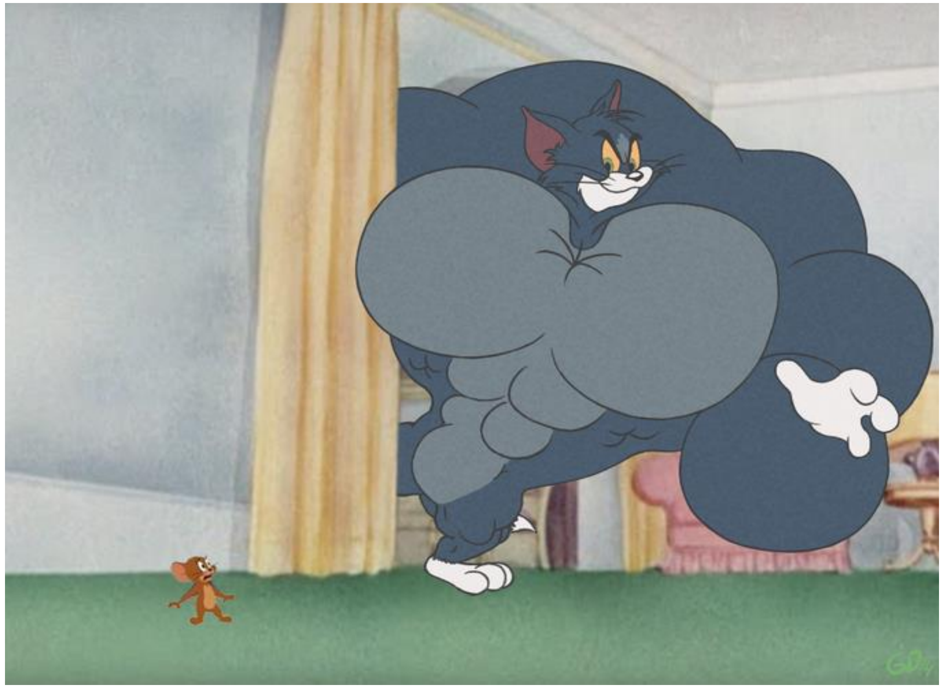
$$STRONGER(Tom, Jerry)$$

STRONGER is a predicate constant.（谓词常项）

Predicate（谓词）

NOTE

Each predicate maps individuals to T or F.



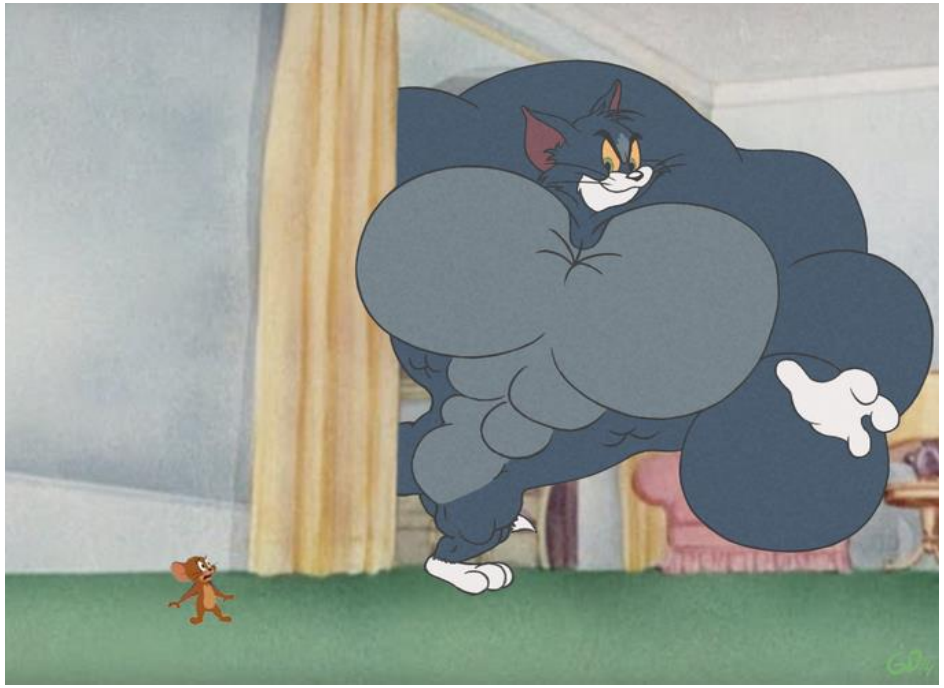
STRONGER maps Tom and Jerry to T or F.

$$STRONGER(Tom, Jerry)$$

Predicate（谓词）

NOTE

Each predicate has an associated arity, a natural number indicating how many arguments it takes.



$$STRONGER(Tom, Jerry)$$

STRONGER takes two arguments.

Predicate（谓词）

NOTE

Each predicate has an associated arity, a natural number indicating how many arguments it takes.

$$P(x_1, ..., x_n)$$

n-ary predicate
（n元谓词）

$$7 = 5$$

Equality is a special predicate of arity 2.

Predicate（谓词）

$$P(x_1, \dots, x_n)$$

It is not a proposition. Because P is a predicate variable（谓词变项） and xs are individual variables（个体变项）. It is a proposition only when they are all constants.

Predicate（谓词）

$$formula\{X1 = [00...01]\}\wedge$$
$$formula\{X2 = [00...00]\}\wedge$$
$$formula\{Y4 = [00...01]\}\wedge$$
$$formula\{Y5 = [00...00]\}\wedge$$
$$\wedge_{i=0,\ldots,31} (X3_i \leftrightarrow (A \wedge X1_i) \vee (\neg A \wedge X2_i))\wedge$$
$$\wedge_{i=0,\ldots,31} (Y6_i \leftrightarrow (B \wedge Y4_i) \vee (\neg B \wedge Y5_i)) \rightarrow$$
$$formula\{X3 + Y6 > 0\}$$

$\Rightarrow$

$$x1 = 1\wedge$$
$$x2 = 0\wedge$$
$$y4 = 1\wedge$$
$$y5 = 0\wedge$$
$$\wedge_{i=0,\ldots,31} (x3_i = (a \wedge x1_i) \vee (\neg a \wedge x2_i))\wedge$$
$$\wedge_{i=0,\ldots,31} (y6_i = (b \wedge y4_i) \vee (\neg b \wedge y5_i)) \rightarrow$$
$$formula\{x3 + y6 > 0\}$$

After introducing predicates

Predicate（谓词）

$$
\begin{aligned}
&formula\{X1 = [00...01]\}\wedge \\
&formula\{X2 = [00...00]\}\wedge \\
&formula\{Y4 = [00...01]\}\wedge \\
&formula\{Y5 = [00...00]\}\wedge \\
&\wedge_{i=0,\dots,31} (X3_i \leftrightarrow (A \wedge X1_i) \vee (\neg A \wedge X2_i))\wedge \\
&\wedge_{i=0,\dots,31} (Y6_i \leftrightarrow (B \wedge Y4_i) \vee (\neg B \wedge Y5_i)) \rightarrow \\
&formula\{X3 + Y6 > 0\}
\end{aligned}
$$

$$\Rightarrow$$

$$
\begin{aligned}
&x1 = 1\wedge \\
&x2 = 0\wedge \\
&y4 = 1\wedge \\
&y5 = 0\wedge \\
&\wedge_{i=0,\dots,31} (x3_i = (a \wedge x1_i) \vee (\neg a \wedge x2_i))\wedge \\
&\wedge_{i=0,\dots,31} (y6_i = (b \wedge y4_i) \vee (\neg b \wedge y5_i)) \rightarrow \\
&formula\{x3 + y6 > 0\}
\end{aligned}
$$

Can we go further?

Function（函数）

DEFINITION

A function maps individuals to an individual instead of T or F.

Functions are represented by lower-case words.

Function with one argument

$$bestfriend(SpongeBob) = PatrickStar$$

predicate

Function（函数）

DEFINITION

A function maps individuals to an individual instead of T or F.

Functions are represented by lower-case words.

Not a proposition

$$\boxed{bestfriend(SpongeBob)} = PatrickStar$$

Function（函数）

DEFINITION

A function maps individuals to an individual instead of T or F.

