Propositional Logic

Adapted From:

教材《数理逻辑与集合论》1.1-1.5、2.1-2.4

Zhaoguo Wang

Proof

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

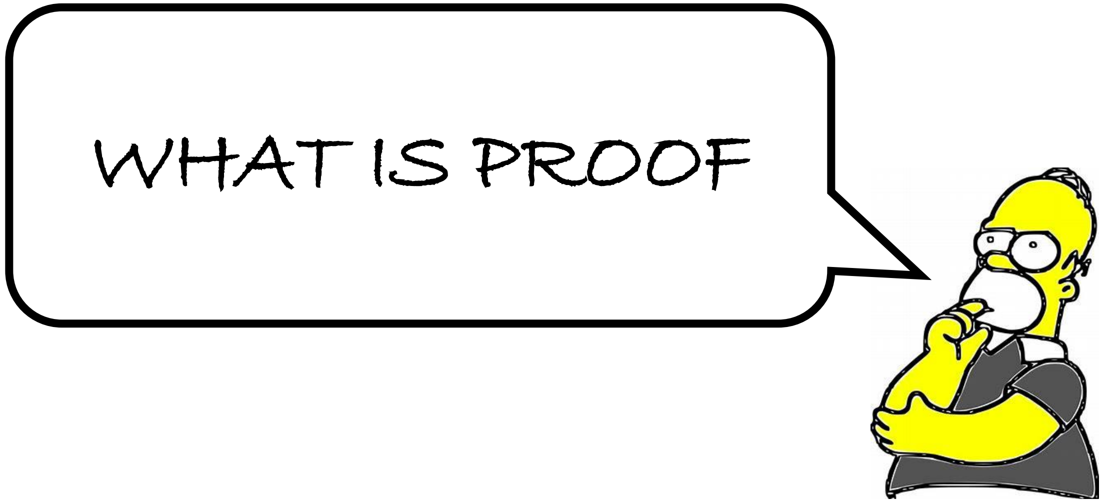
Step 1. Convert program into first order logic formula.

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



Proof

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 program 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

A ***proof*** is an argument that demonstrates why a conclusion is true, subject to certain standards of truth.

Proof

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 program 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

A ***mathematical proof*** is an argument that demonstrates why a mathematical statement is true, following the rules of mathematics.

Proof

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 program into first order logic formula.

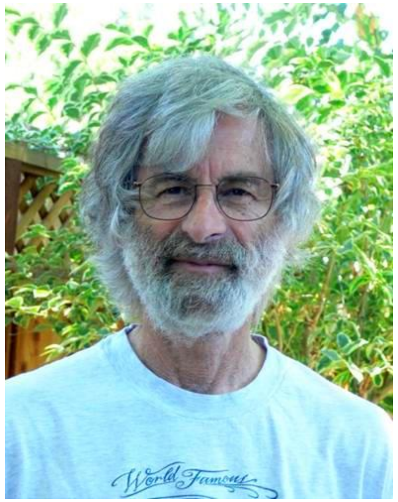
Step 2. Ask the computer to solve the formula.

1.3 Auto-active Proof

Step 1. Axiom system

Step 2. Ask the computer to check the invariants

Modern Proofs



Some 20 years ago, I published an article titled *How to Write a Proof* in a festschrift in honor of the 60^{th} birthday of Richard Palais [5]. In celebration of his 80^{th} birthday, I am describing here what I have learned since then about writing proofs and explaining how to write them.

How to Write a Proof

Leslie Lamport

February 14, 1993

revised December 1, 1993

How to Write a 21^{st} Century Proof

Leslie Lamport

23 November 2011

Minor change on 15 January 2012

Examples – Modern Proofs

Direct Proof.

Case By Case

Proof By Induction.

...

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

DEFINITION

An integer n is ***even*** if there is an integer k such that $n = 2k$.

An integer n is ***odd*** if there is an integer k such that $n = 2k + 1$.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

ASSUMPTIONS FOR NOW

Every integer is either even or odd.

No integer is both even and odd.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that n = 2k.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that $n = 2k$.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that $n = 2k$.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that $n = 2k$.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that $n = 2k$.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

This symbol means "end of proof".

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that n = 2k.

This means that n2 = $(2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since … 2k.

This …

From … k^2)

where …

Therefore, n^2 is even. ■

To prove a statement of the form "If P, then Q".

Assume that P is true, then show that Q must be true as well.

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that n = 2k.

This means that n2 = $(2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that n = 2k.

This means

From this,

where n^2 =

Therefore,

This is the definition of an even integer. When writing a mathematical proof, it's common to call back the definitions,

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that $n = 2k$.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Direct Proof

Theor

Notice how we use the value of k that we obtained above. Giving names to quantities, even if we aren't fully sure what they are, allows us to manipulate them. This is similar to variables in programs.

Let n b

Since n is even, there is some integer k such that n = 2k.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that n = 2k.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Direct Proof

The

Our ultimate goal is to prove that n is even. This means that we need to find some m such that $n^2 = 2m$. Here, we're explicitly showing how we can do that.

Le

Si

This

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Direct Proof

Theorem: If n is an even integer, then n^2 is even.

PROOF

Let n be an even integer.

Since n is even, there is some integer k such that $n = 2k$.

This means that $n^2 = (2k)^2 = 4k^2 = 2(2k^2)$.

From this, we see that there is an integer m (namely, $2k^2$) where $n^2 = 2m$.

Therefore, n^2 is even. ■

Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

DEFINITION

N queens problem: placing N chess queens on an N×N chessboard so that no two queens threaten each other.

Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

DEFINITION

N queens problem: placing N chess queens on an N×N chessboard so that no two queens threaten each other.

Two queens threaten each other: two queens share the same row, column or diagonal.

Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

DEFINITION

2 queens problem: placing 2 chess queens on an 2×2 chessboard so that no two queens share the same row, column or diagonal.

Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

REPHRASE

If placing 2 chess queens on an 2×2 chessboard, then they must share the same row, column or diagonal.

Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

INTUITION

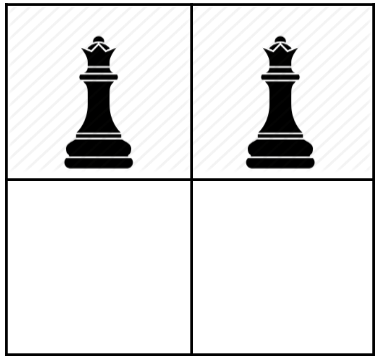
P1	P2
P3	P4

Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

INTUITION

P1	P2
P3	P4

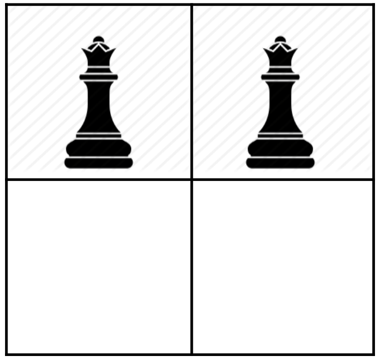


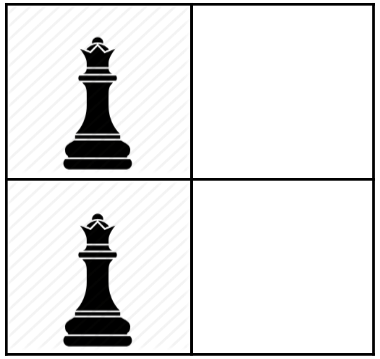
Modern Proofs – Proof By Cases

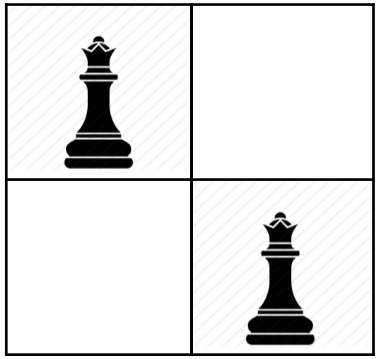
Theorem: Two queens problem has no solution.

INTUITION

P1	P2
P3	P4

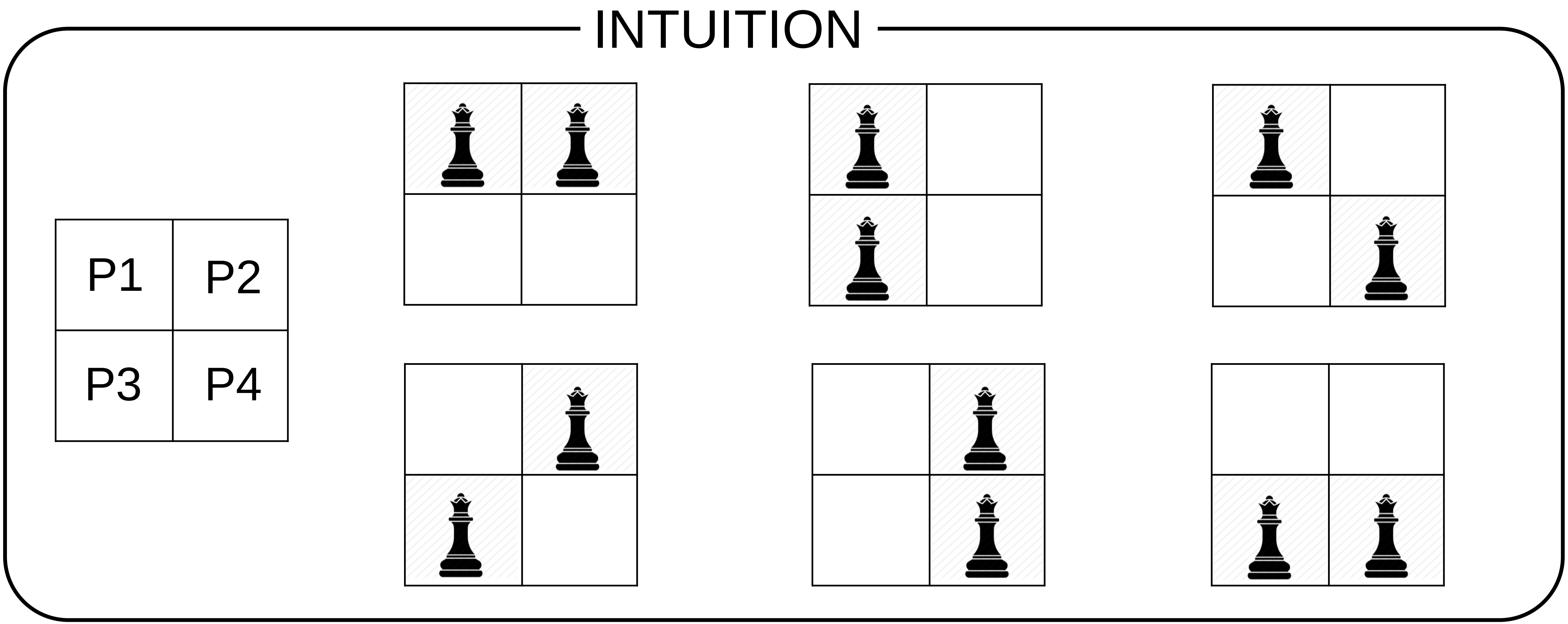






Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.



Modern Proofs – Proof By Cases

Theorem: Two queens problem has no solution.

PROOF

Let's pick two positions P_A and P_B to places these two queens. We will prove that two queens threaten each other by cases (by exhaustion).

Case 1: P_A is P1, and P_B is P2, so two queens share the same row.

Case 2: P_A is P1, and P_B is P3, so two queens share the same column.

Case 3: P_A is P1, and P_B is P4, so two queens share the same diagonal.

Case 4: P_A is P2, and P_B is P3, so two queens share the same diagonal.

Case 5: P_A is P2, and P_B is P4, so two queens share the same column.

Case 6: P_A is P3, and P_B is P4, so two queens share the same row.

Since P_A and P_B are symmetric, above list all cases. In any case, we find two queens threaten each other by definition. ■

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

Since $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$,

Then $\sum_{i=1}^{k+1} i = \left(\sum_{i=1}^{k} i\right) + (k+1) = \frac{k(k+1)}{2} + (k+1) = \frac{(k+1)(k+2)}{2}$. ■

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

Since $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$,

Then $\sum_{i=1}^{k+1} i = \left(\sum_{i=1}^{k} i\right) + (k+1) = \frac{k(k+1)}{2} + (k+1) = \frac{(k+1)(k+2)}{2}$. ■

REALLY??? Tiger, I do not trust you. ☺

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

Since $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$,

Then $\sum_{i=1}^{k+1} i = \left(\sum_{i=1}^{k} i\right) + (k+1) = \frac{k(k+1)}{2} + (k+1) = \frac{(k+1)(k+2)}{2}$. ■

It shows that we can always move to the next step, but we need to start somewhere.

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$ **and** if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$ **and** if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

First, since $\sum_{i=1}^{1} i = 1, and\ \frac{1(1+1)}{2} = 1$, then $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$.

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$ **and** if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

First, since $\sum_{i=1}^{1} i = 1, and\ \frac{1(1+1)}{2} = 1$, then $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$.

Second, since $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \left(\sum_{i=1}^{k} i\right) + (k+1)$

$$= \frac{k(k+1)}{2} + (k+1) = \frac{(k+1)(k+2)}{2}.$$

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$ **and** if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

First, since $\sum_{i=1}^{1} i = 1, and\ \frac{1(1+1)}{2} = 1$, then $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$.

Second, since $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \left(\sum_{i=1}^{k} i\right) + (k+1)$

$$= \frac{k(k+1)}{2} + (k+1) = \frac{(k+1)(k+2)}{2}.$$

Therefore, the theorem is true. ■

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

We will prove this by induction by showing that $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$ **and** if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

The general pattern here is the following:

1. Start by announcing that we're going to use a proof by induction.
2. Explicitly state the base case and the inductive case.
3. Go prove both cases.

Modern Proofs – Proof By Induction

Theorem: If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

PROOF

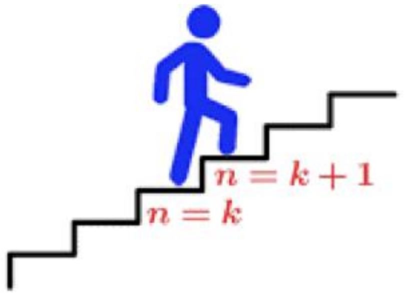
We will prove this by induction by showing that $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$ **and** if $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \frac{(k+1)(k+2)}{2}$.

First, since $\sum_{i=1}^{1} i = 1, and\ \frac{1(1+1)}{2} = 1$, then $\sum_{i=1}^{1} i = \frac{1(1+1)}{2}$.

Base Case : prove the property holds for base elements.

Modern Proofs – Proof By Induction

inductive case : prove the property holds for elements built by elements-building operations according to induction hypothesis.



First, since $\sum_{i=1}^{1} i = 1, and\ \frac{1(1+1)}{2} = 1$, then $\sum_{i=1}^{1} i = \frac{1(}{2}$.

Second, since $\sum_{i=1}^{k} i = \frac{k(k+1)}{2}$, then $\sum_{i=1}^{k+1} i = \left(\sum_{i=1}^{k} i\right) + (k+1)$

$$= \frac{k(k+1)}{2} + (k+1) = \frac{(k+1)(k+2)}{2}.$$

Therefore, the theorem is true. ■

Propositional Logic （命题逻辑）

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 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

Steps of Proving A Program

Program ⇨ Correctness ⇨ Propositional Formula ⇨ Computer

```
bool xor(int v)
{
    return v ^ v;
}
```

The ***xor*** function always returns 0

$$(((\neg v1) \wedge v1) \vee (v1 \wedge (\neg v1)))\vee$$
$$(((\neg v2) \wedge v2) \vee (v2 \wedge (\neg v2)))\vee$$
$$(((\neg v3) \wedge v3) \vee (v3 \wedge (\neg v3)))\vee$$
$$\ldots$$
$$(((\neg v32) \wedge v32) \vee (v32 \wedge (\neg v32)))$$

Steps of Proving A Program

Why?

