# Chapter 2

# Basic Universal Algebra

## 2.1 Algebras

By a (homogeneous) **algebra** we will mean a triple $\mathcal{A} = (A, C, \Phi)$ where

- $A$ is a set, called the **carrier** of $\mathcal{A}$;
- $C$ is a subset of $A$, whose elements are called the **constants** of $\mathcal{A}$; and
- $\Phi$ is a set of operations on $A$.

No restrictions are placed on the cardinalities of $A$, $C$ and $\Phi$. When the two latter sets are finite, $C = \{a_1, \dots, a_n\}$ and $\Phi = \{\phi_1, \dots, \phi_m\}$, then, as a notational convention, we may write $\mathcal{A}$ as $(A; a_1, \dots, a_n; \phi_1, \dots, \phi_m)$, or as $(A; \phi_1, \dots, \phi_m)$ if $C = \emptyset$ . For any $n \geq 1$ we define $\Phi_n \subseteq \Phi$ to contain the $n$-ary operations in $\Phi$, where we may, of course, have $\Phi_n = \emptyset$. Accordingly, $\Phi = \Phi_1 \cup \Phi_2 \cup \cdots$. When the constants and the operations are fixed or irrelevant then we may identify an algebra with its carrier, using the symbols $\mathcal{A}$ and $A$ interchangeably. This could potentially make the symbol '$A$' ambiguous (are we referring to $A$ alone, as a set, or to entire structure $(A, C, \Phi)$?), but in fact the context will always make our intentions clear.

**Example 2.1.1** The set of natural numbers $N = \{0, 1, 2, \dots\}$ is the carrier of many interesting algebras. We single out a few of them below:

$$(N; 0; \,')$$

$$\mathcal{PA} = (N; 0; \,', +, \cdot)$$

$$(N; 0; \,', +, -, \cdot)$$

$$(N; 0; \,', +, -, \cdot, \div)$$

Here we have written $'$ for the successor function on $N$, $n' = n + 1$, and $+, -, \cdot$, and $\div$ for binary addition, subtraction, multiplication, and integer division, respectively. We have singled out the second algebra by giving it the name $\mathcal{PA}$; we will refer to it several times in the sequel.

A number of well-known algebras arise by taking the two-element set of truth values $B = \{\mathbf{t}, \mathbf{f}\}$ as the carrier. A prime example is the algebra

$$(B; \mathbf{t}, \mathbf{f}; ^{-}, +, \cdot)$$

where $^{-}$ stands for negation and $+, \cdot$ for disjunction and conjunction, respectively. These operations are defined by the familiar truth tables, which we do not specify here.

A similar example (for reasons that will become more clear in the sequel) is the algebra

$$(\mathcal{P}(A); \emptyset, A; ^{-}, \cup, \cap)$$

where $\mathcal{P}(A)$ is the power-set of a set $A$, $^{-}$ is set-complementation, and $\cup, \cap$ are set-theoretic union and intersection, respectively.

Another example is $([N \to N]; \circ)$, where $[N \to N]$ is the set of all functions from $N$ to $N$ and $\circ$ is the binary operation of function composition.

An algebra with many applications in computer arithmetic is

$$(N_k; +_k, \cdot_k)$$

for $k \geq 1$, where $N_k = \{0, 1, \ldots, k-1\}$ and $+_k, \cdot_k$ are addition and multiplication modulo $k$.

Our final example is the algebra $(\Sigma^+; \cdot)$, where $\Sigma^+$ is the set of all non-empty finite words over an alphabet $\Sigma$ and $\cdot$ is the binary operation of concatenation. ■

The adjective "homogeneous" in the preceding definition reflects the fact that there is only one carrier. Allowing multiple carriers takes us to heterogeneous (or "multi-sorted") algebras. These are very important in the field of algebraic specifications and we will study them in detail later, but their treatment requires fairly complicated notation. For this reason, we prefer to introduce the basic notions of algebra (morphisms, congruences, etc.) in the homogeneous setting; the subsequent conceptual leap to the heterogeneous case should prove straightforward.

## Concrete, abstract, and universal algebra

Algebra can be studied at three main levels of abstraction. At the lowest level we might focus our attention on one *specific* algebra—out of the enormous space of all possible algebras—and proceed to accumulate a large body of knowledge about it and about its relationship to certain other algebras. Number theory, for instance, may be viewed as the study of the algebra formed by the integers under the few familiar arithmetic operations. In such cases, then, we are dealing with a *concrete algebra*: a specific, fixed carrier along with a number of specific operations on the carrier. However, modern mathematics is never done at such a low level of abstraction. Even when we are working with a particular algebra $\mathcal{A}$, what we are *really* studying is the class of all algebras that are isomorphic to $\mathcal{A}$ (we will shorty give a formal definition of isomorphism); and the results we obtain are applicable to all such algebras.

At a much higher level of generality we find *abstract algebra*. In abstract algebra we are not dealing with any one algebra in particular, but rather with a *class* of algebras whose members bear certain similarities. These similarities are invariably expressible by a collection of postulates that impose certain constraints on the behavior of the constants and/or the operations. The postulates are often

given as formulas in a certain logic, e.g. as equations, or as Horn clauses, or as first-order axioms, etc. They may be quite general and inclusive (for instance, in defining the class of semigroups we only require that the binary operation be associative), or they may be quite specific and restrictive. Of course if the postulates are too general—an extreme case being, say, $x = x$—then we start to degenerate towards universal algebra; and if they are too specific—to the point where the operations are fully defined, in the extreme case—then we degenerate towards concrete algebra. Most cases lie somewhere in the middle, of course. The axioms that define the class of lattices, for example, impose a good deal of structure without being overly constraining.

Finally, at the highest possible level of abstraction we have *universal algebra*, which studies the class of *all* algebras without making any special assumptions whatsoever about the carrier or the constants and operations. This might seem too general to be of any value, but in fact a great many concepts such as subalgebras, generation, morphisms, congruences, freeness, etc., can profitably be formulated in this setting. What we gain is generality: the concepts we define and the results we obtain will be applicable to all algebras. In the next few sections we will study the fundamental ideas of universal algebra.

## 2.2 Subalgebras

For the rest of this section fix an algebra $\mathcal{A} = (A, C, \Phi)$ and let $B \subseteq A$. We say that $B$ is a **subalgebra** of $A$, written $B \leq A$, iff

- $C \subseteq B$, i.e. $B$ contains every constant of $\mathcal{A}$; and
- $B$ is closed under every operation in $\Phi$, i.e., for any $a_1, \ldots, a_n \in B$ and $\phi \in \Phi_n$ we have $\phi(a_1, \ldots, a_n) \in B$.

If $B \leq A$ and $B \neq A$ we say that $B$ is a **proper subalgebra** of $A$ and write $B < A$. We will write $Sub\{A\}$ (or, alternatively, $Sub\{\mathcal{A}\}$) for the collection of all subalgebras of $A$.

Note that it is really an abuse of terminology to say that the *set* $B$ is a subalgebra of $A$. Strictly speaking, what we should say is that the algebra $\mathcal{B} = (B, C, \{\phi \restriction B^n \mid \phi \in \Phi_n, n \geq 1\})$ is a subalgebra of $\mathcal{A}$. But the abuse is consistent with the convention we established in the opening of the previous section, namely that of identifying algebras with their carriers. Again, the context will always make clear whether we are referring to the set $B$ or to the associated algebra—$B$ plus constants plus operations.

