Artificial Intelligence

① อันเก่าเน้นต้องใช้กฎที่มากเดินไปเป็นในทุกการงานหนึ่ง

② ตาม สมการเปลี่ยนแปลงได้ แต่ต่ำแหน่งเดียวกันต่างที่ time step ต่าง

กฎตนละกฎ

Instructor: Kietikul Jearanaitanakij

Department of Computer Engineering

King Mongkut's Institute of Technology Ladkrabang

Lecture 6
First-Order Logic

- First-order logic

- FOL Syntax and Semantics

- Quantifiers

- Connections between ∀ and ∃

- First-order logic for reflex agent

Recall: Problems of the propositional agent

1. There are too many propositions to handle.

   For example, The simple rule "Don't go forward if the Wumpus is in front of the agent" can be stated in propositional logic by 64 rules, i.e., 16 squares x 4 directions = 64.

2. Since the world can change configuration at each time step, we must specify time in the inference rules.

$$A^0_{1,1} \land East^0_A \land \neg W_{2,1} \Rightarrow Forward^0$$

$$A^6_{1,1} \land East^6_A \land \neg W_{2,1} \Rightarrow TurnLeft^6$$

1,4	2,4	3,4	4,4
1,3	2,3	3,3	4,3
1,2	2,2	3,2	4,2
OK			
1,1 A	2,1	3,1	4,1
OK	OK		

First-Order Logic

- Propositional logic is too puny a language to represent knowledge of complex environments in a concise way.

- We examine first-order logic which is sufficiently expressive to represent a knowledge.

- First-order logic can express the properties of entire collections of objects rather than having to enumerate the objects by name.

- Similar to other logics, First-order logic composes of syntax and semantic.
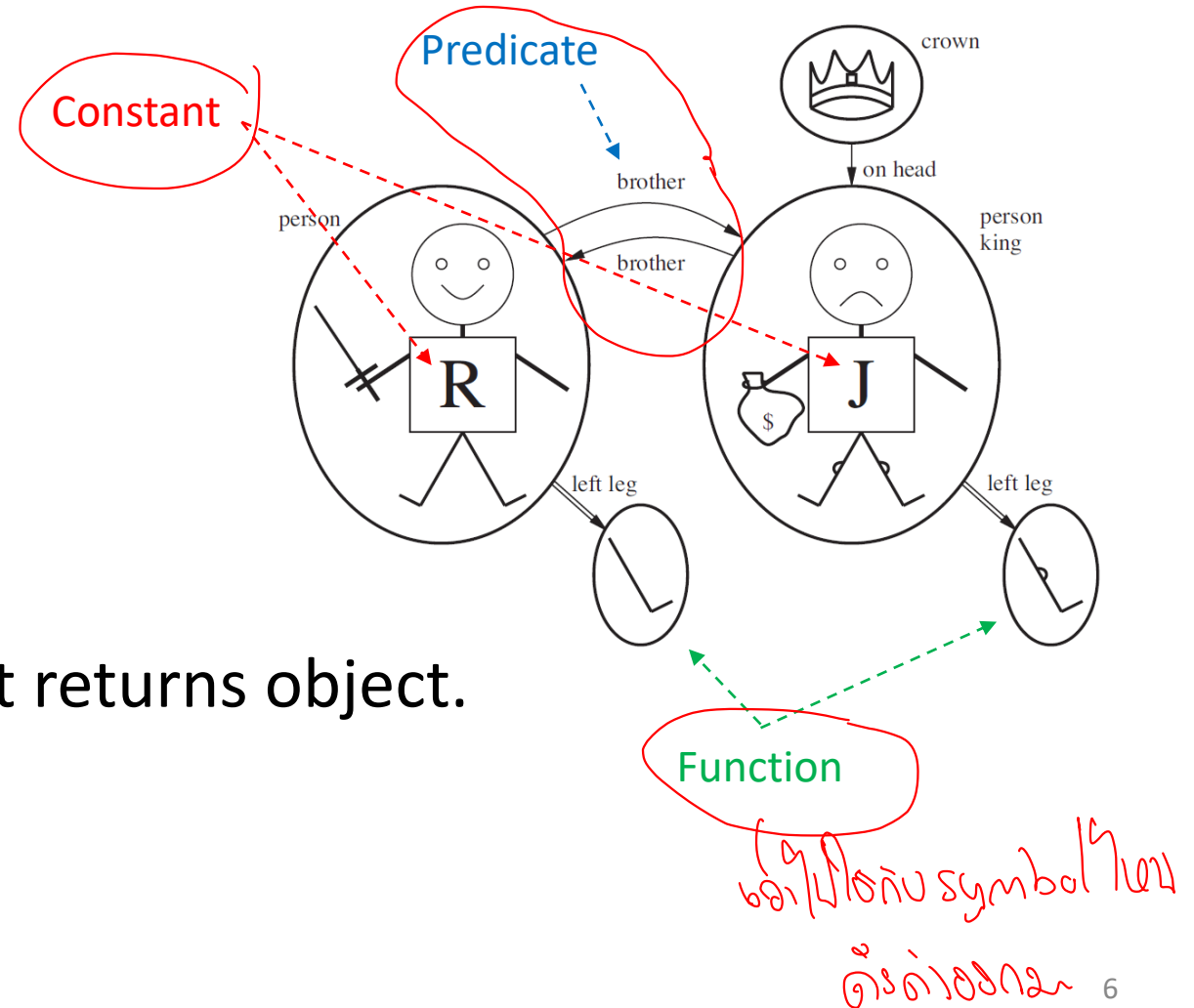
Syntax and Semantics