Functions are represented by lower-case words.

a proposition

$$\boxed{bestfriend(SpongeBob) = PatrickStar}$$

Function（函数）

When working in predicate logic, be careful to keep individuals (actual things) and propositions (true or false) separate.

$$bestfriend(SpongeBob) \xrightarrow{\triangle} PatrickStar$$

Don't use connectives to associate individuals.

$$\overset{\triangle}{not}(T) = F$$

Functions cannot operate on propositions.

The Type-Checking Table

	arguments	results
connectives	propositions	a proposition
predicates	individuals	a proposition
functions	individuals	an individual

Examples

Predicate logic is an expansion of propositional logic.

Propositional logic

- Equality: no
- Predicates: P, Q, R, …
- Functions: no

Propositional variables can be treated as nullary predicates.

Number theory

- Equality: yes
- Predicates: >, <, …
- Functions: +, -, …

Function（函数）

$x1 = 1\wedge$

$x2 = 0\wedge$

$y4 = 1\wedge$

$y5 = 0\wedge$

$\wedge_{i=0,\ldots,31}\ (x3_i = (a \wedge x1_i) \vee (\neg a \wedge x2_i))\wedge$

$\wedge_{i=0,\ldots,31}\ (y6_i = (b \wedge y4_i) \vee (\neg b \wedge y5_i)) \rightarrow$

$formula\{x3 + y6 > 0\}$

➡

Ite: a?x1:x2

$x1 = 1\wedge$

$x2 = 0\wedge$

$y4 = 1\wedge$

$y5 = 0\wedge$

$x3 = ite(a, x1, x2)\wedge$

$y6 = ite(b, y4, y5) \rightarrow$

$sum(x3, y6) > 0$

Function（函数）

$$x1 = 1 \wedge$$
$$x2 = 0 \wedge$$
$$y4 = 1 \wedge$$
$$y5 = 0 \wedge$$
$$\wedge_{i=0,\ldots,31} (x3_i = (a \wedge x1_i) \vee (\neg a \wedge x2_i)) \wedge$$
$$\wedge_{i=0,\ldots,31} (y6_i = (b \wedge y4_i) \vee (\neg b \wedge y5_i)) \rightarrow$$
$$formula\{x3 + y6 > 0\}$$

$\Rightarrow$

$$x1 = 1 \wedge$$
$$x2 = 0 \wedge$$
$$y4 = 1 \wedge$$
$$y5 = 0 \wedge$$
$$x3 = ite(a, x1, x2) \wedge$$
$$y6 = ite(b, y4, y5) \rightarrow$$
$$sum(x3, y6) > 0$$

We reserve logical connectives.

Function（函数）

$$x1 = 1 \wedge$$

$$x2 = 0 \wedge$$

$$y4 = 1 \wedge$$

$$y5 = 0 \wedge$$

$$x3 = ite(a, x1, x2) \wedge$$

$$y6 = ite(b, y4, y5) \rightarrow$$

$$sum(x3, y6) > 0$$

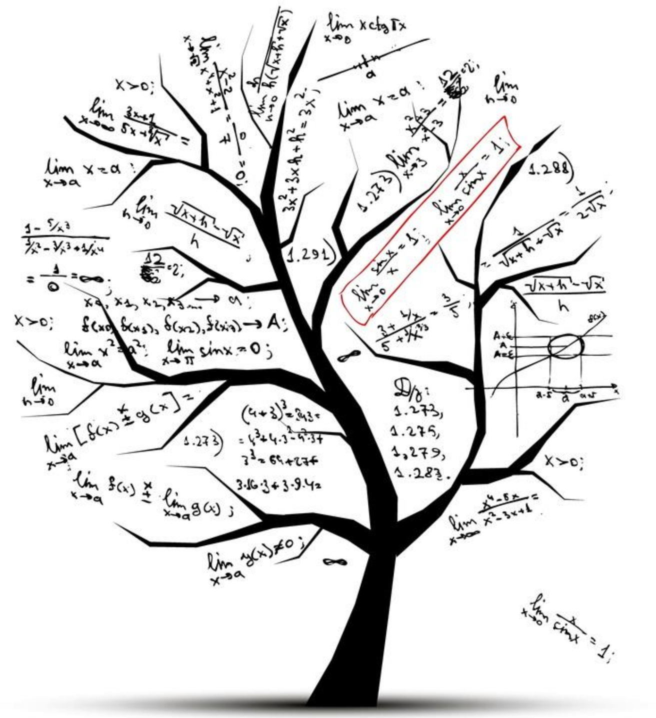
There are some individual variables. It is not a real proposition. But we cannot replace them with constants.

How to express "for all integers"?

Quantifier（量词）

DEFINITION

A quantifier turns a formula about individuals having some property into a formula about the number (quantity) of individuals having the property.

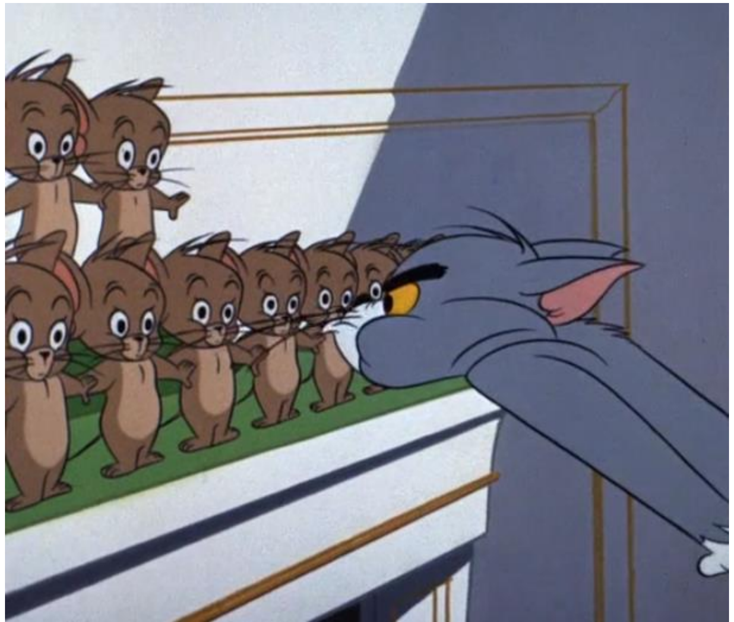


We have met quantifiers in mathematics.

The Universal Quantifier（全称量词）

DEFINITION

The universal quantifier expresses that a proposition can be satisfied by every member of the domain of discourse, which is interpreted as "given any" or "for all".



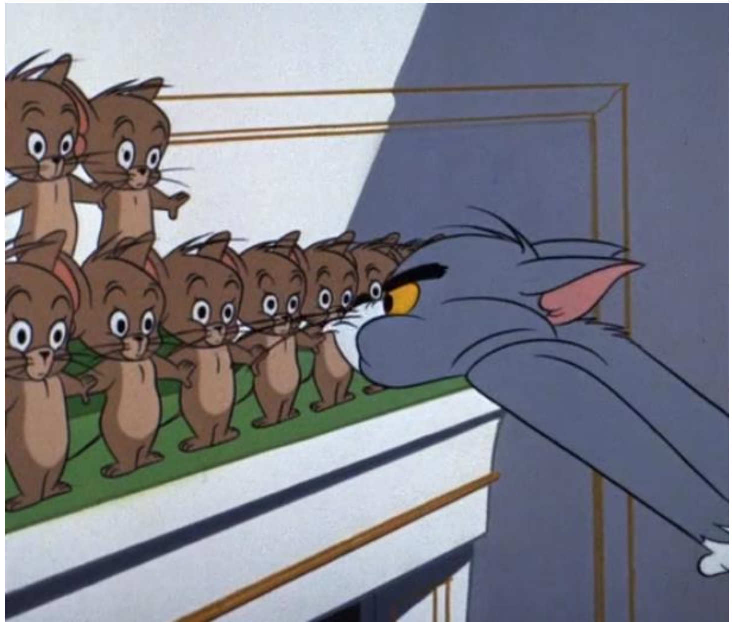
All toys look like Jerry.

$$(\forall x)(LIKEJERRY(x))$$

The Universal Quantifier（全称量词）

DEFINITION

The universal quantifier expresses that a proposition can be satisfied by every member of the domain of discourse, which is interpreted as "given any" or "for all".



