## Exercises

**Exercise 5.5.**
*Prove theorem 5.5.*

**Exercise 5.6.**
*Be* $\Sigma = (S, \Omega)$ *with:*

$$S = \{\, int,\ nonzero,\ pair \,\}$$

$$\begin{array}{lcll} \Omega & = & \{\ / & : \ int \times nonzero \to pair, \\ & & \ \ + & : \ pair \times pair \to pair, \\ & & \ \ + & : \ int \times int \to int\ \} \end{array}$$

*(cont'd at next page)*



$A(int) = \mathbb{Z}$,
$A(nonzero) = \mathbb{Z} - \{0\}$,
$A(pair) = \mathbb{Z} \times (\mathbb{Z} - \{0\})$,
$A(/)(m, n) = (m, n)$,
$A(+)((m, n), (m', n')) = (m * n' + m' * n, n * n')$,
$A(+ : int \times int \to int)$ is the sum in $\mathbb{Z}$.

*Further, let* $B$ *be like* $A$*, except for:*
$B(pair) = \mathbb{Q}$,
$B(/)(m, n) = \frac{m}{n}$,
$B(+ : pair \times pair \to pair)$ is the sum in $\mathbb{Q}$.

*(i) Find a congruence relation* $Q$ *on* $A$ *such that* $A/_Q \simeq B$.
*(ii) Define the quotient homomorphism of the congruence relation found in (i).*
*(iii) Show that there is no congruence relation* $P$ *on* $B$ *such that* $B/_P \simeq A$.

**end of Exercise5.6.**



# Part 6: Particular Classes of Algebras



## Motivation

$\textbf{NAT}^{Mod2} = (S_{NAT}, \Omega_{NAT} \cup \{Mod2 : nat \to nat\})$

consider the (yet informal) specification:

$$\begin{array}{rcl} Mod2(0) & = & 0 \\ Mod2(Succ(0)) & = & Succ(0) \\ Mod2(Succ(Succ(n))) & = & Mod2(n) \end{array}$$

now consider the property $(*)$:

$$\forall x : nat.\ Mod2(x) = 0\ \vee Mod2(x) = Succ(0)$$

## Motivation

an algebra fulfilling $(*)$:

$$\begin{aligned} A(nat) &= \mathbb{N}_0 \\ A(0) &= 0 \\ A(Succ) &: \mathbb{N}_0 \longrightarrow \mathbb{N}_0 \\ A(Mod2) &: \mathbb{N}_0 \longrightarrow \mathbb{N}_0 \end{aligned}$$

with

$$\begin{aligned} A(Succ)(n) &= n+1 \text{ for all } n \in \mathbb{N}_0 \\ A(Mod2)(n) &= \begin{cases} 0 & \text{if } n \in \mathbb{N}_0 \text{ and is odd} \\ 1 & \text{if } n \in \mathbb{N}_0 \text{ and is even} \end{cases} \end{aligned}$$



## Motivation

an algebra not fulfilling $(*)$:

$$\begin{aligned} B(nat) &= \mathbb{N}_0 \cup \{\#\} \\ B(0) &= 0 \\ B(Succ) &: \mathbb{N}_0 \cup \{\#\} \longrightarrow \mathbb{N}_0 \cup \{\#\} \\ B(Mod2) &: \mathbb{N}_0 \cup \{\#\} \longrightarrow \mathbb{N}_0 \cup \{\#\} \end{aligned}$$

with

$$B(Succ)(n) = \begin{cases} n+1 & \text{if } n \in \mathbb{N}_0 \\ \# & \text{if } n = \# \end{cases}$$

$$B(Mod2)(n) = \begin{cases} 0 & \text{if } n \in \mathbb{N}_0 \text{ and is odd} \\ 1 & \text{if } n \in \mathbb{N}_0 \text{ and is even} \\ \# & \text{if } n = \# \end{cases}$$



