CHAPTER 1 STRUCTURES & LANGUAGE

1.2 Languages

$\boxed{\text{TPS}}$

$\mathbb{N} = \{0, 1, 2, \ldots\}$

① Write down a statement about the natural numbers — it must be T or F.

② Did you write it symbolically or did you use words? Try to write it symbolically — what symbols did you need?

③ Everyone repeat ② for (Goldbach's conj.):

Every integer greater than 2 is the sum of two prime numbers.

* My choices: $2, >, \mathbb{P}$, variables OR $2, +, 1, \ldots$

Follow-up: is it true?, can you prove it?

So what symbols might we want to talk about the natural numbers?

Perhaps: $+, \cdot, (,),$ variables, $\forall, \ldots$

Can you use these to express that every element of $\mathbb{N}$ is either even or one more than an even number?

Perhaps $(\forall a)\Big(\big((\exists b)(a = b \cdot (1+1))\big) \vee \big((\exists b)(a = b(1+1) + 1)\big)\Big)$

Def A first-order language $\mathcal{L}$ is an infinite collection of distinct symbols, no one of which is properly contained in another, of one of the following types

always included (implicitly)

1. Parentheses: (,)
2. Connectives: $\vee, \neg$! what about $\wedge$?
3. Quantifier: $\forall$! what about $\exists$?

aiming for some amount of efficiency

4. Variables: $v_1, v_2, \ldots, v_n, \ldots$ (denoted Vars)
5. Equality: $=$
6. Constant symbols: zero or more
7. Function symbols: zero or more n-ary function symbols for each pos. n
8. Relation symbols: zero or more n-ary rel. symbols for each pos. n

! $\exists$ is a placeholder for $\neg\forall\neg$. Similarly for $\wedge$.

Ex Language of Number Theory

$$\mathcal{L}_{NT} = \{0, S, +, \cdot, E, <\}$$

! Remember, symbols of types 1-5 are also implicity included

arity

arity		
	0	— constant
$a(S)=1$	S	— 1-ary (or unary) function
$a(+)=2$	$+$	— binary function
⋮	$\cdot$	— binary function
	E	— binary function
$a(<)=2$	$<$	— binary relation

intended meaning?

* This is all syntax right now! The symbols have no meaning, though they usually have an intended meaning.

TPS write down any sentence in $\mathcal{L}_{NT}$.

* that you haven't already.

TPS If $S(x) = x+1$, write a sentence in $\mathcal{L}_{NT}$ expressing that every natural number is even.

Note: you are assuming $S(x) = x+1$ and 0 is really zero and... so your sentence simply has an intended meaning.

TPS what other symbols might you want to have included to talk about $\mathbb{N}$?
Can you think of an important unary relation on $\mathbb{N}$? Can you think of any 4-ary relation? Can you define them in $\mathcal{L}_{NT}$?

* e.g. $Prime(x)$, $D(x,y,z,w) \Leftrightarrow xw - yz = 0$

Ex Language of Graphs

$\mathcal{L}_{Graph} = \{E\}$, E is a binary relation.

Q: what is the intended meaning of

$$(\forall v_1)(\exists v_2)(v_1 = v_2 \vee \neg E(v_1, v_2))\ ?$$

Ex Language of Set Theory

$\mathcal{L}_{ST} = \{\in\}$, $\in$ is a binary relation.

Q: why not include $\subseteq$?

... $x \subseteq y$ can be expressed by $(\forall v)(v \in x \Rightarrow v \in y)$

so the relation $\subseteq$ can be "defined" from $\mathcal{L}_{ST}$

A: There are choices, and here we tried to minimize the # of symbols.

1.3 Terms & Formulas

Once we pick a language, we can write things down, but the may be completely nonsensical. For example, in $\mathcal{L}_{NT}$ we could write

$$(\forall v_1)((\exists v_2)(v_2 > v_1))$$

or we could write

$$v_{17}) \exists \forall >> ($$

Here we decide which strings will have meaning.

Terms (The nouns of our language.)

These are built up from variables and constants using functions, but not relations.

Def Let $\mathcal{L}$ be a language. A term of $\mathcal{L}$ is a nonempty, finite string t of symbols from $\mathcal{L}$ such that either:

the most primitive formulas

1. t is a variable
2. t is a constant symbol

recursive definition

3. $t :\equiv f t_1 t_2 \ldots t_n$ where f is an n-ary function symbol and each t_i is a term

$\mathcal{L} = \mathcal{L}_{NT} = \{0, S, +, \cdot, E, <\}$

For example "is"

1. $t :\equiv v_{17}$
2. $t :\equiv 0$
3. $t :\equiv S v_2$

OR $t :\equiv S0$

OR $t :\equiv + v_1 v_2$!