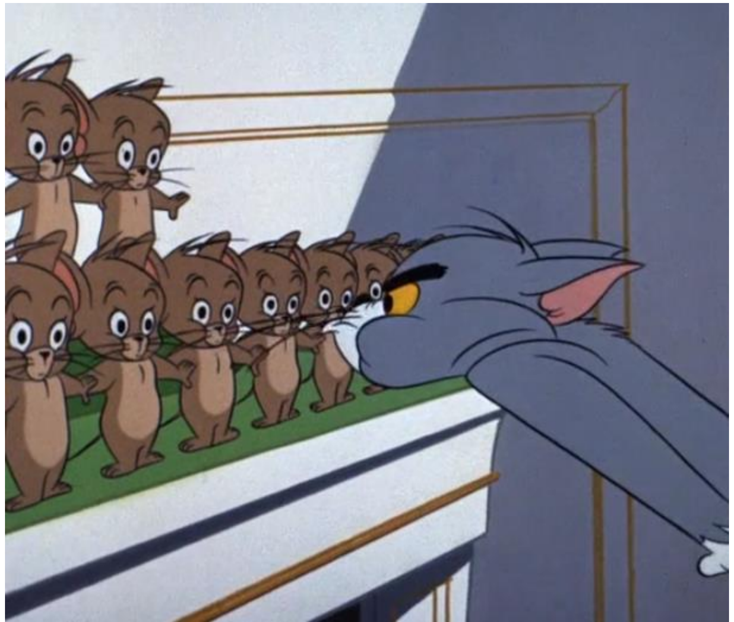
It is the scope（辖域）of the universal quantifier.

$$(\forall x)\boxed{(LIKEJERRY(x))}$$

The Universal Quantifier（全称量词）

DEFINITION

The universal quantifier expresses that a proposition can be satisfied by every member of the domain of discourse, which is interpreted as "given any" or "for all".



x is bound by the universal quantifier.
It is a bound variable（约束变元）.

$$(\forall \boxed{x})(LIKEJERRY(x))$$

The Universal Quantifier（全称量词）

$$(\forall x)(MOUSE(x))$$

The Universal Quantifier（全称量词）




$$(\forall x)(MOUSE(x))$$ FALSE

The Universal Quantifier（全称量词）

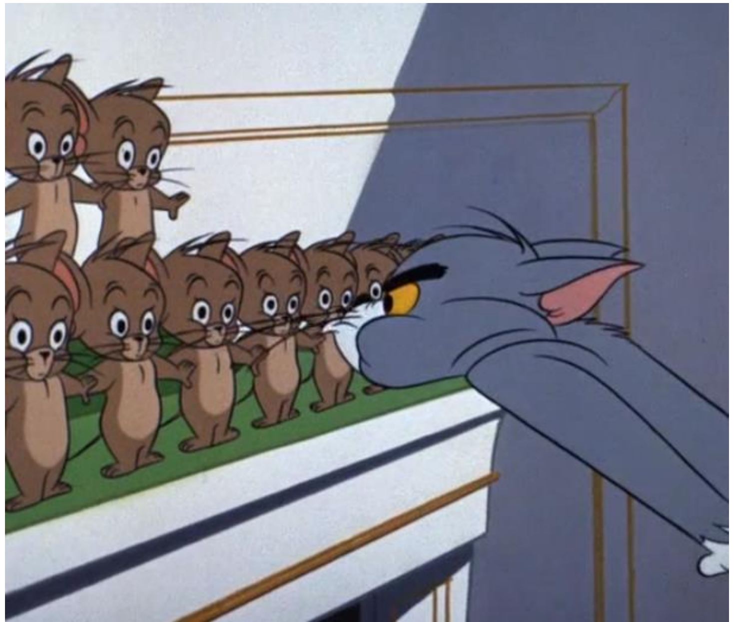
Domain is empty.

$(\forall x)(MOUSE(x))$ TRUE

The Existential Quantifier（存在量词）

DEFINITION

The existential quantifier expresses that a proposition can be satisfied by some member of the domain of discourse, which is interpreted as "there exists", "there is at least one", or "for some".



There exists a real Jerry.

$$(\exists x)(ISJERRY(x))$$

The Existential Quantifier（存在量词）





$$(\exists x)(ELEPHANT(x))$$

The Existential Quantifier（存在量词）

$$(\exists x)(ELEPHANT(x))$$ TRUE

The Existential Quantifier（存在量词）

Domain is empty.

$$(\exists x)(ELEPHANT(x))$$ FALSE

Quantifier（量词）

The blue rectangle is the scope of "exists x".

$$(\exists x)\boxed{((\forall y)\boxed{(x \leqslant y)})}$$

x and y are all bound variables.

The red rectangle is the scope of "for all y".

Quantifier（量词）

Generally, we don't use repeated names.

The red rectangle is the scope of "exists x".

$$(\exists x)\boxed{P(x)} \vee Q(x)$$

The first x is a bound variable.

The second x is a free variable（自由变元）. It is not bound by any quantifiers.

Quantifier（量词）

Proposition should not include free variables

There are two ways to eliminate free variables

- Replace free variables with constants
- Use quantifiers to convert free variables to bound variables

We can change the name of bound variables（变元易名规则）

$$(\forall y)(LIKEJERRY(y)) = (\forall x)(LIKEJERRY(x))$$

Quantifier（量词）

PROPERTY

1) $(\forall x)(\forall y)P(x,y) = (\forall x)((\forall y)P(x,y))$

2) $(\forall x)(\exists y)P(x,y) = (\forall x)((\exists y)P(x,y))$

3) $(\exists x)(\forall y)P(x,y) = (\exists x)((\forall y)P(x,y))$

4) $(\exists x)(\exists y)P(x,y) = (\exists x)((\exists y)P(x,y))$

5) $(\forall x)(\forall y)P(x,y) = (\forall y)(\forall x)P(x,y)$

6) $(\exists x)(\exists y)P(x,y) = (\exists y)(\exists x)P(x,y)$

Quantifier（量词）

To understand quantifiers better, we can discuss quantifiers with finite domain of discourse.

Quantifier（量词）

Assuming that domain include k elements {1, 2, …, k}.

Then

$$(\forall x)P(x) = P(1) \wedge P(2) \wedge ... \wedge P(k)$$
$$(\exists x)P(x) = P(1) \vee P(2) \vee ... \vee P(k)$$

Exercise

Now the domain is {1, 2}.

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

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

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

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

Can you convert these expressions to be quantifier-free?

Quantifier（量词）

$$(\forall x)(\forall y)P(x,y) = P(1,1) \wedge P(1,2) \wedge P(2,1) \wedge P(2,2)$$

$$(\exists x)(\exists y)P(x,y) = P(1,1) \vee P(1,2) \vee P(2,1) \vee P(2,2)$$

$$(\exists x)(\forall y)P(x,y) = (P(1,1) \wedge P(1,2)) \vee (P(2,1) \wedge P(2,2))$$

$$(\forall y)(\exists x)P(x,y) = (P(1,1) \vee P(2,1)) \wedge (P(1,2) \vee P(2,2))$$

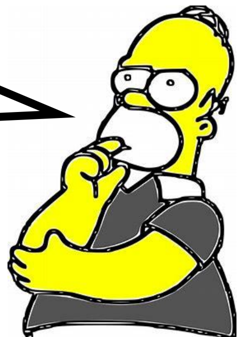
The conversion can help us understand some important properties.

Quantifier（量词）

$$(\exists x)(\forall y)P(x,y) = (P(1,1) \wedge P(1,2)) \vee (P(2,1) \wedge P(2,2))$$

$$(\forall y)(\exists x)P(x,y) = (P(1,1) \vee P(2,1)) \wedge (P(1,2) \vee P(2,2))$$

The two different quantifiers cannot be swapped. But what the relation between them?



