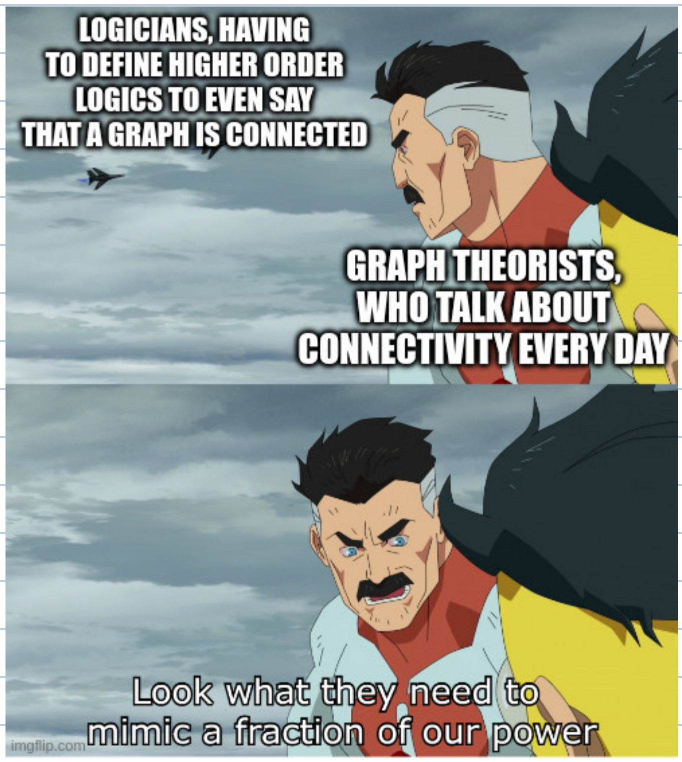
CS 245: TUT 105 - Tutorial 07

Last time:
- First-order logic: Motivation
- FOL syntax: language, signatures, terms, formulas, parse trees,
- (informal) e.g.'s of theories

This time:
- More syntax
- FOL semantics
- More examples of theories

Meme(s) of the week:



Review:

Last time, we introduced *signatures*… these are collections of constant, function, and relation symbols that together specify a given first order language:

$L(\sigma)$

FOLang over σ signature. $\sigma = (\underbrace{\mathcal{C} \cup \mathcal{F} \cup \mathcal{R}}, \overbrace{\mathrm{ary}()})$

arity function

constant, function, relation symbols

We defined FOLangs for a given signature by first defining terms over that signature, then formulas— both were inductive definitions.

We also defined what it means for formulas to be ***atomic*** & ***compound***, ***free*** & ***bound***.

E.g.: $\forall x \forall y \left((x+\pi) = (y+\pi) \rightarrow (P(z) = w + y) \right)$

has bound vars. x & y, free vars z & w.

It is NOT a sentence.

Formulas without free variables (i.e., all vars bound) are called ***sentences***.
Only sentences can be given an *interpretation* (T or F) — see next section.

Formulas which are not sentences can be thought of as specifying definitions for objects (when interpreted).

In other words, if a term subbed into a free variable of a formula gives T when interpreted, then the term can be thought of as "satisfying" the property specified by that formula.

E.g.:

$\mathrm{Prime}(x) := (x>1) \wedge \forall y \forall z \left((x = (y \cdot z)) \rightarrow (y = 1 \vee y = z) \right)$

a shorthand for this formula.

You are free to define formula shorthands like this, and sub them into other formulas or sentences.

Other useful shorthands / conventions when writing formulas:

- Given finite set I, $\{\varphi_i\}_{i \in I} \subseteq L(\sigma)$,

$$\bigwedge_{i \in I} \varphi_i := \varphi_{i_1} \wedge \varphi_{i_2} \wedge \dots \wedge \varphi_{i_{|I|}}$$

$$\bigvee_{i \in I} \varphi_i := \varphi_{i_1} \wedge \varphi_{i_2} \wedge \dots \wedge \varphi_{i_{|I|}} \, .$$

- You are also free to use ellipses, as long as it is clear and unambiguous what formula you are trying to specify— bracket carefully.