Program ⇨ Correctness ⇨ Propositional Formula ⇨ Computer

```
bool xor(int v)
{
    return v ^ v;
}
```

The ***xor*** function always returns 0

$$(((\neg v1) \wedge v1) \vee (v1 \wedge (\neg v1)))\vee$$
$$(((\neg v2) \wedge v2) \vee (v2 \wedge (\neg v2)))\vee$$
$$(((\neg v3) \wedge v3) \vee (v3 \wedge (\neg v3)))\vee$$
$$\ldots$$
$$(((\neg v32) \wedge v32) \vee (v32 \wedge (\neg v32)))$$

Steps of Proving A Program

Program ⇨ Correctness ⇨ Propositional Formula ⇨ Computer

```
bool xor(int v)
{
    return v ^ v;
}
```

The ***xor*** function always returns 0

This is a proposition.

$$(((\neg v1) \wedge v1) \vee (v1 \wedge (\neg v1)))\vee$$
$$(((\neg v2) \wedge v2) \vee (v2 \wedge (\neg v2)))\vee$$
$$(((\neg v3) \wedge v3) \vee (v3 \wedge (\neg v3)))\vee$$
$$\ldots$$
$$(((\neg v32) \wedge v32) \vee (v32 \wedge (\neg v32)))$$

What is Proposition? (命题)

Definition

A ***proposition*** is a statement that is, by itself, either true or false.

Examples

Theorem proved before are all propositions:

If n is an even integer, then n^2 is even.

If n^2 is an even integer, then n is even.

If n is an integer and n > 0, then $\sum_{i=1}^{n} i = \frac{n(n+1)}{2}$

Two queens problem has no solution.

Examples

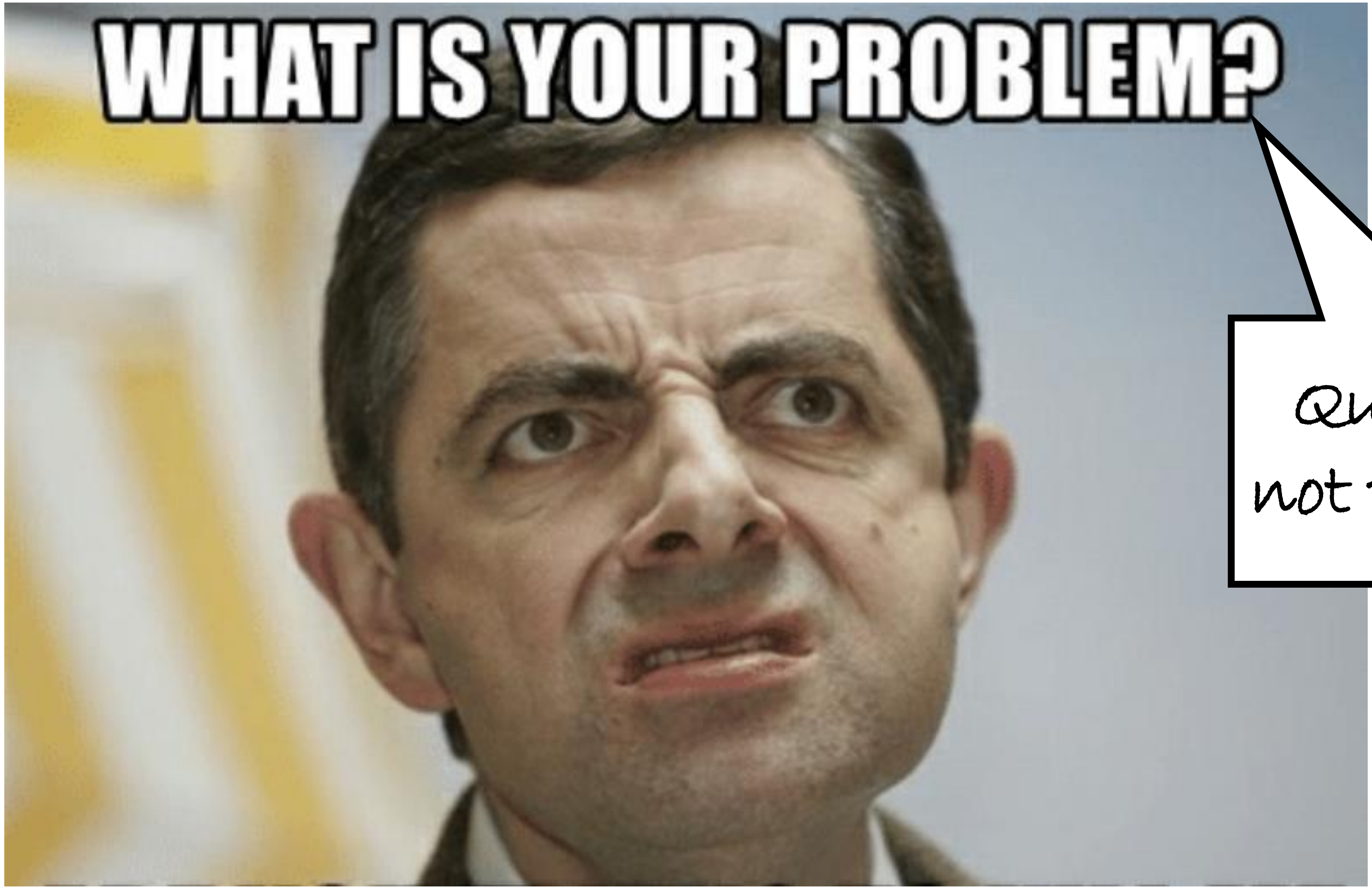
Tom is taller than Jerry ☑

$2 + 2 = 4$ ☑

$N > 6$ ☒

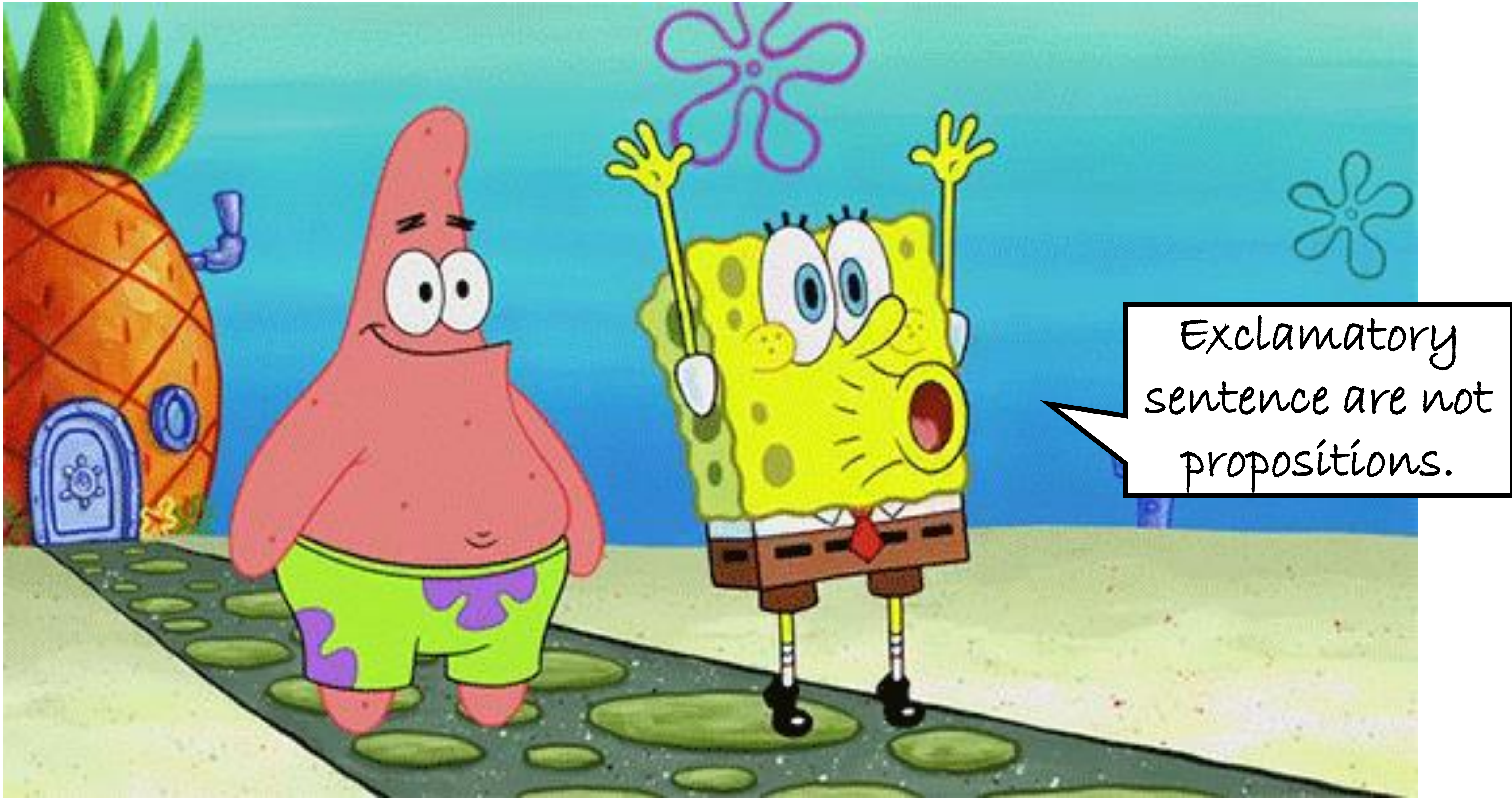
Discrete mathematics is an undergraduate course ☑

3 is odd ☑



Questions are not propositions.

Commands are not propositions.


Exclamatory sentence are not propositions.

Propositional Logic

Propositional logic (命题逻辑) is a mathematical system for reasoning about propositions and how they relate to one another.

To perform reasoning, propositional logic introduces ***propositional logic formula***

Propositional Logic Formula

Every formula in propositional logic consists of ***propositional variables*** combined via ***propositional connectives.***

Each variable represents some proposition, such as “Tom is a cat” or “Jerry is a rat.”

Connectives encode how propositions are related, such as “Tom is a cat ***and*** Jerry is a rat.”

Propositional Variables（命题变项）

A propositional variable is used to represent a simple proposition (简单命题)

Propositional variables are usually represented as upper-case letters

- For example, P and Q.
- We can use P to represent “Tom is taller than Jerry”

Every propositional variable has a truth value

- It is either true or false

Propositional Connectives（命题联结词）

Propositional Connectives are used to

- Connect propositional variables
- Express more complex meaning with existing propositions

P1: “Tom is a cat.”

P2: “Jerry is a rat.”

P3: “Tom and Jerry are enemy.”

⇨ If Tom is a cat and Jerry is a rat, then Tom and Jerry are enemy.

Atomic Proposition (原子命题)

Propositions without any connectives are called atomic propositions or simple propositions（原子命题/简单命题）

P1: “Tom is a cat.”

P2: “Jerry is a rat.”

P3: “Tom and Jerry are enemy.”

Atomic Propositions

If Tom is a cat and Jerry is a rat, then Tom and Jerry are enemy.

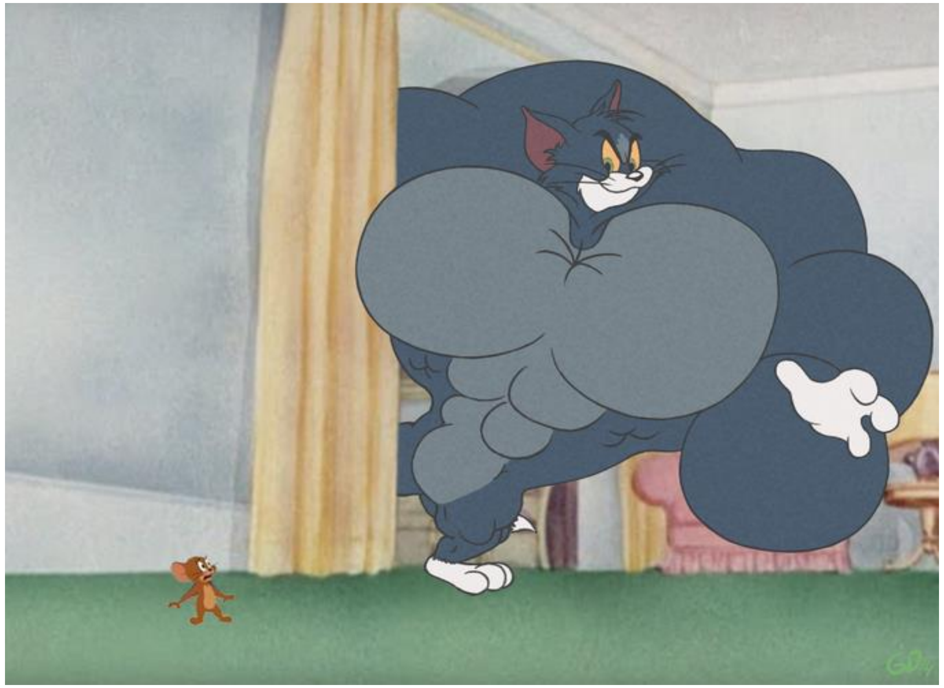
Proposition

NOT (Logical Negation)（否定词）

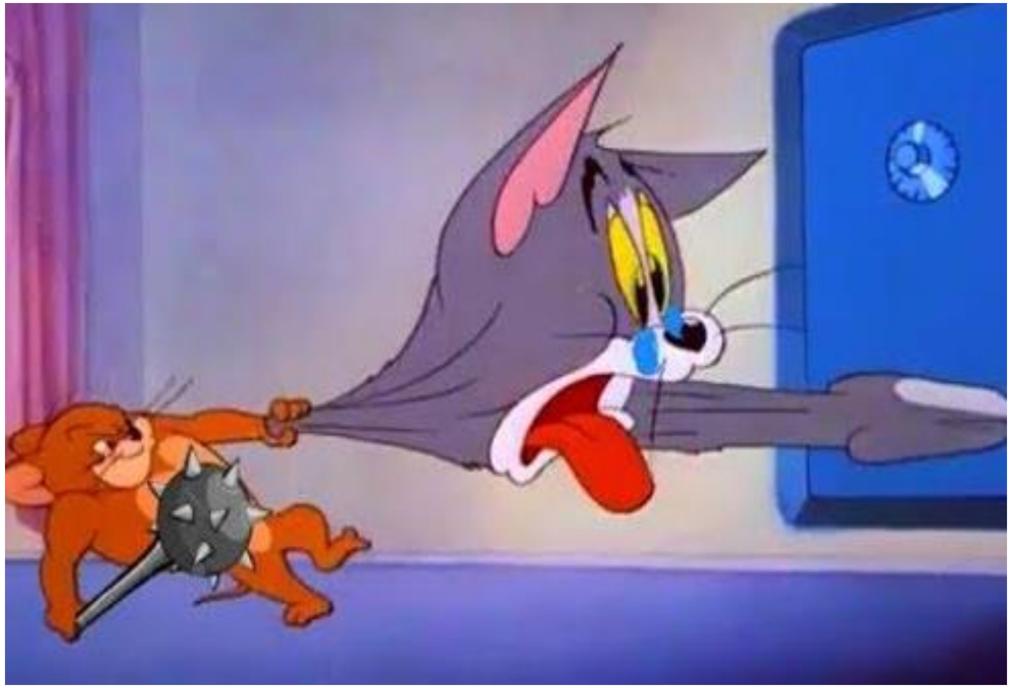
P : Tom is stronger than Jerry

$\neg P$: Tom is **NOT** stronger than Jerry

P



$\neg P$



Semantics of Connectives: Truth Table

A truth table (真值表) is a table

- showing the truth value of a propositional logic formula as a function of its inputs.
- describing the semantic of a propositional connective / formula.