Quantifier（量词）

$$
\begin{aligned}
&(\forall y)(\exists x)P(x,y)\\
=&(P(1,1)\vee P(2,1))\wedge(P(1,2)\vee P(2,2))\\
=&((P(1,1)\vee P(2,1))\wedge P(1,2))\vee((P(1,1)\vee P(2,1))\wedge P(2,2))\\
=&(P(1,1)\wedge P(1,2))\vee(P(2,1)\wedge P(1,2))\vee(P(1,1)\wedge P(2,2))\vee(P(2,1)\wedge P(2,2))\\
=&(P(1,1)\wedge P(1,2))\vee(P(2,1)\wedge P(2,2))\vee(P(2,1)\wedge P(1,2))\vee(P(1,1)\wedge P(2,2))\\
=&(\exists x)(\forall y)P(x,y)\vee(P(2,1)\wedge P(1,2))\vee(P(1,1)\wedge P(2,2))
\end{aligned}
$$

$$\Downarrow$$

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

Quantifier（量词）

$$
\begin{aligned}
&x1 = 1 \wedge \\
&x2 = 0 \wedge \\
&y4 = 1 \wedge \\
&y5 = 0 \wedge \\
&x3 = ite(a, x1, x2) \wedge \\
&y6 = ite(b, y4, y5) \rightarrow \\
&sum(x3, y6) > 0
\end{aligned}
$$

$$
\begin{aligned}
&(\forall x1)(\forall x2)(\forall y4)(\forall x3)(\forall y5)(\forall y6)(\forall a)(\forall b)(\\
&x1 = 1 \wedge \\
&x2 = 0 \wedge \\
&y4 = 1 \wedge \\
&y5 = 0 \wedge \\
&x3 = ite(a, x1, x2) \wedge \\
&y6 = ite(b, y4, y5) \rightarrow \\
&sum(x3, y6) > 0)
\end{aligned}
$$

After introducing quantifiers

Well-formed Formula（合式公式）

We can use predicates, functions, quantifiers, individuals and connectives to build many expressions. But not all of them have legal semantics.

Well-formed Formula（合式公式）

Parameter requirements:

- Predicates can only operate on individuals
- Functions can only operate on individuals
- Quantifiers can only bound individuals

Symbol writing:

- Propositional variables: p, q, r, …
- Individual variables: x, y, z, …
- Individual constants: a, b, c, … or upper-case words
- Predicate variables: P, Q, R, …
- Predicate constants: upper-case words
- Functions: f, g, ...

Well-formed Formula（合式公式）

DEFINITION

Terms are expressions generated from the individuals by the functions.

DEFINITION

An **atomic formula**（原子谓词公式） is an expression of the form

P(x1, . . ., xn) where P is a predicate of arity n and x1,. . . ,xn are terms.

Well-formed Formula (合式公式)

INDUCTIVE DEFINITION of WFF

1). Every atomic formula is in WFF.

2). If A and B are WFFs, so are $(\neg A)$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, and $(A \leftrightarrow B)$. There is no variable which is bounded in one wff and free in the other wff.

3). If A is a WFF and x is free in A, then $(\forall x)A$, $(\exists x)A$ are wffs.

4). No expression is WFF unless forced by 1), 2) or 3).

Well-formed Formula（合式公式）

Propositional Logic

1). Every single proposition (symbol) is in WFF.

2). If *A* and *B* are WFF, so are $(\neg A)$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, and $(A \leftrightarrow B)$.

3). No expression is WFF unless forced by 1) or 2).

Predicate Logic

1). Every atomic formula is in WFF.

2). If A and B are WFFs, so are $(\neg A)$, $(A \wedge B)$, $(A \vee B)$, $(A \rightarrow B)$, and $(A \leftrightarrow B)$. There is no variable which is bound in one wff and free in the other wff.

3). If A is a WFF and x is free in A, then $(\forall x)A$, $(\exists x)A$ are wffs.

4). No expression is WFF unless forced by 1), 2) or 3).

Exercise

$$P(x) \vee (\forall x)Q(x)$$

$$(\forall x)(P(x) \wedge Q(x))$$

$$(\exists x)(\forall x)P(x)$$

$$(\exists x)P(y, z)$$

$$(\forall x)(P(x) \rightarrow (\exists y)Q(x, y))$$

Exercise

$$P(x) \lor (\forall x)Q(x)$$ ✗

$$(\forall x)(P(x) \land Q(x))$$ ✓

$$(\exists x)(\forall x)P(x)$$ ✗

$$(\exists x)P(y, z)$$ ✗

$$(\forall x)(P(x) \rightarrow (\exists y)Q(x, y))$$ ✓

Well-formed Formula（合式公式）

All rational numbers are real numbers.

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

P means x is a rational number.

Q means x is a real number.

Some real numbers are not rational numbers.

A means x is a real number.

B means x is a rational number.

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

Validity（有效性）

```
void f(bool a, bool b)
{
    unsigned x, y;
    if (a)
        x = 1;
    else
        x = 0;
    if (b)
        y = 1;
    else
        y = 0;
    assert(x + y > 0);
}
```

➡

$$(\forall x1)(\forall x2)(\forall y4)(\forall x3)(\forall y5)(\forall y6)(\forall a)(\forall b)($$
$$x1 = 1 \wedge$$
$$x2 = 0 \wedge$$
$$y4 = 1 \wedge$$
$$y5 = 0 \wedge$$
$$x3 = ite(a, x1, x2) \wedge$$
$$y6 = ite(b, y4, y5) \rightarrow$$
$$sum(x3, y6) > 0)$$

We want to prove it is always true. Then the assertion will not fail.

Validity（有效性）

DEFINITION

The interpretation（解释） of predicate formula includes predicate variables, propositional variables, functions and free individual variables.

DEFINITION

If a formula is always true under any interpretations, it is universally valid（普遍有效）. $(\forall x)(P(x) \lor \neg P(x))$

Validity（有效性）

DEFINITION

If a formula is true under some interpretation, it is satisfiable（可满足的）. $(\forall x)P(x)$

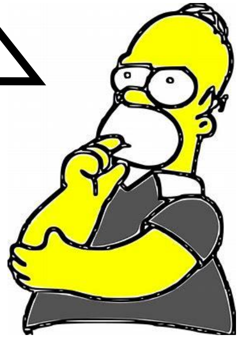
All positive integers are greater than 0.

DEFINITION

If a formula is always false under any interpretations, it is unsatisfiable（不可满足的）. $(\forall x)(P(x) \land \neg P(x))$

Validity（有效性）

What is the relation between universally valid formula and satisfiable formula in predicate logic?



Validity（有效性）

RELATION

1) P is universally valid iff ¬P is unsatisfiable

2) P is satisfiable iff ¬P is not universally valid

$$
\begin{aligned}
&(\forall x1)(\forall x2)(\forall y4)(\forall x3)(\forall y5)(\forall y6)(\forall a)(\forall b)(\\
&x1 = 1 \wedge \\
&x2 = 0 \wedge \\
&y4 = 1 \wedge \\
&y5 = 0 \wedge \\
&x3 = ite(a, x1, x2) \wedge \\
&y6 = ite(b, y4, y5) \rightarrow \\
&sum(x3, y6) > 0)
\end{aligned}
$$

Universally valid

$\Longleftrightarrow$

$$
\begin{aligned}
&\neg(\forall x1)(\forall x2)(\forall y4)(\forall x3)(\forall y5)(\forall y6)(\forall a)(\forall b)(\\
&x1 = 1 \wedge \\
&x2 = 0 \wedge \\
&y4 = 1 \wedge \\
&y5 = 0 \wedge \\
&x3 = ite(a, x1, x2) \wedge \\
&y6 = ite(b, y4, y5) \rightarrow \\
&sum(x3, y6) > 0)
\end{aligned}
$$

unsatisfiable

Decision Problem（判定问题）

DEFINITION

Decision problem in predicate logic is whether there is an effective algorithm to determine if a formula is universally valid.

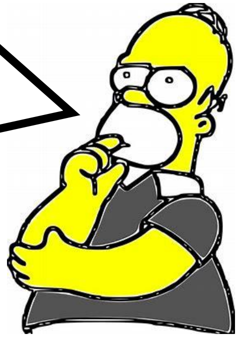
The algorithm must be automatic and terminate in finite steps.