OR $t :\equiv + S0S0$

Q: which of the following are terms of $\mathcal{L}_{NT}$?

① $S+0v_5$ ③ $++S00$ ⑤ $S+E0SS00$

② $+(S0)1$ ④ $0<S0$

TPS write down 3 more terms of $\mathcal{L}_{NT}$, in prefix & infix.

TPS write down 3 terms of $\mathcal{L}_{ST}$.

! So, technically, terms are written with prefix notation, but we will often use infix when the contex is clear. For, example we will say that $v_1+(v_2+v_3)$ is a term of $\mathcal{L}_{NT}$ with the understanding that $v_1+(v_2+v_3)$ is a placeholder for $+v_1+v_2v_3$

Formulas

terms <---> nouns

formulas <---> assertions

Def Let $\mathcal{L}$ be a language. A formula of $\mathcal{L}$ is a nonempty, finite string ϕ of symbols from $\mathcal{L}$ such that either:

atomic formulas

1. $\phi :\equiv = t_1 t_2$ where t_1, t_2 are terms
2. $\phi :\equiv R t_1 t_2 \cdots t_n$ where R is an n-ary relation symbol and $t_1, \dots, t_n$ are terms

recursive def. again

3. $\phi :\equiv (\neg \alpha)$ where α is a formula
4. $\phi :\equiv (\alpha \vee \beta)$ where α, β are formulas
5. $\phi :\equiv \underbrace{(\forall v)(\alpha)}$ where v is a variable and α is a formula

the scope of the quantifier $\forall$ is α

$\boxed{\text{TPS}}$ Write down 4 formulas of $\mathcal{L}_{NT}$ s.t. 2 are atomic and 2 are not.

$\boxed{\text{TPS}}$ Do the same in $\mathcal{L}_{GRAPH} = \{E\}$ binary relation.

Are you missing $\exists$ or $\wedge$?

why did we choose this?

Let us agree that

$\alpha \wedge \beta$ is a placeholder for $(\neg((\neg\alpha) \vee (\neg\beta)))$

$\alpha \rightarrow \beta$ " " " $((\neg\alpha) \vee \beta)$

$(\exists v)(\alpha)$ " " " $(\neg(\forall v)(\neg\alpha))$

$\boxed{\text{TPS}}$ Can you write down a formula of $\mathcal{L}_{NT}$ with whose <u>intended</u> meaning captures the fact that $\mathbb{N}$ has no largest element.

Q: Let $\mathcal{L}$ be <u>any</u> 1st-order language. What is the indended meaning of

$$(\exists v_1)((\exists v_2)((\exists v_3)(\neg((v_1 = v_2) \vee (v_1 = v_3) \vee (v_2 = v_3)))))$$

1.4 Induction

Recall...

Proof by induction Suppose you want to prove

$P(n)$ is true for all $n \in \mathbb{N}$ with $x \geq c$.

where P is a statement about n.

1. (Base case) Prove $P(c)$ is true.
2. (Inductive step). Prove that $P(n) \to P(n+1)$.

complete induction → 2. (Inductive step). Prove that $P(c), P(c+1), \ldots, P(n)$ all together imply $P(n+1)$.

Running Example $P(n)$

Prove $\boxed{1+2+\cdots+n = \frac{n(n+1)}{2}}$ for all $n \geq 1$

Base case: $P(1)$

Prove $1 = \frac{1(1+1)}{2}$. LHS $=1$, RHS $=1$, so done.

Inductive step $P(n) \to P(n+1)$

Assume $1+2+\cdots+n = \frac{n(n+1)}{2}$

Prove $1+2+\cdots+(n+1) = \frac{(n+1)(n+1+1)}{2}$

$$\begin{aligned}
\text{LHS} &= 1+2+\cdots+(n+1) \\
&= 1+2+\cdots+n+(n+1) \\
&= \frac{n(n+1)}{2} + n+1 = \cdots = \frac{(n+1)(n+2)}{2} = \text{RHS}
\end{aligned}$$

$\square$

Proving statements about all formulas or all terms

* usually done "by induction on the complexity of the formula" ... or "complexity of the term"
* let's see this by example.