E.g.: $p_1 \to (p_2 \to (p_3 \to (\cdots \to p_n))\cdots)$.

- You may write "$t_1 \neq t_2$" instead of "$\neg(t_1 = t_2)$", t_1, t_2 tems.
- $\nexists x \varphi$ instead of $\neg(\exists x \varphi)$, etc.

Another syntactic notion that was missed by the notes: **substitution**.

Def.: If $\varphi \in L(\sigma)$, y is a free variable and t a term, then $\phi(t) = \phi_{[y/t]}$ is the formula obtained by subbing t in for all instances of y in φ.

E.g.: If $\varphi = \forall x(x + w = w + x)$, then

$$\varphi_{[w/0]} = \forall x(x + 0 = 0 + x).$$

Now, we would like for the formulas we write down to have some meaning. For this we need *semantics*.

Semantics:

First, we want to be able to interpret our signature as genuine mathematical *structures* encoding constants, functions, and relations.

Def'n: The structure $\mathcal{M}$ for a signature σ specifies:

- The domain D, a non-empty set of elements.
- An elt $c^{\mathcal{M}} \in D$, for each $c \in \mathcal{C}$.
- A function $f^{\mathcal{M}} : D^n \longrightarrow D$, for each n-ary $f \in \mathcal{F}$.
- A subset $P^{\mathcal{M}} \subseteq D^n$, for each n-ary $P \in \mathcal{R}$.

Note that domains can be anything… finite, infinite, countable, uncountabl— as long as they're nonempty. Weird, pathological things happen if we allow empty domains.
Also, we can assign the same value to multiple distinct constant symbols.

E.g.: $\sigma = \{0, S, +, \times\}$ w/ different structures...

"natural" interpretation vs. with fruits.

$\mathcal{M}^{\mathbb{N}} = (\underset{D}{\mathbb{N}}, 0^{\mathcal{M}}, +^{\mathcal{M}}, S^{\mathcal{M}}, \times^{\mathcal{M}})$ vs. $\mathcal{M}^{f} = (\{\text{apple}, \text{banana}, \text{cherry}\}, S^{\mathcal{M}^f}, 0^{\mathcal{M}^f}, +^{\mathcal{M}^f}, \times^{\mathcal{M}^f})$

Def'n: Given a structure $\mathcal{M}$ on $L(\sigma)$, the valuation of a sentence φ with respect $\mathcal{M}$ is:

- *Evaluation of terms.* For any constant c, function f, and term t,

$$\begin{aligned} \mathrm{val}_{\mathcal{M}}(c) &= c^{\mathcal{M}} \\ \mathrm{val}_{\mathcal{M}}(f(t)) &= f^{\mathcal{M}}(\mathrm{val}_{\mathcal{M}}(t)) \end{aligned}$$

- *Evaluation of predicates.* For any terms $t_1, t_2, \ldots, t_n$ and predicate P,

$$\mathrm{val}_{\mathcal{M}}((t_1 = t_2)) = \begin{cases} True & \text{if } \mathrm{val}_{\mathcal{M}}(t_1) = \mathrm{val}_{\mathcal{M}}(t_2) \\ False & \text{otherwise} \end{cases}$$

$$\mathrm{val}_{\mathcal{M}}(P(t_1, \ldots, t_n)) = \begin{cases} True & \text{if } (\mathrm{val}_{\mathcal{M}}(t_1), \ldots, \mathrm{val}_{\mathcal{M}}(t_n))) \in P^{\mathcal{M}} \\ False & \text{otherwise} \end{cases}$$

- *Evaluation of quantifiers.* For any variable v and formula ϕ, if we let $\phi_{[v/d]}$ denote the formula obtained by replacing all free occurrences of v in ϕ with d for some element $d \in D$, then

$$\mathrm{val}_{\mathcal{M}}(\exists v\, \phi) = \begin{cases} True & \text{if } \exists d \in D \text{ such that } \mathrm{val}_{\mathcal{M}}(\phi_{[v/d]}) = True \\ False & \text{otherwise} \end{cases}$$

$$\mathrm{val}_{\mathcal{M}}(\forall v\, \phi) = \begin{cases} True & \text{if } \forall d \in D, \mathrm{val}_{\mathcal{M}}(\phi_{[v/d]}) = True \\ False & \text{otherwise} \end{cases}$$