Is predicate logic decidable?

Decision Problem（判定问题）

P	Q	$P \overline{\vee} Q$
T	T	F
F	T	T
T	F	T
F	F	F

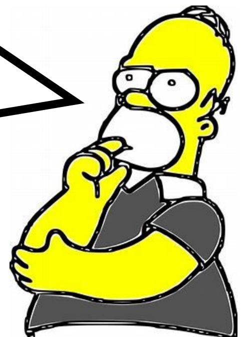
Propositional logic is a special predicate logic. It is decidable because we can use truth table to determine the validity.



Decision Problem（判定问题）

$$\begin{aligned} 2x + y - z &= 8 \\ -3x - y + 2z &= -11 \\ -2x + y + 2z &= -3 \end{aligned} \Rightarrow \left[\begin{array}{ccc|c} 2 & 1 & -1 & 8 \\ -3 & -1 & 2 & -11 \\ -2 & 1 & 2 & -3 \end{array}\right] \Rightarrow \left[\begin{array}{ccc|c} 1 & 0 & 0 & 2 \\ 0 & 1 & 0 & 3 \\ 0 & 0 & 1 & -1 \end{array}\right] \Rightarrow \begin{array}{rcl} x & = & 2 \\ y & = & 3 \\ z & = & -1 \end{array}$$

A system of linear equations is decidable. We can use Gaussian elimination to solve it.



Decision Problem（判定问题）

- Predicate logic is not decidable
- Predicate logic with finite domain is decidable
- Formula with only unary predicate variable is decidable
- The following forms are decidable

$$(\forall x_1)...(\forall x_n)P(x_1, ..., x_n)$$
$$(\exists x_1)...(\exists x_n)P(x_1, ..., x_n)$$

Decision Problem（判定问题）

That is why we need deduction.

We have used deduction in propositional logic

Deduction（推理）

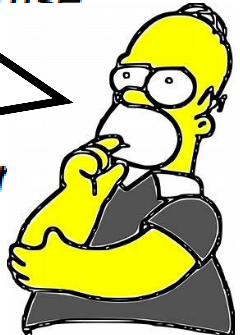
tautology implication

Prove $(P \to (Q \to S)) \wedge (\neg R \vee P) \wedge Q \Rightarrow R \to S$.

inference rule

$(1) \neg R \vee P$	*introduce premise*
$(2) R \to P$	*replacement rule*
$(3) R$	*introduce premise*
$(4) P$	*Hypothetical reasoning on* (2)(3)
$(5) P \to (Q \to S)$	*introduce premise*
$(6) Q \to S$	4)(5)
$(7) Q$	*emise*
$(8) S$	
$(9) R \to S$	*deduction theor*

Many terms, rules and symbols can be introduced from propositional logic.



Deduction formula（推理公式）

There are many important tautology implication expressions.

They can be used to produce new theorems.

$$\neg(P \to Q) \Rightarrow P$$

$$\neg(P \to Q) \Rightarrow \neg Q$$

$$P \wedge (P \to Q) \Rightarrow Q$$

$$(P \to Q) \wedge (Q \to R) \Rightarrow P \to R$$

$$P \wedge Q \Rightarrow P$$

$$P \Rightarrow P \vee Q$$

Tautology implication in propositional logic can also be applied here.

Deduction Formula（推理公式）

$$(\exists x)(P(x) \wedge Q(x)) \Rightarrow (\exists x)P(x) \wedge (\exists x)Q(x)$$

$$(\exists x)P(x) \wedge (\exists x)Q(x) \Rightarrow (\exists x)(P(x) \wedge Q(x))$$

Does it also hold?

Deduction Formula（推理公式）

x is a mouse.

x is a mouse and x is an elephant.

$$(\exists x)P(x) \wedge (\exists x)Q(x) \Rightarrow (\exists x)(P(x) \wedge Q(x))$$

x is an elephant.

The arrow is not bi-directional.

Deduction Formula（推理公式）

$$(\exists x)(P(x) \wedge Q(x)) \Rightarrow (\exists x)P(x) \wedge (\exists x)Q(x)$$

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

$$(\forall x)(P(x) \leftrightarrow Q(x)) \Rightarrow (\exists x)P(x) \leftrightarrow (\exists x)Q(x)$$

$$(\forall x)(P(x) \rightarrow Q(x)) \wedge (\forall x)(Q(x) \rightarrow R(x)) \Rightarrow (\forall x)(P(x) \rightarrow R(x))$$

$$(\forall x)(P(x) \rightarrow Q(x)) \wedge P(a) \Rightarrow Q(a)$$

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

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

They can be understood through semantics. (5.4.3)

Deduction Calculus（推理演算）

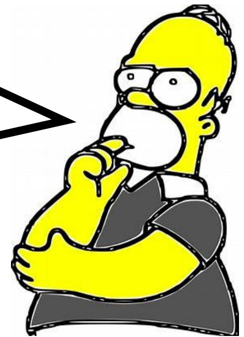
We now introduce an example to explain deduction calculus.

$$Premise: \quad (\forall x)(P(x) \to Q(x)),\ (\forall x)(Q(x) \to R(x))$$

$$Conclusion: \quad (\forall x)(P(x) \to R(x))$$

Proof.

We can still use some inference rules in propositional logic.



Deduction Calculus（推理演算）

We now introduce an example to explain deduction calculus.

$$Premise: \quad (\forall x)(P(x) \to Q(x)),\ (\forall x)(Q(x) \to R(x))$$

$$Conclusion: \quad (\forall x)(P(x) \to R(x))$$

Proof.

$(1)(\forall x)(P(x) \to Q(x))$ $\qquad$ $introduce\ premise$

We can still use some inference rules in propositional logic.

Deduction Calculus（推理演算）

We now introduce an example to explain deduction calculus.

$$Premise: \quad (\forall x)(P(x) \to Q(x)),\ (\forall x)(Q(x) \to R(x))$$

$$Conclusion: \quad (\forall x)(P(x) \to R(x))$$

Proof.

$(1)(\forall x)(P(x) \to Q(x))$ $\qquad$ *introduce premise*

$(2)P(x) \to Q(x)$ $\qquad$ *remove universal quantifier*

（全称量词消去规则）x is an arbitrary individual in the domain and it is free in P and Q.

Deduction Calculus（推理演算）

We now introduce an example to explain deduction calculus.

$$Premise: \quad (\forall x)(P(x) \rightarrow Q(x)),\ (\forall x)(Q(x) \rightarrow R(x))$$

$$Conclusion: \quad (\forall x)(P(x) \rightarrow R(x))$$

Proof.

$$(1)(\forall x)(P(x) \rightarrow Q(x)) \qquad introduce\, premise$$

$$(2)P(x) \rightarrow Q(x) \qquad remove\, universal\, quantifier$$

$$(3)(\forall x)(Q(x) \rightarrow R(x)) \qquad introduce\, premise$$

Deduction Calculus（推理演算）

We now introduce an example to explain deduction calculus.

$$Premise: \quad (\forall x)(P(x) \to Q(x)),\ (\forall x)(Q(x) \to R(x))$$

$$Conclusion: \quad (\forall x)(P(x) \to R(x))$$

Proof.

$(1)(\forall x)(P(x) \to Q(x))$ — *introduce premise*

$(2)P(x) \to Q(x)$ — *remove universal quantifier*

$(3)(\forall x)(Q(x) \to R(x))$ — *introduce premise*

$(4)Q(x) \to R(x)$ — *remove universal quantifier*

全称量词消去规则

Deduction Calculus（推理演算）

We now introduce an example to explain deduction calculus.

$$Premise: \quad (\forall x)(P(x) \to Q(x)),\ (\forall x)(Q(x) \to R(x))$$

$$Conclusion: \quad (\forall x)(P(x) \to R(x))$$

Proof.

$$(1)(\forall x)(P(x) \to Q(x)) \qquad introduce\ premise$$

$$(2)P(x) \to Q(x) \qquad remove\ universal\ quantifier$$

$$(3)(\forall x)(Q(x) \to R(x)) \qquad introduce\ premise$$

$$(4)Q(x) \to R(x) \qquad remove\ universal\ quantifier$$

$$(5)P(x) \to R(x) \qquad syllogism\ on\ (2)(4)$$

三段论

Deduction Calculus（推理演算）

We now introduce an example to explain deduction calculus.

$$Premise: \quad (\forall x)(P(x) \to Q(x)),\ (\forall x)(Q(x) \to R(x))$$

$$Conclusion: \quad (\forall x)(P(x) \to R(x))$$

Proof.