ก่อนจะทำ

ตัวที่เรียกรวมวยเรียกว่า ว:บบ symbol
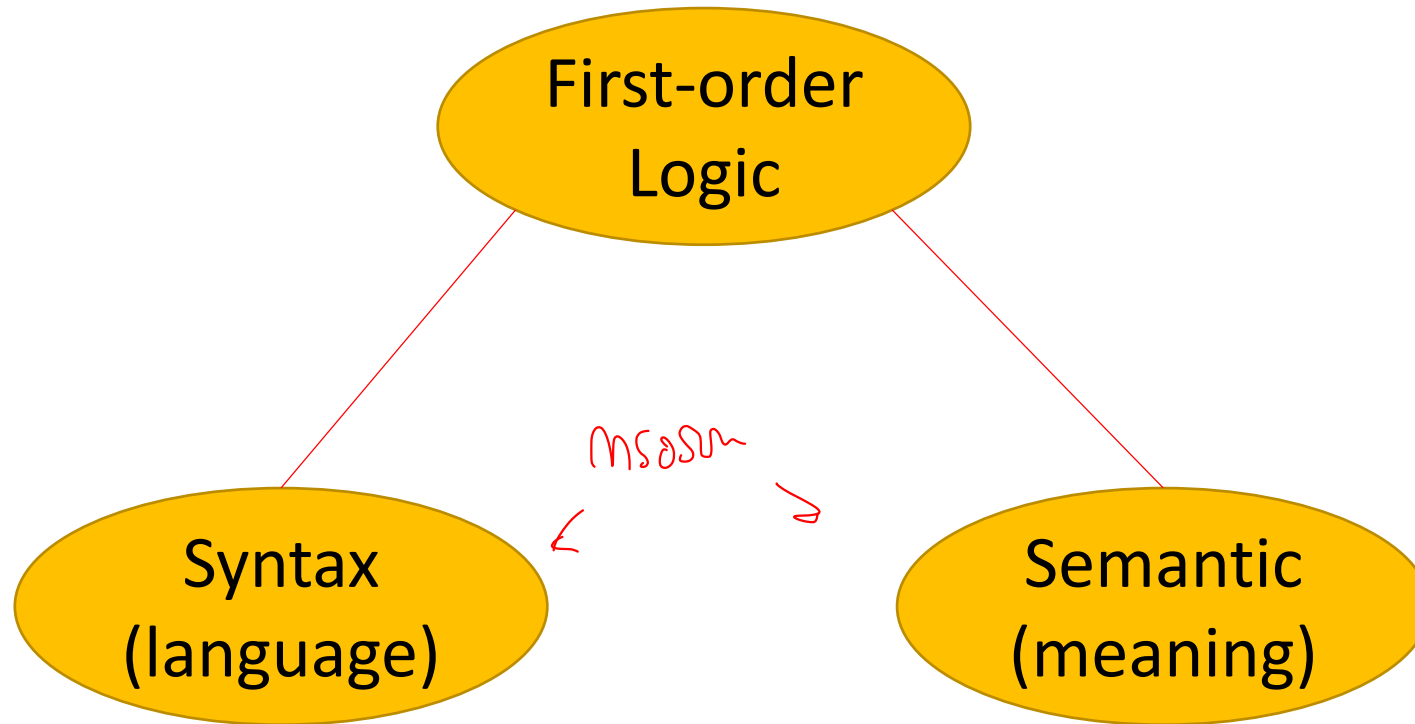
- **Symbol** consists of constant symbol, predicate symbol, and function symbol.