The Semantics of “NOT”

Propositional connective

Propositional variable

P	$\neg P$
T	F
F	T

Propositional formula

Truth Table

$\neg \boldsymbol{P}$ always has a different truth value with P.

AND (Logical Conjunction)（合取词）

P： Tom is shaking hands with Jerry.

Q： Tom is shaking hands with Quacker.

$P \wedge Q$： Tom is shaking hands with Jerry **AND** Tom is shaking hands with Quacker.

$P \wedge Q$

The Semantics of “AND”

P	Q	$P \wedge Q$	$Q \wedge P$
T	T	T	T
T	F	F	F
F	T	F	F
F	F	F	F

Truth Table

They are the same.

$\boldsymbol{P \wedge Q}$ is true if P is true and Q is true.

OR (Logical Disjunction)（析取词）

P：Krusty Krab has burger.

Q：Krusty Krab has pizza.

$P \vee Q$：Krusty Krab has burger **OR** Krusty Krab has pizza.

$P \vee Q$

The Semantics of "OR"

P	Q	$P \vee Q$	$Q \vee P$
T	T	T	T
T	F	T	T
F	T	T	T
F	F	F	F

$\boldsymbol{P} \vee \boldsymbol{Q}$ is true if P is true or Q is true.

Implication（蕴涵词）

P : Jerry fires the firecracker.

Q : The firecracker will explode.

$P \rightarrow Q$: **If** Jerry fires the firecracker, **then (implies)** the firecracker will explode.

$P \rightarrow Q$

The Semantics of "Implication"

P	Q	$P \to Q$
T	T	T
T	F	F
F	T	T
F	F	T

The implication is only false if P is true and Q isn't.

Biconditional Connective（双条件词）

$\boldsymbol{P}$: SpongeBob is on the right of Patrick Star.

$\boldsymbol{Q}$: Patrick Star is on the left of SpongeBob.

$\boldsymbol{P} \leftrightarrow \boldsymbol{Q}$: SpongeBob is on the right of Patrick Star **if and only if** Patrick Star is on the left of SpongeBob.

$\boldsymbol{P} \leftrightarrow \boldsymbol{Q}$

The Semantics of "Biconditional Connective"

P	Q	$P \leftrightarrow Q$
T	T	T
T	F	F
F	T	F
F	F	T

The $P \leftrightarrow Q$ is only true if P and Q have the same truth value.

The Semantics of “Biconditional Connective”

P	Q	$P \leftrightarrow Q$
T	T	T
T	F	F
F	T	F
F	F	T

It seems $\boldsymbol{P} \leftrightarrow \boldsymbol{Q}$ means “P implies Q” and “Q implies P”. But can you prove it?

The Semantics of "Biconditional Connective"

P	Q	$P \leftrightarrow Q$	$(P \to Q) \wedge (Q \to P)$
T	T	T	T
T	F	F	F
F	T	F	F
F	F	T	T

They have the same truth values.

A Quick Recap

A propositional formula includes variables and connectives.

Propositional variables: P, Q …

Propositional connectives

Name	Symbol	#Variables	Example
NOT	$\neg$	1	$\neg P$
AND	$\wedge$	2	$P \wedge Q$
OR	$\vee$	2	$P \vee Q$
Implication	$\rightarrow$	2	$P \rightarrow Q$
Biconditional Connective	$\leftrightarrow$	2	$P \leftrightarrow Q$

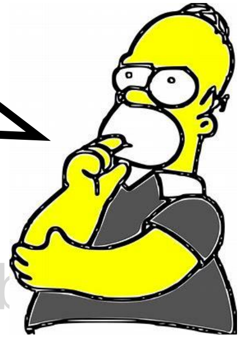
A Quick Recap

A p… …iables and connectives.

Pr…

Pr…

How to connect 3 or more variables?



Name	Symbol	#Variab…	Example
Biconditional Connective	↔	2	P ↔ Q

INTUITION

With multiple connectives.

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

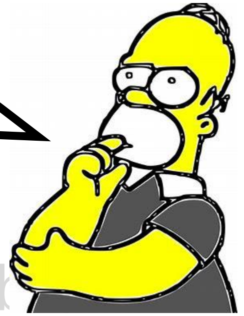
Well-Formed Formulas (合式公式)

A p… iables and connectives.

Pr…

Pr…

How to connect 3 or more variables legally?



Name	Symbol	#Varial…	Example
Im…			…Q
Biconditional Connective	↔	2	P ↔ Q

Well-Formed Formulas (合式公式)

If a formula is a well-formed formula (WFF), then it is legal.

Well-Formed Formulas (合式公式)

INDUCTIVE DEFINITION of WFF

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

A Quick Recap

A propositional formula includes variables and connectives.

Propositional variables: P, Q …

Propositional connectives: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$

WFF: construct the legal propositional formula.

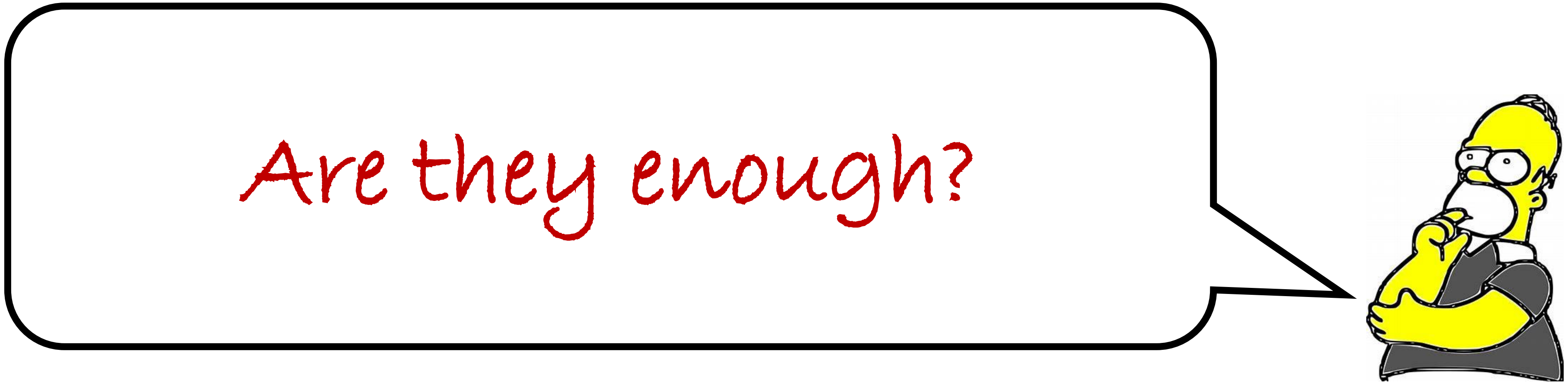
A Quick Recap

A propositional formula includes variables and connectives.

Propositional variables: P, Q …

Propositional connectives: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$

WFF: construct the legal propositional formula.



Can We Introduce New Connective?

We are able to define a new connective, and use truth table to describe its semantic.

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

Can We Introduce New Connective?

We are able to define a new connective, and use truth table to describe its semantic.

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

$$P \overline{\nabla} Q = ((P \wedge (\neg Q)) \vee ((\neg P) \wedge Q))$$

Because it can be represented with a WFF.

Completeness Of Connectives（联结词完备性）

DEFINITION

A group of connectives are **complete** if

1). Every formula can be converted to an equivalent formula and

2). The formula only includes connectives in the group.

Completeness Of Connectives（联结词完备性）

OBSERVATION

A formula with n variables can be treated to be a function:

1) The input is the truth values of n propositional variables.

2) The output is the truth value of the formula.

Completeness Of Connectives （联结词完备性）

OBSERVATION

A formula with n variables can be treated to be a function:

1) The input is the truth values of n propositional variables.

If every such function can be converted to a WFF, we can prove completeness.

Completeness Of Connectives （联结词完备性）

THEOREM

Let G be an n-place Boolean function, $n \geq 1$. There exists a *WFF* α such that α realizes the function G.

DEFINITIONS

n-place Boolean function: a function with n parameters and a Boolean return value.

α realizes the function G: α and G have the same truth values.

Completeness Of Connectives （联结词完备性）

THEOREM

Let G be an n-place Boolean function, $n \geq 1$. There exists a *WFF* α such that α realizes the function G.

How to reason it out?

n-place Boolean fu… …ean return value.

α realizes the function G: α and G have the same truth value

Completeness Of Connectives（联结词完备性）

THEOREM

Let G be an n-place Boolean function, $n \geq 1$. There exists a *WFF* α such that α realizes the function G.

Let's start with a simple case (i.e. $n = 2$).

n-place Boolean fu... ean return value.

α realizes the function G: α and G have the same truth value

Completeness Of Connectives
（联结词完备性）

How many connectives we can define when n is 2.

Completeness Of Connectives （联结词完备性）

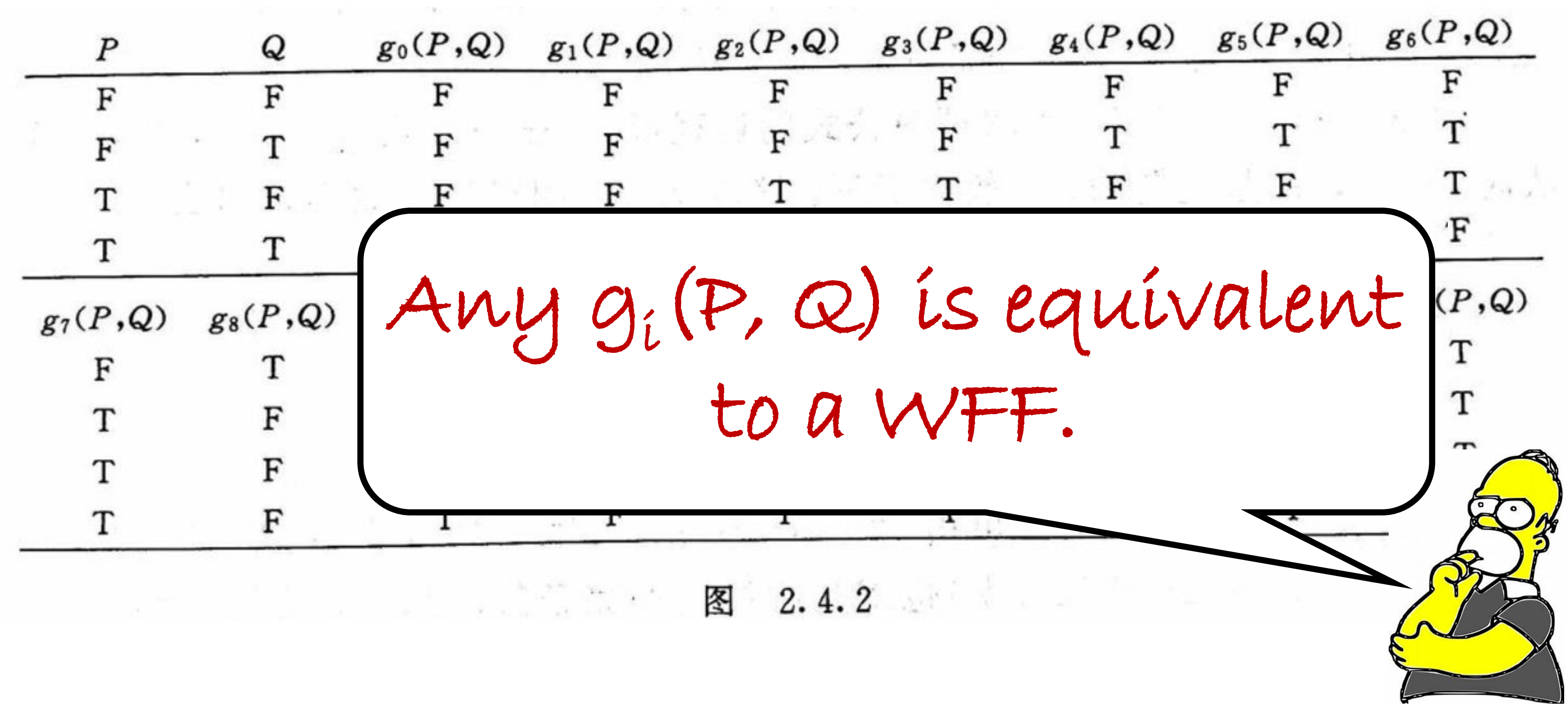


图 2.4.2

Completeness Of Connectives （联结词完备性）

P	Q	$g_0(P,Q)$	$g_1(P,Q)$	$g_2(P,Q)$	$g_3(P,Q)$	$g_4(P,Q)$	$g_5(P,Q)$	$g_6(P,Q)$
F	F	F	F	F	F	F	F	F
F	T	F	F	F	F	T	T	T
T	F	F	F	T	T	F	F	T
T	T	F	T	F	T	F	T	F

$g_7(P,Q)$	$g_8(P,Q)$	$g_9(P,Q)$	$g_{10}(P,Q)$	$g_{11}(P,Q)$	$g_{12}(P,Q)$	$g_{13}(P,Q)$	$g_{14}(P,Q)$	$g_{15}(P,Q)$
F	T	T	T	T	T	T	T	T
T	F	F	F	F	T	T	T	T
T	F	F	T	T	F	F	T	T
T	F	T	F	T	F	T	F	[illegible]

Exercise, What about g_{10} ?

Completeness Of Connectives
（联结词完备性）

P	Q	$g_0(P,Q)$	$g_1(P,Q)$	$g_2(P,Q)$	$g_3(P,Q)$	$g_4(P,Q)$	$g_5(P,Q)$	$g_6(P,Q)$
F	F	F	F	F	F	F	F	F
F	T	F	F	F	F	T	T	T
T	F	F	F	T	T	F	F	T
T	T	F	T	F	T	F	T	F

$g_7(P,Q)$	$g_8(P,Q)$	$g_9(P,Q)$	$g_{10}(P,Q)$	$g_{11}(P,Q)$	$g_{12}(P,Q)$	$g_{13}(P,Q)$	$g_{14}(P,Q)$	$g_{15}(P,Q)$
F	T	T	T	T	T	T	T	T
T	F	F	F	F	T	T	T	T
T	F	F	T	T	F	F	T	T
T	F	T	F	T	F	T	F	[illegible]

$$g_{10} = (\sim P \wedge \sim Q) \vee (P \wedge \sim Q)$$

Completeness Of Connectives（联结词完备性）

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

More exercises...

Completeness Of Connectives
（联结词完备性）

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

Can we derive general rules ???

Completeness Of Connectives

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	$\boxed{\text{T}}$	T
F	T	$\boxed{\text{T}}$	T
T	F	F	F
T	T	$\boxed{\text{T}}$	F

Step1: find all true rows

Completeness Of Connectives

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	$\boxed{T}$	T
F	T	$\boxed{T}$	T
T	F	F	F
T	T	$\boxed{T}$	F

Step1: find all true rows

Step2: generate a formula for every row

$(\neg P) \wedge (\neg Q)$ $\quad (\neg P) \wedge Q$ $\quad P \wedge Q$

Completeness Of Connectives

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	$\boxed{\text{T}}$	T
F	T	$\boxed{\text{T}}$	T
T	F	F	F
T	T	$\boxed{\text{T}}$	F

Step1: find all true rows

Step2: generate a formula for every row

Step3: use "or" to connect these formulas

$$g_0(P, Q) = (((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)) \vee (P \wedge Q)$$

Completeness Of Connectives

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	T	$\boxed{T}$
F	T	T	$\boxed{T}$
T	F	F	F
T	T	T	F

What about g1(P,Q)?