**Example 2.2.1** Consider $(N; +, \cdot)$, the natural numbers with addition and multiplication. The set of even numbers $\{2n \mid n \in N\}$ is a (proper) subalgebra of $N$ because it is closed under both operations. By contrast, the set of odd numbers is not a subalgebra of $N$ because it is not closed under addition. ■

**Example 2.2.2** Consider the algebra $(\mathcal{P}(A); \emptyset, A; ^-, \cup, \cap)$ of Ex. 2.1.1. Here the set $\{\emptyset, A\}$ is a subalgebra because it contains the two constants $\emptyset$ and $A$ and is closed under all three operations. One can intuitively see that this is the "smallest" subalgebra there is in this case. It is also "essentially the same" as the algebra $(B; \mathbf{t}, \mathbf{f}; ^-, +, \cdot)$ of truth values, with $\emptyset$ playing the role of $\mathbf{f}$, $A$ for $\mathbf{t}$, and $^-, \cup, \cap$ as negation, disjunction, and conjunction, respectively. In mathematical parlance, the two algebras are *isomorphic*, an important concept that will soon be made precise. ■

The following is an important observation:

**Proposition 2.2.1** *The intersection of any number of subalgebras of $A$ is itself a subalgebra of $A$.*

**Proof**: Let $\{B_i \mid i \in I\}$ be any $I$-indexed collection of subalgebras of $A$ and set $B = \cap\{B_i \mid i \in I\}$. Every constant $a \in C$ is in every $B_i$, hence $a \in B$ as well. Further, pick any $\phi \in \Phi_n$ and let $a_1, \ldots, a_n \in B$, so that $a_1, \ldots, a_n \in B_i$ for every $i \in I$. Because all the $B_i$ are closed under $\phi$, the element $\phi(a_1, \ldots, a_n)$ is in every $B_i$, hence it is also in $B$. Therefore, $B \leq A$. ■

Moreover, because we trivially have $A \leq A$, we conclude that $Sub\{A\}$ is a closure system on $A$:

**Proposition 2.2.2** *The collection of all subalgebras of $A$ is a closure system on $A$.*

Closure systems of this kind are called **algebraic**. That is, a closure system is algebraic iff it consists of all and only the subalgebras of some algebra. Algebraic closure systems are very important in Computer Science for reasons that we will discuss in Sec. 2.8.

The associated closure operator $[]_{Sub\{A\}}$ maps an arbitrary subset $X \subseteq A$ to the intersection of all subalgebras of $A$ that include $X$, i.e. , to the smallest subalgebra of $A$ that includes $X$. As usual, when $A$ is understood or immaterial we drop the subscript and simply write $[X]$. We call $[X]$ the **subalgebra generated by** $X$. If $[X] = A$ we say that the algebra $A$ itself is generated by $X$, and we call $X$ a **generating set** of $A$. If $X$ is finite we say that $A$ is **finitely generated**. As a notational convention, when $X$ contains finitely many elements $x_1, \ldots, x_k$, we will write $[x_1, \ldots, x_k]$ instead of $[\{x_1, \ldots, x_k\}]$.

The subalgebra $[X]$ has an alternative "bottom-up" characterization that helps to explain the constructive connotation of the phrase "generated by $X$". In particular, we define an operator $Con_A : \mathcal{P}(A) \to \mathcal{P}(A)$ as

$$Con_A(X) = X \cup C \cup \{\phi(a_1, \ldots, a_n) \mid \phi \in \Phi_n, a_1, \ldots, a_n \in X\}$$

for any $X \subseteq A$. Thus $Con_A(X)$ is $X$ augmented with the constants of $A$ and all the elements that can be obtained by applying an operation of $A$ to members of $X$. The subscript $A$ will be omitted when there is no risk of confusion. As usual, the iterated composition of $Con$ is defined as

$$\begin{aligned} Con^0(X) &= X \\ Con^{i+1}(X) &= Con(Con^i(X)). \end{aligned}$$

Finally we set

$$Con^\infty(X) = \bigcup_{i \in N} Con^i(X).$$

Thus $Con^\infty(X)$ can be thought of as the limit of a construction process that proceeds in discrete stages: we begin at stage 0 with $X$, the "basis" elements. Then at each new stage we incorporate the constants in $C$ along with whatever elements can be obtained by applying operations in $\Phi$ to elements of the set from the previous stage. If, at some finite stage, no new elements can be obtained in this manner, then we have reached a fixed point and we are done—we have "closed" $X$ in a finite number of steps. Otherwise the construction proceeds *ad infinitum*. We will now prove that $Con^\infty(X)$ is none other than $[X]$, the smallest subalgebra that includes $X$:

**Theorem 2.2.1**

$$[X] = Con^{\infty}(X).$$

**Proof**: In one direction it is easy to check that $Con^{\infty}(X)$ is a subalgebra of $A$ that includes $X$, and hence, since $[X]$ is the least such subalgebra, $[X] \subseteq Con^{\infty}(X)$. In the other direction we use induction to prove that for all $i \in N$, $Con^{i}(X) \subseteq [X]$. The basis case $i = 0$ is trivial. For the inductive step, assume that $Con(X)^{i} \subseteq [X]$ for some $i \in N$ and consider any $a \in Con^{i+1}(X)$. By definition, either $a \in Con^{i}(X)$, or $a \in C$, or else $a = \phi(a_1, \ldots, a_n)$ for some $\phi \in \Phi_n$ and $a_1, \ldots, a_n \in Con^{i}(X)$. In the first case, $a \in [X]$ by the inductive hypothesis. The second case is also immediate, since $C \subseteq [X]$ by the definition of a subalgebra. In the third case, $a_1, \ldots, a_n$ are also in $[X]$ (again by the inductive hypothesis), and since $[X]$ is closed under $\phi$, we must have $\phi(a_1, \ldots, a_n) = a \in [X]$. Thus in all possible cases $a \in [X]$, and we may conclude $Con^{i+1} \subseteq [X]$, completing the induction. It now easily follows that $Con^{\infty}(X) \subseteq [X]$, and having shown subset inclusion in both directions we can finally infer the claimed equality. ■

A simpler proof can be given using Tarski's fixed-point theorem. The main observation is quite simple:

**Proposition 2.2.3** *A subset $X \subseteq A$ is a subalgebra of $A$ iff it is a fixed point of $Con_A$.*

Next we observe that $Con_A$ is continuous:

**Lemma 2.2.1** *$Con_A$ is a continuous operator.*

**Proof**: It follows directly from its definition that $Con_A$ is monotonic and compact. Hence, by Th. **??**, it is continuous as well. ■

Finally:

*Alternative proof of Th. 2.2.1.* From the fixed-point theorem (Th. **??**), we know that $Con^{\infty}(X)$ is the least fixed point of $Con$ that includes $X$. But, from Lem. 2.2.1, this is the least subalgebra of $A$ that includes $X$, namely $[X]$. ■

**Example 2.2.3** All of the algebras of Ex. 2.1.1 with $N$ as their carrier are finitely generated by $\emptyset$, that is, $\emptyset = [N]$. This can be understood via Th. 2.2.1: $Con^{1}(\emptyset)$ contains 0, $Con^{2}(\emptyset)$ contains $0' = 1$, $Con^{3}(\emptyset)$ contains $0'' = 2$, and so on. Thus $Con^{\infty}(\emptyset) = [\emptyset] = N$. ■

**Example 2.2.4** The algebra $(\Sigma^{+}; \cdot)$ is finitely generated by $\Sigma$, i.e. $[\Sigma] = \Sigma^{+}$. In terms of the constructive characterization of $[\Sigma]$: $Con^{0}(\Sigma)$ contains all one-letter words, while

$$Con^{i+1}(\Sigma) = Con(Con^{i}(\Sigma)) = \{u \cdot v \mid u, v \in Con^{i}(\Sigma)\}$$

contains all words that can be formed by joining two words in $Con^{i}(\Sigma)$. It is a simple exercise to show that for all $i \in N$,

$$Con^{i}(\Sigma) = \{w \in \Sigma^{+} \mid |w| \leq 2^{i}\}. \ ■$$

### Minimal algebras

An algebra that has no proper subalgebras is called **minimal**. Minimal algebras are appealing because they are simple: they have as few elements as possible and satisfy as few laws as possible. For many purposes, when an algebra does have a proper subalgebra then the elements which are not in that subalgebra are likely to be redundant. As an example of minimality consider any algebra that has $N$ as its carrier and which includes zero as a constant and the successor function as an operation. Any such algebra is minimal because any subalgebra $[X]$ must include 0 and $0+1$ and $(0+1)+1$, and so on, and thus $[X] = N$—we cannot have $[X] < N$. The following offers a simple characterization of minimality:

**Proposition 2.2.4**

(a) $[C] \subseteq [X]$ *for any* $X \subseteq A$.

(b) $A$ *is minimal iff* $A = [\emptyset]$, *or equivalently, iff* $A = [C]$.