  - Constant symbol: object

    Ex: Richard , John, etc.

  - Predicate symbol: relation

    Ex: Brother, Friend, etc.

  - Function symbol: function that returns object.

    Ex: LeftLeg, etc.

เอาไปใช้กับ symbol ในเนน

ด้วยอย่างกัน

Syntax and Semantics

เอา symbol จากโจรมารวมกัน → term

- **Term** is a logical expression that refers to an object; e.g., KingJohn, LeftLegOf(John). Constant symbols are also terms.

  Fun   conster

- **Atomic sentence** represents a relationship between objects :- formed from a predicate symbol followed by a parenthesized list of terms.

  For example,

  pre   Fun   con   Fun   con
  Brother(Richard, John), Married(FatherOf(Richard), MotherOf(John))

  Predic   con   Con

- **Complex sentence** :- logical connectives of atomic sentences.

  เอา Yogic Con จากรวมกัน

  atomic

  Brother(Richard, John) ∧ Brother(John, Richard),

  ¬King(Richard) ⇒ King(John)

The syntax of first-order logic with equality, specified in Backus–Naur form.

กลุ่มๆ Proposition

$$Sentence \rightarrow AtomicSentence \mid ComplexSentence$$

$$AtomicSentence \rightarrow Predicate \mid Predicate(Term, \ldots) \mid Term = Term$$

$$ComplexSentence \rightarrow (\; Sentence \;) \mid [\; Sentence \;]$$
$$\mid \; \neg \; Sentence$$
$$\mid \; Sentence \land Sentence$$
$$\mid \; Sentence \lor Sentence$$
$$\mid \; Sentence \Rightarrow Sentence$$
$$\mid \; Sentence \Leftrightarrow Sentence$$
$$\mid \; Quantifier \; Variable, \ldots \; Sentence$$

$$Term \rightarrow Function(Term, \ldots)$$
$$\mid \; Constant$$
$$\mid \; Variable$$

มีสิ่งนี้เข้ามาบวก เตี๋ยวอธิบาย

$$Quantifier \rightarrow \forall \mid \exists$$
$$Constant \rightarrow A \mid X_1 \mid John \mid \cdots$$
$$Variable \rightarrow a \mid x \mid s \mid \cdots$$
$$Predicate \rightarrow True \mid False \mid After \mid Loves \mid Raining \mid \cdots$$
$$Function \rightarrow Mother \mid LeftLeg \mid \cdots$$

ตัวจะ return obj

Predicate gives you true or false based on your input(s). While, a function gives you an output per your input(s).
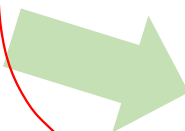
$$\textsc{Operator Precedence} \;:\; \neg, =, \land, \lor, \Rightarrow, \Leftrightarrow$$

8

- **Quantifier** : express the properties of entire collections of objects, rather than having to enumerate the objects by name.
  1. **Universal Quantification (∀)**

ตัวที่ค่าเป็น ถึงจะมากากวร

ทุกๆ obj ที่อยู่ใน domain

$$\forall_x \; Cat(x) \Rightarrow Mammal(x)$$

For all object x, if x is a cat then x is mammal.