$(1)(\forall x)(P(x) \to Q(x))$

$(2)P(x) \to Q(x)$

$(3)(\forall x)(Q(x) \to R(x))$

$(4)Q(x) \to R(x)$

$(5)P(x) \to R(x)$

$(6)(\forall x)(P(x) \to R(x))$ — *introduce universal quantifier*

（全称量词引入规则）If any element in the domain satisfies the property and x is free in P and R, we can use the rule.

Deduction Calculus（推理演算）

We can also remove existential quantifier（存在量词消去规则）.

$$(\exists x)P(x) \Rightarrow P(c)$$

c is an individual constant.

$$(\exists x)P(x) = (\exists x)(c < x)$$

P should not include c.

$$(\exists x)P(x) = (\exists x)(x > y)$$

P should not include free variables.

Deduction Calculus（推理演算）

We can also introduce existential quantifier（存在量词引入规则）.

$$P(c) \Rightarrow (\exists x)P(x)$$

c is an individual constant.

$$P(c) = (\exists x)(x > 0) \quad \boxtimes$$

P should not include x.

Exercise

$$Premise:(\exists x)P(x) \to (\forall x)((P(x) \vee Q(x)) \to R(x))$$

$$(\exists x)P(x)$$

$$Conclusion: \ (\exists x)(\exists y)(R(x) \wedge R(y))$$

(1) $(\exists x)P(x)\to(\forall x)((P(x)\vee Q(x))\to R(x))$	前提
(2) $(\exists x)P(x)$	前提
(3) $(\forall x)((P(x)\vee Q(x))\to R(x))$	(1),(2) 分离
(4) $P(c)$	(2) 存在量词消去
(5) $P(c)\vee Q(c)\to R(c)$	(3) 全称量词消去
(6) $P(c)\vee Q(c)$	(4)
(7) $R(c)$	(5),(6) 分离
(8) $(\exists x)R(x)$	(7) 存在量词引入
(9) $(\exists y)R(y)$	(7) 存在量词引入
(10) $(\exists x)R(x)\wedge(\exists y)R(y)$	(8),(9)
(11) $(\exists x)(\exists y)(R(x)\wedge R(y))$	(10) 置换

Resolution Reasoning(归结推理法)

Deduction before depends on many proof skills. Resolution reasoning is more automatic.

A Quick Recap

DEFINITION

For formula $A = P \vee Q$ and $B = \neg P \vee R$, $R(A, B) = Q \vee R$ is a resolvent(归结式) of A and B.

- To prove $\boldsymbol{A} \Longrightarrow \boldsymbol{B}$, we just need to prove $\boldsymbol{A} \wedge \neg \boldsymbol{B}$ is a contradiction
- Convert $\boldsymbol{A} \wedge \neg \boldsymbol{B}$ to CNF P1∧P2∧… ∧Pn
- Repeatly produce resolvents from Pi
- Find contradiction and proof ends

$(\boldsymbol{P} \vee \boldsymbol{Q}) \wedge (\neg \boldsymbol{P} \vee \boldsymbol{R})$
$\Longrightarrow \boldsymbol{Q} \vee \mathbf{R}$

Now the main problem is how to eliminate quantifiers?

A Quick Recap

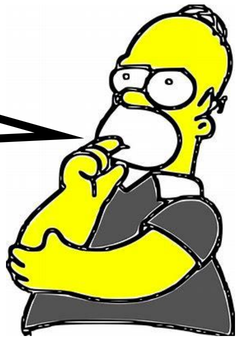
DEFINITION

For formula $A = P \vee Q$ and $B = \neg P \vee R$, $R(A, B) = Q \vee R$ is a resolvent(归结式) of A and B.

- To prove $\boldsymbol{A} \Longrightarrow \boldsymbol{B}$, we just need to prove $\boldsymbol{A} \wedge \neg \boldsymbol{B}$ is a contradiction
- Convert $\boldsymbol{A} \wedge \neg \boldsymbol{B}$ to CNF P1∧P2∧… ∧Pn
- Repeatly produce resolvents from Pi
- Find contradiction and proof ends

$(\boldsymbol{P} \vee \boldsymbol{Q}) \wedge (\neg \boldsymbol{P} \vee \boldsymbol{R})$
$\Longrightarrow \boldsymbol{Q} \vee \mathbf{R}$

We can firstly move all quantifiers to the left.



Prenex Normal Form(前束范式)

DEFINITION

A formula of the predicate calculus is in prenex normal form(PNF) if it is written as a string of quantifiers and bound variables, followed by a quantifier-free part, called the matrix（母式/基式）.

$$(Q_1 x_1) \ldots (Q_n x_n) M(x_1, \ldots, x_n)$$

quantifiers

Prenex Normal Form(前束范式)

DEFINITION

A and B are equivalent if A and B have the same truth value under any interpretation.

$A \leftrightarrow B$ is universally valid / $A = B$ / $A \Leftrightarrow B$

THEOREM

Every formula in predicate logic is equivalent to a formula in prenex normal form.

How to convert a formula to PNF? We need some equations.

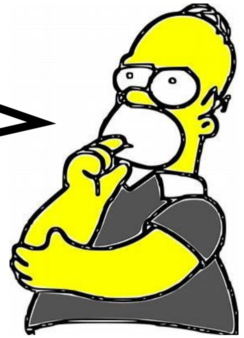
Equivalence(等值)

double negation law

$$\neg\neg(\forall x)P(x) = (\forall x)P(x)$$

$$(\forall x)P(x) \rightarrow (\exists x)Q(x) = \neg(\forall x)P(x) \vee (\exists x)Q(x)$$

Some equations in propositional logic can be directly applied.



Equivalence(等值)

$$(\forall x)P(x) \to (\exists x)Q(x) = \boxed{\neg(\forall x)P(x)} \vee (\exists x)Q(x)$$

We need to move "not" inside.

THEOREM

$$\neg(\forall x)P(x) = (\exists x)\neg P(x)$$

$$\neg(\exists x)P(x) = (\forall x)\neg P(x)$$

No x satisfies P.

Every x doesn't satisfy P.

Equivalence(等值)

If the domain is {1, 2}

$$
\begin{aligned}
&\neg(\forall x)P(x) \\
=&\neg(P(1) \wedge P(2)) \\
=&\neg P(1) \vee \neg P(2) \\
=&(\exists x)\neg P(x)
\end{aligned}
$$

Morgan's law

Equivalence(等值)

We can prove through semantics.

$$\neg(\forall x)P(x) = (\exists x)\neg P(x)$$

$$\neg(\forall x)P(x) = T$$
$$\Downarrow$$
$$P(x0) = F$$
$$\Downarrow$$
$$\neg P(x0) = T$$
$$\Downarrow$$
$$(\exists x)\neg P(x) = T$$

$$(\exists x)\neg P(x) = T$$
$$\Downarrow$$
$$\neg P(x0) = T$$
$$\Downarrow$$
$$P(x0) = F$$
$$\Downarrow$$
$$\neg(\forall x)P(x) = T$$

Equivalence(等值)

If there are many quantifiers, we need to flip every quantifiers and move “not” inside.

$$\neg(\forall x)(\forall y)P(x,y) = (\exists x)(\exists y)\neg P(x,y)$$

Equivalence(等值)

THEOREM

Distributive Law（分配律）：

$$(\forall x)(P(x) \vee q) = (\forall x)P(x) \vee q$$
$$(\exists x)(P(x) \vee q) = (\exists x)P(x) \vee q$$
$$(\forall x)(P(x) \wedge q) = (\forall x)P(x) \wedge q$$
$$(\exists x)(P(x) \wedge q) = (\exists x)P(x) \wedge q$$

x has nothing to do with q.