**Proof**: (a) An easy induction on $i$ will show that $Con^i(C) \subseteq Con^{i+1}(X)$ for all $i \in N$. For $i = 0$ this is immediate. Assume the claim holds for some $i \in N$. Then

$$\begin{aligned} Con^{i+1}(C) &= Con^i(C) \cup C \cup \{\phi(a_1, \ldots, a_n) \mid a_1, \ldots, a_n \in Con^i(C), \phi \in \Phi_n\} \\ Con^{i+2}(X) &= Con^{i+1}(X) \cup C \cup \{\phi(a_1, \ldots, a_n) \mid a_1, \ldots, a_n \in Con^{i+1}(X), \phi \in \Phi_n\} \end{aligned}$$

and the inclusion follows from the inductive hypothesis. Thus $Con^\infty(C) \subseteq Con^\infty(X)$, i.e. $[C] \subseteq [X]$. (b) Clearly, if $[\emptyset] \neq A$ then $[\emptyset]$ is a proper subalgebra of $A$ and hence the latter is not minimal. Conversely, if $A$ is not minimal then $[X] < A$ for some $X \subseteq A$. But $[\emptyset] \subseteq [X]$ (since $\emptyset \subseteq X$), thus $[\emptyset] \subset A$ and hence $[\emptyset] \neq A$. Moreover, if $A \neq C$ then, again, $A$ cannot be minimal. And conversely, if $A$ is not minimal then $[X] < A$ for some $X \subseteq A$. But, from (a), $[C] \subseteq [X]$, hence $[C] \subset A$ and $A \neq [C]$. ■

## 2.3 Homomorphisms

By a **translation** $T$ from an algebra $(A, C, \Phi)$ to an algebra $(A', C', \Phi')$ we will mean a pair of surjective mappings $\sigma : C \to C'$ and $\tau : \Phi \to \Phi'$ such that $\tau(\phi) \in \Phi'_n$ for every $\phi \in \Phi_n$. Thus a translation preserves arities: an $n$-ary operation in $\Phi$ must get mapped to an $n$-ary operation in $\Phi'$. The surjectivity of $\sigma$ and $\tau$ ensures that every constant in $C'$ and every operation in $\Phi'$ is the image of some constant in $C$ or operation in $\Phi$, respectively. However, neither $\sigma$ nor $\tau$ are required to be injective: two different constants in $C$ might be "translated" into the same constant in $C'$, and likewise for operations. Hence a translation is not necessarily invertible. Usually, however, $\sigma$ and $\tau$ *are* injective, and hence bijective as well, and thus there exists an inverse translation from $(A', C', \Phi')$ to $(A, C, \Phi)$ given by $\sigma^{-1}$ and $\tau^{-1}$. When such a bijective translation exists we say that the two algebras are **similar**. Of course many different bijective translations might exist between a pair of similar algebras, depending on how we choose to pair up constants and operations. As a convention, when the mappings $\sigma$ and $\tau$ are fixed or immaterial, we will write $c'$ and $\phi'$ instead of $\sigma(c)$ and $\tau(\phi)$, respectively.

**Example 2.3.1** A translation from the algebra $([N \to N]; \circ)$ of Ex. 2.1.1 to $(\Sigma^+; \cdot)$ is given by the mapping $\circ \mapsto \cdot$. It is obvious that this is the only possible translation. It is also a bijective translation, hence the two algebras are similar. ■

**Example 2.3.2** A translation from $(B; \mathbf{t}, \mathbf{f}; ^-, +, \cdot)$ to $(\mathcal{P}(A); \emptyset, A; ^-, \cup, \cap)$ is given by the mappings $\{\mathbf{t} \mapsto A, \mathbf{f} \mapsto \emptyset\}$ and $\{^- \mapsto ^-, + \mapsto \cup, \cdot \mapsto \cap\}$. This is also a bijective translation, hence the two algebras are similar. In this case there are three other translations, namely

$$\{\mathbf{t} \mapsto A, \mathbf{f} \mapsto \emptyset\} \text{ and } \{^- \mapsto ^-, + \mapsto \cap, \cdot \mapsto \cup\}$$

$$\{\mathbf{t} \mapsto \emptyset, \mathbf{f} \mapsto A\} \text{ and } \{^- \mapsto ^-, + \mapsto \cup, \cdot \mapsto \cap\}$$

$$\{\mathbf{t} \mapsto \emptyset, \mathbf{f} \mapsto A\} \text{ and } \{^- \mapsto ^-, + \mapsto \cap, \cdot \mapsto \cup\}. \blacksquare$$

A very fundamental concept relating different algebras is that of a *homomorphism*. Consider two algebras $(A, C, \Phi)$ and $(A', C', \Phi')$, along with a translation $T = (\sigma, \tau)$ from the former to the latter. A function $h : A \to A'$ is said to be a **homomorphism with respect to** $T$ iff for all $a \in C, \phi \in \Phi_n$, and $a_1, \ldots, a_n \in A$, we have

$$h(a) = a' \tag{2.1}$$

$$h(\phi(a_1, \ldots, a_n)) = \phi'(h(a_1), \ldots, h(a_n)). \tag{2.2}$$

where, as we agreed above, we are writing $a'$ for $\sigma(a)$ and $\phi'$ for $\tau(\phi)$. We say that $h$ **respects** or **preserves** the operations of $\mathcal{A}$ (with respect to $T$). The relationship (2.2) is graphically represented by the following diagram:

$$\begin{tikzcd}
A^n \arrow[r, "\phi"] \arrow[d, "h^n"'] & A \arrow[d, "h"] \\
A'^n \arrow[r, "\phi'"] & A'
\end{tikzcd}$$

Diagrams such as the above are said to *commute*, meaning that any two paths which begin and end at the same respective points represent identical mappings—they yield the same result when applied to the same arguments. In this case, the diagram says that the same result will be obtained regardless of whether we

1. first apply $\phi$ to a tuple of $n$ arguments in $A$ and then apply $h$ to the result $\phi(a_1, \ldots, a_n)$; or
2. first apply $h$ to each of the $n$ arguments and then apply $\phi'$ to the resulting tuple $h(a_1), \ldots, h(a_n)$.

When the tranlation $T$ is fixed or irrelevant we will omit the qualification "with respect to $T$" and speak of $h$ directly as a homomorphism.

**Example 2.3.3** Consider the algebras $(\Sigma^*; \cdot)$, strings over $\Sigma$ with concatenation, and $(N; +)$, the natural numbers with addition. These two algebras are similar under the obvious translation $\cdot \mapsto +$. Let $l : \Sigma^* \to N$ be the length function that maps a string $w \in \Sigma^*$ to $|w|$. It should be clear that $l$ "respects" concatenation:

$$l(u \cdot v) = l(u) + l(v)$$

for all $u, v \in \Sigma^*$. Therefore, $l$ is a homomorphism. ■

**Example 2.3.4** Consider the algebras $(B; \mathbf{t}, \mathbf{f}; \ , +, \cdot)$ and $(\mathcal{P}(A); \emptyset, A; \ , \cup, \cap)$ of Ex. 2.1.1. These two algebras are similar under the correspondence $\mathbf{f} \mapsto \emptyset, \mathbf{t} \mapsto A, \ \mapsto \ , + \mapsto \cup, \cdot \mapsto \cap$.[1] Define $h : \{\mathbf{f}, \mathbf{t}\} \to \mathcal{P}(A)$ as $h(\mathbf{f}) = \emptyset$, $h(\mathbf{t}) = A$. We claim that $h$ is a morphism with respect to this translation. For the constants of the two algebras the morphism property follows directly from the definition of $h$. For the negation operation $^-$ we have

$$h(\bar{\mathbf{f}}) = h(\mathbf{t}) = A = \overline{\emptyset} = \overline{h(\mathbf{f})}$$

and

$$h(\bar{\mathbf{t}}) = h(\mathbf{f}) = \emptyset = \overline{A} = \overline{h(\mathbf{t})}.$$

Next we check the disjunction operation $+$. Pick any $b_1, b_2 \in \{\mathbf{f}, \mathbf{t}\}$. Either $b_1 = b_2 = \mathbf{f}$ or not. In the first case

$$h(b_1 + b_2) = h(\mathbf{f}) = \emptyset = \emptyset \cup \emptyset = h(b_1) \cup h(b_2).$$

In the second case assume—without loss of generality—that $b_1 = \mathbf{t}$. Then

$$h(b_1 + b_2) = h(\mathbf{t}) = A = A \cup h(b_2) = h(b_1) \cup h(b_2).$$

Thus in either case we have $h(b_1 + b_2) = h(b_1) \cup h(b_2)$.