ถูกต้อง #

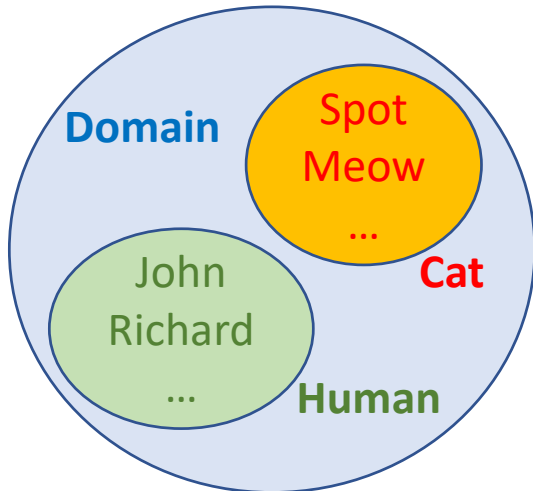**If there is no any cat in our domain, it is fine, i.e., the sentence is still true.**

ถ้าเขียนอีกอันละ

ถ้าเขียนแบบนี้แทนละ

Problem: If we represent "All cats are mammal" by $\forall_x \; Cat(x) \wedge Mammal(x),$

we will not capture what we need since the representing sentence means the following:

apply ได้กับทุก obj ในdomain

มากากมวอก

ลองเทสบแบบไม่รู้นิวเมอก

$$[Cat(Spot) \wedge Mammal(Spot)] \wedge \checkmark$$
$$[Cat(Meow) \wedge Mammal(Meow)] \wedge \checkmark \quad True$$
$$[Cat(John) \wedge Mammal(John)] \wedge \quad \Big\} \; false$$
$$[Cat(Richard) \wedge Mammal(Richard)] \wedge$$
$$.... \quad .... \quad ....$$

มากากอก

Too strong sentence (Always false)

(TRUE) AND (FALSE)
FALSE

9

2. Existential Quantification (∃)

for some (บางตัวของ) [handwritten]

$$\exists_x \, Sister(x, spot) \wedge Cat(x)$$

ไม่มีปัญหาแบบด้านบน [handwritten]

If there is no sister of spot who is a cat, this sentence must be false.

There is a sister of spot who is a cat.

ถ้าใน domain ไม่มี sister เลย → เป็น False $F \vee F \vee F \vee ... \vee F$ ← ธรรมดาสิ่งของสิน [handwritten]

<u>Problem</u>: If we represent "There is a sister of spot who is a cat" by

ถ้าเกิดเรียงตาม [handwritten]

$$\exists_x Sister(x, spot) \Rightarrow Cat(x),$$

$F \to anything = \boxed{TRUE}$ [handwritten]

we will not capture what we need since the representing sentence means the following:

$T \vee T \vee ... \vee T$ [handwritten]