<u>Theorem 1.4.2</u> Let ϕ be any formula in a language $\mathcal{L}$. Then the number of left parentheses in ϕ is equal to the number of right parentheses.

<u>pf</u>
we proceed by induction on the number of connectives and quantifiers in ϕ.

<u>Base case</u>: Assume ϕ has 0 connectives and quant.
Then ϕ is atomic. Thus, either

$$\phi :\equiv \; = t_1 t_2$$

OR

$$\phi :\equiv R t_1 \dots t_n$$

with $t_1, \dots, t_n$ terms and R an n-ary rel. sym.

As terms have no parentheses, ϕ has no parentheses, so ϕ certainly has an equal # of L and R parens.

<u>Inductive step</u> Assume Thm is true for formulas w/ k connectives/quantifiers. Assume ϕ has $k+1$ con/quan. Thus ϕ is <u>not</u> atomic, so either

$$\phi :\equiv (\neg \alpha)$$

OR

$$\phi :\equiv (\alpha \vee \beta)$$

OR

$$\phi :\equiv (\forall v)(\alpha)$$

where α, β are formulas; v is a variable

Note that α, β have at most k quantifiers

So the num. of their L and R paren. are equal.

Thus, the same is true for ϕ.

$\square$

1.5 Sentences

Ex Determine if each $\mathcal{L}_{NT}$-formula is True, False, or indeterminate (when interpreted the usual way)

(a) $(\exists x)(\forall y)(y < x)$

(b) $(\forall x)(\exists y)(x < y)$

(c) $\forall x(x < y)$ ← y is a free variable

Def Let v be a variable and ϕ a formula. We say that v is free in ϕ if either

1. ϕ is atomic and v occurs in ϕ
2. $\phi :\equiv \neg\alpha$ and v is free in α
3. $\phi :\equiv \alpha \vee \beta$ and v is free in α or β
4. $\phi :\equiv (\forall u)(\alpha)$ and v is free in α **AND** $v \neq u$

$\mathcal{L}_{NT}$

1. $v = 0$ or $v < 0$
2. $\neg(v = 0)$
3. $(\neg(v=0)) \vee 0=0$
4. $\forall u(v = 0)$ OR $\forall u(v = u)$ BUT NOT $\forall v(v=0)$

* Notice that this covers $\wedge, \rightarrow, \exists, \ldots$ too

! If v is free in ϕ, we may write $\phi(v)$ instead of ϕ.

$\boxed{\text{TPS}}$ Determine if x is free in

$$(\forall x(x E 0 = S0)) \vee (x = 0)$$

Def A <u>sentence</u> is a formula with no free variables.

binary $x \cdot y$ unary x^{-1} constant

$\boxed{\text{TPS}}$ Let $\mathcal{L}_G = \{\cdot, ^{-1}, 1\}$, write three strings of symbols s.t. one is <u>NOT</u> an $\mathcal{L}_G$-formula, one is an $\mathcal{L}_G$-formula that is <u>NOT</u> a sentence, and one that is an $\mathcal{L}_G$-sentence.

$\boxed{\text{TPS}}$ Is $\sin^2 x + \cos^2 x = 1$ a <u>sentence</u> of $\mathcal{L} = \{\sin^2, \cos^2, 1\}$? If not, alter it

unary unary constant

to create an actual sentence that captures the intended meanging of the original.

1.6 Structures

... the beginning of semantics.

<u>Def</u> Let $\mathcal{L}$ be a language. An <u>$\mathcal{L}$-structure</u> $\mathcal{M}$ is a nonempty set M (the <u>universe</u> of $\mathcal{M}$) together with:

1. an element $c^{\mathcal{M}}$ of M for each constant symbol c,
2. an n-ary function $f^{\mathcal{M}} : M^n \to M$ for each n-ary function symbol f

see the difference?

3. an n-ary relation $R^{\mathcal{M}} \subseteq M^n$ for each n-ary relation symbol R

book occasionally uses $\mathcal{L}$-model.

<u>Ex</u> Let $\mathcal{L} = \{0, f, S\}$ with

0 – constant

f – binary function

S – unary relation

Define

* $M = \{a, b, \Delta\}$
* $0^{\mathcal{M}} = a$
* $f^{\mathcal{M}}$ with the following table:

$f^{\mathcal{M}}$	a	b	Δ
a	Δ	b	b
b	a	Δ	b
Δ	a	a	Δ

$f(a,b) = \Delta$

* $S^{\mathcal{M}} = \{a, b\} \subseteq M$

Then $\mathcal{M} = (M, 0^{\mathcal{M}}, f^{\mathcal{M}}, S^{\mathcal{M}})$ is an $\mathcal{L}$-structure.

