Logic in Computer Science (ECS666U/7018P/U)

Nikos Tzevelekos

Lecture 8
Predicate Logic

Schedule for Weeks 8-12

We cover three subjects:

- Predicate Logic (weeks 8 & 9)
- Hoare Logic (week 10)
- Prolog (week 11)

Test on week 12:
- is on the material of weeks 8-11
- assumes knowledge of prop. logic

	Tutorial	Lecture
Week 8	Prolog Demo	Predicate Logic
Week 9	Predicate Logic	Predicate Logic II
Week 10	Predicate Logic II	Hoare Logic
Week 11	Hoare Logic	Introduction to Prolog
Week 12	*Mock Test*	End-of-term Test

Propositional Logic

So far we looked at *propositional* logic:

- propositions combined with *boolean* connectives
- we extended it with *temporal* connectives

For example:

S1: If it rains then I take an umbrella

S2: If I take an umbrella then it rains

S3: If I do not take an umbrella then it is not raining

Questions we looked at:

→ do S1 and S2 have the same meaning?

→ do S1 and S3 have the same meaning?

Reasoning about individuals and their properties

Propositional logic is somewhat restricted in that:

- we cannot reason about properties of an individual subject
- we cannot make statements about all/some individuals

For example, here is a classical logical syllogism (going back to Aristotle):

– *all humans are mortal*
– *Socrates is a human*
– *Socrates is mortal*

propositional logic is too weak for this!

Logic in Computer Science

Sometimes we need to go beyond propositions:

– *Users can access their account only if they provide the right password*

– *A database API operation preserves the integrity of the database*

– *Users cannot access files they do not have permissions for*

and, in Programming:

– *if the length of the array A is 0 then the list L is empty* (e.g. `listFromArray`)

– *if x is not 0 then* `divide(y,x)` *terminates with no errors*

– *if x is not 0 then* `divide(y,x)` *terminates and returns y divided by x*

Predicate Logic formulas

Ingredients:

- constants (*A*, *L*, etc.)
- functions (*size*, *length*, etc.)
- relations (*elementOf*, =, etc.)

- variables (*x*, *y*, etc.)
- quantifiers: $\forall$ (*for all*) $\exists$ (*there exists*)
- connectives: $\wedge$ $\vee$ $\rightarrow$ $\neg$

Predicate Logic

A logic of *predicates* (relations, functions) over individuals

– *if the length of the array A is 0 then the list L is empty*

– *the length of the array A is bounded by the size of the list L*

– *each element of the array A appears in the list L*

Predicate Logic formulas

Ingredients:

- constants (*A*, *L*, etc.)
- functions (*size*, *length*, etc.)
- relations (*elementOf*, =, etc.)

how are these different?

- variables (*x*, *y*, etc.)
- quantifiers: $\forall$ (*for all*) $\exists$ (*there exists*)
- connectives: $\wedge$ $\vee$ $\rightarrow$ $\neg$

A function or relation is:

- *nullary* if it has 0 arguments
- *unary* if it has 1 argument
- *binary* if it has 2 arguments
- etc.

We also say that the function or relation has *arity* 0, 1, 2, etc.

A constant is simply a nullary function (i.e. needs no args)

Anatomy of a formula

We first form terms – individuals the logic talks about

- variables: x, y, etc.
- functions applied to (other) terms: $\mathit{size}(x)$, A, $\mathit{length}(A)$, etc.

Anatomy of a formula

We first form terms – individuals the logic talks about

- variables: x, y, etc.
- functions applied to (other) terms: $size(x)$, A, $length(A)$, etc.

We then form formulas – statements about individuals

- relations applied to terms: $elementOf(A,x)$, $t_1 = t_2$, etc.
- use prop. connectives: $elementOf(A,x) \rightarrow elementOf(\mathsf{L},x)$
- apply quantifiers: $\forall x.\ (elementOf(A,x) \rightarrow elementOf(\mathsf{L},x))$

Example formulas

1. Every student is younger than some instructor
2. Every module has at least one lecturer
3. Every module has exactly one module organizer
4. For every integer one can find a larger integer
5. If an integer is even then it is divisible by 4
6. If an integer is divisible by 4 then it is even
7. for some positive integer n: $2^n < n^2$
8. for all positive integers n: $2^n \geq n^2$

Example formulas

1. $\forall x.\ \mathit{student}(x) \rightarrow \exists y.\ \mathit{instructor}(y) \land \mathit{younger}(x,y)$

2. $\forall x.\ \mathit{module}(x) \rightarrow \exists y.\ \mathit{lecturer}(y,x)$

3. $\forall x.\ \mathit{module}(x) \rightarrow \exists y.\ \mathit{organizer}(y,x) \land \forall z.\ \mathit{organizer}(z,x) \rightarrow z = y$

4. $\forall x.\ \mathit{integer}(x) \rightarrow \exists y.\ \mathit{integer}(y) \land x < y$

5. $\forall x.\ (\mathit{integer}(x) \land \mathit{even}(x)) \rightarrow \mathit{divisible}(x,4)$

6. $\forall x.\ (\mathit{integer}(x) \land \mathit{divisible}(x,4)) \rightarrow \mathit{even}(x)$

7. $\exists n.\ \mathit{integer}(x) \land n > 0 \land \exp(2,n) < \exp(n,2)$

8. $\forall n.\ (\mathit{integer}(x) \land n > 0) \rightarrow (\exp(2,n) \geq \exp(n,2))$

Predicate includes Propositional

Why?

- *E.g. here is a formula in propositional logic:* $A \rightarrow B$
- *and in predicate logic?*

Predicate includes Propositional

Why?