Finally we check the conjunction operation. Again pick any two $b_1, b_2 \in \{\mathbf{f}, \mathbf{t}\}$. Either $b_1 = b_2 = \mathbf{t}$ or not. In the first case

$$h(b_1 \cdot b_2) = h(\mathbf{t}) = A = A \cap A = h(b_1) \cap h(b_2).$$

In the second case assume, without loss of generality, that $b_1 = \mathbf{f}$. Then

$$h(b_1 \cdot b_2) = h(\mathbf{f}) = \emptyset = \emptyset \cap h(b_2) = h(b_1) \cap h(b_2).$$

Thus in either case $h(b_1 \cdot b_2) = h(b_1) \cap h(b_2)$, and $h$ is a morphism. ■

**Example 2.3.5** Consider the algebras $(N; +)$, the natural numbers with addition, and $(N_3; +_3)$, the set $\{0, 1, 2\}$ with addition modulo 3, under the obvious translation. The function $h : N \to N_3$ defined as $h(n) = n \bmod 3$ is a homomorphism, i.e.,

$$h(n + m) = h(n) +_3 h(m) \tag{2.3}$$

for all $n, m \in N$. The situation is depicted by the commuting diagram

[1] Of course they are similar under other translations as well, as shown in Ex. 2.3.2. For instance, we could pair up $\mathbf{f}$ with $A$ and $\mathbf{t}$ with $\emptyset$. But then the function $h$ defined in this example might not be a homomorphism; see Exer. **??**.

$$\begin{tikzcd}
N \times N \arrow[r, "+"] \arrow[d, "h^2"'] & N \arrow[d, "h"] \\
N_3 \times N_3 \arrow[r, "+_3"] & N_3
\end{tikzcd}$$

which says that the same result is obtained whether we

1. first add two numbers in $N$ and then take the remainder of their sum modulo 3, or
2. first compute the remainder of each of the two numbers and then apply $+_3$ to the results.

Consider 4 and 11, for instance. In one direction, $h(4+11) = h(15) = 0$, while in the other direction, $h(4) +_3 h(11) = 1 +_3 2 = 0$.

To *prove* that (2.3) holds, let $r_1 = n \bmod 3$ and $r_2 = m \bmod 3$, so that $n = 3q_1 + r_1$ and $m = 3q_2 + r_2$ for some numbers $q_1, q_2$. Now

$$\begin{aligned}
h(n+m) &= h(3q_1 + r_1 + 3q_2 + r_2) \\
&= h(3(q_1 + q_2) + (r_1 + r_2)) \\
&= h(r_1 + r_2) \\
&= (r_1 + r_2) \mod 3
\end{aligned}$$

while $h(n) +_3 h(m) = r_1 +_3 r_2 = (r_1 + r_2) \mod 3$, which proves that $h$ is indeed a morphism. ■

A computer scientist might well intuit a connection between the defining property of a morphism and recursion. We will see later that there is indeed a connection; the following example offers a preview:

**Example 2.3.6** Let **GA** be the language generated by the following context-free grammar:

$$E ::= \texttt{0} \mid \texttt{s(}E\texttt{)} \mid \texttt{(}E\texttt{+}E\texttt{)} \mid \texttt{(}E\texttt{*}E\texttt{)}$$

and consider the algebra $\mathcal{G} = (\mathbf{GA}; \phi_{\texttt{0}}; \phi_{\texttt{s}}, \phi_{\texttt{+}}, \phi_{\texttt{*}})$, where the constant $\phi_{\texttt{0}}$ is simply the symbol $\texttt{0}$, while the operations $\phi_{\texttt{s}}, \phi_{\texttt{+}}$ and $\phi_{\texttt{*}}$ are defined as follows:

$$\begin{aligned}
\phi_{\texttt{s}}(E) &= \texttt{s(}E\texttt{)} \\
\phi_{\texttt{+}}(E_1, E_2) &= \texttt{(}E_1 \texttt{ + } E_2\texttt{)} \\
\phi_{\texttt{*}}(E_1, E_2) &= \texttt{(}E_1 \texttt{ * } E_2\texttt{)}.
\end{aligned}$$

Note that $\mathcal{G}$ is similar to $\mathcal{PA}$ under the natural translation $\phi_{\texttt{0}} \mapsto 0, \phi_{\texttt{s}} \mapsto {}', \phi_{\texttt{+}} \mapsto +, \phi_{\texttt{*}} \mapsto \cdot$.

Now let us define a semantics for **GA** via a "meaning function" $\mathcal{M}$ that maps expressions $E \in \mathbf{GA}$ to the natural numbers:

$$\begin{array}{rcl} \mathcal{M}[\mathtt{0}] & = & 0 \\ \mathcal{M}[\mathtt{s}(E)] & = & \mathcal{M}[E]' \\ \mathcal{M}[(E_1 \ \mathtt{+} \ E_2)] & = & \mathcal{M}[E_1] + \mathcal{M}[E_2] \\ \mathcal{M}[(E_1 \ \mathtt{*} \ E_2)] & = & \mathcal{M}[E_1] \cdot \mathcal{M}[E_2]. \end{array}$$

It follows directly from this definition that $\mathcal{M}$ is a homomorphism from **GA** to $\mathcal{PA}$ (with respect to the aforementioned translation). E.g. for addition we have

$$\begin{tikzcd} \mathbf{GA} \times \mathbf{GA} \arrow[r, "\phi_+"] \arrow[d, "\mathcal{M}^2"'] & \mathbf{GA} \arrow[d, "\mathcal{M}"] \\ N \times N \arrow[r, "+"] & N \end{tikzcd}$$

In fact $\mathcal{M}$ is an *epimorphism* (see the discussion below) since it is onto, and thus for the purposes of addition and multiplication the natural numbers can be seen as an abstract model of **GA**, with each number corresponding to a unique element in the kernel relation of $\mathcal{M}$. This is encountered time and again in Computer Science: most "meaning" functions (a.k.a. "evaluation" functions) are directly expressible as epimorphisms from "term algebras" (such as $\mathcal{G}$) to appropriate similar algebras (such as $\mathcal{PA}$). The morphism property is simply a formal manifestation of the so-called *compositionality principle* of semantics, which states that "the meaning" of an expression $E$ should be a function of the meanings of one or more proper subexpressions of $E$. ■

## A classification of morphisms

Let $h : A \rightarrow B$ be a morphism from an algebra $A$ to an algebra $B$, with respect to some unspecified translation. We say that $h$ is

- a **monomorphism** if $h$ is 1-1;
- an **epimorphism** if $h$ is onto;
- an **isomorphism** if $h$ is a bijection (both 1-1 and onto);
- an **endomorphism** if $A = B$; and
- an **automorphism** if $A = B$ and $h$ is a bijection.

Isomorphisms and epimorphisms are the most important concepts from the above list.

**Isomorphisms**

If there is an isomorphism $h : A \to B$ from an algebra $A$ to an algebra $B$ then we say that $A$ and $B$ are **isomorphic**, written $A \cong B$. Isomorphic algebras are "essentially the same", because any element $a \in A$ (respectively, $b \in B$) may be identified with its unique mirror element $h(a) \in B$ (respectively, $h^{-1}(b) \in A$). The two may be identified because, by the morphism property, $a$ behaves in $A$ precicely as $b$ behaves in $B$. A simple example is given by $(N, ')$, the natural numbers with the successor function, and $(\{2n \mid n \in N\}, f)$, the even natural numbers with $f(n) = n + 2$, with the obvious translation. The two are isomorphic under the mapping $h = 2n$, as for any $n \in N$,

$$h(n') = h(n + 1) = 2(n + 1) = 2n + 2 = h(n) + 2 = f(h(n)).$$

Thus 0 may be identified with 2, 1 with 4, 3 with 6, etc. In general, the elements of $B$ can be seen as "different names" for the elements of $A$, with $h$ providing the renaming. Or, equivalently, the elements of $A$ can be seen as different names for those of $B$, with $h^{-1}$ providing the renaming.