Equivalence(等值)

THEOREM

Distributive Law（分配律）：

$$(\forall x)(P(x) \to q) = (\exists x)P(x) \to q$$
$$(\exists x)(P(x) \to q) = (\forall x)P(x) \to q$$
$$(\forall x)(p \to Q(x)) = p \to (\forall x)Q(x)$$
$$(\exists x)(p \to Q(x)) = p \to (\exists x)Q(x)$$

x has nothing to do with p and q.

Exercise

$$(\forall x)(p \rightarrow Q(x)) = p \rightarrow (\forall x)Q(x)$$

$$\begin{aligned}
&(\forall x)(p \rightarrow Q(x)) \\
&\quad = (\forall x)(\neg p \vee Q(x)) \\
&\quad = \neg p \vee (\forall x)Q(x) \\
&\quad = p \rightarrow (\forall x)Q(x)
\end{aligned}$$

Can you prove it with laws before?

Equivalence(等值)

THEOREM

Distributive Law（分配律）：

$$(\forall x)(P(x) \land Q(x)) = (\forall x)P(x) \land (\forall x)Q(x)$$

$$(\exists x)(P(x) \lor Q(x)) = (\exists x)P(x) \lor (\exists x)Q(x)$$

$$(\forall x)(P(x) \lor Q(x)) = (\forall x)P(x) \lor (\forall x)Q(x)$$

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

Are they true?

Equivalence(等值)

Assuming the domain is {1, 2} and you will understand it better.

$$
\begin{aligned}
&(\forall x)(P(x) \vee Q(x)) \\
=&(P(1) \vee Q(1)) \wedge (P(2) \vee Q(2)) \\
=&((P(1) \vee Q(1)) \wedge P(2)) \vee ((P(1) \vee Q(1)) \wedge Q(2)) \\
=&(P(1) \wedge P(2)) \vee (Q(1) \wedge P(2)) \vee (P(1) \wedge Q(2)) \vee (Q(1) \wedge Q(2)) \\
=&(\forall x)P(x) \vee (\forall x)Q(x) \vee (Q(1) \wedge P(2)) \vee (P(1) \wedge Q(2))
\end{aligned}
$$

Equivalence(等值)

We can change the name of bound variables.

THEOREM

Distributive Law（分配律）：

$$(\forall x)(\forall y)(P(x) \vee Q(y)) = (\forall x)P(x) \vee (\forall x)Q(x)$$

$$(\exists x)(\exists y)(P(x) \wedge Q(y)) = (\exists x)P(x) \wedge (\exists x)Q(x)$$

Proof

$$\begin{aligned} &(\forall x)P(x) \vee (\forall x)Q(x) \\ =&(\forall x)P(x) \vee (\forall y)Q(y) \\ =&(\forall x)(P(x) \vee (\forall y)Q(y)) \\ =&(\forall x)(\forall y)(P(x) \vee Q(y)) \end{aligned}$$

Prenex Normal Form(前束范式)

Now we can convert formulas to PNF.

$$\neg((\forall x)(\exists y)P(a,x,y) \to (\exists x)(\neg(\forall y)Q(y,b) \to R(x)))$$

Prenex Normal Form(前束范式)

Now we can convert formulas to PNF.

$$\neg((\forall x)(\exists y)P(a,x,y) \rightarrow (\exists x)(\neg(\forall y)Q(y,b) \rightarrow R(x)))$$
$$=\neg(\neg(\forall x)(\exists y)P(a,x,y) \vee (\exists x)(\neg\neg(\forall y)Q(y,b) \vee R(x)))$$

Step1: eliminate arrows.

Prenex Normal Form(前束范式)

Now we can convert formulas to PNF.

$$\neg((\forall x)(\exists y)P(a,x,y) \to (\exists x)(\neg(\forall y)Q(y,b) \to R(x)))$$

$$=\neg(\neg(\forall x)(\exists y)P(a,x,y) \vee (\exists x)(\neg\neg(\forall y)Q(y,b) \vee R(x)))$$

$$=(\forall x)(\exists y)P(a,x,y) \wedge (\forall x)((\exists y)\neg Q(y,b) \wedge \neg R(x))$$

Step1: eliminate arrows.

Step2: move "not" inside.

Prenex Normal Form(前束范式)

Now we can convert formulas to PNF.

$$\neg((\forall x)(\exists y)P(a,x,y) \to (\exists x)(\neg(\forall y)Q(y,b) \to R(x)))$$

$$=\neg(\neg(\forall x)(\exists y)P(a,x,y) \vee (\exists x)(\neg\neg(\forall y)Q(y,b) \vee R(x)))$$

$$=(\forall x)(\exists y)P(a,x,y) \wedge (\forall x)((\exists y)\neg Q(y,b) \wedge \neg R(x))$$

$$=(\forall x)((\exists y)P(a,x,y) \wedge (\exists y)\neg Q(y,b) \wedge \neg R(x))$$

Step1: eliminate arrows.

Step2: move "not" inside.

Step3: move quantifiers to the left.

Prenex Normal Form(前束范式)

Now we can convert formulas to PNF.

$$
\begin{aligned}
&\neg((\forall x)(\exists y)P(a,x,y) \to (\exists x)(\neg(\forall y)Q(y,b) \to R(x))) \\
=&\neg(\neg(\forall x)(\exists y)P(a,x,y) \vee (\exists x)(\neg\neg(\forall y)Q(y,b) \vee R(x))) \\
=&(\forall x)(\exists y)P(a,x,y) \wedge (\forall x)((\exists y)\neg Q(y,b) \wedge \neg R(x)) \\
=&(\forall x)((\exists y)P(a,x,y) \wedge (\exists y)\neg Q(y,b) \wedge \neg R(x)) \\
=&(\forall x)((\exists y)P(a,x,y) \wedge (\exists z)\neg Q(z,b) \wedge \neg R(x))
\end{aligned}
$$

Step1: eliminate arrows.

Step2: move "not" inside.

Step3: move quantifiers to the left.

Step4: change names of bound variables.

Prenex Normal Form(前束范式)

Now we can convert formulas to PNF.

$$
\begin{aligned}
&\neg((\forall x)(\exists y)P(a,x,y) \to (\exists x)(\neg(\forall y)Q(y,b) \to R(x))) \\
=&\neg(\neg(\forall x)(\exists y)P(a,x,y) \vee (\exists x)(\neg\neg(\forall y)Q(y,b) \vee R(x))) \\
=&(\forall x)(\exists y)P(a,x,y) \wedge (\forall x)((\exists y)\neg Q(y,b) \wedge \neg R(x)) \\
=&(\forall x)((\exists y)P(a,x,y) \wedge (\exists y)\neg Q(y,b) \wedge \neg R(x)) \\
=&(\forall x)((\exists y)P(a,x,y) \wedge (\exists z)\neg Q(z,b) \wedge \neg R(x)) \\
=&(\forall x)(\exists y)(\exists z)(P(a,x,y) \wedge \neg Q(z,b) \wedge \neg R(x))
\end{aligned}
$$

Step1: eliminate arrows.

Step2: move "not" inside.

Step3: move quantifiers to the left.

Step4: change names of bound variables.

Move quantifiers to the left.

Prenex Normal Form(前束范式)

Now we can convert formulas to PNF.

$$
\begin{aligned}
&\neg((\forall x)(\exists y)P(a,x,y) \to (\exists x)(\neg(\forall y)Q(y,b) \to R(x))) \\
=&\neg(\neg(\forall x)(\exists y)P(a,x,y) \vee (\exists x)(\neg\neg(\forall y)Q(y,b) \vee R(x))) \\
=&(\forall x)(\exists y)P(a,x,y) \wedge (\forall x)((\exists y)\neg Q(y,b) \wedge \neg R(x)) \\
=&(\forall x)((\exists y)P(a,x,y) \wedge (\exists y)\neg Q(y,b) \wedge \neg R(x)) \\
=&(\forall x)((\exists y)P(a,x,y) \wedge (\exists z)\neg Q(z,b) \wedge \neg R(x)) \\
=&(\forall x)(\exists y)(\exists z)(P(a,x,y) \wedge \neg Q(z,b) \wedge \neg R(x)) \\
=&(\forall x)(\exists y)(\exists z)S(a,b,x,y,z)
\end{aligned}
$$

Step1: eliminate arrows.