## Exercises

**Exercise 6.1.**
*Extend the 'specification' of algebras for $\boldsymbol{NAT}^{Mod2}$ to exclude elements which are not 'accessable' by the terms in $T_\Sigma$.*



## An Unsound Induction Proof?

**MARKEXAMS** $= (S_{MARKEXAMS}, \Omega_{MARKEXAMS})$

with:

$$\begin{aligned} S_{MARKEXAMS} &= \{nat, grades\} \\ \Omega_{MARKEXAMS} &= \{\, 0 : \;\to nat, \\ &\quad\;\; Succ : nat \to nat, \\ &\quad\;\; G : \;\to grades, \\ &\quad\;\; U : \;\to grades, \\ &\quad\;\; Mark : nat \to grades\} \end{aligned}$$

## An Unsound Induction Proof?

consider the (yet informal) specification:

$$\begin{array}{rcl} Mark(0) & = & G \\ Mark(Succ(x)) & = & Mark(x) \end{array}$$

now consider the property $(*)$:

$$\forall x : nat.\ Mark(x) = G$$

Now:

- we give an inductive proof for $(*)$ (blackboard)
- we give an algebra in which $(*)$ does not hold (blackboard)



## No Junk

The user of an algebra can access carriers *only* by writing ground terms for them.

⇒ Carriers which are not the value of any term are ‘useless’.

⇒ can be called junk.

Algebras without junk are called *generated*.

Interesting: only these algebras ‘jusify’ induction.



## Constructors and Constructor Terms

Let $\Sigma = (S, \Omega)$ be a signature.

We then can select any subset $\Omega_c \subseteq \Omega$ and call its elements *constructors*.

Put $\Sigma_c = (S, \Omega_c)$.

An element of $T_{\Sigma_c,s}$ is then called *constructor term of sort $s$*.

$T_{\Sigma_c} = (T_{\Sigma_c,s})_{s \in S}$ is called *the constructor terms w.r.t. $\Omega_c$*.



## Generated Algebras

Let $\Sigma = (S, \Omega)$ be a signature, and $\Omega_c \subseteq \Omega$ a set of constructors.

Then, a $\Sigma$-Algebra $A$ is said to be:

- *generated by $\Omega_c$*, if:
  for each $s \in S$ and each $a \in A(s)$,
  there exists at least one constructor term $t \in T_{\Sigma_c,s}$ with $a = A(t)$
- *generated*, if it is generated by $\Omega$

# Reminder

**BOOL**-Algebra $A$

with:

$$\begin{aligned} A(bool) &= \{\mathsf{true}, \mathsf{false}\} \\ A(True) &= \mathsf{true} \\ A(False) &= \mathsf{false} \\ A(\neg)(n) &= \begin{cases} \mathsf{true} & \text{if } n = \mathsf{false} \\ \mathsf{false} & \text{otherwise} \end{cases} \\ A(\wedge)(n,m) &= \begin{cases} \mathsf{true} & \text{if } n = m = \mathsf{true} \\ \mathsf{false} & \text{otherwise} \end{cases} \end{aligned}$$



# Reminder

**BOOL**-Algebra $B$

with:

$$\begin{aligned} B(bool) &= \{\#\} \\ B(True) &= \# \\ B(False) &= \# \\ B(\neg)(\#) &= \# \\ B(\wedge)(\#,\#) &= \# \end{aligned}$$



# Reminder

**BOOL**-Algebra $D$

with:

$$\begin{aligned} D(bool) &= \mathbb{N}_0 \\ D(True) &= 1 \\ D(False) &= 0 \\ D(\neg)(n) &= \begin{cases} n+1 & \text{if } n \text{ is even} \\ n-1 & \text{otherwise} \end{cases} \\ D(\wedge)(n,m) &= n * m \end{aligned}$$