As another example, consider the morphism $l$ from $(\Sigma^*; \cdot)$ to $(N; +)$ in Ex. 2.3.3. Now in general $l$ is not an isomorphism, as several different strings might have the same length. But when $\Sigma$ contains only one element, say the tally $|$, then $l$ is a bijection and the two algebras are isomorphic: a number $n$ is identified with $|^n$, a string of $n$ occurences of $|$. Informally this means that, for purposes of counting, the sets $\{|\}^*$ and $N$ are indistinguishable.

As these two examples hint, only similar algebras can be isomorphic:

**Proposition 2.3.1** *If $h : A \to B$ is an isomorphism with respect to some translation $(\sigma, \tau)$ then $\sigma$ and $\tau$ are bijections, and hence $A$ and $B$ are similar algebras. Further, $h^{-1} : B \to A$ is an isomorphism with respect to the inverse translation $(\sigma^{-1}, \tau^{-1})$.*

**Proof**. The mappings $\sigma$ and $\tau$ are surjective by definition, so we only need to show that they are injective. This follows because $h$ is injective. Specifically, for any two distinct constants $a_1, a_2$ of $A$ we have $h(a_1) = \sigma(a_1) \neq h(a_2) = \sigma(a_2)$, since $h$ is injective. Likewise, pick any two distinct $n$-ary operations $\phi_1, \phi_2$ of the algebra $A$. Since $\phi_1 \neq \phi_2$ there exists a tuple $(a_1, \ldots, a_n) \in A^n$ such that $\phi_1(a_1, \ldots, a_n) \neq \phi_2(a_1, \ldots, a_n)$. Consequently, $h(\phi_1(a_1, \ldots, a_n)) \neq h(\phi_2(a_1, \ldots, a_n))$. But $h(\phi_1(a_1, \ldots, a_n)) = \tau(\phi_1)(h(a_1), \ldots, h(a_n))$ and $h(\phi_2(a_1, \ldots, a_n)) = \tau(\phi_2)(h(a_1), \ldots, h(a_n))$, hence $\tau(\phi_1)(h(a_1), \ldots, h(a_n)) \neq \tau(\phi_2)(h(a_1), \ldots, h(a_n))$, which entails $\tau(\phi_1) \neq \tau(\phi_2)$. Thus the inverse translation $(\sigma^{-1}, \tau^{-1})$ exists, and it is now straightforward to show that $h^{-1}$ is an isomorphism. ■

Let $P$ be a property that an algebra might have. We say that $P$ is **algebraic** if whenever it holds for an algebra $A$ and $A \cong B$, then it also holds for $B$. That is, $P$ is algebraic if it is "closed under isomorphism". If we extensionalize, identifying a property with the class of algebras for which it holds, then we might say that a class of algeras $\mathbf{K}$ is *algebraic* iff $B \in \mathbf{K}$ whenever $A \in \mathbf{K}$ and $A \cong B$. For instance, the property "the carrier contains the number 3" is not an algebraic property as there are algebras $A$ that have this property and there are others, isomorphic to $A$, that do not (the preceding example of $(N; ')$ and $(\{2n \mid n \in N\}; \lambda n.n + 2)$ being a case in point). By contrast, the property "the carrier is finite" is easily seen to be algebraic, as no finite algebra can be isomorphic to an infinite one. Philosophically, we might view algebraic and non-algebraic properties as "essential" and "accidental", respectively, in Aristotle's terminology. Modern mathematics is mostly concerned with algebraic properties.

**Epimorphisms**

When $h : A \to B$ is an epimorphism we say that $B$ is a **homomorphic image** of $A$. Epimorphisms are frequently encountered in denotational semantics, and we have already seen an example of this with the "meaning function" $\mathcal{M}$ from the strings in **GA** to the natural numbers. Later we will see that this is just an instance of the more general phenomenon that *any* algebra can be undestood as the homomorphic image of an appropriate "term algebra".

Because an epimorphism $h : A \to B$ usually "crams" $A$ into $B$ by mapping several distinct elements of $A$ to the same image in $B$, for many purposes we might view an element $b \in B$ as an "approximation" or "abstract description" of all the different $a \in A$ such that $h(a) = b$. (Of course if $h$ is 1-1 then the approximation is perfect—$h$ is an isomorphism.) This can have useful practical applications if the operations in $B$ are "easier" to perform than those in $A$. Then instead of computing in $A$, whose operations might be very expensive—or even uncomputable—we compute in $B$ with the approximations of the relevant objects. This is more or less the idea behind *abstract interpretation*, a technique that has been used mainly for progral analysis, but which, in principle, is applicable in any situation in which we wish to simulate a complex system by a simpler abstraction.

As another example, the epimorphism $h(n) = n \mod 3$ of Ex. 2.3.5 can be seen as approximating infinite subsets of $N$ by a single number: $\{0, 3, 6, 9, 12, \ldots\}$ by 0, $\{1, 4, 7, 10, \ldots\}$ by 1, and $\{2, 5, 8, 11, \ldots\}$ by 2. This is, of course, a very coarse approximation (too many elements get crammed into the same slot), but even with morphisms of this type useful approximate information can be achieved by performing modular instead of perfect arithmetic.

## 2.4 Signatures and $\mathcal{F}$-algebras

Translations allow us to interrelate different algebras with concepts such as morphisms, by establishing a correspondence from the constants and operations of one algebra to those of another. A convenient way of establishing such correspondences is to index the constants and operations of the algebras we wish to interrelate by the symbols of a certain *signature*, as explained in the next two sections.

### 2.4.1 Signatures

By a **signature** we will mean a pair $(\mathcal{F}, r)$ consisting of a set $\mathcal{F}$ of *function symbols* and a function $r : \mathcal{F} \to N$ that maps each symbol $f \in \mathcal{F}$ to a unique natural number $r(f)$ known as the **arity** of $f$. This should be thought of as the number of arguments that must be supplied in an application of a function denoted by $f$. Function symbols of arity zero will be called *constant symbols*. For any $n \in N$ we define $\mathcal{F}_n = \{f \in \mathcal{F} \mid r(f) = n\}$. Thus $\mathcal{F}_n$ contains those symbols in $\mathcal{F}$ that have arity $n$. Accordingly,

$$\mathcal{F} = \bigcup_{n \in N} \mathcal{F}_n.$$

We will write $\mathcal{F}_+$ for $\mathcal{F} - \mathcal{F}_0$, i.e., for the set of all function symbols in $\mathcal{F}$ that have positive arity.

The arity function $r$ is often specified informally and not mentioned explicitly. In general, $r$ will be omitted whenever it is fixed or immaterial, and we will then speak of $\mathcal{F}$ directly as a signature. We will use the letters $c$ and $d$ (possibly with subscripts, etc.) to denote constant symbols, and the letters $f$ and $g$ to denote function symbols of arbitrary arity (possibly zero).

As a convention, we will often introduce a signature with an expression of the form

$$\mathbf{sig}\,[f_1^1, \ldots, f_{k_1}^1 : d_1; \cdots; f_1^m, \ldots, f_{k_m}^m : d_m]$$

with the understanding that $f_1^i, \ldots, f_{k_i}^i$ are all and only the symbols of arity $d_i$, for $i = 1, \ldots, m$. Thus, more formally, the signature defined by an expression of the above form is

$$(\{f_1^1, \ldots, f_{k_1}^1, \ldots, f_1^m, \ldots, f_{k_1}^m\}, r)$$

where $r(f_j^i) = d_i, 1 \leq j \leq k_i, i = 1, \ldots, m$. Usually the symbols will be listed in order of increasing arity, so that $d_1 < \cdots < d_m$.