- *E.g. here is a formula in propositional logic:* $A \to B$
- *and in predicate logic?*
    - $A \to B$ *where* A *and* B *are relations with 0 arguments*

But in predicate logic we can express more:

- $\forall x.\ A(x) \to B(x)$

Formal definition

Given a **vocabulary** V of function and relation symbols, where

- each symbol has an *arity* (i.e. takes a specific number of arguments)

a **term** t over V is:

a **formula** φ over V is:

Formal definition

Given a **vocabulary** V of function and relation symbols, where

- each symbol has an *arity* (i.e. takes a specific number of arguments)

a **term** t over V is:

- a variable: $x, \; y, \; z, \; \ldots$
- a function symbol applied to terms: $f(t_1, \ldots, t_n)$

a **formula** φ over V is:

- a relation symbol applied to terms: $P(t_1, \ldots, t_n)$
- a propositional connective applied to formulas: $\varphi_1 \wedge \varphi_2, \; \varphi_1 \vee \varphi_2, \; \varphi_1 \rightarrow \varphi_2, \; \neg \varphi_1$
- a quantification of a formula: $\forall x . \varphi_1$, $\exists x . \varphi_1$

Formal definition

$$t \;=\; x \;\mid\; f(t,\ldots,t)$$

$$\varphi \;=\; P(t,\ldots,t) \mid \neg\varphi \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid \varphi \to \varphi \mid \forall x.\varphi \mid \exists x.\varphi$$

Equality

A relation we typically want to have in our vocabulary is equality:

$$t_1 = t_2$$

The intended meaning is: the terms on the two sides of the equality symbol are equal

The relation here is “=”. But we write “$t_1 = t_2$” and not “$=(t_1, t_2)$”

Predicate logic example

Consider the statement:

> *For all x and all y, if x is the father of m and if y is a daughter of x, then y is a sister of m*

Predicate logic encoding using $V_1 = \{m, f, S, D, =\}$

Predicate logic example

Consider the statement:

For all x and all y, if x is the father of m and if y is a daughter of x, then y is a sister of m

Predicate logic encoding using $V_1 = \{m, f, S, D, =\}$:

- m is a constant symbol (i.e. a nullary function), representing an individual (i.e. a person)
- f is a unary function symbol , representing the Father function
- S, D and $=$ are binary relation symbols (Sister, Daughter, equality)
- Possible encoding:
  - $\forall x,y.\ (x = f\,(m) \wedge D(y, x)) \to S(y, m)$

    note: $\forall x,y.$ is just an abbreviation for $\forall x.\forall y.$

Predicate logic example

Consider the statement:

For all x and all y, if x is the father of m and if y is a daughter of x, then y is a sister of m

Predicate logic encoding using $V_2 = \{m, f, S, D\}$:

- these have the same intended meaning as before

Predicate logic example

Consider the statement:

> *For all x and all y, if x is the father of m and if y is a daughter of x, then y is a sister of m*

Predicate logic encoding using $V_2 = \{m, f, S, D\}$:

- these have the same intended meaning as before
- Possible encoding:
  - $\forall y.\ D(y, f(m)) \rightarrow S(y, m)$

Variables

Let us focus more on the role of variables.

- A variable stands for some individual that we do not specify.
- We then use a quantifier to *bind* that variable, i.e. specify that we meant it to stand for all ($\forall$) or for some ($\exists$) individual.

E.g. in this formula:

$$\forall x. \forall y.\ (x = f(m)) \wedge D(y, x) \rightarrow S(y, m)$$

the variables x and y are **bound** by the $\forall$ quantifier.

Variables

Let us focus more on the role of variables.

- A variable stands for some individual that we do not specify.
- We then use a quantifier to *bind* that variable, i.e. specify that we meant it to stand for all (∀) or for some (∃) individual.

free and bound variables in Maths and CS:

Algebra: $\sum_{i=1}^{n} 2^{i+k}$
n and k are free variables, i is a bound variable
Calculus: $\int_a^b c + sin(2t+1)dt$
a, b, c are free variables, t is a bound variable
Programming: global and local variables

E.g. in this formula:

$$\forall x.\forall y.\ (x = f(m)) \wedge D(y, x) \rightarrow S(y, m)$$

the variables x and y are **bound** by the ∀ quantifier.

A variable that is not bound is called **free**. E.g. x is free in:

$$\forall y.\ (x = f(m)) \wedge D(y, x) \rightarrow S(y, m)$$

Semantics of formulas

We will next look at the **semantics** of predicate logic formulas

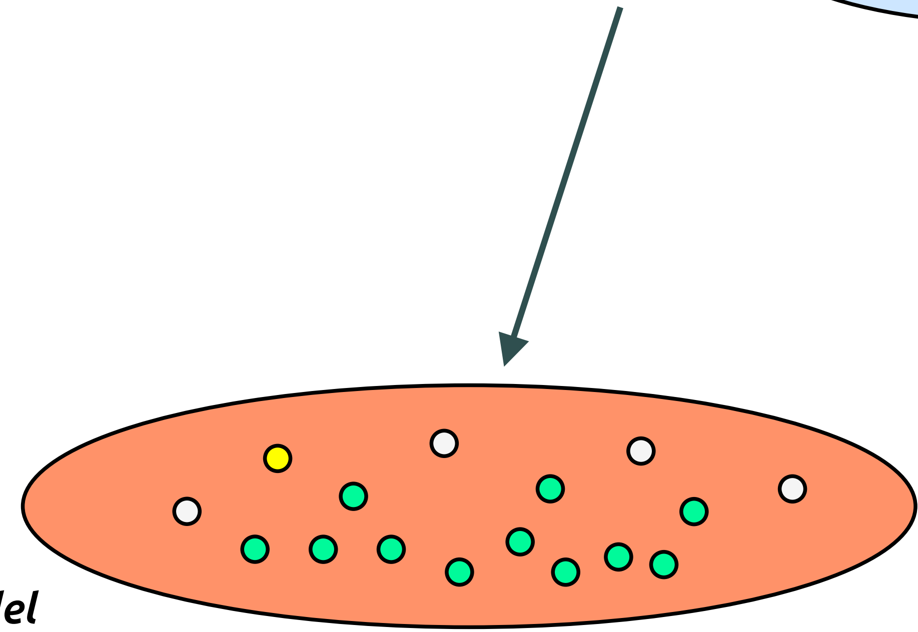
Logic

- vocabulary V
- formulas

$$t \;=\; x \;\mid\; f(t,\ldots,t)$$

$$\varphi \;=\; P(t,\ldots,t) \mid \neg\varphi \mid \varphi\wedge\varphi \mid \varphi\vee\varphi \mid \varphi\rightarrow\varphi \mid \forall x.\varphi \mid \exists x.\varphi$$

$$\forall x.\ student(x) \rightarrow \exists y.\ instructor(y) \wedge younger(x,y)$$