- *Evaluation of connectives.* The evaluation of compound formulas follows the same rules as in propositional logic:

$$\begin{aligned} \mathrm{val}_{\mathcal{M}}(\phi \wedge \psi) &= \mathrm{val}_{\mathcal{M}}(\phi) \wedge \mathrm{val}_{\mathcal{M}}(\psi) \\ \mathrm{val}_{\mathcal{M}}(\phi \vee \psi) &= \mathrm{val}_{\mathcal{M}}(\phi) \vee \mathrm{val}_{\mathcal{M}}(\psi) \\ \mathrm{val}_{\mathcal{M}}(\neg\phi &= \neg \mathrm{val}_{\mathcal{M}}(\phi) \\ \mathrm{val}_{\mathcal{M}}(\phi \to \psi) &= \mathrm{val}_{\mathcal{M}}(\phi) \to \mathrm{val}_{\mathcal{M}}(\psi) \\ \mathrm{val}_{\mathcal{M}}(\phi \leftrightarrow \psi) &= \mathrm{val}_{\mathcal{M}}(\phi) \leftrightarrow \mathrm{val}_{\mathcal{M}}(\psi).. \end{aligned}$$

NOTE: Formulas with free variables do not get truth valuations.

Validity, Satisfiability, Consequence:

Def'n:

- $\mathcal{M}$ satisfies the sentence φ, denoted $\mathcal{M} \models \varphi$, when $\mathrm{val}_{\mathcal{M}}(\varphi) = T$.
- φ is satisfiable if there exists an $\mathcal{M}$ s.t. $\mathcal{M} \models \varphi$.
- φ is (logically) valid when every $\mathcal{M}$ satisfies φ.
  I.t.c., we write $\models \varphi$.

E.g.: $\sigma = \{S, 0, +, \times\}$ $\forall x_1 \forall x_2((x_1 + x_2) = (x_2 + x_1))$

satisfiable (via $\mathcal{M}^{\mathbb{N}}$), but NOT valid (e.g., $\mathcal{M}^{\text{fruit}}$)

__Def'n:__ φ is a __logical consequence__ of $\Gamma \subseteq L(\sigma)$, denoted

$$\Gamma \models \varphi,$$

if every structure that satisfies Γ satisfies φ.

__Axioms, models, theories:__

- Sets of sentences in a FOLang are called <u>***axioms***</u>.
- The <u>***(first-order) theory***</u> corresponding to a set of axioms is the set of sentences that are logical consequences of the set of axioms.

__Def'n:__ A __model__ for a theory T corresponding to a set of axioms Γ is a structure $\mathcal{M}$ which satisfies all elements of T.

~ End Review ~

Practice Problems:

① **The Theory of Graphs.** Let $\sigma_G := \{E\}$ ← binary predicate.

Let $\mathcal{M}_G := (V, E^G = E)$ where

$V = \{$(some set of vertices)$\}$

$E = \{$(some set of ordered pairs (u,v) : $u, v \in V$, denoting edges b/t vertices$\}$

Each graph $G = (V, E)$ gives a distinct σ_G-structure $\mathcal{M}_G$.

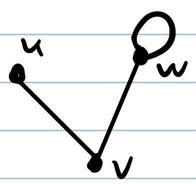
For each of the following English sentences,

- Translate it into FOL.
- Find a model for it (i.e., draw a graph satisfying that property).
- Write down the formal definition of the structure for this model.

(a) "The graph has ≥ 2 vertices."

$\varphi_a : \exists u \exists v (\neg (u = v))$.

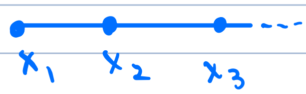
G:



$\mathcal{M}_G$: $V = \{u, v, w\}$

$E = \{(u,v), (v,w), (w,w)\}$

Alt.: G:



(infinite snake graph)

$\mathcal{M}_G$: $V = \{x_1, x_2, x_3, x_4, \ldots\}$

$E = \bigcup_{i=1}^{\infty} \{(x_i, x_{i+1})\}$

(b) "The graph has exactly one vertex."