Step2: move “not” inside.

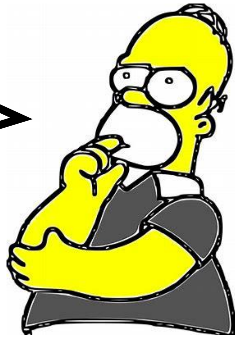
Step3: move quantifiers to the left.

Step4: change names of bound variables.

Move quantifiers to the left.

Skolem Normal Form(Skolem标准形)

Can we eliminate
existential quantifier?



Skolem Normal Form(Skolem标准形)

DEFINITION

A formula is in Skolem normal form if it is in prenex normal form with only universal quantifiers.

THEOREM

Every formula A can be converted to corresponding Skolem normal form B. A is unsatisfiable iff B is unsatisfiable.

This transformation does not preserve semantics but it does preserve satisfiability, which is consistent with resolution method.

Skolem Normal Form(Skolem标准形)

$$(\forall x)(\exists y)(\exists z)S(a,b,x,y,z)$$

$$\Downarrow$$

y depends on x

$$(\forall x)(\exists z)S(a,b,x,f(x),z)$$

$$\Downarrow$$

$$(\forall x)(\exists z)S(a,b,x,f(x),g(x))$$

z depends on x

If x doesn't depend on any variables, we replace x with a constant.

Exercise

$$(\exists x)(\forall y)(\forall z)(\exists u)(\forall v)(\exists w)P(x, y, z, u, v, w)$$

$$(\forall y)(\forall z)(\forall v)P(a, y, z, f(y, z), v, g(y, z, v)).$$

Resolution Reasoning(归结推理法)

DEFINITION

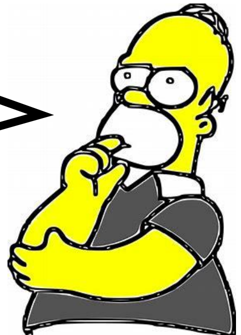
For formula $A = P \vee Q$ and $B = \neg P \vee R$, $R(A, B) = Q \vee R$ is a resolvent(归结式) of A and B.

$$\left.\begin{array}{l} P(x) \vee Q(x) \quad \Rightarrow \quad P(a) \vee Q(a) \\ \\ \neg P(a) \vee R(y) \end{array}\right\} \Rightarrow Q(a) \vee R(y)$$

Resolution Reasoning(归结推理法)

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \Rightarrow (\forall x)(P(x) \to R(x))$$

Now all in one.



Example

$$(\forall x)(P(x) \to Q(x)) \land (\forall x)(Q(x) \to R(x)) \Rightarrow (\forall x)(P(x) \to R(x))$$

Example

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \Rightarrow (\forall x)(P(x) \to R(x))$$

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \wedge \neg(\forall x)(P(x) \to R(x))$$

Prove it is unsatisfiable

Step 1. Prove its “not” is unsatisfiable.

Example

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \Rightarrow (\forall x)(P(x) \to R(x))$$

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \wedge \neg(\forall x)(P(x) \to R(x))$$

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \wedge (\exists x)(P(x) \wedge \neg R(x))$$

Prove it is unsatisfiable

Move "not" inside

Step 2. Convert it into Skolem form.

Example

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \Rightarrow (\forall x)(P(x) \to R(x))$$

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \wedge \neg(\forall x)(P(x) \to R(x))$$

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \wedge (\exists x)(P(x) \wedge \neg R(x))$$

$$(\forall x)(P(x) \to Q(x)) \wedge (\forall x)(Q(x) \to R(x)) \wedge (P(a) \wedge \neg R(a))$$

Prove it is unsatisfiable

Move "not" inside

Skolemization

Step 2. Convert it into Skolem form.

Example

$$(\forall x)(P(x) \rightarrow Q(x)) \wedge (\forall x)(Q(x) \rightarrow R(x)) \Rightarrow (\forall x)(P(x) \rightarrow R(x))$$

$$(\forall x)(P(x) \rightarrow Q(x)) \wedge (\forall x)(Q(x) \rightarrow R(x)) \wedge \neg(\forall x)(P(x) \rightarrow R(x))$$

Prove it is unsatisfiable

$$(\forall x)(P(x) \rightarrow Q(x)) \wedge (\forall x)(Q(x) \rightarrow R(x)) \wedge (\exists x)(P(x) \wedge \neg R(x))$$

Move "not" inside

$$(\forall x)(P(x) \rightarrow Q(x)) \wedge (\forall x)(Q(x) \rightarrow R(x)) \wedge (P(a) \wedge \neg R(a))$$

Skolemization

$$\Downarrow$$

$$\neg P(x) \vee Q(x) \qquad \neg Q(x) \vee R(x) \qquad P(a) \qquad \neg R(a)$$

Example

$$\neg P(x) \vee Q(x) \qquad \neg Q(x) \vee R(x) \qquad P(a) \quad \neg R(a)$$

$$(1) \neg P(x) \vee Q(x)$$

$$(2) \neg Q(x) \vee R(x)$$

$$(3) P(a)$$

$$(4) \neg R(a)$$

Example

$$\neg P(x) \vee Q(x) \qquad \neg Q(x) \vee R(x) \qquad P(a) \quad \neg R(a)$$

$(1) \neg P(x) \vee Q(x)$

$(2) \neg Q(x) \vee R(x)$

$(3) P(a)$

$(4) \neg R(a)$

$(5) Q(a) \qquad\qquad resolution\ on\ (1)(3)$

Example

$$\neg P(x) \vee Q(x) \qquad \neg Q(x) \vee R(x) \qquad P(a) \quad \neg R(a)$$

$$
\begin{array}{ll}
(1)\neg P(x) \vee Q(x) & \\
(2)\neg Q(x) \vee R(x) & \\
(3) P(a) & \\
(4)\neg R(a) & \\
(5) Q(a) & resolution\ on\ (1)(3) \\
(6) R(a) & resolution\ on\ (2)(5)
\end{array}
$$

Example

$$\neg P(x) \vee Q(x) \qquad \neg Q(x) \vee R(x) \qquad P(a) \qquad \neg R(a)$$

$$
\begin{array}{ll}
(1) \neg P(x) \vee Q(x) & \\
(2) \neg Q(x) \vee R(x) & \\
(3) P(a) & \\
(4) \neg R(a) & \\
(5) Q(a) & resolution\ on\ (1)(3) \\
(6) R(a) & resolution\ on\ (2)(5) \\
(7) \Box & resolution\ on\ (4)(6)
\end{array}
$$

Exercise

$$A1 = (\exists x)(P(x) \wedge (\forall y)(D(y) \rightarrow L(x,y)))$$

$$A2 = (\forall x)(P(x) \rightarrow (\forall y)(Q(y) \rightarrow \neg L(x,y)))$$

$$B = (\forall x)(D(x) \rightarrow \neg Q(x))$$

Prove $A1 \wedge A2 \Rightarrow B$

证明 不难建立 A_1 的子句集为 $\{P(a), \neg D(y) \vee L(a,y)\}$，$A_2$ 的子句集为 $\{\neg P(x) \vee \neg Q(y) \vee \neg L(x,y)\}$，$\neg B$ 的子句集为 $\{D(b), Q(b)\}$. 求并集得子句集 S，进而建立归结过程：

(1) $P(a)$	
(2) $\neg D(y) \vee L(a,y)$	
(3) $\neg P(x) \vee \neg Q(y) \vee \neg L(x,y)$	
(4) $D(b)$	
(5) $Q(b)$	
(6) $L(a,b)$	(2)(4) 归结
(7) $\neg Q(y) \vee \neg L(a,y)$	(1)(3) 归结
(8) $\neg L(a,b)$	(5)(7) 归结
(9) $\square$	(6)(8) 归结

Thanks & Questions