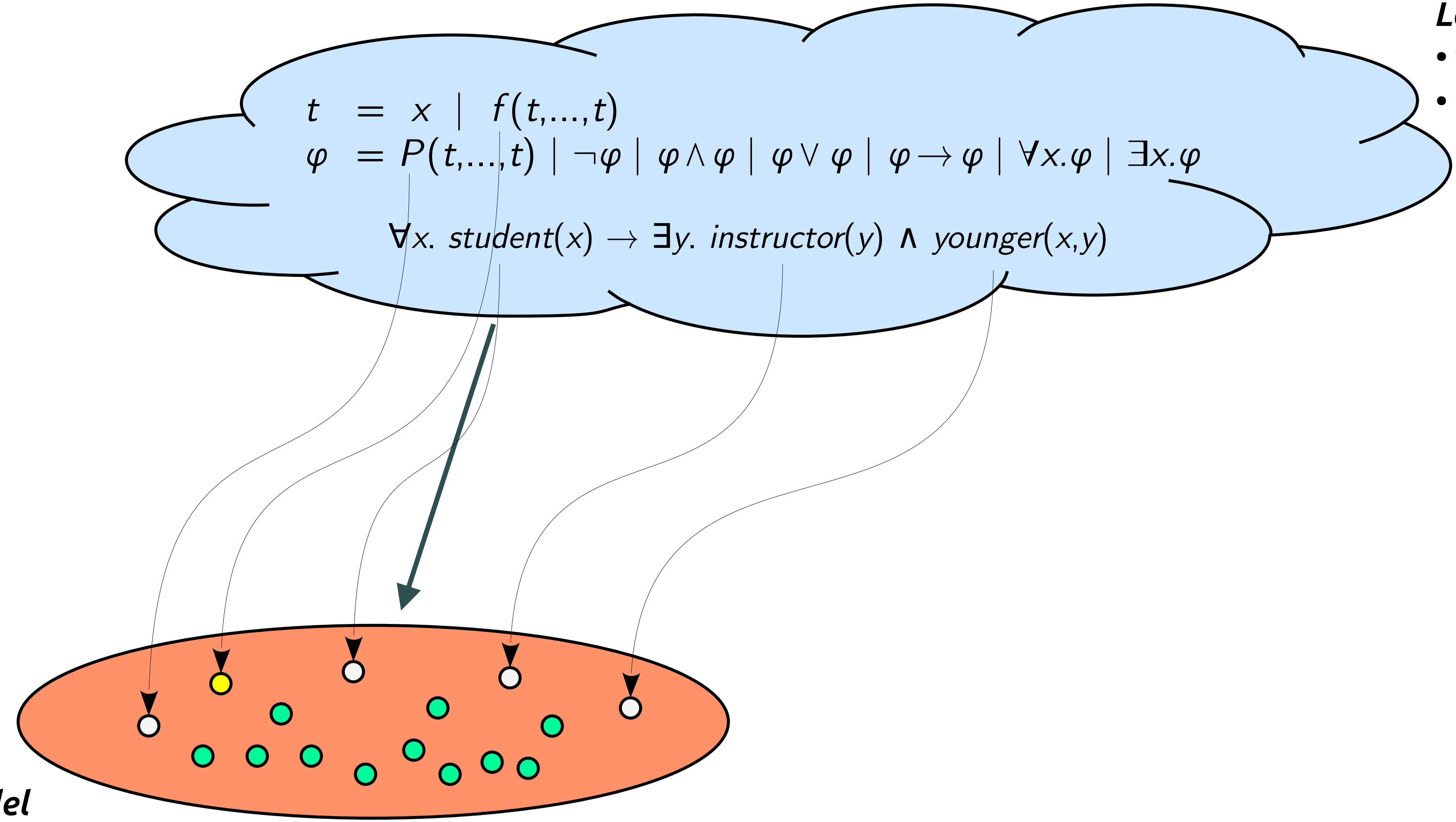


Model

- universe U
- func/rel interpretations

Semantics of formulas

We will next look at the **semantics** of predicate logic formulas



Logic
- vocabulary V
- formulas

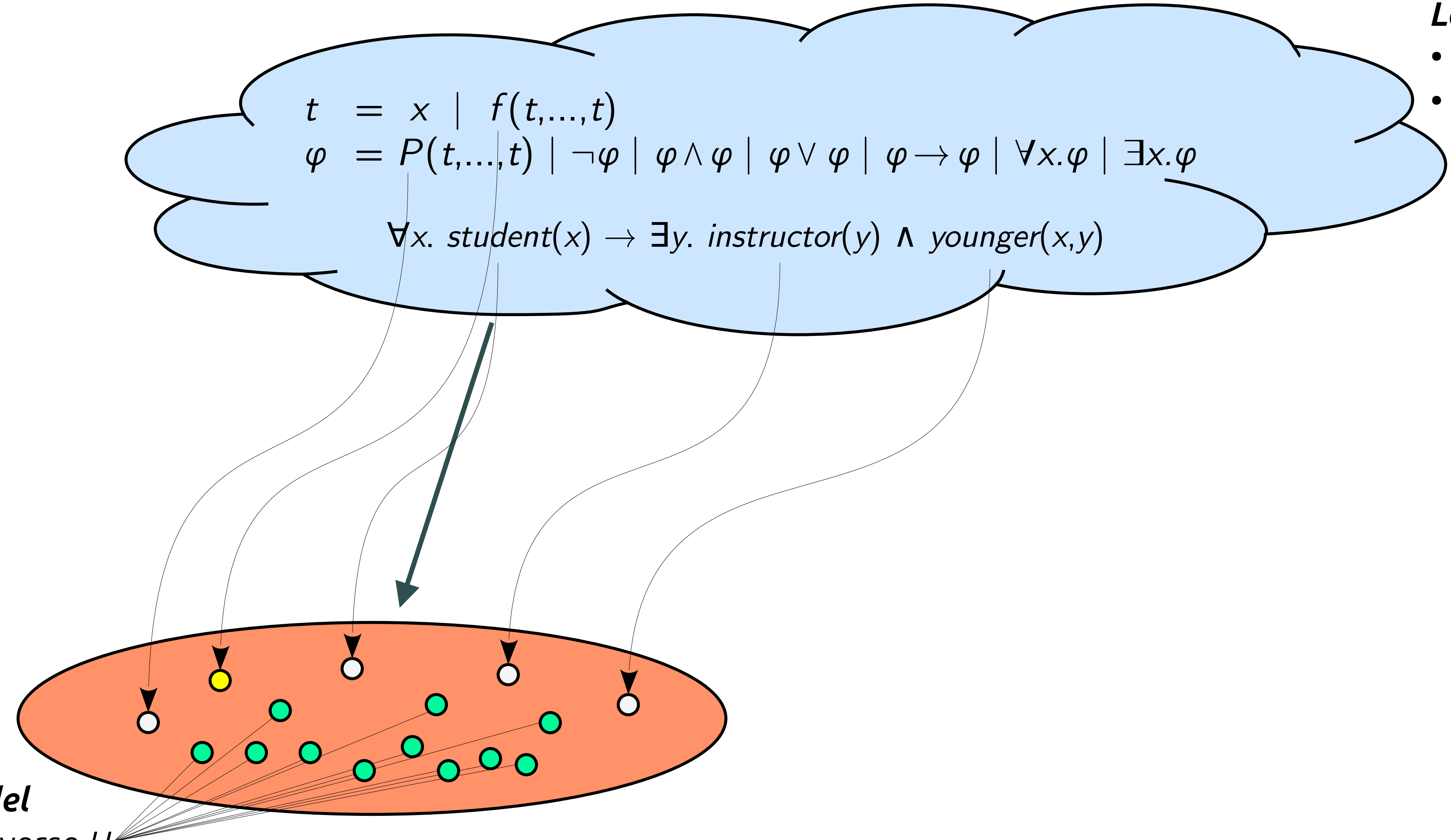
Model
- universe U
- func/rel interpretations