Ex Let $\mathcal{L}$ be as above.

Define $\mathcal{R} = (\mathbb{R}, 0^{\mathcal{R}}, f^{\mathcal{R}}, S^{\mathcal{R}})$ by

$0^{\mathcal{R}} = \pi/2$, $f^{\mathcal{R}}(a,b) = a \cdot b$, $S^{\mathcal{R}} = \mathbb{Z}$.

Then this is also an $\mathcal{L}$-structure

TPS Recall: $\mathcal{L}_{Graph} = \{E\}$ where E is a binary relation. Define an $\mathcal{L}$-structure w/ 4 elements.

Back to firm ground...

Ex Recall $\mathcal{L}_{NT} = \{0, S, +, \cdot, E, <\}$

Define $\mathcal{N} = (\mathbb{N}, 0^{\mathcal{N}}, S^{\mathcal{N}}, +^{\mathcal{N}}, \cdot^{\mathcal{N}}, E^{\mathcal{N}}, <^{\mathcal{N}})$ as follows

$0^{\mathcal{N}} = 0$ ← the actual number!

symbol on the paper

$S^{\mathcal{N}}: \mathbb{N} \to \mathbb{N}$ by $S^{\mathcal{N}}(x) = x+1$

$+^{\mathcal{N}}: \mathbb{N}^2 \to \mathbb{N}$ by $+^{\mathcal{N}}(x,y) = x+y$ OR $x +^{\mathcal{R}} y = x+y$

$\cdot^{\mathcal{N}}: \mathbb{N}^2 \to \mathbb{N}$ by $\cdot^{\mathcal{N}}(x,y) = x \cdot y$

$E^{\mathcal{N}}: \mathbb{N}^2 \to \mathbb{N}$ by $E(x,y) = x^y$

$<^{\mathcal{N}} \subseteq \mathbb{N}^2$ by $(x,y) \in <^{\mathcal{N}} \iff x < y$.

Really: $<^{\mathcal{N}} = \{(0,1), (0,2), \dots, (1,2), \dots\}$

Then $\mathcal{N}$ is the so-called "standard" $\mathcal{L}_{NT}$-structure.

! we often omit the superscripts and write

$\mathcal{N} = (\mathbb{N}, 0, S, +, \cdot, E, <)$ is an $\mathcal{L}_{NT}$-struct.

Ex Let $\mathcal{L} = \{\ell, +\}$ with ℓ a constant and $+$ a 2-ary funct. symbol.

Define the $\mathcal{L}$-structure $\mathcal{S} = (S, \ell, +)$ by

* S is the set of finite strings from the spanish alphabet.
* $\ell^{\mathcal{S}} = ab\tilde{n}z$.
* $+^{\mathcal{S}} : S \times S \to S$ is "concatenation"

e.g. $\ell^{\mathcal{S}} +^{\mathcal{S}} rrr = ab\tilde{n}zrrr$

1.7 Truth in a Structure

<u>Def – Informal</u> Let ϕ be an $\mathcal{L}$-formula and $\mathfrak{M}$ an $\mathcal{L}$-structure. We say $\mathfrak{M}$ is a <u>model</u> of ϕ or $\mathfrak{M}$ <u>satisfies</u> ϕ, denoted $\mathfrak{M} \models \phi$, provided:

- <u>if ϕ is a sentence</u>

  ϕ is a true statement about $\mathfrak{M}$ with the standard interpretations of the quantifiers and connectives

- <u>if ϕ has free variables among $v_1, \dots, v_n$</u>

  $(\forall v_1, v_2, \dots, v_n)(\phi)$ is a true statement about $\mathfrak{M}$.

<u>Ex</u> Let $\mathcal{L}_R = \{\underbrace{0, 1}_{C}, \underbrace{-}_{F^1}, \underbrace{+, \cdot}_{F^2}\}$. Consider

$\mathbb{Q} = \mathfrak{Q} = (\mathbb{Q}, 0, 1, -, +, \cdot)$ with the standard interp.

e.g. $-^{\mathfrak{Q}}(x) = -x$ (the neg. of x)

$\mathbb{Z} = \mathfrak{Z} = (\mathbb{Z}, 0, 1, -, +, \cdot)$ with the stand. interp.