Step1: find all true rows

Step2: generate a formula for every row

Step3: use "or" to connect these formulas

$$g_0(P, Q) = (((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)) \vee (P \wedge Q)$$

$$g_1(P, Q) = ((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)$$

Completeness Of Connectives

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

Another algorithm

Completeness Of Connectives

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

Step1: find all false rows

Completeness Of Connectives

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

Step1: find all false rows

Step2: generate a formula for every row

$((\neg P) \vee Q)$ $\quad ((\neg P) \vee (\neg Q))$

Completeness Of Connectives

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	T	T
F	T	T	T
T	F	F	$\boxed{F}$
T	T	T	$\boxed{F}$

Step1: find all false rows

Step2: generate a formula for every row

Step3: use "and" to connect these formulas

$$g_1(P,Q) = ((\neg P) \vee Q) \wedge ((\neg P) \vee (\neg Q))$$

Completeness Of Connectives

P	Q	$g_0(P, Q)$	$g_1(P, Q)$
F	F	T	T
F	T	T	T
T	F	$\boxed{\text{F}}$	F
T	T	T	F

Step1: find all false rows

Step2: generate a formula for every row

Step3: use "and" to connect these formulas

What about g0(P,Q)?

$$g_1(P, Q) = ((\neg P) \vee Q) \wedge ((\neg P) \vee (\neg Q))$$

$$g_0(P, Q) = ((\neg P) \vee Q)$$

Completeness Of Connectives

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	$\boxed{\text{T}}$	$\boxed{\text{T}}$
F	T	$\boxed{\text{T}}$	$\boxed{\text{T}}$
T	F	F	F
T	T	$\boxed{\text{T}}$	F

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

$$g_0(P,Q) = (((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)) \vee (P \wedge Q)$$

$$g_1(P,Q) = ((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)$$

Completeness Of Connectives

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	T	T
F	T	T	T
T	F	F	F
T	T	T	F

$$g_0(P,Q) = (((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)) \vee (P \wedge Q)$$

$$g_1(P,Q) = ((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)$$

$$g_0(P,Q) = ((\neg P) \vee Q)$$

$$g_1(P,Q) = ((\neg P) \vee Q) \wedge ((\neg P) \vee (\neg Q))$$

Completeness Of Connectives

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	$\boxed{\text{T}}$	$\boxed{\text{T}}$
F	T	$\boxed{\text{T}}$	$\boxed{\text{T}}$
T	F	F	F
T	T	$\boxed{\text{T}}$	F

$$g_0(P,Q) = (((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)) \vee (P \wedge Q)$$

$$g_1(P,Q) = ((\neg P) \wedge (\neg Q)) \vee ((\neg P) \wedge Q)$$

M1. According to the T rows

P	Q	$g_0(P,Q)$	$g_1(P,Q)$
F	F	T	T
F	T	T	T
T	F	$\boxed{\text{F}}$	$\boxed{\text{F}}$
T	T	T	$\boxed{\text{F}}$

$$g_0(P,Q) = ((\neg P) \vee Q)$$

$$g_1(P,Q) = ((\neg P) \vee Q) \wedge ((\neg P) \vee (\neg Q))$$

M2. According to the F rows

Completeness Of Connectives （联结词完备性）

THEOREM

Let G be an n-place Boolean function, $n \geq 1$. There exists a *WFF* α such that α realizes the function G.

DEFINITIONS

n-… arameters and a Boolean

ret…

α re… same t… lues.

We now give a proof

Completeness Of Connectives（联结词完备性）

THEOREM

Let G be an n-place Boolean function, $n \geq 1$. There exists a *WFF* α such that α realizes the function G.

Proof

If G always return F, then $\alpha = P \wedge (\neg P)$. It is clear α realizes the function G.

Otherwise, G returns true sometimes. Suppose there are k cases where G returns true.

Completeness Of Connectives

Proof

There are k inputs that can make G return true:

G($X_{11}, X_{12}, \ldots, X_{1n}$) = T

G($X_{21}, X_{22}, \ldots, X_{2n}$) = T

…

G($X_{k1}, X_{k2}, \ldots, X_{kn}$) = T

Then we can construct α in the following way.

$$\beta_{ij} = \begin{cases} P_j & if\ \ X_{ij} = T \\ \neg P_j & if\ \ X_{ij} = F \end{cases}$$

$$\gamma_i = \beta_{i1} \land \cdots \land \beta_{in}$$

$$\alpha = \gamma_1 \lor \cdots \lor \gamma_k$$

Looks like algorithm above

Completeness Of Connectives

Proof

When G returns true, it is clear that α is true.
When α is true, there must be some γ_i is true.
There is only one assignment that can make γ_i is true.
Under this assignment, G returns true.

G($X_{11}, X_{12}, \dots, X_{1n}$) = T
G($X_{21}, X_{22}, \dots, X_{2n}$) = T
…
G($X_{k1}, X_{k2}, \dots, X_{kn}$) = T

$$\beta_{ij} = \begin{cases} P_j & if \ \ X_{ij} = T \\ \neg P_j & if \ \ X_{ij} = F \end{cases}$$

$$\gamma_i = \beta_{i1} \wedge \cdots \wedge \beta_{in}$$

$$\alpha = \gamma_1 \vee \cdots \vee \gamma_k$$

A Quick Recap

A propositional formula includes variables and connectives.

Propositional variables: P, Q …

Propositional connectives: $\neg$, $\wedge$, $\vee$, $\rightarrow$, $\leftrightarrow$

WFF: construct the legal propositional formula.

Are they optimal???

Completeness Of Connectives

We already know that

$$P \leftrightarrow Q = (P \rightarrow Q) \wedge (Q \rightarrow P)$$

The group becomes {¬, ∧, ∨, →}

What about "→" ?

Completeness Of Connectives

P	Q	$P \to Q$	$(\neg P) \vee Q$
T	T	T	T
T	F	F	F
F	T	T	T
F	F	T	T

Implication can be expressed by "not" and "or"

The group becomes $\{\neg, \wedge, \vee\}$

Completeness Of Connectives

P	Q	$P \wedge Q$	$\neg((\neg P) \vee (\neg Q))$
T	T	T	T
T	F	F	F
F	T	F	F
F	F	F	F

"and" can be expressed by "not" and "or"

P	Q	$P \vee Q$	$\neg((\neg P) \wedge (\neg Q))$
T	T	T	T
T	F	T	T
F	T	T	T
F	F	F	F

"or" can be expressed by "not" and "and"

Finally, $\{\neg, \wedge\}$ and $\{\neg, \vee\}$ are all complete
We cannot eliminate elements from them.

Now, can you use the propositional logic to formalize the "2 queens problem theorem?"

Example

P1	P2
P3	P4

Theorem: Two queens problem has no solution.

REPHRASE

2 queens problem: placing 2 chess queens on an 2×2 chessboard so that no two queens share the same row, column or diagonal.

How to formalize the problem has no solution?

Example

P1	P2
P3	P4

Theorem: Two queens problem has no solution.

REPHRASE

2 queens problem: placing 2 chess queens on an 2×2 chessboard so that no two queens share the same row, column or diagonal.

There are four positions on the chessboard. We now define four propositional variables $P1$, $P2$, $P3$ and $P4$.

$P1$ signifies that there is a queen on position 1. $\neg P1$ signifies there is not a queen on position 1 and so on.

Example

P1	P2
P3	P4

Theorem: Two queens problem has no solution.

REPHRASE

2 queens problem: placing 2 chess queens on an 2×2 chessboard so that no two queens share the same row, column or diagonal.

First, we formalize the requirement:

$$(((((\neg(P1 \wedge P2)) \wedge (\neg(P1 \wedge P3)))\wedge$$
$$(\neg(P1 \wedge P4))) \wedge (\neg(P2 \wedge P3)))\wedge$$
$$(\neg(P2 \wedge P4))) \wedge (\neg(P3 \wedge P4)))$$

Example

P1	P2
P3	P4

Theorem: Two queens problem has no solution.

REPHRASE

2 queens problem: placing 2 chess queens on an 2×2 chessboard so that no two queens share the same row, column or diagonal.

Second, we formalize the placement:

$$\Big(\big((P1 \wedge P2) \wedge (\neg P3)\big) \wedge (\neg P4)\Big) \vee \Big(\big((P1 \wedge P3) \wedge (\neg P2)\big) \wedge (\neg P4)\Big) \vee$$
$$\Big(\big((P1 \wedge P4) \wedge (\neg P2)\big) \wedge (\neg P3)\Big) \vee \Big(\big((P2 \wedge P3) \wedge (\neg P1)\big) \wedge (\neg P4)\Big) \vee$$
$$\Big(\big((P2 \wedge P4) \wedge (\neg P3)\big) \wedge (\neg P1)\Big) \vee \Big(\big((P3 \wedge P4) \wedge (\neg P1)\big) \wedge (\neg P2)\Big)$$

Example

P1	P2
P3	P4

Theorem: Two queens problem has no solution.

REPHRASE

2 queens problem: placing 2 chess queens on an 2×2 chessboard so that no two queens share the same row, column or diagonal.

$$\left[\begin{array}{l}(((((\neg(P1 \wedge P2)) \wedge (\neg(P1 \wedge P3)))\wedge \\ (\neg(P1 \wedge P4))) \wedge (\neg(P2 \wedge P3)))\wedge \\ (\neg(P2 \wedge P4))) \wedge (\neg(P3 \wedge P4)))\end{array} \wedge \begin{array}{l}(((P1 \wedge P2) \wedge (\neg P3)) \wedge (\neg P4))\vee \\ (((P1 \wedge (\neg P2)) \wedge P3) \wedge (\neg P4))\vee \\ (((P1 \wedge (\neg P2)) \wedge (\neg P3)) \wedge P4)\vee \\ ((((\neg P1) \wedge P2) \wedge P3) \wedge (\neg P4))\vee \\ ((((\neg P1) \wedge P2) \wedge (\neg P3)) \wedge P4)\vee \\ ((((\neg P1) \wedge (\neg P2)) \wedge P3) \wedge P4)\end{array}\right] \leftrightarrow F$$

$$\left(\begin{array}{l} (((((\neg(P1 \land P2)) \land (\neg(P1 \land P3)))\land \\ (\neg(P1 \land P4))) \land (\neg(P2 \land P3)))\land \\ (\neg(P2 \land P4))) \land (\neg(P3 \land P4))) \end{array} \land \begin{array}{l} (((P1 \land P2) \land (\neg P3)) \land (\neg P4))\lor \\ (((P1 \land (\neg P2)) \land P3) \land (\neg P4))\lor \\ (((P1 \land (\neg P2)) \land (\neg P3)) \land P4)\lor \\ ((((\neg P1) \land P2) \land P3) \land (\neg P4))\lor \\ ((((\neg P1) \land P2) \land (\neg P3)) \land P4)\lor \\ ((((\neg P1) \land (\neg P2)) \land P3) \land P4) \end{array} \right) \leftrightarrow F$$

How to prove it?

$$\left(\begin{array}{l}(((((\neg(P1 \land P2)) \land (\neg(P1 \land P3)))\land \\ (\neg(P1 \land P4))) \land (\neg(P2 \land P3)))\land \\ (\neg(P2 \land P4))) \land (\neg(P3 \land P4)))\end{array} \quad \land \quad \begin{array}{l}(((P1 \land P2) \land (\neg P3)) \land (\neg P4))\lor \\ (((P1 \land (\neg P2)) \land P3) \land (\neg P4))\lor \\ (((P1 \land (\neg P2)) \land (\neg P3)) \land P4)\lor \\ ((((\neg P1) \land P2) \land P3) \land (\neg P4))\lor \\ ((((\neg P1) \land P2) \land (\neg P3)) \land P4)\lor \\ ((((\neg P1) \land (\neg P2)) \land P3) \land P4)\end{array}\right) \leftrightarrow F$$

How to prove it?
Truth table.

Truth Table

There are 4 propositional variables.

P_1	...	P_4	$g(P_1, P_2, P_3, P_4)$
T	...	T	T
...	...	...	...
...	...	...	...
F	...	F	T

??? ROWS

Truth Table

There are 4 propositional variables.

P_1	...	P_4	$g(P_1, P_2, P_3, P_4)$
T	...	T	T
...	...	...	...
...	...	...	...
F	...	F	T

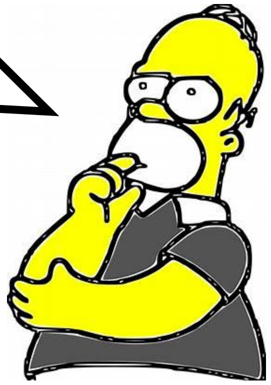
16 ROWS

If there are n variables, how many rows in the truth table?

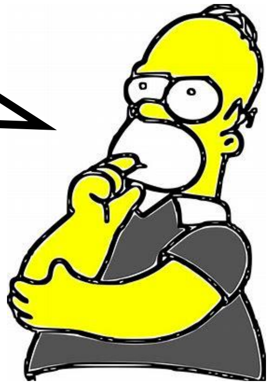
$\mathbf{2^n}$!!!

What will affect the computation cost?

The computation cost depends on #variables and the complexity of the formula.



Can we simplify the formula???



$$\left[\begin{array}{l}(((((\neg(P1 \wedge P2)) \wedge (\neg(P1 \wedge P3)))\wedge \\ (\neg(P1 \wedge P4))) \wedge (\neg(P2 \wedge P3)))\wedge \\ (\neg(P2 \wedge P4))) \wedge (\neg(P3 \wedge P4)))\end{array} \wedge \begin{array}{l}(((P1 \wedge P2) \wedge (\neg P3)) \wedge (\neg P4))\vee \\ (((P1 \wedge (\neg P2)) \wedge P3) \wedge (\neg P4))\vee \\ (((P1 \wedge (\neg P2)) \wedge (\neg P3)) \wedge P4)\vee \\ ((((\neg P1) \wedge P2) \wedge P3) \wedge (\neg P4))\vee \\ ((((\neg P1) \wedge P2) \wedge (\neg P3)) \wedge P4)\vee \\ ((((\neg P1) \wedge (\neg P2)) \wedge P3) \wedge P4)\end{array}\right] \leftrightarrow F$$

Can we eliminate them firstly
without affecting the semantic?

Operator Precedence

$$((P \vee Q) \wedge R) \quad \textbf{F}$$

$$P \vee Q \wedge R$$

$$P = T, Q = F, R = F$$

$$(P \vee (Q \wedge R)) \quad \textbf{T}$$

Operator Precedence

P1	P2
P3	P4

Recap: what does $\neg(P1 \land P2)$ mean in two queens problem?

Two queens are not on the first row at the same time

But, what does $\neg P1 \land P2$ mean?

Operator Precedence

We introduce operator precedence like that in C, C++ and python

$$\neg \quad > \quad \wedge \quad > \quad \vee \quad > \quad \rightarrow \quad > \quad \leftrightarrow$$

$$\neg P \wedge Q \iff (\neg P) \wedge Q$$

$$P \vee Q \wedge R \iff P \vee (Q \wedge R)$$

Operator Precedence

Only operator precedence is not enough

$$P \to Q \to R$$

$\to$

$\to$

Operator Precedence

The real meaning is

$$P \rightarrow (Q \rightarrow R)$$

$\rightarrow$ (

$\rightarrow$

)

Operator Precedence

But there is can be a different interpretation

$(P \to Q) \to R$

Operator Precedence

Left-associative（左结合）

✓ The same precedence are evaluated in order from left to right.

$$P \rightarrow Q \rightarrow R$$

Operator Precedence

With operator precedence and association, we cannot write the following formula with less parentheses.