Semantics of formulas

We will next look at the **semantics** of predicate logic formulas

Logic

- vocabulary V
- formulas



Model

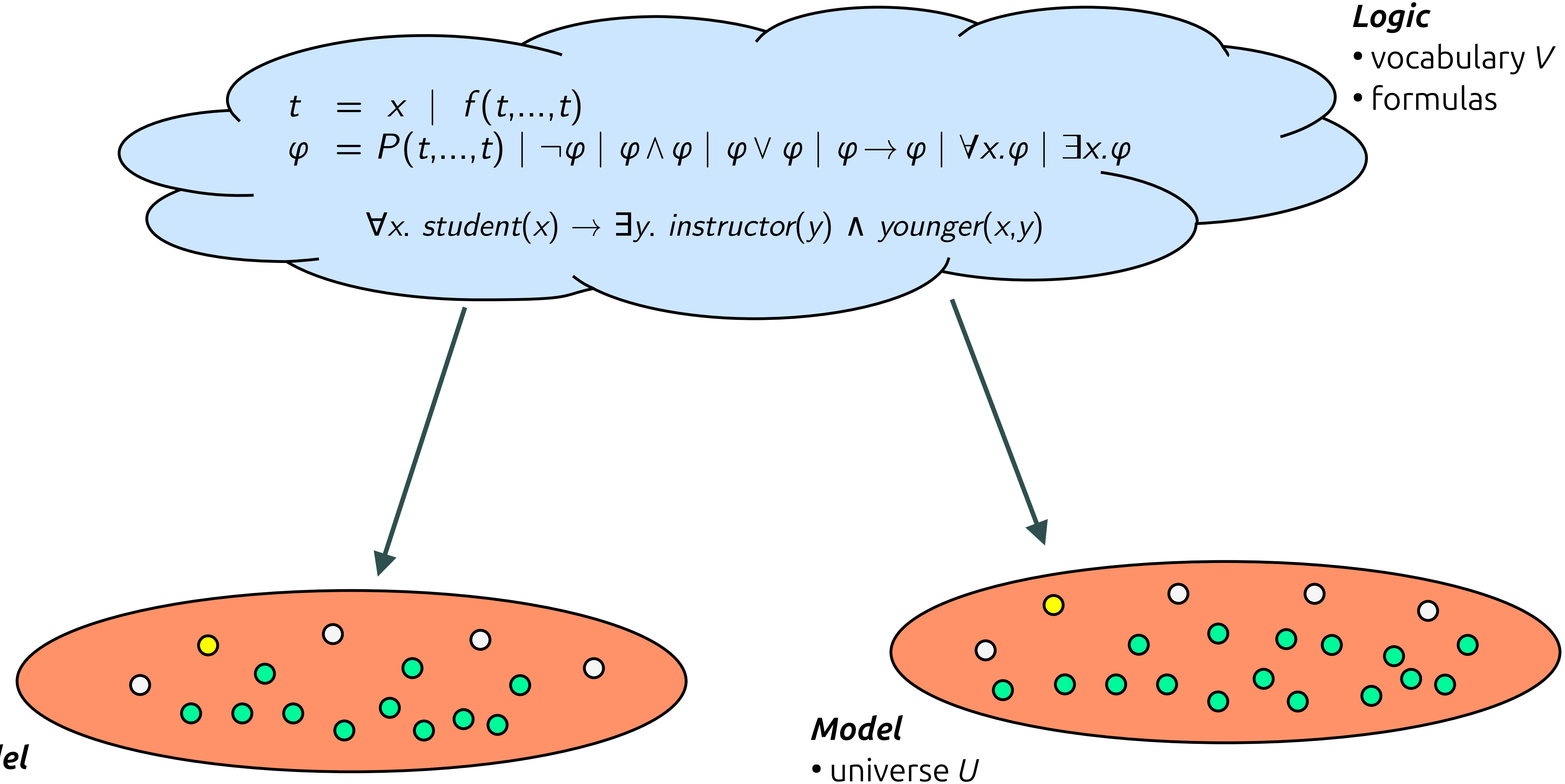
- universe U
- func/rel interpretations

Semantics of formulas

We will next look at the **semantics** of predicate logic formulas

$$t \;=\; x \;\mid\; f(t,\ldots,t)$$
$$\varphi \;=\; P(t,\ldots,t) \mid \neg\varphi \mid \varphi \wedge \varphi \mid \varphi \vee \varphi \mid \varphi \rightarrow \varphi \mid \forall x.\varphi \mid \exists x.\varphi$$

$$\forall x.\ student(x) \rightarrow \exists y.\ instructor(y) \wedge younger(x,y)$$