Thus, $\mathbb{Q}$ & $\mathbb{Z}$ are $\mathcal{L}_R$-structures.

① Let $\phi :\equiv (\forall x)[\underbrace{(x \neq 0)}_{\neg(x=0)} \rightarrow ((\exists y)(x \cdot y = 1))]$

Then, $\mathbb{Q} \models \phi$ but $\mathbb{Z} \not\models \phi$.

this notation highlights the free variables.

2 Let

$$\psi(x) :\equiv (x \neq 0) \rightarrow [(\exists y)(x \cdot y = 1)]$$

$$\rho(x) :\equiv (\exists y)(x \cdot y = 1)$$

Then,

$$\mathbb{Q} \models \psi(x) \quad \text{b/c} \quad \mathbb{Q} \models (\forall x)(\psi(x))$$

$$\mathbb{Z} \not\models \psi(x)$$

Also,

$$\mathbb{Q} \not\models \rho(x) \quad \text{b/c} \quad \mathbb{Q} \not\models (\forall x)(\rho(x))$$

$$\mathbb{Z} \not\models \rho(x)$$

* Finally, we do have $\mathbb{Q} \models \rho(-1)$ and $\mathbb{Z} \models \rho(-1)$

substitute -1 for x in ρ yields a sentence.

TPS $\mathbb{R}$ and $\mathbb{C}$ are also $\mathcal{L}_R$-structures in the obvious way. Find an $\mathcal{L}_R$-sentence true in $\mathbb{C}$ but not in $\mathbb{R}$.

Onward to the details...

Assigning Values to Variables

- $\mathcal{L}$ a language
- $\mathcal{M}$ an $\mathcal{L}$-structure

$\boxed{\text{Running Example}}$ $\mathcal{L}_{NT}$, standard $\mathcal{L}_{NT}$-struct. $\mathbb{N}$.

Def Any function $s : \text{Vars} \longrightarrow M$ is called a variable assignment function into $\mathcal{M}$.

$\boxed{\text{Ex}}$ In $\mathbb{N}$...

$s_1 : \text{Vars} \longrightarrow \mathbb{N} \; : \; s_1(v_i) = 2i+1$

e.g. $s_1(v_3) = 7$, $s_1(v_6) = 13$

$s_2 : \text{Vars} \longrightarrow \mathbb{N} \; : \; s_2(v_i) = 13$

e.g. $s_2(v_3) = 13$, $s_2(v_6) = 13$

From Vars to $\mathcal{L}$-terms

Def Let $s : \text{Vars} \longrightarrow M$ be any variable assignment function. Define $\bar{s} : \{\mathcal{L}\text{-terms}\} \longrightarrow M$ by

1. if t is a variable, then $\bar{s}(t) = s(t)$
2. if t is a constant, then $\bar{s}(c) = c^{\mathcal{M}}$
3. (inductively) if $t :\equiv f t_1 \cdots t_n$, then

$$\bar{s}(t) = f^{\mathcal{M}}(\bar{s}(t_1), \dots, \bar{s}(t_n)).$$

$\boxed{\text{Ex}}$ In $\mathbb{N}$... let $t :\equiv E(v_3, SS0) + (v_1 \cdot v_3) + v_5$

$s_1(v_i) = 2i+1 \Longrightarrow \bar{s}_1(t) = E(7,2) + 3 \cdot 7 + 11 = 81$

$s_3(v_i) = 3 \Longrightarrow \bar{s}_3(t) = E(3,2) + 3 \cdot 3 + 3 = 21$

$\boxed{\text{Ex}}$ Let $t' :\equiv S0$. Explain how $\bar{s}_1(t')$ is evaluated.

$$\bar{s}_1(t') = S^{\mathfrak{M}}(\bar{s}_1(0)) = S^{\mathfrak{M}}(0^{\mathfrak{M}}) = 1$$

then, $\bar{s}_1(0) = 0^{\mathfrak{M}}$

Sometimes we want to fix some outputs of s.

Def Let $s : \text{Vars} \longrightarrow M$ be a var. assign. function.

Define

$$s[x|a](v) = \begin{cases} a & \text{if } v \text{ is the variable } x \\ s(v) & \text{otherwise} \end{cases}.$$

This is an <u>x-modification</u> of s.

$\boxed{\text{Ex}}$ Let $t :\equiv E(v_3, SS0) + (v_1 \cdot v_3) + v_5$.

$s_1(v_i) = 2i+1$.

