

**Exercise 20 (Models)**

- a) Let  $G = \{f(x, f(y, z)) \approx f(f(x, y), z)\}$ .
- i) Construct a model of  $G$ .
  - ii) Construct an  $f$ -algebra  $\mathcal{A}$  such that  $\mathcal{A}$  is not a model of  $G$ .
- b) Let  $H = \{f(x, f(y, z)) \approx f(f(x, y), z), f(e, x) \approx x, f(x, i(x)) \approx e\}$ .
- i) Construct a model of  $H$  where the additional equation  $f(x, e) \approx x$  also holds.
  - ii) Show that  $f(x, e) \approx_H x$  is not a semantic consequence of  $H$ , by constructing a model of  $H$  where  $f(x, e) \approx x$  does *not* hold.
  - iii) Why is this not a contradiction to Exercise 15 from last week's sheet?

**Solution**

- a) i)  $\mathcal{A} = \mathbb{N}, f^{\mathcal{A}}(x, y) = x \cdot y$   
 ii)  $\mathcal{B} = \mathbb{N}, f^{\mathcal{A}}(x, y) = x - y$
- b) i)  $\mathcal{A} = \mathbb{Z}, e^{\mathcal{A}} = 0, i^{\mathcal{A}}(x) = -x, f^{\mathcal{A}}(x, y) = x + y$ .  
 or  
 $\mathcal{B} = \{*\}, e^{\mathcal{B}} = *, i^{\mathcal{B}}(*) = *, f^{\mathcal{B}}(*, *) = *$ .
- ii)  $\mathcal{A} = \{1, 2\}, e^{\mathcal{A}} = 1, i^{\mathcal{A}}(x) = 1, f^{\mathcal{A}}(x, y) = y$ .  
 or

$$\mathcal{B} = \{0, 1, 2, 3\}, e^{\mathcal{B}} = 1, i^{\mathcal{B}} = (0, 2, 0, 2), f^{\mathcal{B}} =$$

$f$	0	1	2	3
0	0	1	2	3
1	3	2	0	1
2	0	1	2	3
3	3	2	0	1

- iii) The above specification does not characterize groups. Rather, either the identity axiom or the inverse element axiom need to be flipped.

### Exercise 21 (Equality)

Let  $\Sigma = \{f\}$  where  $f$  is a binary function symbol, and

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

Is the question of whether two terms  $s$  and  $t$  are equivalent with respect to  $E$  decidable?

*Hint:* Figure out how to calculate, for any term  $s$ , the smallest term equivalent to  $s$ .

### Solution

A smallest term  $t'$  that is equivalent to  $t \in T(\Sigma, V)$  contains any variable occurring in  $t$  exactly once, ordered by its first occurrence in  $t$ , w.r.t. left-to-right depth-first order. Moreover,  $t'$  is unique up to associativity.

Thus, we can decide equivalence of  $t$  and  $s$  by comparing the first occurrences of the variables in  $s$  and  $t$ . As terms are finite, the problem is decidable.

The proposition about equivalence can be shown as follows:

Let  $V = \{x_1, x_2, \dots\}$  be the set of variables for  $\Sigma$ . Let  $t \in T(\Sigma, V)$  and let  $x_{i_1}, \dots, x_{i_k}$  be the variables in  $t$  in order of their occurrence. We define  $g : T(\Sigma, V) \rightarrow \text{List}(V)$  by

$$g(t) = [x_{i_1}, \dots, x_{i_k}].$$

Now we can express our equivalence proposition in terms of  $g$ :

$$s \longleftrightarrow_E^* t \iff g(s) = g(t)$$

“ $\implies$ ”: We have to show that  $g(s) = g(t)$  is an invariant w.r.t.  $E$ . I.e., if  $t \longrightarrow_E t'$ , then  $g(t) = g(t')$ . This has to be shown for each equation in  $E$ . By induction over the derivation of  $s \longleftrightarrow_E^* t$ , we can show  $g(s) = g(t)$ .

“ $\impliedby$ ”: We sketch an algorithm that, for two terms  $s$  and  $t$  with  $g(s) = g(t)$ , computes a term  $r$  with  $s \longrightarrow_E^* r \longleftarrow_E^* t$ . This term  $r$  contains all occurring variables exactly once, and is parenthesized to the left, i.e., the associativity rule  $f(x, f(y, z)) \longrightarrow f(f(x, y), z)$  cannot be applied any more.

a) As long as  $s$  contains two occurrences of a variable  $w$ , transform  $s$  by associativity in one of the forms:

- $f(\dots f(f(w, s_1), w) \dots)$
- $f(\dots f(w, w) \dots)$

b) Erase the second occurrence of  $w$  by applying the first or second rule in  $E$ .

c) If no variable occurs twice in  $s$ , use the associativity rule to parenthesize the term to the left.

If  $g(s) = g(t)$ , the the resulting terms are equal, because

- The algorithm does not change the order of variables.
- All occurrences of a variable except for the first one are erased.

## Homework 22 (Programming Assignment)

You are assigned to implement equational logic in Haskell based on the inference rules described in the book in §3.1 (p. 42). Additional instructions can be found on the lecture web page.

## Homework 23 (Commutativity)

Let  $\Sigma = \{0, s, +\}$  and  $E = \{x + 0 \approx x, x + s(y) \approx s(x) + y\}$ .

- a) Show that  $E \models x + s(s(0)) \approx s(s(x))$  and  $E \models s^i(0) + s^j(0) \approx s^j(0) + s^i(0)$ .
- b) Show that  $E \not\models x + y \approx y + x$ .

*Hint:* Give a model of  $E$  where  $+$  is not commutative.