# Reminder

**BOOL**-Algebra $E$

with:

$$\begin{aligned} E(bool) &= \mathbb{N}_0 \\ E(True) &= 1 \\ E(False) &= 0 \\ E(\neg)(n) &= n+1 \\ E(\wedge)(n,m) &= n+m \end{aligned}$$

## Generated Algebras: Examples

**Fact 6.1.**

1. *$A$ is generated by $\{True, False\}$. (!! not by* {true, false} *!!)*
2. *$B$ is generated by $\{True\}$ as well as by $\{False\}$.*
3. *$D$ is* not *generated (by anything) because the carriers $2, 3, \ldots$ fail to be values of ground terms.*
4. *$E$ is generated by $\{False, \neg\}$ as well as by $\{True, False, \wedge\}$.*

⋆



## Generated Algebras: Examples

Consider the $\mathbf{NAT}^{Mod2}$-Algebras $A$ and $B$.

**Fact 6.2.**

1. *$A$ is generated.*
2. *$A$ is even generated by the constructors $\{0, Succ\}$.*
3. *$B$ is* not *generated (by anything) because the carrier $\#$ fails to be value of a ground term.*

⋆



## Proofs by Induction (informally only)

- Each carrier of a generated algebra can be represented by a ground term.

  ⇒ In order to prove a property for all carriers of a generated algebra, it suffices to prove the property for the (value of) each ground term.

  This can be done by *structural induction over the whole signature*.

- Each carrier of an algebra (already) generated by some constructors $\Omega_c$ can be represented by a constructor term.

  ⇒ In order to prove a property for all carriers of a generated algebra, it suffices to prove the property for the (value of) each constructor term.

  This can be done by *structural induction over the constructors*.



## Proofs by Induction: Example (informally only)

As the $\mathbf{NAT}^{Mod2}$-Algebra $A$ is generated,
we can prove a property on $A$ by structural induction over $0$, $Succ$, and $Mod2$.

As the $\mathbf{NAT}^{Mod2}$-Algebra $A$ is already generated by the constructors $\{0, Succ\}$,
we can prove a property of $A$ already by structural induction over $0$ and $Succ$.

## Proofs by Induction (informally only)

note

inductive proofs are ony valid for algebras which are (known to be) generated

If we want to prove something for a whole class of algebras, we can use induction only if *all* algebras in the class are generated.



## Classes of Generated Algebras

Let $\Sigma = (S, \Omega)$ be a signature, $A$ be a $\Sigma$-Algebra and $\Omega_c \subseteq \Omega$.

Then the classes $Gen(\Sigma, \Omega_c)$ and $Gen(\Sigma)$ are defined to be:

$$\begin{aligned} Gen(\Sigma, \Omega_c) &= \{ A \in Alg(\Sigma) \mid A \text{ is generated by } \Omega_c \} \\ Gen(\Sigma) &= Gen(\Sigma, \Omega) \end{aligned}$$



## Generated Algebras and Homomorphisms

**Theorem 6.3.**
*Let $h : A \longrightarrow B$ be a $\Sigma$-homomorphism for some signature $\Sigma$.*

1. *If $A$ is generated, then $h$ is the only homomorphism from $A$ to $B$.*
2. *If $B$ is generated, then $h$ is surjective.* ⋆

**Corollary 6.4.**
*Let $A$ and $B$ be $\Sigma$-Algebras for some signature $\Sigma$.*

1. *If $A$ is generated and $A \simeq B$, then $B$ is also generated.*
2. *If $A$ and $B$ are generated, and there exists homomorphisms $h : A \longrightarrow B$ and $g : B \longrightarrow A$, then $A \simeq B$.* ⋆



## Exercises

**Exercise 6.2.**
*Prove theorem 6.3 and/or corollary 6.4.*

## Freely Generated Algebras

special case of generated algebras:

Let $\Sigma = (S, \Omega)$ be a signature, and $\Omega_c \subseteq \Omega$ a set of constructors.