$\bar{s}_1(t) = 81$ (from before).

$\overline{s_1[v_3|2]}(t) = E(2,2) + (3 \cdot 2) + 11 = 21$

Def Let $s : \text{Vars} \longrightarrow M$ be a var. assign. function.

Let ϕ be an $\mathcal{L}$-formula. We say $\mathfrak{M}$ <u>satisfies ϕ</u> with assignment s, denoted $\mathfrak{M} \models \phi[s]$, provided:

1. <u>if $\phi :\equiv t_1 = t_2$</u> then $\bar{s}(t_1) = \bar{s}(t_2)$
2. <u>if $\phi :\equiv R t_1 \dots t_n$</u> then $(\bar{s}(t_1), \dots, \bar{s}(t_2)) \in R^{\mathfrak{M}}$

inductive

3. <u>if $\phi :\equiv (\neg \alpha)$</u>, then $\mathfrak{M} \not\models \alpha[s]$
4. <u>if $\phi :\equiv (\alpha \vee \beta)$</u>, then $\mathfrak{M} \models \alpha[s]$ <u>OR</u> $\mathfrak{M} \models \beta[s]$
5. <u>if $\phi :\equiv (\forall x)(\alpha)$</u>, then $\forall m \in M$ $\mathfrak{M} \models \alpha[s(x|m)]$

* This also covers the other quantifiers and connectives.

<u>Ex</u> In $\mathcal{L}_{NT}$, let $\phi :\equiv (\forall v_1)\left[(v_1 = 0) \vee (\forall v_2)(v_2 < v_1 + v_2)\right]$

Show that for any v.a.f. s, $\mathbb{N} \models \varphi[s]$.

Observe,

$\mathbb{N} \models \varphi[s]$ iff $\forall m \in \mathbb{N}$, $\left[(v_1 = 0) \vee (\forall v_2)(v_2 < v_1 + v_2)\right][s[v_1|m]]$

iff $\forall m \in \mathbb{N}$, $\mathbb{N} \models (v_1 = 0)[s[v_1|m]]$ OR

$\mathbb{N} \models (\forall v_2)(v_2 < v_1 + v_2)[s[v_1|m]]$

iff $\forall m \in \mathbb{N}$, $\Big(m = 0^{\mathbb{N}}$ <u>OR</u>

$\forall r \in \mathbb{N}$, $\mathbb{N} \models (v_2 < v_1 + v_2)[s[v_1|m][v_2|r]]\Big)$

iff $\forall m \in \mathbb{N} \left(m = 0^{\mathbb{N}} \text{ OR } \forall r \in \mathbb{N}\, (r <^{\mathbb{N}} m +^{\mathbb{N}} r)\right)$

The final statement is true, so $\mathbb{N} \models \varphi[s]$.

<u>Def</u> For ϕ an $\mathcal{L}$-formula and $\mathfrak{M}$ an $\mathcal{L}$-structure we write $\mathfrak{M} \models \phi$ iff $\mathfrak{M} \models \phi[s]$ for <u>every</u> v.a.f. s. Also, if Γ is a set of $\mathcal{L}$-formulas we write $\mathfrak{M} \models \Gamma$ if $\mathfrak{M} \models \phi$ for all $\phi \in \Gamma$.

* we read $\mathfrak{M} \models \phi$ as "$\mathfrak{M}$ models ϕ" OR "$\mathfrak{M}$ satisfies ϕ"

So, in the last example, $\mathbb{N} \models \phi$, i.e. $\mathbb{N}$ models ϕ.

A few results...

OPTIONAL

Lemma 1.7.6 Suppose s_1, s_2 are v.a.f. into a structure $\mathfrak{M}$.

If $s_1(v) = s_2(v)$ for every variable v that occurs in the term t, then $\bar{s}_1(t) = \bar{s}_2(t)$.

pf we use induction on the complexity of t.

Base case:

- $t :\equiv v$, v a variable. Then $\bar{s}_1(t) = s_1(v) = s_2(v) = \bar{s}_2(t)$
- $t :\equiv c$, c a constant. Then $\bar{s}_1(t) = c^{\mathfrak{M}} = \bar{s}_2(t)$.

Ind. case:

Assume $t = f t_1 \ldots t_n$ and $\bar{s}_1(t_i) = \bar{s}_2(t_i)$.