 $\rightarrow$ ( $\rightarrow$ )

Exercise

How to parse this statement?

$$\neg X \rightarrow Y \vee Z \rightarrow X \vee Y \wedge Z$$

Operator precedence

$$\neg \quad > \quad \wedge \quad > \quad \vee \quad > \quad \rightarrow \quad > \quad \leftrightarrow$$

All operators are left-associative

Exercise

How to parse this statement?

$$(\neg X) \rightarrow Y \vee Z \rightarrow X \vee Y \wedge Z$$

Operator precedence

$$\neg \quad > \quad \wedge \quad > \quad \vee \quad > \quad \rightarrow \quad > \quad \leftrightarrow$$

All operators are left-associative

Exercise

How to parse this statement?

$$(\neg X) \to Y \vee Z \to X \vee (Y \wedge Z)$$

Operator precedence

$$\neg \;>\; \wedge \;>\; \vee \;>\; \to \;>\; \leftrightarrow$$

All operators are left-associative

Exercise

How to parse this statement?

$$(\neg X) \rightarrow (Y \vee Z) \rightarrow (X \vee (Y \wedge Z))$$

Operator precedence

$$\neg \;>\; \wedge \;>\; \vee \;>\; \rightarrow \;>\; \leftrightarrow$$

All operators are left-associative

Exercise

How to parse this statement?

$$((\neg X) \to (Y \vee Z)) \to (X \vee (Y \wedge Z))$$

Operator precedence

$$\neg \quad > \quad \wedge \quad > \quad \vee \quad > \quad \to \quad > \quad \leftrightarrow$$

All operators are left-associative

$$\left[\begin{array}{lcl} & & (((P1 \wedge P2) \wedge (\neg P3)) \wedge (\neg P4)) \vee \\ (((((\neg(P1 \wedge P2)) \wedge (\neg(P1 \wedge P3))) \wedge & & (((P1 \wedge (\neg P2)) \wedge P3) \wedge (\neg P4)) \vee \\ (\neg(P1 \wedge P4))) \wedge (\neg(P2 \wedge P3))) \wedge & \wedge & (((P1 \wedge (\neg P2)) \wedge (\neg P3)) \wedge P4) \vee \\ (\neg(P2 \wedge P4))) \wedge (\neg(P3 \wedge P4))) & & ((((\neg P1) \wedge P2) \wedge P3) \wedge (\neg P4)) \vee \\ & & ((((\neg P1) \wedge P2) \wedge (\neg P3)) \wedge P4) \vee \\ & & ((((\neg P1) \wedge (\neg P2)) \wedge P3) \wedge P4) \end{array}\right] \leftrightarrow F$$

Which parentheses can be eliminated?

$$\left[\begin{array}{l} (((((\neg(P1 \wedge P2)) \wedge (\neg(P1 \wedge P3)))\wedge \\ (\neg(P1 \wedge P4))) \wedge (\neg(P2 \wedge P3)))\wedge \\ (\neg(P2 \wedge P4))) \wedge (\neg(P3 \wedge P4))) \end{array} \wedge \begin{array}{l} (((P1 \wedge P2) \wedge (\neg P3)) \wedge (\neg P4))\vee \\ (((P1 \wedge (\neg P2)) \wedge P3) \wedge (\neg P4))\vee \\ (((P1 \wedge (\neg P2)) \wedge (\neg P3)) \wedge P4)\vee \\ ((((\neg P1) \wedge P2) \wedge P3) \wedge (\neg P4))\vee \\ ((((\neg P1) \wedge P2) \wedge (\neg P3)) \wedge P4)\vee \\ ((((\neg P1) \wedge (\neg P2)) \wedge P3) \wedge P4) \end{array} \right] \leftrightarrow F$$

$$(((P1 \wedge P2) \wedge (\neg P3)) \wedge (\neg P4)) \Rightarrow P1 \wedge P2 \wedge \neg P3 \wedge \neg P4$$

What about this? $\neg(P1 \wedge P2)$

Propositional Equivalences（等值）

Theorem

Propositional Equivalences（等值）： If two formula P and Q has the same truth value under any assignment, they are equivalent. It is denoted by $P = Q$ or $P \Leftrightarrow Q$.

Equivalence Theorem（等值定理）： $P = Q$ iff $P \leftrightarrow Q$ is always true.

De Morgan‘s Laws（摩根律）

Theorem

De Morgan’s Laws:

$$\neg(P \wedge Q) = \neg P \vee \neg Q, \neg(P \vee Q) = \neg P \wedge \neg Q$$

$\neg(P1 \wedge P2)$ can be replaced by $\neg P1 \vee \neg P2$.

More Laws

Theorem

Double Negation（双重否定律）：

$$\neg\neg P = P$$

More Laws

P	Q	$P \to Q$
T	T	T
T	F	$\boxed{\text{F}}$
F	T	T
F	F	T

According to F rows, $P \longrightarrow Q = \neg P \vee Q$

More Laws

Theorem

Associative Law（结合律）：

$$(P \vee Q) \vee R = P \vee (Q \vee R)$$
$$(P \wedge Q) \wedge R = P \wedge (Q \wedge R)$$
$$(P \leftrightarrow Q) \leftrightarrow R = P \leftrightarrow (Q \leftrightarrow R)$$

$$(P \to Q) \to R \neq P \to (Q \to R)$$

Warning!

More Laws

Theorem

Commutative Law（交换律）：

$$P \vee Q = Q \vee P$$
$$P \wedge Q = Q \wedge P$$
$$P \leftrightarrow Q = Q \leftrightarrow P$$

$P \to Q = Q \to R$?

It is wrong. Can you give a counterexample?

More Laws

Theorem

Commutative Law（交换律）：

$$P \vee Q = Q \vee P$$
$$P \wedge Q = Q \wedge P$$
$$P \leftrightarrow Q = Q \leftrightarrow P$$

$$P = T, Q = F$$

$$\underbrace{\boxed{P \rightarrow Q}}_{\text{F}} \neq \underbrace{\boxed{Q \rightarrow R}}_{\text{T}}$$

More Laws

Theorem

Idempotent Law（等幂率）：

$$P \vee P = P$$
$$P \wedge P = P$$
$$P \leftrightarrow P = T$$
$$P \to P = T$$

Theorem

Identity Law（同一律）：

$$P \vee F = P$$
$$P \wedge T = P$$
$$T \to P = P$$
$$T \leftrightarrow P = P$$
$$P \to F = \neg P$$
$$F \leftrightarrow P = \neg P$$

More Laws

Theorem

Complementary Law（补余律）：

$$P \vee \neg P = T$$
$$P \wedge \neg P = F$$
$$P \to \neg P = \neg P$$
$$\neg P \to P = P$$
$$P \leftrightarrow \neg P = F$$

Zero Law （零律）：

$$P \vee T = T$$
$$P \wedge F = F$$
$$P \to T = T$$
$$F \to P = T$$

More Laws

Theorem

Distributive Law（分配律）：

$$P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$$
$$P \wedge (Q \vee R) = (P \wedge Q) \vee (P \wedge R)$$

Absorption Law（吸收律）：

$$P \vee (P \wedge Q) = P$$
$$P \wedge (P \vee Q) = P$$

$$P \wedge (Q_1 \vee Q_2 \vee \ldots \vee Q_n) = ?$$

More Laws

Theorem

Distributive Law（分配律）：

$$P \vee (Q \wedge R) = (P \vee Q) \wedge (P \vee R)$$
$$P \wedge (Q \vee R) = (P \wedge Q) \vee (P \wedge R)$$

Absorption Law（吸收律）：

$$P \vee (P \wedge Q) = P$$
$$P \wedge (P \vee Q) = P$$

$$P \wedge (Q_1 \vee Q_2 \vee \ldots \vee Q_n) = (P \wedge Q_1) \vee (P \wedge Q_2) \vee \ldots \vee (P \wedge Q_n)$$

Example

$$
\begin{array}{l}
(\\
\quad (((P1 \land P2) \land (\neg P3)) \land (\neg P4)) \lor \\
\quad (((P1 \land (\neg P2)) \land P3) \land (\neg P4)) \lor \\
\quad (((P1 \land (\neg P2)) \land (\neg P3)) \land P4) \lor \\
\quad ((((\neg P1) \land P2) \land P3) \land (\neg P4)) \lor \\
\quad ((((\neg P1) \land P2) \land (\neg P3)) \land P4) \lor \\
\quad ((((\neg P1) \land (\neg P2)) \land P3) \land P4) \\
) \land \\
(\ (((((\neg (P1 \land P2)) \land (\neg (P1 \land P3))) \land \\
\quad (\neg (P1 \land P4))) \land (\neg (P2 \land P3))) \land \\
\quad (\neg (P2 \land P4))) \land (\neg (P3 \land P4))) \\
\ \leftrightarrow F
\end{array}
$$

Eliminate redundant parentheses

Example

$$
\begin{aligned}
&(\\
&\quad (P1 \wedge P2 \wedge \neg P3 \wedge \neg P4) \vee \\
&\quad (P1 \wedge \neg P2 \wedge P3 \wedge \neg P4) \vee \\
&\quad (P1 \wedge \neg P2 \wedge \neg P3 \wedge P4) \vee \\
&\quad (\neg P1 \wedge P2 \wedge P3 \wedge \neg P4) \vee \\
&\quad (\neg P1 \wedge P2 \wedge \neg P3 \wedge P4) \vee \\
&\quad (\neg P1 \wedge \neg P2 \wedge P3 \wedge P4) \\
&) \wedge \\
&\quad \neg(P1 \wedge P2) \wedge \neg(P1 \wedge P3) \wedge \\
&\quad \neg(P1 \wedge P4) \wedge \neg(P2 \wedge P3) \wedge \\
&\quad \neg(P2 \wedge P4) \wedge \neg(P3 \wedge P4) \\
&\leftrightarrow F
\end{aligned}
$$

We can Eliminate more Parentheses. But we retain them for readability.

Example

$$
\begin{aligned}
&(\\
&\quad (P1 \wedge P2 \wedge \neg P3 \wedge \neg P4) \vee \\
&\quad (P1 \wedge \neg P2 \wedge P3 \wedge \neg P4) \vee \\
&\quad (P1 \wedge \neg P2 \wedge \neg P3 \wedge P4) \vee \\
&\quad (\neg P1 \wedge P2 \wedge P3 \wedge \neg P4) \vee \\
&\quad (\neg P1 \wedge P2 \wedge \neg P3 \wedge P4) \vee \\
&\quad (\neg P1 \wedge \neg P2 \wedge P3 \wedge P4) \\
&) \wedge \\
&\ \neg(P1 \wedge P2) \wedge \neg(P1 \wedge P3) \wedge \\
&\quad \neg(P1 \wedge P4) \wedge \neg(P2 \wedge P3) \wedge \\
&\quad \neg(P2 \wedge P4) \wedge \neg(P3 \wedge P4) \\
&\ \leftrightarrow F
\end{aligned}
$$

Now we can use laws to convert the formula on the left to F. Try to use distributive law first.

Example

$$
\begin{aligned}
&\boxed{(P1 \land P2)} \land \neg P3 \land \neg P4 \land \\
&\boxed{\neg(P1 \land P2)} \land \neg(P1 \land P3) \land \\
&\neg(P1 \land P4) \land \neg(P2 \land P3) \land \\
&\neg(P2 \land P4) \land \neg(P3 \land P4) \lor \\
&\ldots\ldots \lor \\
&\neg P1 \land \neg P2 \land \boxed{(P3 \land P4)} \land \\
&\neg(P1 \land P2) \land \neg(P1 \land P3) \land \\
&\neg(P1 \land P4) \land \neg(P2 \land P3) \land \\
&\neg(P2 \land P4) \land \boxed{\neg(P3 \land P4)}
\end{aligned}
$$

Now commutative law
and associative law.

Example

$$\boxed{(P1 \land P2) \land \neg(P1 \land P2)} \land$$
$$\neg P3 \land \neg P4 \land \neg(P1 \land P3) \land$$
$$\neg(P1 \land P4) \land \neg(P2 \land P3) \land$$
$$\neg(P2 \land P4) \land \neg(P3 \land P4) \lor$$
$$\ldots\ldots \lor$$
$$\boxed{\neg(P3 \land P4) \land (P3 \land P4)} \land$$
$$\neg(P1 \land P2) \land \neg(P1 \land P3) \land$$
$$\neg(P1 \land P4) \land \neg(P2 \land P3) \land$$
$$\neg(P2 \land P4) \land \neg P1 \land \neg P2$$

Now Complementary law.

Example

$$F \vee F \vee F \vee F \vee F \vee F$$

$$\Downarrow$$

$$F$$

$$F \leftrightarrow F$$

We have successfully convert the left to F.

Exercise

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

Can you prove it by laws instead of truth table?

Exercise

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

$$\begin{aligned}
& P \to Q \\
=& (\neg P) \vee Q \\
=& Q \vee (\neg P) \\
=& (\neg(\neg Q)) \vee (\neg P) \\
=& (\neg Q) \to (\neg P)
\end{aligned}$$

Exercise

$$P \rightarrow (Q \rightarrow R) = Q \rightarrow (P \rightarrow R)$$

Can you prove it by laws
instead of truth table?