**Example 2.4.1** A signature that will come up often in the sequel is

$$\mathcal{F}_P = \mathbf{sig}\,[0 : 0; s : 1; \mathit{plus}, \mathit{times} : 2].$$

Here 0 is the only constant symbol (of arity zero), $s$ has arity one, and *plus*, *times* have arity 2. Two other useful signatures are

$$\mathcal{F}_{B_1} = \mathbf{sig}\,[\top, \bot : 0; \neg : 1; \wedge, \vee : 2]$$

and

$$\mathcal{F}_{B_2} = \mathbf{sig}\,[\top, \bot : 0; \neg : 1; \wedge, \vee, \Rightarrow, \Leftrightarrow : 2]. \quad \blacksquare$$

### 2.4.2 $\mathcal{F}$-algebras

Now let $\mathcal{F}$ be a signature. An **$\mathcal{F}$-algebra** is a triple $(\mathcal{A}, \sigma, \tau)$ consisting of an algebra $\mathcal{A} = (A, C, \Phi)$ and two surjective *realization assignments* $\sigma : \mathcal{F}_0 \to C$ and $\tau : \mathcal{F}_+ \to \Phi$ such that $\tau(f) \in \Phi_n$ for every $f \in \mathcal{F}_n, n \in N_+$. Thus a function symbol of arity $n$ gets mapped to an $n$-ary operation on $A$. Following custom, we will denote $\sigma(c)$ and $\tau(f)$ by $c^{\mathcal{A}}$ and $f^{\mathcal{A}}$, respectively. We call $c^{\mathcal{A}}$ and $f^{\mathcal{A}}$ the **realizations** (or *interpretations*) of $c$ and $f$ in $\mathcal{A}$, respectively. When $\mathcal{A}$ can safely be identified with the carrier $A$ we might write $c^A$ and $f^A$ interchangeably with $c^{\mathcal{A}}$ and $f^{\mathcal{A}}$. The mappings $\sigma$ and $\tau$ are required to be surjective to ensure that every constant and operation of $\mathcal{A}$ is named by some symbol in $\mathcal{F}$. On the other hand, neither $\sigma$ nor $\tau$ are required to be injective, which means that different symbols might receive the same realization as long as the arities are respected. For instance, if $\mathcal{F}$ contains the binary symbols $+$ and $*$, it is possible to interpret both of these by the same binary operation (say, multiplication, if the carrier happens to be $N$). This is rather uncommon, of course, since in most contexts we prefer to denote a given object by a single name. Accordingly, an $\mathcal{F}$-algebra with injective realization assignments will be called **normal**. Consequently, in a normal $\mathcal{F}$-algebra the realization assignments are bijections: every constant and operation has a unique name. It follows that any two normal $\mathcal{F}$-algebras are similar (in the technical sense of similarity defined in Sec. **??**).

The assignments $\sigma$ and $\tau$ are usually specified informally. They may be omitted altogether whenever they are understood or irrelevant, and in that case we may speak of the algebra $\mathcal{A}$ by itself as an $\mathcal{F}$-algebra. And, as usual, whenever the contents of $C$ and $\Phi$ are understood or immaterial, $\mathcal{A}$ and $A$ may be identified, and then we might refer directly to the set $A$ as an $\mathcal{F}$-algebra. We will write $\mathrm{Alg}(\mathcal{F})$ for the class of all $\mathcal{F}$-algebras, for any signature $\mathcal{F}$.

Observe that, because no restrictions are placed on the cardinality of a signature, *any* algebra $(A, C, \Phi)$ can be seen as an $\mathcal{F}$-algebra for some appropriate signature $\mathcal{F}$. For after all we can always take $\mathcal{F}_0 = C$, $\mathcal{F}_+ = \Phi$, and name every constant and operation by itself! (Although admittedly this does not agree with out intuitive notion of a symbol.)

Because the realization assignments of an $\mathcal{F}$-algebra $A$ are surjective, we will be able to "iterate" through the constants and operations of $A$ by iterating through the symbols of $\mathcal{F}$. That is, we will say things like "for any $c \in \mathcal{F}_0$ such and such holds of $c^A$", and "for any $f \in \mathcal{F}_+$ such and such holds of $f^A$". By examining $c^A$ for *all* $c \in \mathcal{F}_0$, we are sweeping through every constant in the algebra, because every constant is named by some $c \in \mathcal{F}_0$. Likewise, by examining $f^A$ for all $f \in \mathcal{F}_+$, we are sweeping through every operation because every operation is named by some $f \in \mathcal{F}_+$.

As a convention, when $\mathcal{F}$ is a finite set with $\mathcal{F}_0 = \{c_1, \ldots, c_k\}$ and $\mathcal{F}_+ = \{f_1, \ldots, f_m\}$, we may represent an $\mathcal{F}$-algebra by writing $(A; c_1^A, \ldots, c_k^A; f_1^A, \ldots, f_m^A)$, where $c_i^A$ is understood to be a constant element of $A$ and $f_j^A$ an operation on $A$ of the same arity as the symbol $f_j$, $1 \leq i \leq k, 1 \leq j \leq m$.

**Example 2.4.2** A prominent example of a $\mathcal{F}_P$-algebra, where $\mathcal{F}_P$ is the signature defined in Ex. 2.4.1, is the algebra $(N; 0; ', +, \cdot)$ of Ex. 2.1.1, with 0 as the realization of 0, $'$ as the realization of $s$, and $+, \cdot$ as the realizations of *plus* and *times*, respectively. Observe the dual role that 0 plays here. On the one hand we are treating 0 as standing for a symbol, and on the other as standing for a mathematical object that becomes the referrent of the aforementioned symbol under the relative realization assignment.

Of course the realizations can be chosen differently as long as the arities are respected, e.g. we can choose to interpret *plus* by the multiplication operation $\cdot$ and *times* by the addition operation $+$. We might even choose to interpret *plus* and *times* by the same binary operation, although the result would then not be a normal $\mathcal{F}_P$-algebra. ■

**Example 2.4.3** The algebra $(B; \mathbf{t}, \mathbf{f}; ^-, +, \cdot)$ is a $\mathcal{F}_{B_1}$-algebra under the realization assignments $\{\top \mapsto \mathbf{t}, \bot \mapsto \mathbf{f}\}$ and $\{\neg \mapsto ^-, \wedge \mapsto \cdot, \vee \mapsto +\}$. Again, different realization assignments can be chosen, e.g. $\{\top \mapsto \mathbf{f}, \bot \mapsto \mathbf{t}\}$ and $\{\neg \mapsto ^-, \wedge \mapsto +, \vee \mapsto \cdot\}$. Another $\mathcal{F}_{B_1}$-algebra is

$$(N; \top^N, \bot^N; \neg^N, \wedge^N, \vee^N)$$

where $\top^N = 0, \bot^N = 58$, $\neg^N$ is the factorial function, and $\wedge^N, \vee^N$ are the integer division and exponentiation operations, respectively. Both of these are normal $\mathcal{F}_{B_1}$-algebras, as different symbols are assigned different realizations. ■

Now given any $\mathcal{F}$-algebras $A$ and $B$, a unique translation from $A$ to $B$ arises naturally as follows: $c^A$ is mapped to $c^B$, and $f^A$ is mapped to $f^B$, for all $c \in \mathcal{F}_0, f \in \mathcal{F}_+$. We will refer to such translations as **$\mathcal{F}$-translations**. An $\mathcal{F}$-translation from $A$ to $B$ might fail to exist if $A$ is a non-normal $\mathcal{F}$-algebra while $B$ is normal. For suppose that $c_1^A = c_2^A$ for two distinct symbols $c_1, c_2 \in \mathcal{F}_0$, whereas, $B$ being normal, $c_1^B \neq c_2^B$. Then shall we translate $c_1^A = c_2^A$ to $c_1^B$ or to $c_2^B$? Clearly, no $\mathcal{F}$-translation from $A$ to $B$ exists in such a case. But it is clear that if an $\mathcal{F}$-translation does exist, it is unique, allowing us to speak of *the* $\mathcal{F}$-translation from $A$ to $B$. It is also easy to verify that the $\mathcal{F}$-translation from $A$ to $B$ always exists if $A$ is normal.

Hereafter whenever we speak of "a homomorphism from an $\mathcal{F}$-algebra $A$ to a $\mathcal{F}$-algebra $B$" it will always be with respect to the $\mathcal{F}$-translation from $A$ to $B$ (assuming, of course, that the translation

exists). Specifically, we define an $\mathcal{F}$-**homomorphism** (or simply "$\mathcal{F}$-morphism") from $A$ to $B$ as a function $h : A \to B$ that is a morphism with respect to the $\mathcal{F}$-translation from $A$ to $B$, i.e. such that

$$\begin{aligned} h(c^A) &= c^B \\ h(f^A(a_1, \ldots, a_n)) &= f^B(h(a_1), \ldots, h(a_n)) \end{aligned}$$

for all $c \in \mathcal{F}_0, f \in \mathcal{F}_n, n \in N_+$, and $a_1, \ldots, a_n \in A$. As we said above, we will usually omit the reference to $\mathcal{F}$ and simply speak of "a morphism from $A$ to $B$". It vacuously follows that if the $\mathcal{F}$-translation from $A$ to $B$ does not exist (e.g. because $A$ is normal and $B$ is not, as we discussed above) then no $\mathcal{F}$-morphisms from $A$ to $B$ exist either.