Then $\bar{s}_1(t) = f^{\mathfrak{M}}(\bar{s}_1(t_1), \ldots, \bar{s}_1(t_n))$

$= f^{\mathfrak{M}}(\bar{s}_2(t_1), \ldots, \bar{s}_2(t_n)) = \bar{s}_2(t)$. □

Prop. 1.7.7 Suppose s_1, s_2 are v.a.f. into a structure $\mathfrak{M}$. Let ϕ be a formula. If $s_1(v) = s_2(v)$ for every free variable v in ϕ, then

$$\mathfrak{M} \models \phi[s_1] \text{ iff } \mathfrak{M} \models \phi[s_2].$$

pf By induction — see the book.

Cor. 1.7.8 If ϕ is a sentence, then

$$\mathfrak{M} \models \phi \text{ iff } \mathfrak{M} \models \phi[s] \text{ for } \underline{\text{some}} \text{ v.a.f. } s.$$

1.9 Logical Implication (1.8 is postoned)

Most theorems are of the form

"If V is a vector space, then ..."

"If G is a graph, then ..."

"If G is a group, then ..."

That is, assuming a structure satisfies some axioms (sentences), then some other sentences are true.

<u>Def</u> Let Δ and and Γ be sets of $\mathcal{L}$-formulas. We say that Δ <u>logically implies</u> Γ if for all $\mathcal{L}$-structures $\mathfrak{M}$, $\mathfrak{M} \models \Delta \rightarrow \mathfrak{M} \models \Gamma$.

<u>Ex</u> Let $\mathcal{L} = \{e, \cdot\}$ with e a constant symbol and $\cdot$ a binary function symbol. Let

$$\Delta = \{ (\forall x, y, z)[(xy)z = x(yz)], \quad ① \quad \phi_1$$
$$(\forall x)(xe = ex = x), \quad ② \quad \phi_2$$
$$(\forall x)(\exists y)(xy = yx = e), \quad ③ \quad \phi_3$$
$$(\forall x)(x \cdot x = e) \} \quad ④ \quad \phi_4$$

$$\Gamma = \{ xy = yx \} \quad \psi$$

Prove that $\Delta \models \Gamma$.

(we do this informally this time; contrast w/ book.)

Let $\mathcal{M}$ be any $\mathcal{L}$-structure. We must show

$$\mathcal{M} \models \Delta \implies \mathcal{M} \models \Gamma$$

$$\mathcal{M} \models xy = yx \iff \mathcal{M} \models (\forall x, y)(xy = yx)$$

Assume $\mathcal{M} \models \Delta$. We want to prove $(\forall x, y)(xy = yx)$.

Let $a, b \in M$. Then

* there is a $a' \in M$ s.t. $a'a = e$
* " " a $b' \in M$ s.t. $bb' = e$

$$\begin{aligned}
a \cdot b \in M \implies & (a \cdot b) \cdot (a \cdot b) = e^{\mathcal{M}} \quad 4 \\
\implies & a \cdot (b \cdot (a \cdot b)) = e^{\mathcal{M}} \quad 1 \\
\implies & a'(a(b(a\,b))) = a' e^{\mathcal{M}} \\
\implies & (a'a) \cdot (b(ab)) = a' \quad 1 \\
\implies & e^{\mathcal{M}} \cdot (b(ab)) = a' \quad 3 \\
\implies & b(ab) = a' \quad 2 \\
\implies & b\,a\,b\,e^{\mathcal{M}} = a' \quad 2 \\
\implies & b\,a\,b\,b' = a'b' \\
\implies & b\,a\,e^{\mathcal{M}} = a'b' \quad 3 \\
\implies & ba = a'b' \quad 2 \\
\implies & ba = e^{\mathcal{M}} a' b' e^{\mathcal{M}} \quad 2 \\
\implies & ba = a\underbrace{a\,a'}\,\underbrace{b'\,b}\,b \\
\implies & ba = a\,e^{\mathcal{M}} e^{\mathcal{M}} b \quad 3 \\
\implies & ba = ab \quad 2
\end{aligned}$$

Now that we see how 1 works, lets ignore parentheses

Thus, $\Delta \models \Gamma$

Def Let ϕ be an $\mathcal{L}$-formula. If $\emptyset \models \phi$, then every $\mathcal{L}$-structure satisfies ϕ, and we say ϕ is <u>logically valid</u>.

* we write $\models \phi$ instead of $\emptyset \models \phi$.