Exercise

$$P \to (Q \to R) = Q \to (P \to R)$$

$$\begin{aligned}
& P \to (Q \to R) \\
=& (\neg P) \vee (Q \to R) \\
=& (\neg P) \vee ((\neg Q) \vee R) \\
=& (\neg Q) \vee ((\neg P) \vee R) \\
=& Q \to ((\neg P) \vee R) \\
=& Q \to (P \to R)
\end{aligned}$$

Exercise

$$P \to (Q \to R) = (P \land Q) \to R$$

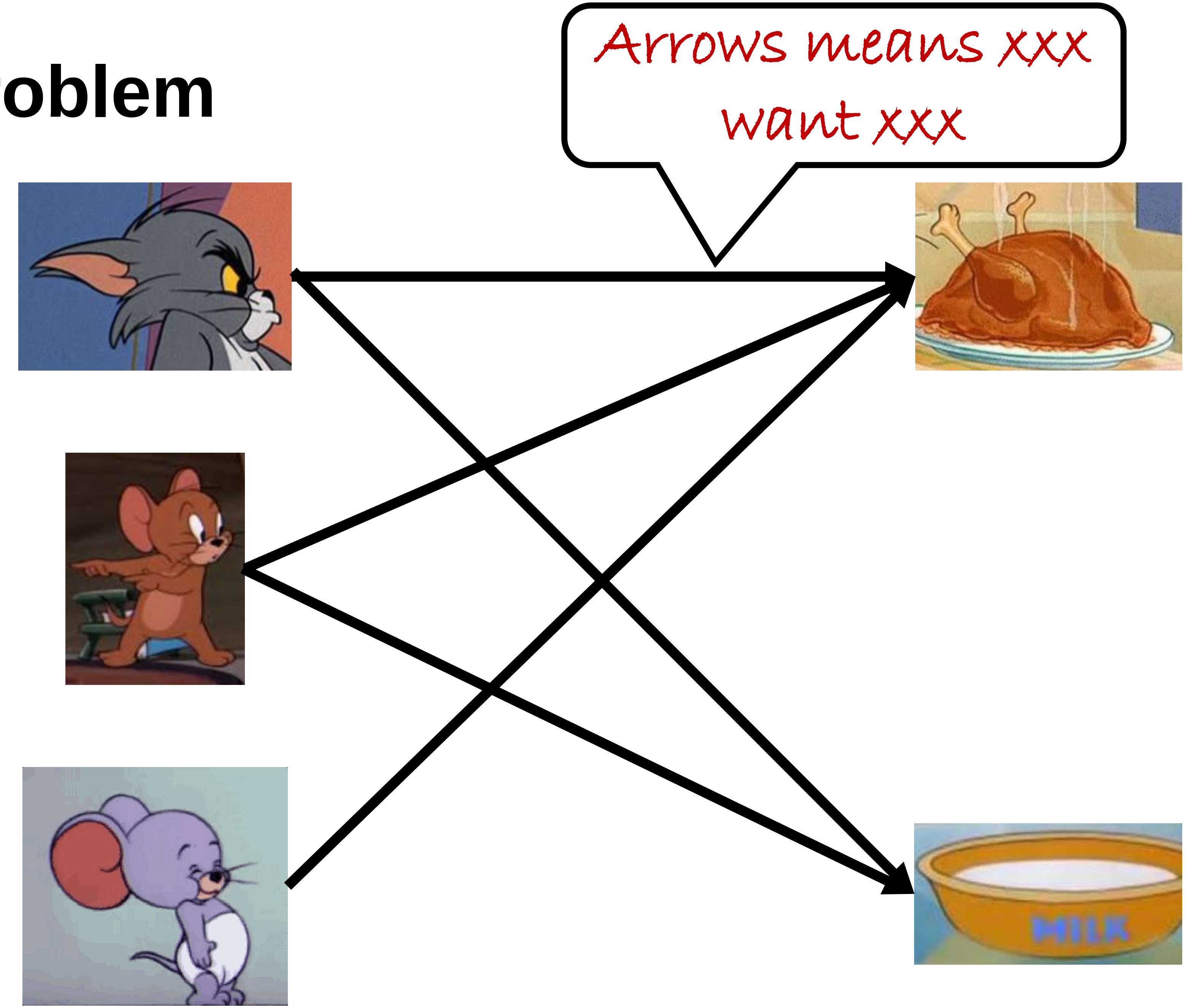
Can you prove it by laws instead of truth table?

Exercise

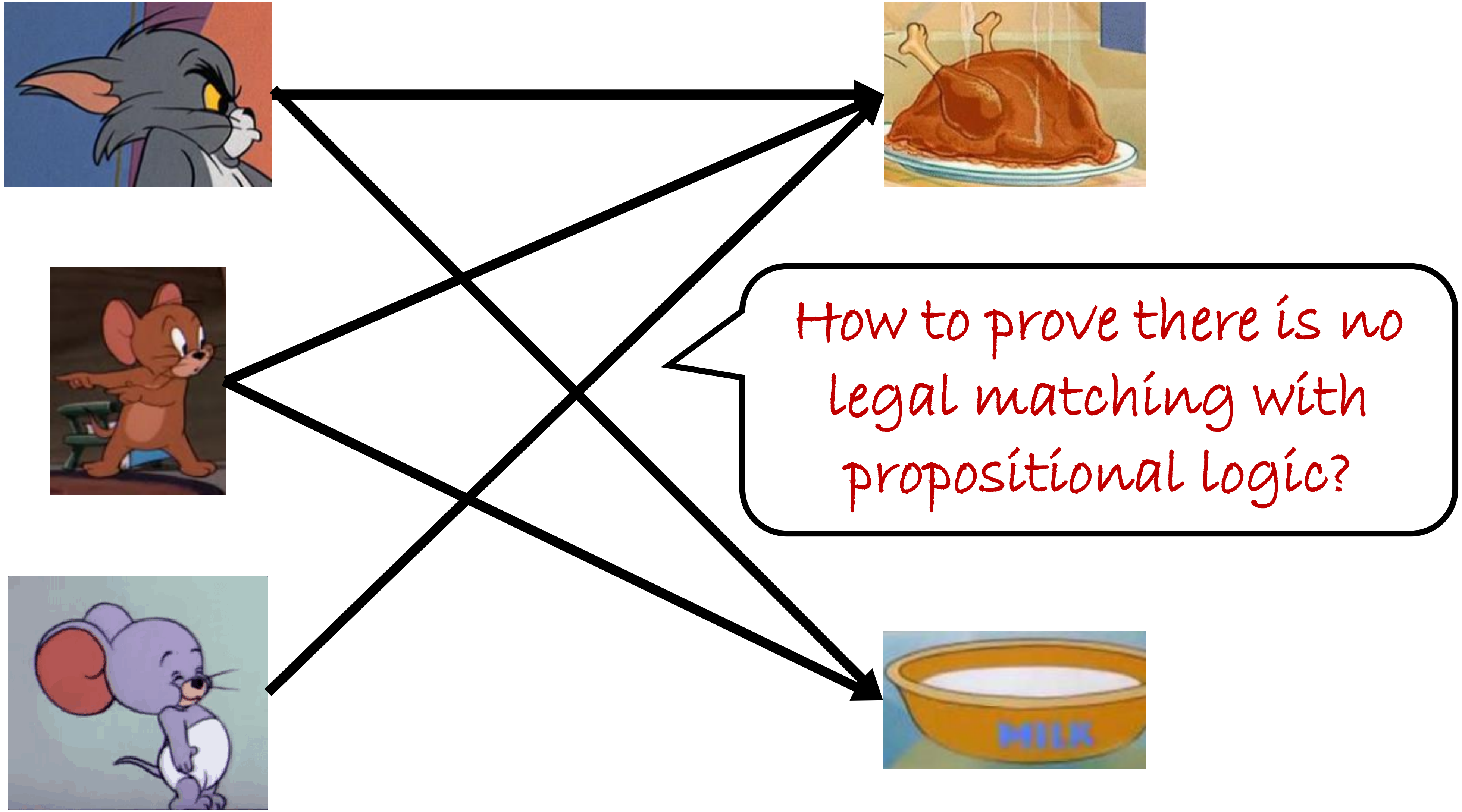
$$P \to (Q \to R) = (P \land Q) \to R$$

$$\begin{aligned}
&P \to (Q \to R) \\
=&(\neg P) \lor (Q \to R) \\
=&(\neg P) \lor ((\neg Q) \lor R) \\
=&((\neg P) \lor (\neg Q)) \lor R \\
=&(\neg (P \land Q)) \lor R \\
=&(P \land Q) \to R
\end{aligned}$$

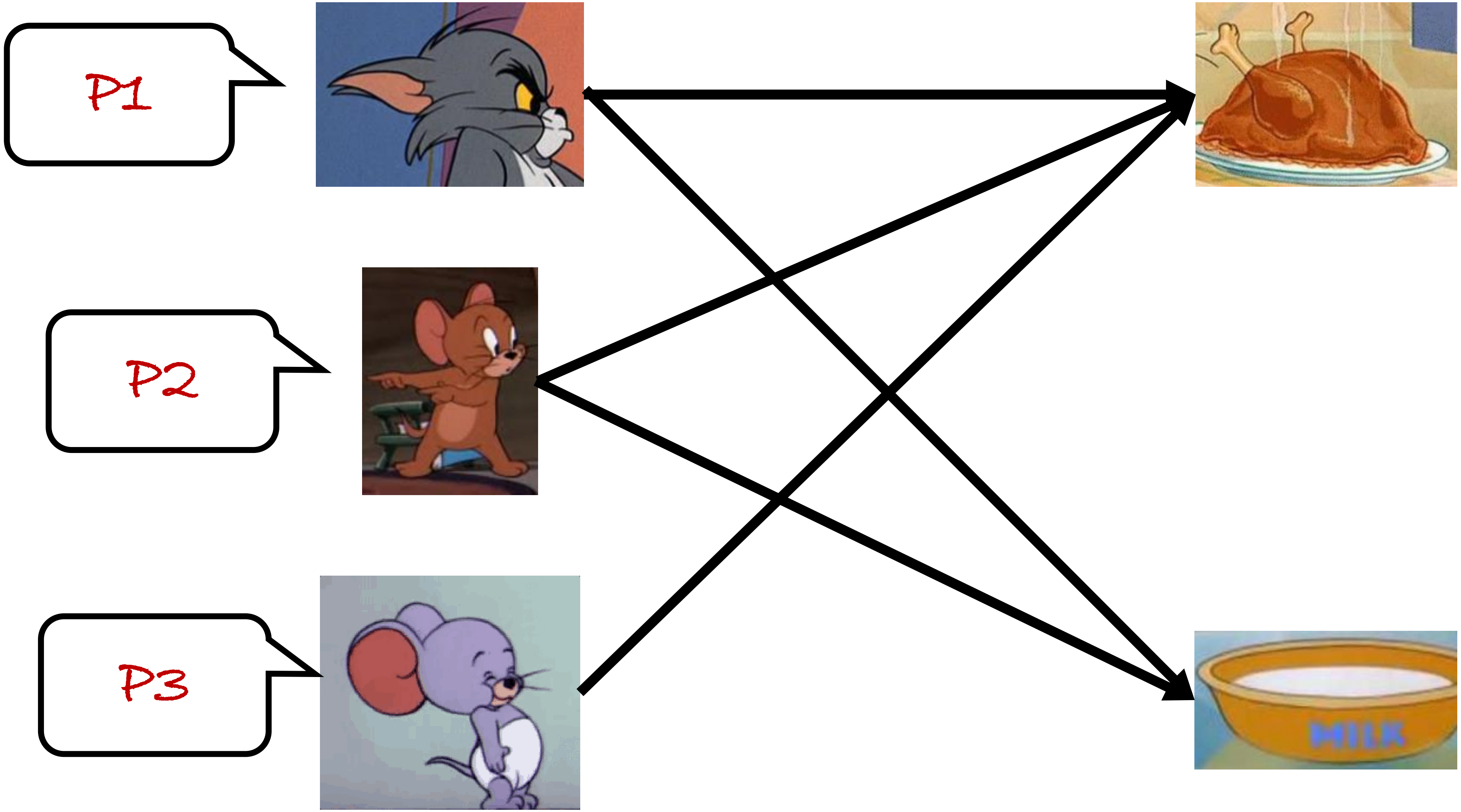
Matching Problem



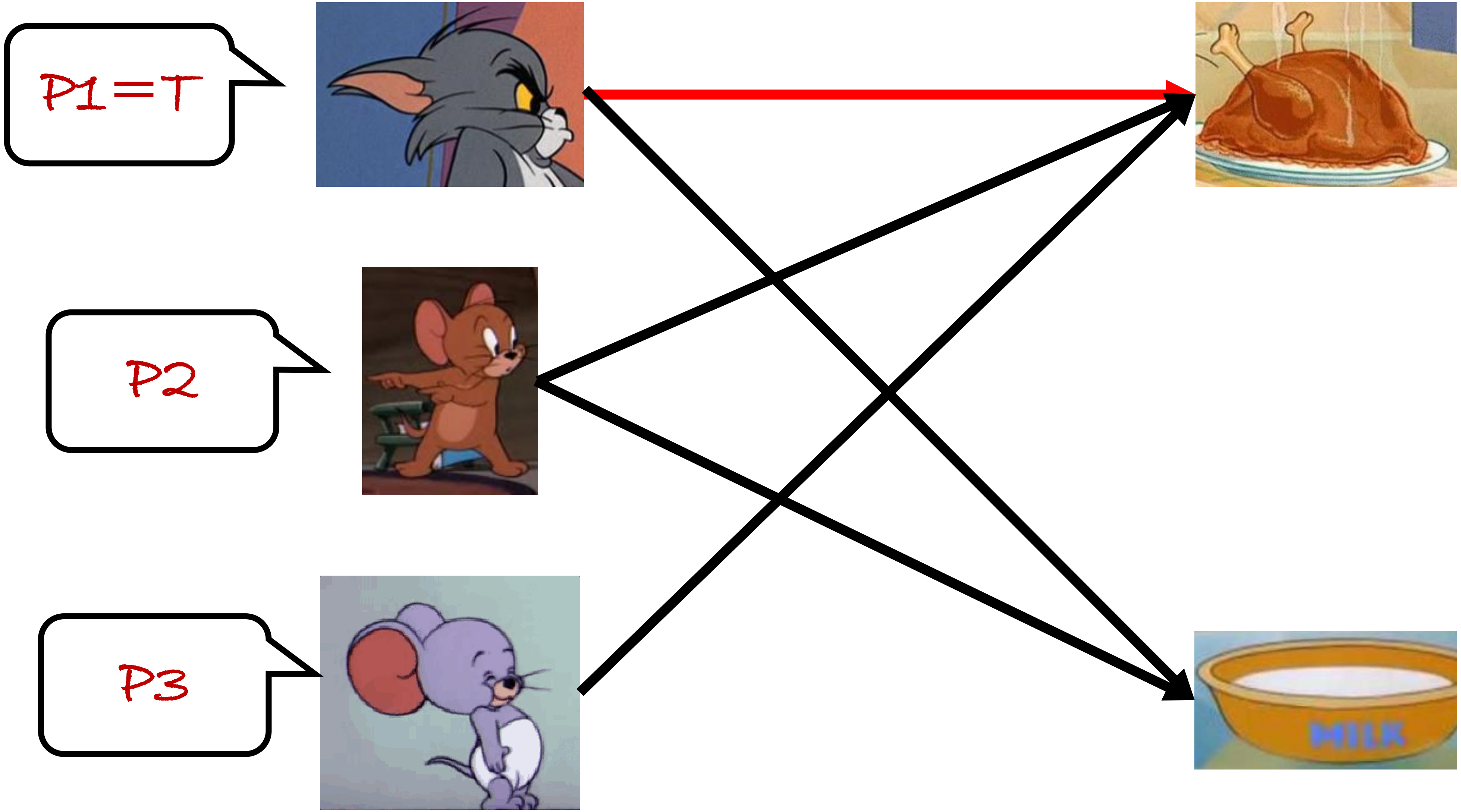
Matching Them With Food And Ensuring No Conflict