Finally we remark that a subalgebra $B$ of an $\mathcal{F}$-algebra $A$ will itself be considered an $\mathcal{F}$-algebra under the obvious realization assignments: $c^B = c^A, f^B = f^A \restriction B$.

## 2.5 Congruences

Consider an equivalence relation $\equiv$ on an $\mathcal{F}$-algebra $A$. If every operation $f^A$, $f \in \mathcal{F}_+$, is *compatible* with $\equiv$, that is, if

$$f^A(a_1, \ldots, a_n) \equiv f^A(a'_1, \ldots, a'_n) \text{ whenever } a_1 \equiv a'_1, \ldots, a_n \equiv a'_n \tag{2.4}$$

then we say then we say that $\equiv$ is a **congruence** on the $\mathcal{F}$-algebra $A$. Condition (2.4) is a generalization of the ancient[2] principle of *substitutivity of equals for equals*. For suppose that the equivalence relation at hand is the identity. Clearly, if $a_i = a'_i, i = 1, \ldots, n$, then $f^A(a_1, \ldots, a_n) = f^A(a'_1, \ldots, a'_n)$ because equals may be substituted for equals in any given context (here the context is provided by the application of $f^A$). The notion of a congruence is obtained by generalizing this idea from the identity relation to an arbitrary equivalence relation $\equiv$ that satisfies (2.4).

Given a congruence relation $\equiv$ on an $\mathcal{F}$-algebra $A$, we can derive a new $\mathcal{F}$-algebra, denoted A/$\equiv$, through the important *quotient construction*. In particular,

- the carrier A/$\equiv$ is the set of all equivalence classes of $\equiv$, i.e., the set of all "blocks" arising from the partitioning of $A$ by $\equiv$;
- for each $c \in \mathcal{F}_0$, the constant $c^{A/\equiv}$ is defined as $[c^A]$, i.e., as the equivalence class of $c^A$; and
- for each $f \in \mathcal{F}_n$ the operation $f^{A/\equiv}$ is defined as

  $$f^{A/\equiv}([a_1], \ldots, [a_n]) = [f^A(a_1, \ldots, a_n)]$$

  for any blocks $[a_1], \ldots, [a_n] \in A/\equiv$. That is, the value of $f^{A/\equiv}$ for $n$ given blocks $[a_1], \ldots, [a_n]$ is the block containing the result $f^A(a_1, \ldots, a_n)$.

As is usual when defining operations on equivalence classes using representatives of the classes, it is not immediately evident that our definition of $f^{A/\equiv}$ is legitimate. Specifically, we have to prove that the value we have specified for $f^{A/\equiv}([a_1], \ldots, [a_n])$, namely $[f^A(a_1, \ldots, a_n)]$, does not in any way

[2] First formulated by Euclid.

depend upon *how we choose the representatives* $a_1, \ldots, a_n$. Formally, we need to show that for any elements $a_i, a'_i$ in the same equivalence class, $i = 1, \ldots, n$, we have

$$f^{A/\equiv}([a_1], \ldots, [a_n]) = f^{A/\equiv}([a'_1], \ldots, [a'_n])$$

i.e. $[f^A(a_1, \ldots, a_n)] = [f^A(a'_1, \ldots, a'_n)]$. Now since the $a_i$ and $a'_i$ are in the same block we have $a_i \equiv a'_i, i = 1, \ldots, n$, and since $\equiv$ is a congruence on $A$ we have $f^A(a_1, \ldots, a_n) \equiv f^A(a'_1, \ldots, a'_n)$, which is to say that $[f^A(a_1, \ldots, a_n)] = [f^A(a'_1, \ldots, a'_n)]$. We have thus shown that A/$\equiv$ is indeed an $\mathcal{F}$-algebra. We will call it the **quotient algebra** of $A$ induced by the congruence $\equiv$, or simply the $\equiv$-quotient algebra of $A$.

## 2.6 Fundamental morphism results

We begin by showing that the composition of two $\mathcal{F}$-morphisms is again an $\mathcal{F}$-morphism:

**Lemma 2.6.1** *If $A, B, C$ are $\mathcal{F}$-algebras and $h_1 : A \to B, h_2 : B \to C$ are morphisms then the composition $h_2 \circ h_1 : B \to C$ is a morphism.*

**Proof**. For $c \in \mathcal{F}_0, h_2 \circ h_1(c^A) = h_2(h_1(c^A)) = h_2(c^B) = c^C$. For $f \in \mathcal{F}_n, n \in N_+$, and $a_1, \ldots, a_n \in A$,

$$\begin{aligned} h_2 \circ h_1(f^A(a_1, \ldots, a_n)) &= h_2(h_1(f^A(a_1, \ldots, a_n))) \\ &= h_2(f^B(h_1(a_1), \ldots, h(a_n))) \\ &= f^C(h_2(h_1(a_1)), \ldots, h_2(h_1(a_n))) \\ &= f^C(h_2 \circ h_1(a_1), \ldots, h_2 \circ h_1(a_n)) \end{aligned}$$

which proves that $h_2 \circ h_1$ is a morphism. ■

The next result shows that homomorphisms are completely determined by their restrictions to generating sets, an observation that will be put to good use later on.

**Proposition 2.6.1** *If $A$ is an $\mathcal{F}$-algebra generated by a subset $X \subseteq A$ and $h_1, h_2$ are morphisms from $A$ to an $\mathcal{F}$-algebra $B$ such that $h_1 \restriction X = h_2 \restriction X$, then $h_1 = h_2$.*

**Proof**: We must show that $h_1(a) = h_2(a)$ for all $a \in A = [X]$. Since $[X] = Con^\infty(X) = Con^0(X) \cup Con^1(X) \cup \cdots$ (Th. 2.2.1) it suffices to show that for every $i \in N$ we have $h_1(a) = h_2(a)$ for all $a \in Con^i(X)$. We use induction on $i$. For $i = 0$ the equality holds by supposition. For the inductive step, assume the claim for some $i \in N$ and pick any $a$ in

$$Con^{i+1}(X) = Con^i(X) \cup \{c^A \mid c \in \mathcal{F}_0\} \cup \{f^A(a_1, \ldots, a_n) \mid a_1, \ldots, a_n \in Con^i(X), f \in \mathcal{F}_n\}.$$

If $a \in Con^i(X)$ the claim holds by virtue of the inductive hypothesis. If $a = c^A$ then it holds by the definition of a morphism—both $h_1$ and $h_2$ must map $c^A$ to its unique mirror element $c^B \in B$. Finally, if $a = f^A(a_1, \ldots, a_n)$ for some $a_1, \ldots, a_n \in Con^i(X)$ then

$$\begin{aligned} h_1(f^A(a_1,\ldots,a_n)) &= f^B(h_1(a_1),\ldots,h_1(a_n)) && \text{(morphism property)} \\ &= f^B(h_2(a_1),\ldots,h_2(a_n)) && \text{(inductive hypothesis)} \\ &= h_2(f^A(a_1,\ldots,a_n)) && \text{(morphism property).} \blacksquare \end{aligned}$$

We will shortly see how the principle of *structural induction* can be used to express such proofs in a more succinct and elegant manner.