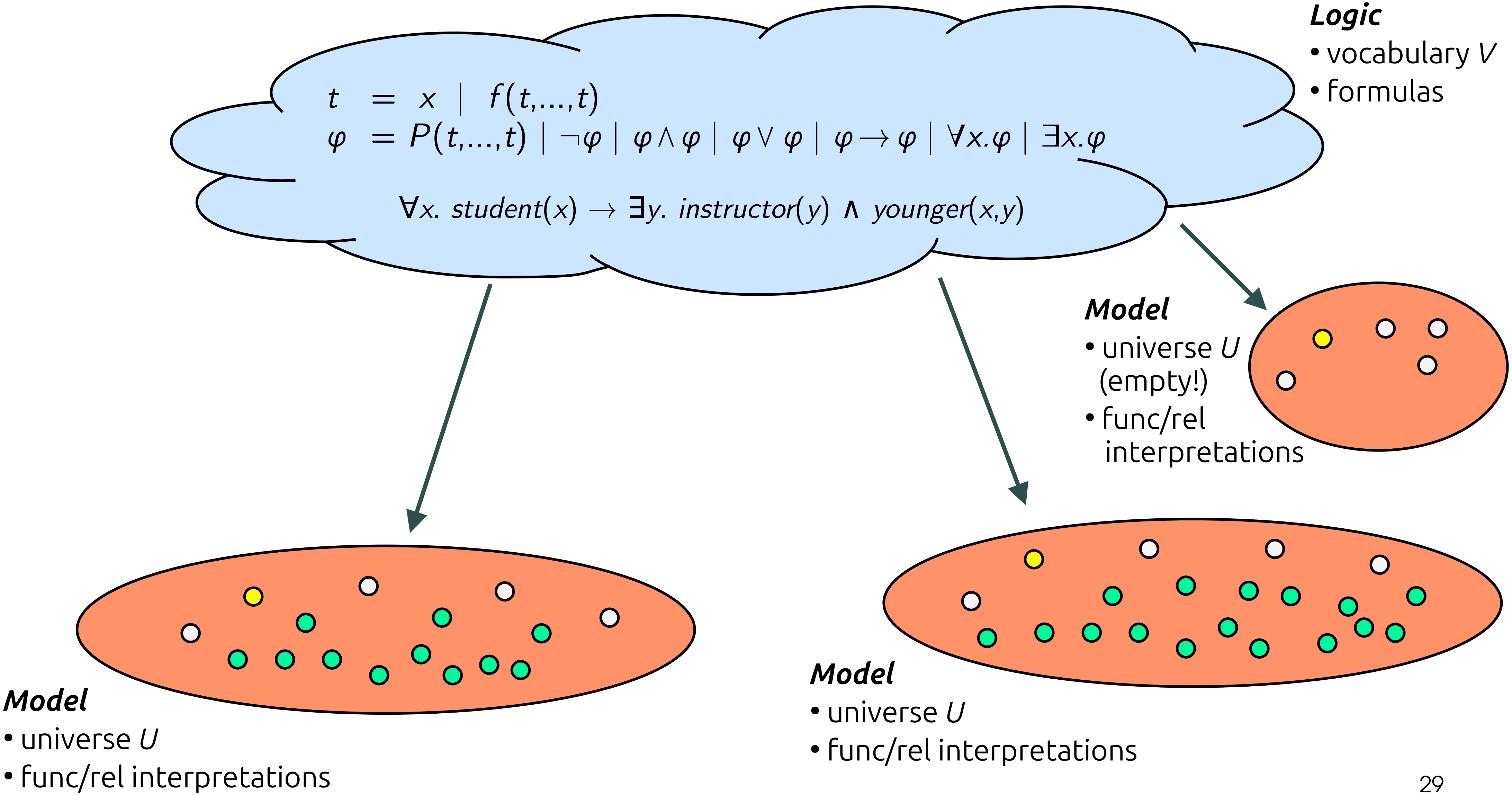


Semantics of formulas

We will next look at the **semantics** of predicate logic formulas

$$t \;=\; x \mid f(t,\ldots,t)$$
$$\varphi \;=\; P(t,\ldots,t) \mid \neg\varphi \mid \varphi \land \varphi \mid \varphi \lor \varphi \mid \varphi \to \varphi \mid \forall x.\varphi \mid \exists x.\varphi$$

$$\forall x.\ student(x) \to \exists y.\ instructor(y) \land younger(x,y)$$



Semantics of formulas

We will next look at the **semantics** of predicate logic formulas:

- this means *the meaning* of formulas, and can be true or false
- but it depends on the *context*, e.g. is this true?

$$\forall x.\ student(x) \rightarrow \exists y.\ instructor(y) \wedge younger(x,y)$$

well, it depends on:

- who the students and the instructors are
- and what is their age!

So, to give meaning to predicate logic formulas we first need to specify the setting we are talking about – we call this a **model**

Example models

$$\forall x.\ student(x) \rightarrow \exists y.\ instructor(y) \land younger(x,y)$$

with the vocabulary $V = \{\ student,\ instructor,\ younger\ \}$

1. one model could be:

    - $U = \{$ Nikos, Pasquale, Adam, Maz, Aisha $\}$
    - instructors: Nikos, Pasquale
    - students: Adam, Aisha, Maz
    - younger relation: (Adam, Nikos), (Adam, Pasquale), (Adam, Maz), (Aisha, Adam), (Aisha, Nikos), (Maz, Nikos)

is the formula true in this model?

Example models

$$\forall x.\ \mathit{student}(x) \rightarrow \exists y.\ \mathit{instructor}(y)\ \wedge\ \mathit{younger}(x,y)$$

with the vocabulary $V = \{\ \mathit{student},\ \mathit{instructor},\ \mathit{younger}\ \}$

2. another model could be:

- $U = \{$ Arthur, Adam, Maz, Aisha $\}$
- instructors: Arthur
- students: Adam, Aisha, Maz
- younger relation: (Adam, Arthur), (Adam, Maz), (Aisha, Adam), (Aisha, Arthur)

is the formula true in this model?

Models formally

Formally, given a vocabulary V, a **model** M for V consists of:

- a set U of individuals – we call this the *universe*
- an *interpretation* of each function and relation in V, i.e:
  - for each function symbol, we have an *actual function* over the universe U
  - for each relation symbol, we have an *actual relation* over the universe U

Models formally

Formally, given a vocabulary V, a **model** M for V consists of:

- a set U of individuals – we call this the *universe*
- an *interpretation* of each function and relation in V, i.e:
  - for each function symbol f with arity k, we have a function

    $$f_M\colon U^k \to U$$

  - for each relation symbol P with arity k, we have a relation

    $$P_M \subseteq U^k$$

So, the model is : $M = (U,\ f_M\ ,\ \ldots,\ P_M\ ,\ \ldots\ )$

Example models

$$\forall x.\ \mathit{student}(x) \rightarrow \exists y.\ \mathit{instructor}(y) \wedge \mathit{younger}(x,y)$$

with the vocabulary $V = \{\ \mathit{student},\ \mathit{instructor},\ \mathit{younger}\ \}$

1. one model could be M with:

- $U = \{$ Nikos, Pasquale, Adam, Maz, Aisha $\}$
- $\mathit{instructor}_M = \{$ Nikos, Pasquale $\}$
- $\mathit{student}_M = \{$ Adam, Aisha, Maz $\}$
- $\mathit{younger}_M = \{$ (Adam, Nikos), (Adam, Pasquale), (Adam, Maz), (Aisha, Adam), (Aisha, Nikos), (Maz, Nikos) $\}$

Example models

$$\forall x.\ student(x) \rightarrow \exists y.\ instructor(y) \wedge younger(x,y)$$

with the vocabulary $V = \{\ student,\ instructor,\ younger\ \}$

2. another model could be M with:

- $U = \{$ Arthur, Adam, Maz, Aisha $\}$
- $instructor_M = \{$ Arthur $\}$
- $student_M = \{$ Adam, Aisha, Maz $\}$
- $younger_M = \{$ (Adam, Arthur), (Adam, Maz), (Aisha, Adam), (Aisha, Arthur) $\}$

Functions and Relations

- A function $g_1 : U \rightarrow U$ maps elements of U to elements of U
- A function $g_2 : U^2 \rightarrow U$ maps pairs of elements of U to elements of U
- *etc.*
- a relation $P_1 \subseteq U$ is a set of elements of U
- a relation $P_2 \subseteq U^2$ is a set of pairs of elements of U
- *etc.*