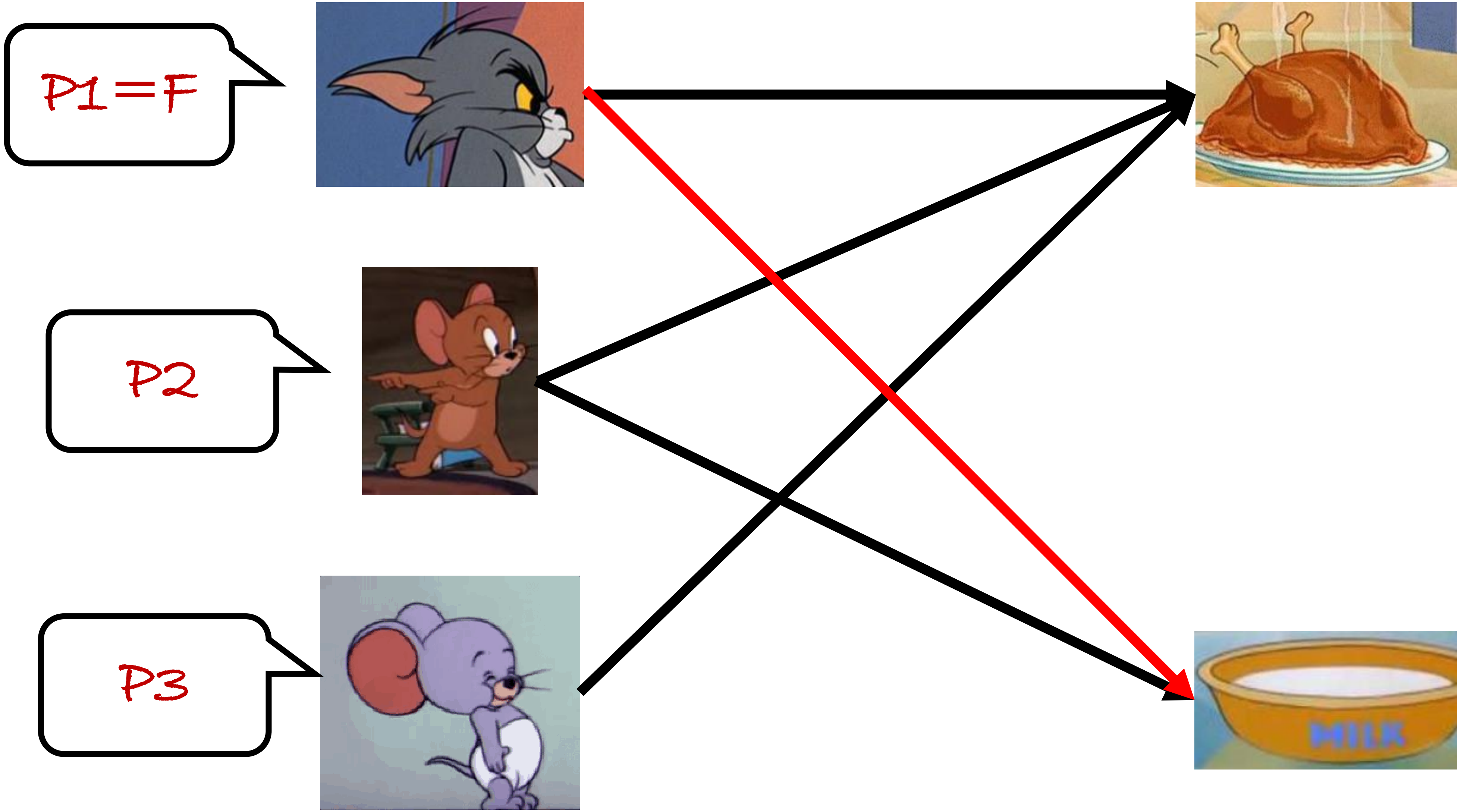
Matching Problem



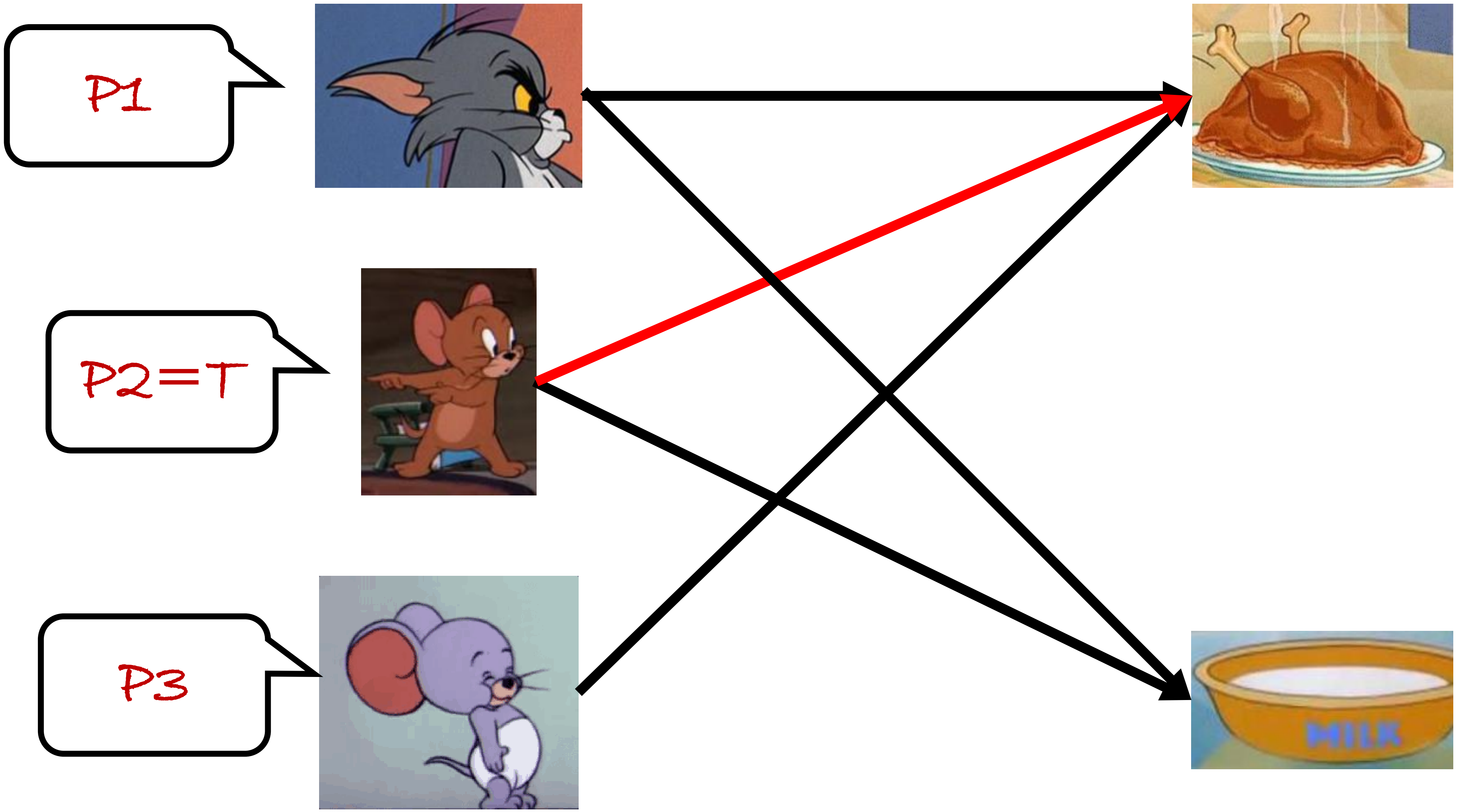
Matching Problem



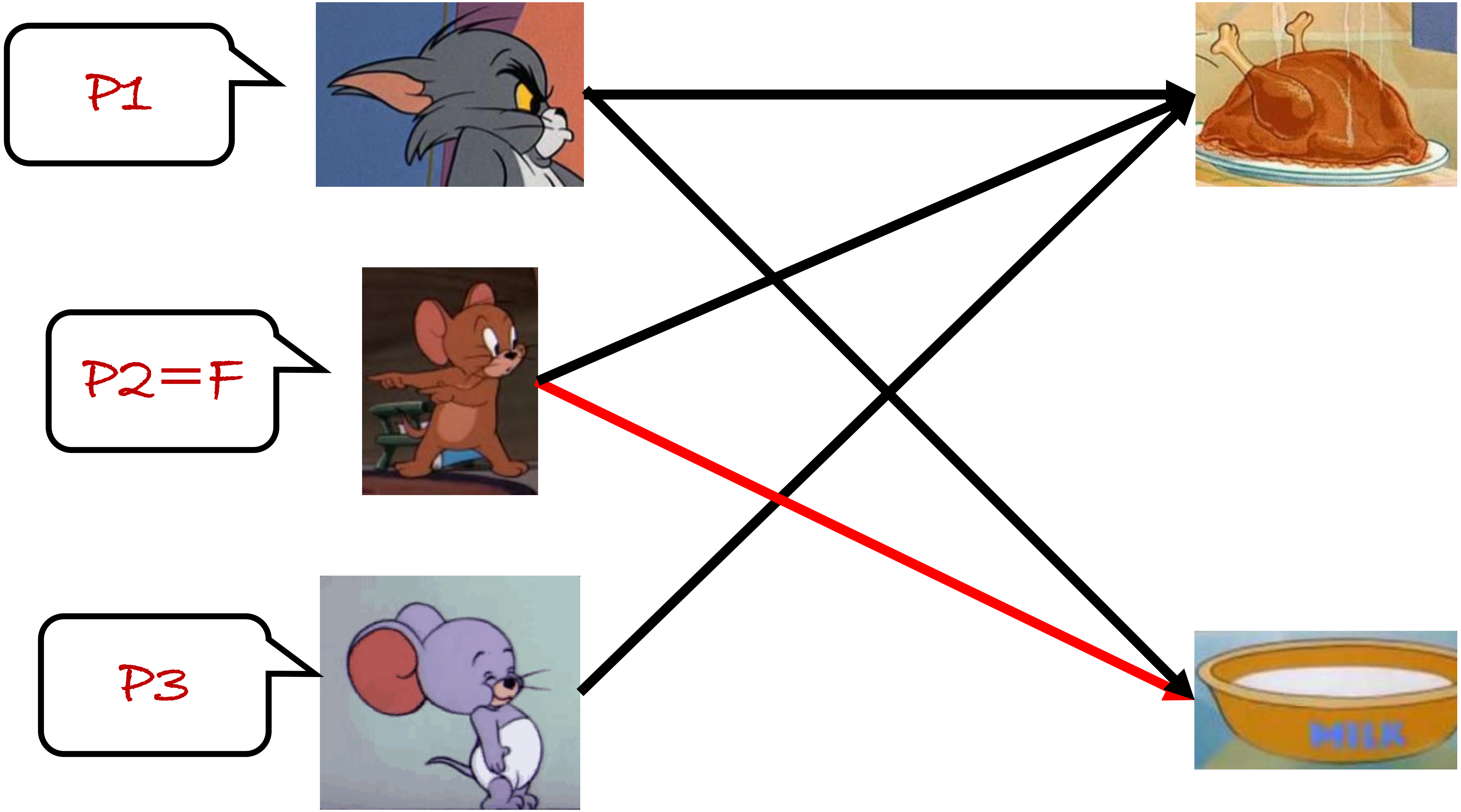
Matching Problem



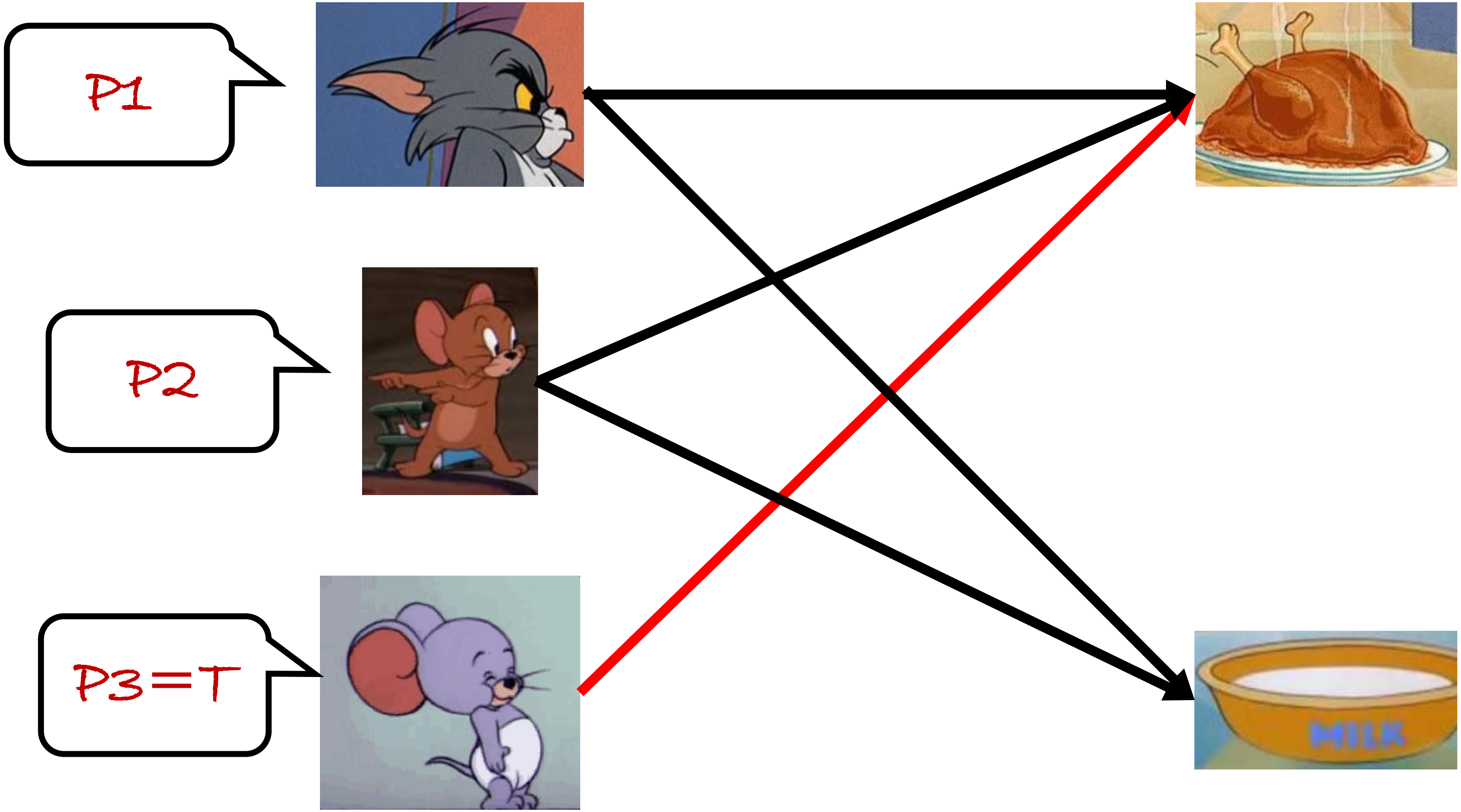
Matching Problem



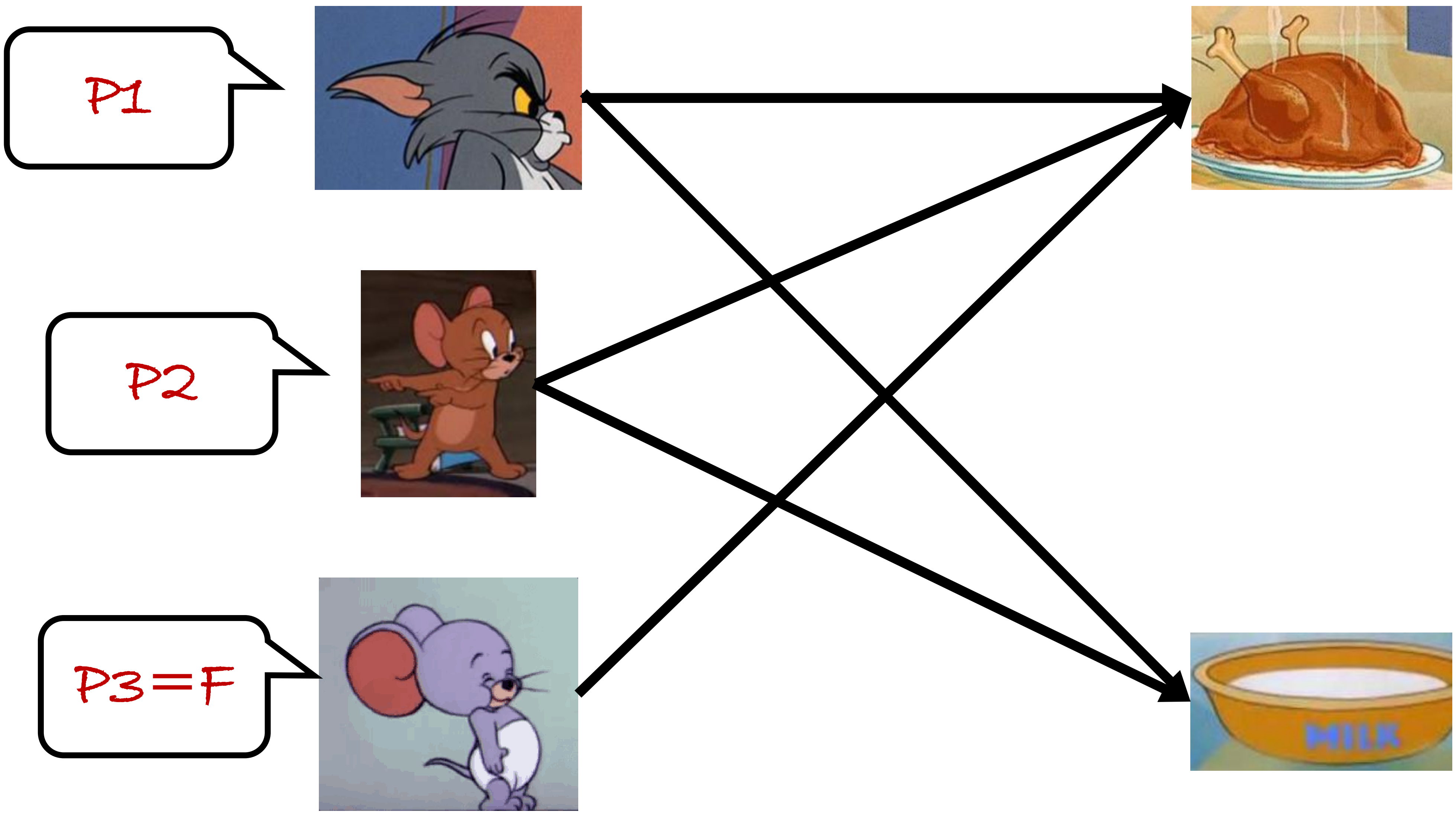
Matching Problem



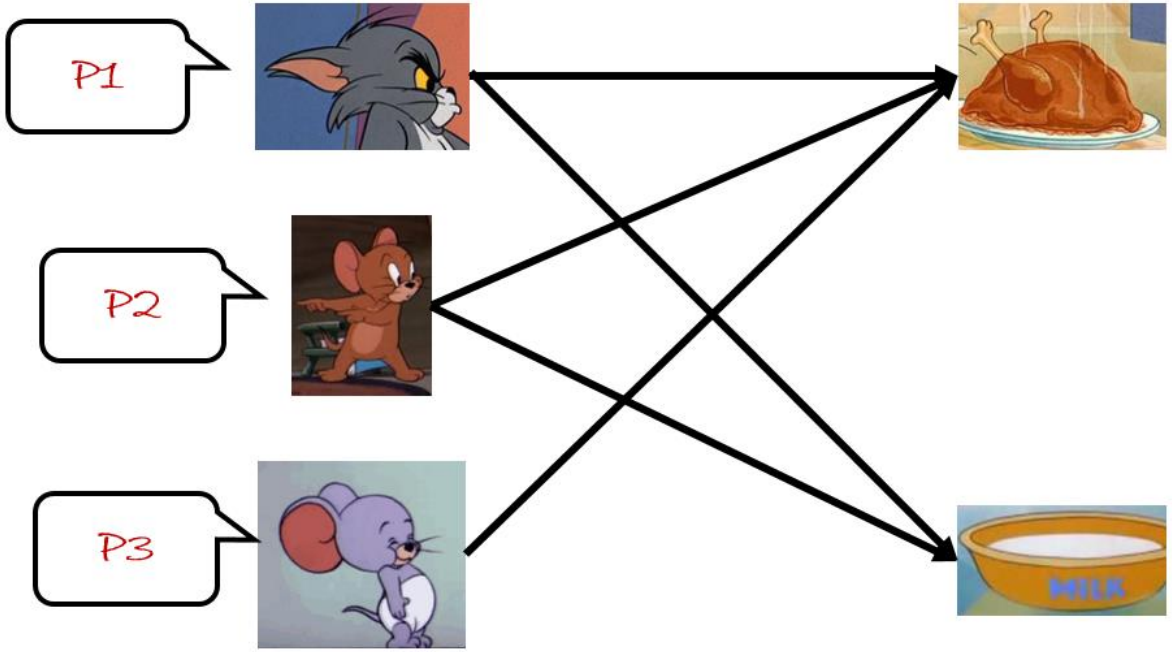
Matching Problem



Matching Problem



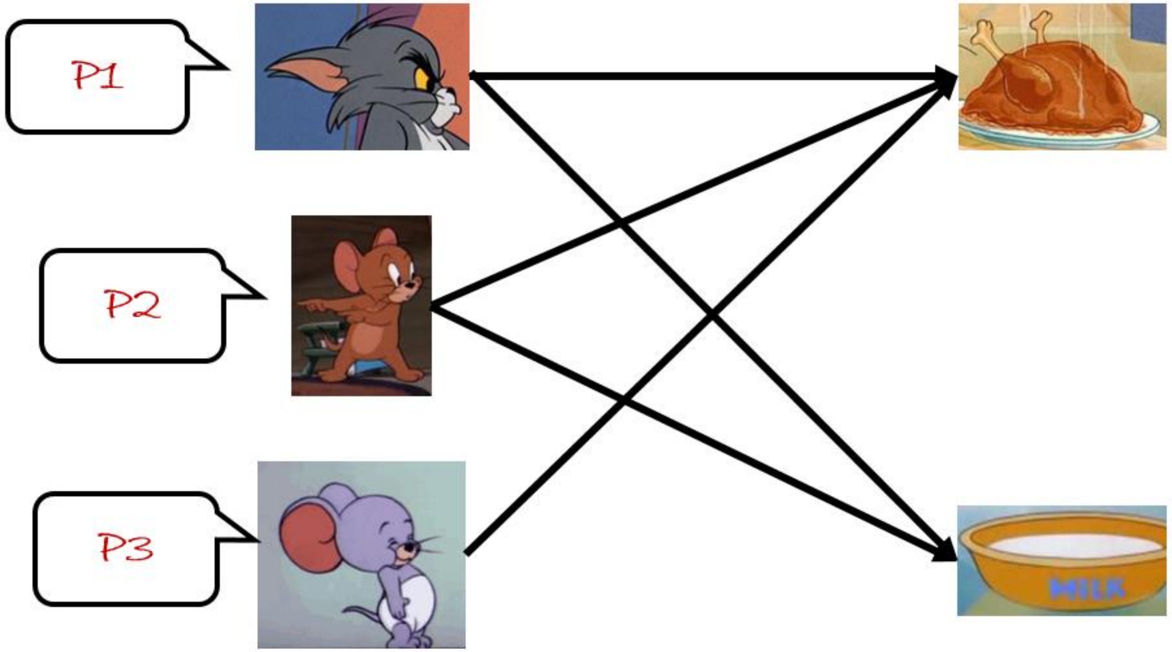
Matching Problem



Tuffy must eat chicken.

$$P3 \wedge \neg(P1 \wedge P2) \wedge \neg(P2 \wedge P3) \wedge \neg(P1 \wedge P3) \wedge \neg(\neg P1 \wedge \neg P2)$$

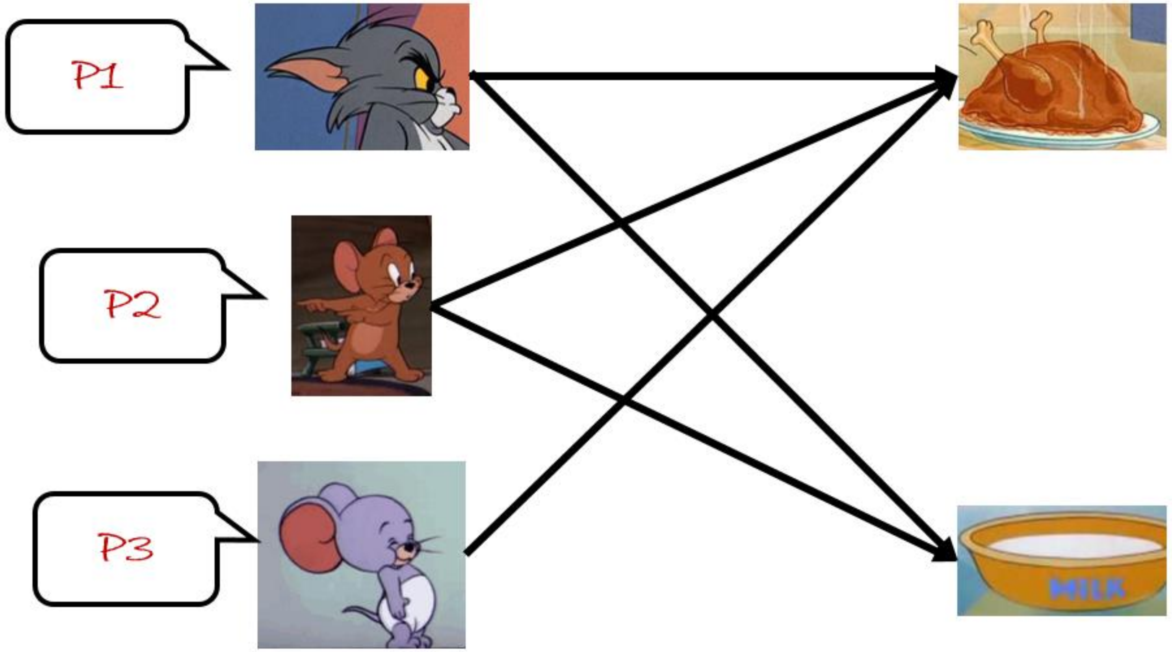
Matching Problem



Chicken cannot be shared.

$$P3 \land \neg(P1 \land P2) \land \neg(P2 \land P3) \land \neg(P1 \land P3) \land \neg(\neg P1 \land \neg P2)$$

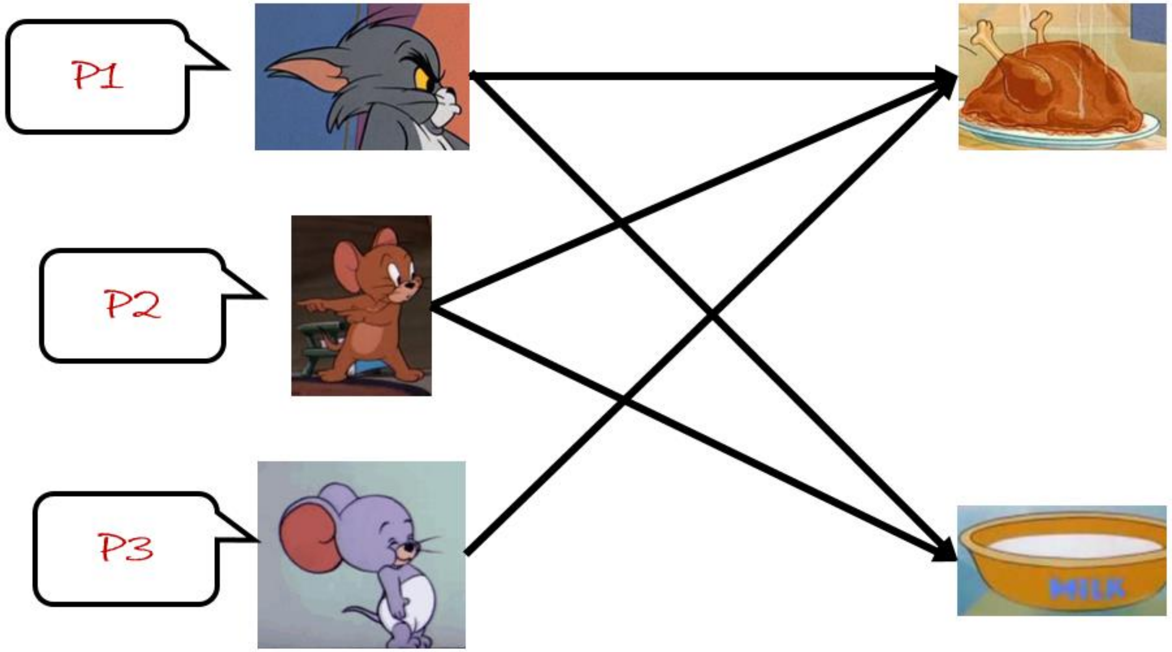
Matching Problem



$$P3 \wedge$$
$$\neg(P1 \wedge P2) \wedge \neg(P2 \wedge P3) \wedge \neg(P1 \wedge P3) \wedge$$
$$\neg(\neg P1 \wedge \neg P2)$$

Milk cannot be shared.

Matching Problem



$$P3 \wedge$$
$$\neg(P1 \wedge P2) \wedge \neg(P2 \wedge P3) \wedge \neg(P1 \wedge P3) \wedge$$
$$\neg(\neg P1 \wedge \neg P2)$$

Now convert it to F.

Matching Problem

$$
\begin{aligned}
& P3 \wedge \boxed{\neg(P1 \wedge P2)} \wedge \boxed{\neg(P2 \wedge P3)} \wedge \boxed{\neg(P1 \wedge P3)} \wedge \boxed{\neg(\neg P1 \wedge \neg P2)} \\
= \; & P3 \wedge \boxed{(\neg P1 \vee \neg P2) \wedge (\neg P2 \vee \neg P3)} \wedge (\neg P1 \vee \neg P3) \wedge (P1 \vee P2) \\