$\varphi_b: \forall x \forall y (x = y)$

Note that because of the requirement that our domain is nonempty, we don't need to specify the clause that says there exists at least one vertex; this is automatic when interpreted.

OR

$\exists u [(u = u) \wedge \neg (\exists v (u \neq v))]$

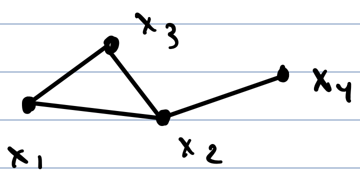
G: 

$\mathcal{M}_G$: $V = \{u\}$

$E = \emptyset$

(c) "The graph contains a triangle."

$\varphi_c: \exists u \exists v \exists w (E(u,v) \wedge E(v,w) \wedge E(w,u))$

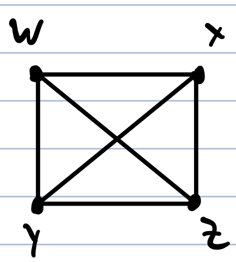
G: 

$\mathcal{M}_G$: $V = \{x_1, \dots, x_4\}$

$E = \{(x_1,x_2), (x_2,x_3), (x_3, x_1), (x_2, x_4)\}$

(d) "The graph is a complete graph, i.e. every vertex is connected to every other vertex."

$\varphi_d: \forall x \forall y ((x=y) \vee E(x,y))$

G: 

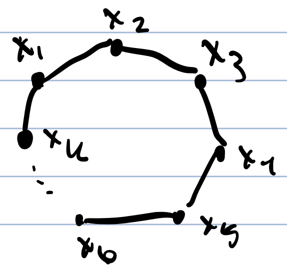
$\mathcal{M}_G$: $V = \{w, x, y, z\}$

$E = \{(w,x), (w,y), (w,z), (x,y), (x,z), (y,z)\}$

(e) "The graph has exactly k vertices."

$$\varphi_e: \quad \exists x_1 \exists x_2 \cdots \exists x_k \left[\left(\bigwedge_{1 \le i < j \le k} x_i \neq x_j \right) \wedge \forall u \left(\bigvee_{i=1}^{k} u = x_i \right) \right].$$

G:



$$\mathcal{M}_G: \quad V = \bigcup_{i=1}^{k} \{x_i\}$$

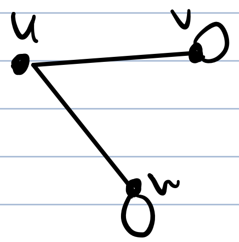
$$E = \bigcup_{i=1}^{k-1} \{(x_i, x_{i+1})\} \cup \{(x_k, x_1)\}.$$

(f) "The graph is k-regular, i.e. every vertex has exactly k edges." ← including self-loops!

$$\forall v \exists x_1 \exists x_2 \cdots \exists x_k \left[\bigwedge_{i=1}^{k} E(v, x_i) \wedge \nexists u \left(E(u,v) \wedge \bigwedge_{i=1}^{k} u \neq x_i \right) \right].$$

(w/ $k=2$)

G:



$\mathcal{M}_G$: ... Ez.

(g) "The graph doesn't have a self-loop."

$$\varphi_g: \quad \forall x \neg (E(x,x)).$$

G, $\mathcal{M}_G$: ez.

2.) Proof Question:

Is it possible to formulate the following sentences in our FOLang?

- "The graph has infinitely many vertices."
- "The graph is connected."
- The graph is *acyclic*, i.e. has no (non-self) loops."

Decide on your answer to these questions, and provide an (intuitive) proof.

Sol.: No. Intuitively, to address the first sentence, "The graph has infinitely many vertices," one would need to say that these vertices exist and are all pairwise distinct. This necessarily requires a formula of infinite length, which isn't allowed.

A similar argument applies for the other statements.

We will see how to prove these statements formally in a couple weeks, using the Compactness Theorem.

$\square$

~ Fin. ~