Functions and Relations

- A function $g_1 : U \rightarrow U$ maps elements of U to elements of U
- A function $g_2 : U^2 \rightarrow U$ maps pairs of elements of U to elements of U
- *etc.*
- a relation $P_1 \subseteq U$ is a set of elements of U (i.e. $P_1 : U \rightarrow \{\text{true, false}\}$)
- a relation $P_2 \subseteq U^2$ is a set of pairs of elements of U (i.e. $P_2 : U^2 \rightarrow \{\text{true, false}\}$)
- *etc.*

Functions and Relations

- A function $g_1 : U \to U$ maps elements of U to elements of U
- A function $g_2 : U^2 \to U$ maps pairs of elements of U to elements of U
- *etc.*
- a relation $P_1 \subseteq U$ is a set of elements of U (i.e. $P_1 : U \to \{\text{true, false}\}$)
- a relation $P_2 \subseteq U^2$ is a set of pairs of elements of U (i.e. $P_2 : U^2 \to \{\text{true, false}\}$)
- *etc.*

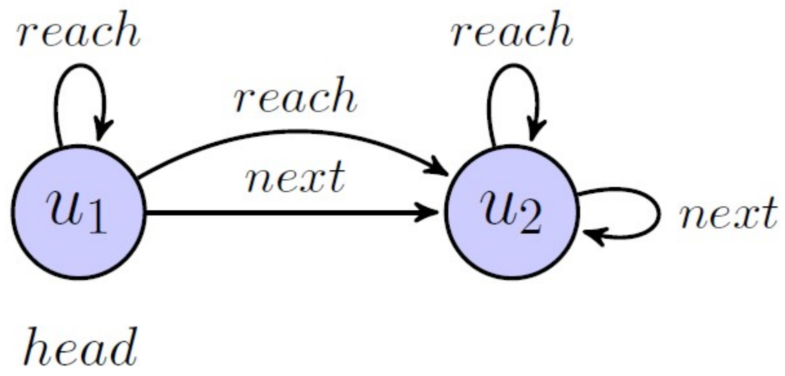
In any model $M = (U, \ldots)$, unless specified otherwise, the equality relation is interpreted to equality in U:

$$=_M \; = \; \{(u,\ u) \mid u \text{ in } U\}$$

A CS example (cyclic lists)

Consider the vocabulary $V = \{\ \mathit{head},\ \mathit{next},\ \mathit{reach}\ \}$ where:

- *head* is a constant (meaning the head, or first element, of a list)
- *next* is a unary function (giving the next element in the list)
- *reach* is a binary relation (*reach*(x,y) means we can reach y from x in the list)

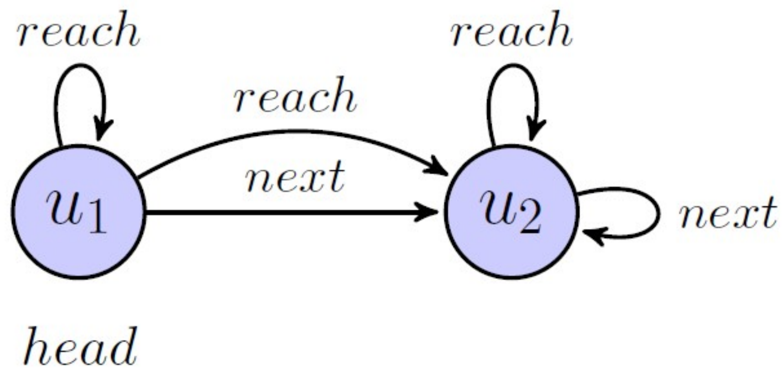


A CS example (cyclic lists)

Consider the vocabulary $V = \{ \ head, \ next, \ reach \ \}$ where:

- *head* is a constant (meaning the head, or first element, of a list)
- *next* is a unary function (giving the next element in the list)
- *reach* is a binary relation (*reach*(x,y) means we can reach y from x in the list)

Here is a model U:



- $U = \{ \ u_1, u_2 \ \}$
- $head_M = u_1$
- $next_M$ is such that $next_M(u_1) = next_M(u_2) = u_2$
- $reach_M$ is given by $reach_M = \{ \ (u_1,u_1),(u_1,u_2),(u_2,u_2) \ \}$

In such cases, it helps to draw the model, as on the right above!

Semantics formally

$$\forall x.\ \mathit{student}(x) \rightarrow \exists y.\ \mathit{instructor}(y) \land \mathit{younger}(x,y)$$

We gave informal arguments of why the formula is/isn't true in given models. Can this be made formal?

Yes, there is a *recursive* way to go through the formula in order to establish its truth value.

Given a model M and a formula φ, we write:

$$M \vDash \varphi$$

to mean that φ is true in M. We may also say that M satisfies φ.

We next look at how to define $M \vDash \varphi$.

The need for environments

We next look at how to define $M \vDash \varphi$.