Domain
Spot
Meow
...
John
Richard
...
Cat
Human
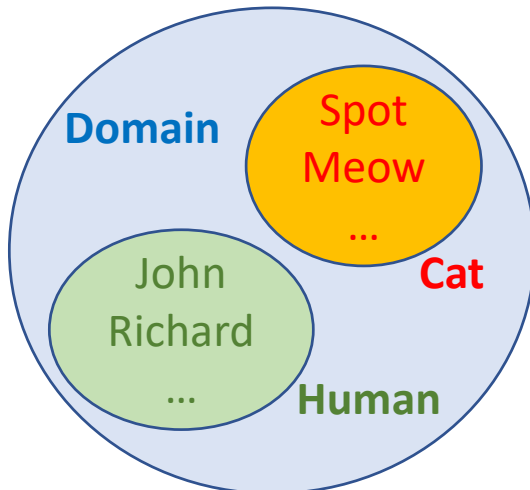
$[Sister(Spot, Spot) \Rightarrow Cat(Spot)] \vee$
$[Sister(Meow, Spot) \Rightarrow Cat(Meow)] \vee$
$[Sister(John, Spot) \Rightarrow Cat(John)] \vee$
$[Sister(Richard, Spot) \Rightarrow Cat(Richard)] \vee$
....

Too weak sentence
(Always true)

$\vee \to \wedge$ [handwritten]
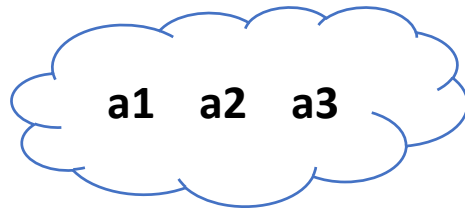$\exists \to \vee$ [handwritten]

ใช้ตามเหมาะสมของเรากัน [handwritten]

10

- Example: "Some cats are intelligent".

$$\exists_x Cat(x) \Rightarrow Intel(x)$$ ผิด $\exists_x Cat(x) \wedge Intel(x)$

What's wrong with the above FOL sentence ?

ก็เป็นแบบที่ จลดก็เป็นนธรรม

Domain

a1   a2   a3

FU...⊕N F

Name	Cat / Dog ?	Intelligent
a1	Cat	No
a2	Cat	No
a3	Dog	Yes

False

อัตโลอึกว่า)ได

$$[Cat(a1) \Rightarrow Intel(a1)] \vee [Cat(a2) \Rightarrow Intel(a2)] \vee [Cat(a3) \Rightarrow Intel(a3)]$$
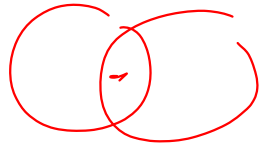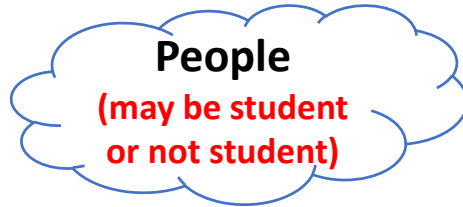
False   True

True

This FOL sentence always true even though there is no cat that is intelligent.

11

- Example: "Every student in this class has visited Africa or America".

Domain

People
(may be student
or not student)

student(x) : x is student in this class

vaf(x) : x has visited Africa

vam(x) : x has visited America

Find FOL sentence for the above sentence.

$\forall_x \text{student}(x) \rightarrow [\text{vaf}(x) \text{ or } \text{vam}(x)]$

- Example: "Some prime number is even number".

Domain

Numbers

prime(x) : x is prime number

even(x) : x is even number

Find FOL sentence for the above sentence.

$$\exists x \; prime(x) \wedge even(x)$$

3. Nested Quantification

- "For all x and all y, if x is the parent of y then y is a child of x"

$$\forall_{x,y} \; Parent(x,y) \Rightarrow Child(y,x)$$

ทุก x, y

- "Everybody loves somebody"

$$\forall_x \exists_y \; Loves(x,y)$$

ทุก x มี y

The order of quantification can change the meaning of the sentence.