Then, a $\Sigma$-Algebra $A$ is said to be:

- *freely* generated by $\Omega_c$, if:
  for each $s \in S$ and each $a \in A(s)$,
  there exists exactly one constructor term $t \in T_{\Sigma_c,s}$ with $a = A(t)$
- *freely* generated, if it is freely generated by $\Omega$



## Freely Generated Algebras: Examples

consider the **BOOL**-Algebras $A$, $B$, and $E$.

**Fact 6.5.**

1. *$A$ is freely generated by $\{True, False\}$.*
2. *$B$ is freely generated by $\{True\}$ as well as by $\{False\}$.*
3. *$E$ is freely generated by $\{False, \neg\}$, but not by $\{True, False, \wedge\}$.*
4. *none of them is freely generated (by $\Omega$).*

⋆



## Freely Generated Algebras: Examples

Consider the **NAT**$^{Mod2}$-Algebra $A$.

**Fact 6.6.**
*$A$ is freely generated by $\{0, Succ\}$, but not by $\{0, Succ, Mod2\}$.* ⋆



## Another Example: Integers

**INT** $= (S_{INT}, \Omega_{INT})$

with:

$$\begin{array}{lcll} S_{INT} & = & \{ \; int \; \} & \\ \Omega_{INT} & = & \{ \; 0 & : \; \to int, \\ & & \quad Succ & : \; int \to int, \\ & & \quad Pred & : \; int \to int \; \} \end{array}$$

# Another Example: Integers

Be $A$ the "classical" **INT**-Algebra with:

$$A(int) = \mathbb{Z}$$

$$\begin{aligned} A(0) &= 0 \\ A(Succ)(n) &= n+1 \\ A(Pred)(n) &= n-1 \end{aligned}$$

There exists no $\Omega_c \subseteq \Omega_{INT}$ that *freely* generates $A$.



# Importance of Freely Generated Algebras

Beside allowing inductive proofs (generatedness):

Freely generated algebras allow to safely use 'inductive definitions' in the specification.



# NATSTACK again

reconsider:

**NATSTACK** $= (S_{NATSTACK}, \Omega_{NATSTACK})$

with:

$$S_{NATSTACK} = \{\ nat, bool, stack\ \}$$

$$\begin{aligned} \Omega_{NATSTACK} = \ & \Omega_{NATBOOL} \\ \cup \{ \ & Emptystack : \ \to stack, \\ & Push : stack \times nat \to stack, \\ & Pop : stack \to stack, \\ & Top : stack \to nat \ \} \end{aligned}$$



# NATSTACK again

reconsider the 'specification' from the first lecture:

$$\begin{aligned} Isempty(Emptystack) &= True \\ Isempty(Push(n,s)) &= False \\ Pop(Push(n,s)) &= s \\ Top(Push(n,s)) &= n \\ Del(n, Emptystack) &= Emptystack \\ Del(n, Push(n,s)) &= s \\ n \neq m \to Del(n, Push(m,s)) &= Push(m, Del(n,s)) \end{aligned}$$

discuss: constructors, (freely) generatedness

# Freely is Expressible

As you know (by now):

'generatedness' is *not expressible* by (first-order or simpler) formulas.

But:

$\boxed{\text{'freely' is expressible}}$ (honestly!)

**Exercise 6.3.**
*Let* $\Sigma = (S, \Omega)$ *be an arbitrary signature and* $\Omega_c \subseteq \Omega$.
*Now consider the class* $Gen(\Sigma, \Omega_c)$ *of algebras generated by* $\Omega_c$.
*How can one derive from* $\Sigma$ *and* $\Omega_c$ *a set of formulas, specifying the algebras from* $Gen(\Sigma, \Omega_c)$ *which are* freely generated *by* $\Omega_c$*?*



# Freely Generate Exercise

**Exercise 6.4.**
*Reconsider the specification you developed for exercise 1.1. Are the 'intended' algebras (freely) generated? If yes: by which subset of the operations?*