We need just one more ingredient:

- recall that our definition will go down the formula φ
- e.g. if φ is of the form $\forall x.\varphi_1$, then we would like to say:
  - $M \vDash \forall x.\varphi_1$ holds if $M \vDash \varphi_1$ holds for *every value* that x can take (from the universe)
  - so, we need to remember what values our variables are supposed to take
- we therefore use an *environment* E that maps variables to elements of the universe U (think of this as a look-up table)

Semantics formally

We therefore define $M, E \models \varphi$. We look at the form of φ:

- if φ is $P(t_1, \ldots, t_n)$ then interpret $t_1, \ldots, t_n$ into elements $u_1, \ldots, u_n$ of U, by:
  - replacing each variable x with its stored value $E(x)$
  - replacing each function f with its interpretation f_M

  Then, $M, E \models P(t_1, \ldots, t_n)$ holds if $(u_1, \ldots, u_n)$ is in P_M

Semantics formally

We therefore define $M,E \vDash \varphi$. We look at the form of φ:

- if φ is $P(t_1,\ldots,t_n)$ then interpret $t_1,\ldots,t_n$ into elements $u_1,\ldots,u_n$ of U, [...]

  Then, $M,E \vDash P(t_1,\ldots,t_n)$ holds if $(u_1,\ldots,u_n)$ is in P_M

- if φ is one of $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$, $\varphi_1 \rightarrow \varphi_2$, $\neg\varphi_1$ then:
  - $M,E \vDash \neg\varphi_1$ holds if $M,E \vDash \varphi_1$ does not hold
  - $M,E \vDash \varphi_1 \wedge \varphi_2$ holds if $M,E \vDash \varphi_1$ and $M,E \vDash \varphi_2$ hold
  - $M,E \vDash \varphi_1 \vee \varphi_2$ holds if $M,E \vDash \varphi_1$ or $M,E \vDash \varphi_2$ holds
  - $M,E \vDash \varphi_1 \rightarrow \varphi_2$ holds if $M,E \vDash \varphi_1$ doesn't hold or $M,E \vDash \varphi_2$ holds

Semantics formally

We therefore define $M,E \vDash \varphi$. We look at the form of φ:

- if φ is $P(t_1,\ldots,t_n)$ then interpret $t_1,\ldots,t_n$ into elements $u_1,\ldots,u_n$ of U, [...]

  Then, $M,E \vDash P(t_1,\ldots,t_n)$ holds if $(u_1,\ldots,u_n)$ is in P_M

- if φ is one of $\varphi_1 \wedge \varphi_2$, $\varphi_1 \vee \varphi_2$, $\varphi_1 \to \varphi_2$, $\neg\varphi_1$ then use propositional logic rules
- if φ is a quantification of a formula, i.e. $\forall x.\varphi_1$ or $\exists x.\varphi_1$ then:
  - $M,E \vDash \forall x.\varphi_1$ holds if, for all u in U: $M,E[x \mapsto u] \vDash \varphi_1$ holds
  - $M,E \vDash \exists x.\varphi_1$ holds if, for some u in U: $M,E[x \mapsto u] \vDash \varphi_1$ holds

Example revisited

$$\varphi \;=\; \forall x.\; student(x) \rightarrow \exists y.\; instructor(y) \wedge younger(x,y)$$

Consider M as below and let $[\,]$ be the empty environment.

- $U = \{$ Nikos, Pasquale, Adam, Maz, Aisha $\}$
- $instructor_M = \{$ Nikos, Pasquale $\}$
- $student_M = \{$ Adam, Aisha, Maz $\}$
- $younger_M = \{$ (Adam, Nikos), (Adam, Pasquale), (Adam, Maz), (Aisha, Adam), (Aisha, Nikos), (Maz, Nikos) $\}$

so $M,[\,] \vDash \varphi \;\Leftrightarrow\;$ for all u in U,

$$M,[x \mapsto u] \vDash student(x) \rightarrow \exists y.\; instructor(y) \wedge younger(x,y)$$

and now we can check that this is true by examining each u in U

e.g. $M,[x \mapsto \text{Nikos}] \vDash student(x) \rightarrow \exists y.\; instructor(y) \wedge younger(x,y)$

holds because $M,[x \mapsto \text{Nikos}] \vDash student(x)$ does not hold

why not? Because Nikos is not in $student_M$!

Example revisited

$$\varphi \;=\; \forall x.\ student(x) \rightarrow \exists y.\ instructor(y) \wedge younger(x,y)$$

Consider M as below and let $[\ ]$ be the empty environment.

- $U = \{$ Nikos, Pasquale, Adam, Maz, Aisha $\}$
- $instructor_M = \{$ Nikos, Pasquale $\}$
- $student_M = \{$ Adam, Aisha, Maz $\}$
- $younger_M = \{$ (Adam, Nikos), (Adam, Pasquale), (Adam, Maz), (Aisha, Adam), (Aisha, Nikos), (Maz, Nikos) $\}$

so $M,[\ ] \vDash \varphi \;\Leftrightarrow\;$ for all u in U,

$$M,[x \mapsto u] \vDash student(x) \rightarrow \exists y.\ instructor(y) \wedge younger(x,y)$$

and now we can check that this is true by examining each u in U

e.g. $M,[x \mapsto \text{Adam}] \vDash student(x) \rightarrow \exists y.\ instructor(y) \wedge younger(x,y)$

holds because $M,[x \mapsto \text{Adam}] \vDash \exists y.\ instructor(y) \wedge younger(x,y)$ holds

which holds because $M,[x \mapsto \text{Adam}, y \mapsto \text{Nikos}] \vDash instructor(y) \wedge younger(x,y)$

holds (as Nikos is in $instructor_M$ and (Adam,Nikos) is in $younger_M$)!

Summary

We introduced predicate logic, which allows us to talk about individuals and their properties, and to quantify over them

We looked at different vocabularies and thus different predicate logic syntaxes

We examined the semantics of a predicate logic formula and we saw that this is defined with respect to a specific model, which needs to be given; we also looked at different models

With models defined, we defined the semantics of a predicate logic formula (formally and informally), and looked at some examples

More examples in the exercises!