- "There is someone who is loved by everybody"

$$\exists_y \forall_x \; Loves(x,y)$$ การอยู่ในวงเล็บต้องไม่อยู่ กระจายออก function

- "Everybody is loved by somebody"

$$\forall_x \exists_y \; loves(y,x)$$

- "Somebody loves everybody"

$$\exists_x \forall_y \; loves(x,y) \longrightarrow หรือ \quad \exists_y \forall_x \; love(y,x)$$

- "Nobody loves everyone"

Everybody doesnot love some body

$$\forall_x \exists_y \sim love(x,y)$$

14

Exercise 1

- Give FOL sentence for

    "Gold and silver ornaments are precious"

Let   G(x) : x is a gold ornament,

        S(X) : x is a silver ornament,

        P(X) : x is precious.

A) $\forall_x (P(x) \implies G(x) \wedge S(x))$

B) $\forall_x (G(x) \wedge S(x) \implies P(x))$

C) $\exists_x (G(x) \wedge S(x) \implies P(x))$

D) $\forall_x (G(x) \vee S(x) \implies P(x))$

15

Exercise 2

- Find FOL for the following sentence

"Every teacher is liked by some students"

ถ้ามองเห็น logic

$\forall_x teacher \rightarrow F$

$[\sim teacher] \wedge$

$T \rightarrow F, T$ มีจุด $\wedge$ กับ

A) $\forall_x\ teacher\ (x) \Longrightarrow \exists_y(student(y) \Longrightarrow Likes(y,x))$

t

F ไม่จบ #

B) $\forall_x\ teacher\ (x) \Longrightarrow \exists_y(student(y) \wedge Likes(y,x))$

ถ้ามองการใช้ร่วม

C) $\exists_y\forall_x\ teacher\ (x) \Longrightarrow\ (student(y) \wedge Likes(y,x))$

$\exists_x\forall_y teacher\ (y) \rightarrow (student\ (x) \wedge likes\ (x,y)) \rightarrow$ some student like every teacher

D) $\forall_x\ (\ teacher\ (x) \wedge \exists_y[student(y) \Longrightarrow Likes(y,x)]\ )$

F, T $\wedge$ T ไม่ว่าจะเกิน $\rightarrow$ รวมทั้งหมดได้ (F)

Exercise 3

- Find FOL for the following sentence

   "Some boys are taller than all girls."

A) $\exists_x \, boy \, (x) \Rightarrow \forall_y (girl(y) \land taller(x, y))$

B) $\exists_x \, boy \, (x) \land \forall_y (girl(y) \land taller(x, y))$

C) $\exists_x \, boy \, (x) \Rightarrow \forall_y (girl(y) \Rightarrow taller(x, y))$

D) $\exists_x \, boy \, (x) \land \forall_y [girl(y) \Rightarrow taller(x, y)]$

Exercise 4

ข้อนี้ฉันไม่เข้าจะยเท่าไหร่

ก็ให้ใส่ not →ตัววรณนสีภาว

ว่ามันเป็นแบบนี้น

ในทดลักษร
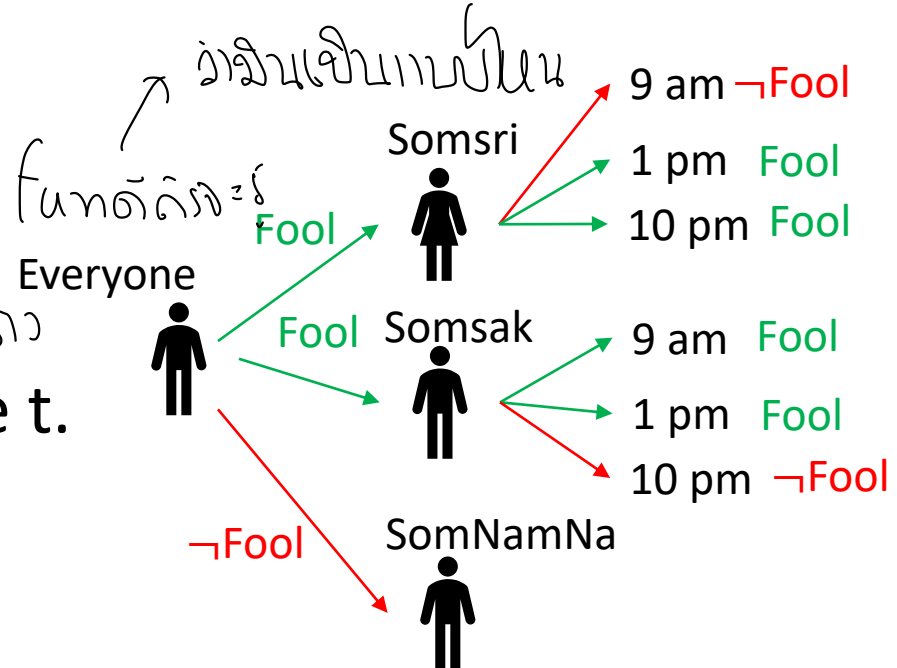
- Let F(x,y,t) : person x can fool person y at time t.