Ex In the previous example, we had

$$\underbrace{\Delta}_{\{\phi_1, \phi_2, \phi_3, \phi_4\}} \models \underbrace{\Gamma}_{\{\psi\}}$$

Then, if $\phi :\equiv (\phi_1 \wedge \phi_2 \wedge \phi_3 \wedge \phi_4) \longrightarrow \psi$, then ϕ is logically valid, so $\models \phi$.

OPTIONAL

Ex Suppose $\mathcal{L}$ contains a unary relation P. Carefully prove that $\phi :\equiv (\forall x) P(x) \longrightarrow (\exists x) P(x)$ is logically valid.

Let $\mathfrak{M}$ be any $\mathcal{L}$-structure and s any v.a.f. We must show $\mathfrak{M} \models \phi[s]$.

- If $\mathfrak{M} \not\models \big((\forall x) P(x)\big)[s]$ then $\mathfrak{M} \models \phi[s]$ by def. of $\longrightarrow$.
- Suppose $\mathfrak{M} \models \big((\forall x) P(x)\big)[s]$. So, for every $m \in M$ $\mathfrak{M} \models \big(P(x)\big)[s[x|m]]$; thus, for some $m \in M$, $\mathfrak{M} \models \big(P(x)\big)[s[x|m]]$. Hence $\mathfrak{M} \models \big((\exists x) P(x)\big)[s]$, so $$\mathfrak{M} \models \phi[s].$$

1.8 Substitutions and Substitutability

Let $\phi :\equiv (\exists y)\neg(x=y)$. Note that ϕ is true in any structure with at least 2 elements. For example $\mathbb{N} \models \phi$.

- suppose you replace x with u. Is it true that $\phi \models \phi^x_u$?

$$\phi^x_u :\equiv (\exists y)\neg(u=y)$$

Yes—clearly.

- Suppose we are in $\mathcal{L}_{NT}$ and replace x with the term $u+v$. Is $\phi \models \phi^x_{u+v}$ true?

$$\phi^x_{u+v} :\equiv (\exists y)\neg(u+v=y)$$

Yup.

- Suppose you replace x with y. $\phi \models \phi^x_y$?

$$\phi^x_y :\equiv (\exists y)\neg(y=y)$$

No! ... ϕ^x_y is now false in every structure.

Def Suppose x is a variable and t is a term.

- If u is a term, we define u^x_t ("u with x replaced by t") as you expect. See book
- If ϕ is a formula, we define ϕ^x_t as follows

1. if $\phi :\equiv \; = t_1 t_2$, then $\phi^x_t :\equiv \; = (t_1)^x_t \, (t_2)^x_t$
2. if $\phi :\equiv R t_1 \dots t_n$, then $\phi^x_t :\equiv R (t_1)^x_t \dots (t_n)^x_t$
3. if $\phi :\equiv \neg(\alpha)$ ...
4. if $\phi :\equiv \alpha \vee \beta$ ...

similar

5. if $\phi :\equiv (\forall y)(\alpha)$, then

$$\phi^x_t :\equiv \begin{cases} (\forall y)(\alpha^x_t) & \text{if } x \neq y \\ \phi & \text{if } x = y \end{cases}$$

Ex Work in $\mathcal{L}_{NT}$. Let

$$\phi :\equiv (\forall y)(x + y = z) \vee (\forall x)(x \cdot x = x)$$

Then, if $t :\equiv y + w$,

$$\phi^x_t :\equiv (\forall y)(y + w + y = z) \vee (\forall x)(x \cdot x = x)$$

but want to avoid this

Same thing would have happened with $\exists x$ here.

<u>Def</u> Hyp. as before. We say <u>t is substitutable for x in ϕ</u> if

1. ϕ is atomic,
2. $\phi :\equiv \neg(\alpha)$ and t is sub. for x in α,
3. $\phi :\equiv (\alpha \vee \beta)$ and t is sub. for x in <u>both</u> α and β.
4. $\phi :\equiv (\forall y)(\alpha)$ and either
   - x is not free in ϕ OR
   - (x is free and) y does not occur in the term t and t is sub. for x in α.

<u>Ex</u> Determine if $t :\equiv yz + z$ is sub. for x. (in $\mathcal{L}_{NT}$)

① $\phi :\equiv (\forall y)(Sx = y)$ No!

② $\phi :\equiv (\forall y)\big(y = 0 \vee (\forall x)(x = y)\big)$ Yes!

③ $\phi :\equiv (\underbrace{y = x}_{\text{atomic}}) \vee (\forall w)(\underbrace{E(w,x) > w}_{\text{atomic}})$ Yes