= \; & \boxed{P3 \wedge (\neg P2 \vee \neg P3)} \wedge (\neg P1 \vee \neg P2) \wedge (\neg P1 \vee \neg P3) \wedge (P1 \vee P2) \\
= \; & P3 \wedge \neg P2 \wedge (\neg P1 \vee \neg P2) \wedge (\neg P1 \vee \neg P3) \wedge \boxed{(P1 \vee P2)} \\
= \; & P3 \wedge \boxed{\neg P2 \wedge (P1 \vee P2)} \wedge (\neg P1 \vee \neg P2) \wedge (\neg P1 \vee \neg P3) \\
= \; & P3 \wedge \neg P2 \wedge P1 \wedge \boxed{(\neg P1 \vee \neg P2) \wedge (\neg P1 \vee \neg P3)} \\
= \; & P3 \wedge \neg P2 \wedge \boxed{P1 \wedge (\neg P1 \vee \neg P3)} \wedge (\neg P1 \vee \neg P2) \\
= \; & \boxed{P3} \wedge \neg P2 \wedge P1 \wedge \boxed{\neg P3} \wedge (\neg P1 \vee \neg P2) \\
= \; & \boxed{P3 \wedge \neg P3} \wedge \neg P2 \wedge P1 \wedge (\neg P1 \vee \neg P2) \\
= \; & F
\end{aligned}
$$

It is a special proposition
which is always false

Tautology（重言式/永真式）

DEFINITION

Tautology is a proposition which is always true under any interpretation.

$$P3 \land \neg(P1 \land P2) \land \neg(P2 \land P3) \land \neg(P1 \land P3) \land \neg(\neg P1 \land \neg P2) \leftrightarrow F$$

The previous problem is to prove the formula is a tautology.

Contradiction（矛盾式/永假式）

DEFINITION

Contradiction is a proposition which is always false under any interpretation.

$$P3 \wedge \neg(P1 \wedge P2) \wedge \neg(P2 \wedge P3) \wedge \neg(P1 \wedge P3) \wedge \neg(\neg P1 \wedge \neg P2)$$

The previous problem is to prove the formula is a contradiction.

Example

$P \vee \neg P$	tautology
$P \wedge \neg P$	contradiction
$P \wedge F$	contradiction
$P \vee T$	tautology
$\neg(A \wedge B) \leftrightarrow (\neg A \vee \neg B)$	tautology