$$\forall_x \exists_y \exists_t \, (\neg F(x, y, t))$$

A) Everyone can fool some person at some time.

B) No one can fool everyone all the time.

↳ ไม่มีใครหลอกลวงทุกคนได้ตลอดเวล

C) Everyone can not fool some person all the time. → ถ้าดูรูปจากการจะนี้ถูก แต่ไม่ตรรงประโยค

↳ ถ้าตาม logic จะเป็น some time

D) No person can fool some person at some time.

Everyone

Fool → **Somsri** → 9 am ¬Fool
1 pm Fool
10 pm Fool

Fool → **Somsak** → 9 am Fool
1 pm Fool
10 pm ¬Fool

¬Fool → **SomNamNa**

18

Exercise 5

ใช้ประกอบ x คือคน y คือสัตว์ตา z คือคนๆ

- Find FOL sentence for

ใช้ประกอบอัง "Everyone who loves all animal is loved by someone."

A) $\forall_{x,y} [Animal(y) \Rightarrow Loves(x,y)] \Rightarrow \exists_z Loves(z,x)$

$\square \rightarrow \top$ จริงเลย

สำหรับทุกๆ สัตว์ที่คนนั้นรัก $\rightarrow$ จะ ถูกรักโดยใครบางคน → love by someon

B) $\forall_{x,y} [Animal(y) \land Loves(x,y)] \Rightarrow \exists_z Loves(z,x)$

$\square \rightarrow \top$ ถ้าคน

ทุกๆคนที่รักสัตว์

สัตว์และคนที่รักสัตว์ทุกตัว ถูกรักโดย   love by sm   same

C) $\forall_{x,y}[Animal(y) \Rightarrow Loves(x,y)] \land \exists_z Loves(z,x)$

$\square \land \top$ จริงไหม

ทุกๆคนที่รักสัตว์และ ถูกรักโดยใครบางคน

meaning is so weird

Connections between ∀ and ∃

Two quantifiers are actually intimately connected with each other through negation.

$$\forall x \ \neg Likes(x, Parsnips) \quad \text{is equivalent to} \quad \neg \exists x \ Likes(x, Parsnips) .$$

ทุกๆคนไม่ชอบหัวผักกาด.

ไม่มีใครชอบหัวผักกาด

$$\forall x \ Likes(x, IceCream) \quad \text{is equivalent to} \quad \neg \exists x \ \neg Likes(x, IceCream) .$$

ไม่มีใครที่ไม่ชอบ

The De Morgan rules for quantified and unquantified sentences are as follows:

ทุกๆคนไม่ชอบไอติม   *ไม่มีใครเลยชอบไอ*

$$\forall x \ \neg P \ \equiv \ \neg \exists x \ P \qquad\qquad \neg(P \vee Q) \ \equiv \ \neg P \wedge \neg Q$$

$$\neg \forall x \ P \ \equiv \ \exists x \ \neg P \qquad\qquad \neg(P \wedge Q) \ \equiv \ \neg P \vee \neg Q$$

$$\forall x \ P \ \equiv \ \neg \exists x \ \neg P \qquad\qquad P \wedge Q \ \equiv \ \neg(\neg P \vee \neg Q)$$

$$\exists x \ P \ \equiv \ \neg \forall x \ \neg P \qquad\qquad P \vee Q \ \equiv \ \neg(\neg P \wedge \neg Q) .$$

ดูไม่ออกว่าจะ ใช้ใน

การยืนยัน

ไม่มีทุกกรณี

มีในกรณีเดียว

จำมาเปลี่ยนขวาก $\forall_x \equiv \neg \exists_x$ *อย่างเดียวก็ได*

Equity → สมบัติเครื่องหมาย = ได้

We can use the equality symbol to signify that two terms refer to the same object.

- **Father (John)=Henry**, says that the object referred to by Father (John) and the object referred to by Henry are the same.

→ ถ้าไม่ได้จากการนับเลข

We can also use equality for counting in first-order logic

- "Spot has at least two sisters" → เปรียบเทียบ → การเปรียบเทียบๆ

$$\exists_{x,y}\ Sister(Spot, x) \wedge Sister(Spot, y) \wedge \neg(x = y)$$

มี x, y บางตัวที่ x, y ไม่เหมือนกัน 2 ตัว

Kinship domain → เหมือน relationship

Kinship domain represents the relationship between objects.

- Domain mother

$$\forall_{m,c}\ Mother(c) = m \Longleftrightarrow Female(m) \wedge Parent(m, c)$$

จะตรวจเช็ค

domain ของ mother (c)

แปลความว่า

Assertions and queries in first-order logic

- Sentences are added to a knowledge base using TELL, exactly as in propositional logic.

- Such sentences are called assertions.

- For example, we can assert that John is a king, Richard is a person, and all kings are persons:

$$\text{TELL}(KB, King(John)) .$$
$$\text{TELL}(KB, Person(Richard)) .$$
$$\text{TELL}(KB, \forall x \; King(x) \Rightarrow Person(x)) .$$

- We can ask questions of the knowledge base using ASK. For example,

$$\text{ASK}(KB, King(John)) \quad \text{returns true.}$$
$$\text{ASK}(KB, Person(John)) \quad \text{also returns true.}$$

First-order logic for reflex agent *ความเอาไปใช้กับ wampus world*

We will take a look how first-order logic represents the rules in a Wumpus world.

Rules : $Percept \Rightarrow Action$

Example:

   "If the agent senses a glitter, it should do a grab in order to pick up the gold" can be represented by the sentence in first-order logic as follow:

$$\forall_{s,b,u,c,t} \ Percept([s, b, Glitter, u, c], t) \Rightarrow Action(Grab, t)$$

ถ้าเขียนครบ action ที่ 2×2×2×2×100

We can reduce some facts into a shorter representation and then use it in the following rules.

$$\forall_{s,b,u,c,t} \ Percept([s, b, Glitter, u, c], t) \Rightarrow At_{Gold}(t)$$

$$\forall_t \ At_{Gold}(t) \Rightarrow Action(Grab, t)$$

สามารถเขียนเป็น predicate สั้นๆก่อนได้

เรียกเอาไปใช้ง่าย

โดยไม่ต้องเขียน logic ก่อนหน้าใหม่

23

Deducing Hidden Properties of the World

1. Causal Rules

   "Squares adjacent to Wumpus are smelly"

   "Squares adjacent to pit are breezy"

   These two rules can be represented by the following causal rules:

   $$\forall_{l1,l2,t} \; At(Wumpus, l, t) \wedge Adjacent(l1, l2) \Rightarrow Smelly(l2)$$

   $$\forall_{l1,l2,t} \; At(Pit, l, t) \wedge Adjacent(l1, l2) \Rightarrow Breezy(l2)$$

2. Diagnostic Rules

   $$\forall_{l,t} \; At(Agent, l, t) \wedge Breezy(l) \Rightarrow Action(TurnLeft, t)$$