Next we show that morphisms preserve subalgebras:

**Proposition 2.6.2** *For any $\mathcal{F}$-algebras $A, B$ and morphism $h : A \to B$, if $A' \leq A$ then $h(A') \leq B$. Conversely, if $B' \leq B$ then $h^{-1}(B') \leq A$.*

**Proof**. In one direction, suppose $A' \leq A$. We must show that $h(A')$ (i) contains every constant $c^B$, $c \in \mathcal{F}_0$, and that (ii) it is closed under every operation $f^B$, $f \in \mathcal{F}_+$. Because $A' \leq A$, $A'$ contains $c^A$, hence (i) follows because $h$ must map $c^A$ to $c^B$. For (ii), consider any $b_1,\ldots,b_n \in h(A')$, so that $b_i = h(a_i)$ for some $a_i \in A', i = 1,\ldots,n$. Because $A'$ is closed under $f^A$, we have $f^A(a_1,\ldots,a_n) \in A'$, hence $h(f^A(a_1,\ldots,a_n)) \in h(A')$. But

$$h(f^A(a_1,\ldots,a_n)) = f^B(h(a_1),\ldots,h(a_n)) = f^B(b_1,\ldots,b_n)$$

thus $f^B(b_1,\ldots,b_n) \in h(A')$, proving that $h(A')$ is closed under $f^B$, for any $f \in \mathcal{F}_n$. A similar argument establishes the converse proposition. ■

For any congruence $\equiv$ on a $\mathcal{F}$-algebra $A$ we define a function $\mathrm{nat}_\equiv : A \to A/\equiv$ as $\mathrm{nat}_\equiv(a) = [a]$. Thus each element of $A$ gets mapped to its (unique) equivalence class under $\equiv$. We call $\mathrm{nat}_\equiv$ the **natural map** of $\equiv$. When $\equiv$ is understood we may drop the subscript and simply write nat. An interesting observation is that $\mathrm{nat}_\equiv$ is a morphism from $A$ to the quotient $\mathcal{F}$-algebra $\mathrm{A}/\equiv$. It is in fact an epimorphism, since its image includes every block in $\mathrm{A}/\equiv$.

**Proposition 2.6.3** *Let $\equiv$ be a conguence on an $\mathcal{F}$-algebra $A$. The map $nat_\equiv : A \to A/\equiv$ is an epimorphism.*

**Proof**. For $c \in \mathcal{F}_0$ we have $\mathrm{nat}_\equiv(c^A) = [c^A] = c^{A/\equiv}$. For $f \in \mathcal{F}_n, n \in N_+$, and $a_1,\ldots,a_n \in A$, we have

$$\begin{aligned} \mathrm{nat}_\equiv(f^A(a_1,\ldots,a_n)) &= [f^A(a_1,\ldots,a_n)] \\ &= f^{A/\equiv}([a_1],\ldots,[a_n]) \\ &= f^{A/\equiv}(\mathrm{nat}_\equiv(a_1),\ldots,\mathrm{nat}_\equiv(a_n)) \end{aligned}$$

and since nat is surjective, it is an epimorphism. ■

Recall that for any function $f : A \to B$, the *kernel relation* of $f$ is the relation $\equiv_f \subseteq A^2$ which obtains between two elements of $A$ iff $f$ maps both of them to the same value in $B$. We already know that $\equiv_f$ is an equivalence relation on $A$ (Lem. **??**). The next result shows that the kernel relation of a morphism is in fact a congruence on the morphism's domain.

**Proposition 2.6.4** *Let $h : A \to B$ be a morphism, for any $\mathcal{F}$-algebras $A$ and $B$. Then $\equiv_h$ is a congruence on $A$.*

**Proof**: Suppose that $a_1 \equiv_h a'_1, \ldots, a_n \equiv_h a'_n$ and $f \in \mathcal{F}_n$. Then

$$\begin{aligned} h(f^A(a_1, \ldots, a_n)) &= f^B(h(a_1), \ldots, h(a_n)) && \text{(morphism property)} \\ &= f^B(h(a'_1), \ldots, h(a'_n)) && \text{(since } a_i \equiv_h a'_i) \\ &= h(f^A(a'_1), \ldots, f^A(a'_n)) && \text{(morphism property)} \end{aligned}$$

which is to say that $f^A(a_1, \ldots, a_n) \equiv_h f^A(a'_1, \ldots, a'_n)$. ■

The next result might well be *the* fundamental theorem of universal algebra. It says that for any epimorphism $h : A \to A'$, $A'$ is isomorphic (and hence for all algebraic purposes identical) to $A/\equiv_h$:

**Theorem 2.6.1 (Fundamental Homomorphism Theorem)** *If $A, B$ are $\mathcal{F}$-algebras and $h : A \to B$ is an epimorphism then $A/\equiv_h \cong B$.*

**Proof**: Define $g : A/\equiv_h \to B$ as $g([a]) = h(a)$. We note that

- $g$ is a well-defined function because $[a]$ only contains elements that get mapped by $h$ to the same value in $B$, and hence $g([a])$ does not depend on the choice of $a$;
- $g$ is surjective because $h$ is surjective; and
- $g$ is injective because for any two *distinct* blocks $[a_1]$ and $[a_2]$, we have $h(a_1) \neq h(a_2)$ and hence $g([a_1]) \neq g([a_2])$.

Now for $c \in \mathcal{F}_0$ we have $g([c^A]) = h(c^A) = c^B$, while for $f \in \mathcal{F}_n$, $n \in N_+$, and $a_1, \ldots, a_n \in A$, we have

$$\begin{aligned} g(f^{A/\equiv_h}([a_1], \ldots, [a_n])) &= g([f^A(a_1, \ldots, a_n)]) \\ &= h(f^A(a_1, \ldots, a_n)) \\ &= f^B(h(a_1), \ldots, h(a_n)) \\ &= f^B(g([a_1]), \ldots, g([a_n])). \end{aligned}$$

Therefore, $g$ is an isomorphism, and since it is also a bijection, the two algebras are isomorphic. ■

## 2.7 Free algebras

Let $A$ be an $\mathcal{F}$-algebra (for some signature $\mathcal{F}$) and let $\mathbf{K}$ be a class of $\mathcal{F}$-algebras, possibly including $A$. Further, let $X$ be any given set and let $\gamma : X \to A$ be any mapping from $X$ to $A$. We say that the algebra $A$ is **free for K relative to** $\gamma$ iff for every algebra $B \in \mathbf{K}$ and every mapping $h : X \to B$ there exists a unique $\mathcal{F}$-homomorphism $\hat{h} : A \to B$ such that

$$(\forall x \in X)\, h(x) = \hat{h}(\gamma(x)) \tag{2.5}$$

i.e., such that the following diagram commutes:

$$\begin{tikzcd} X \arrow[rr, "h"] \arrow[dr, "\gamma"'] & & B \\ & A \arrow[ur, "\hat{h}"'] & \end{tikzcd}$$

If $A$ is in fact a member of $\mathbf{K}$, we say that $A$ is free **in K** relative to $\gamma$.

We will assume hereafter that $\mathbf{K}$ is non-trivial in the sense that it contains at least one algebra with more than one element. Under this assumption we can infer that the mapping $\gamma : X \to A$ must be injective:

**Lemma 2.7.1** *If $A$ is free for $\mathbf{K}$ relative to a mapping $\gamma : X \to A$ then $\gamma$ is injective.*

**Proof**. Let $x_1, x_2$ be any two distinct elements of $X$. Choose an algebra $B \in \mathbf{K}$ that contains at least two elements $b_1, b_2$, and define $h : X \to B$ so that $h(x_1) = b_1, h(x_2) = b_2$. Then $h(x_1) \neq h(x_2)$, hence $\hat{h}(\gamma(x_1)) \neq \hat{h}(\gamma(x_2))$, hence $\gamma(x_1) \neq \gamma(x_2)$. ■

**Lemma 2.7.2** *Let $A$ be free for $\mathbf{K}$ relative to $\gamma : X \to A$ and suppose that $\delta : Y \to X$ is a bijection. Then $A$ is free for $\mathbf{K}$ relative to $\gamma \circ \delta : Y \to A$.*

**Proof**: Pick any algebra $B \in \mathbf{K}$ and mapping $h : Y \to B$. Define $g : X \to B$ as $g = h \circ \delta^{-1}$. Since $A$ is free for $\mathbf{K}$ relative to $\gamma$, there is a unique morphism $\widehat{g} : A \to B$ such that $\widehat{g}(\gamma(x)) = g(x)$ for all $x \in X$. Therefore, for any $y \in Y$,

$$\widehat{g}(\gamma(\delta(y))) = g(\delta(y)) = h(\delta^{-1}(\delta(y))) = h(y).$$

Now suppose there is another morphism $g_1 : A \to B$ such that $g_1(\gamma(\delta(y))) = h(y)$ for all $y \in Y$. Then for any $x \in X$ we would have

$$g_1(\gamma(x)) = g_1(\gamma(\delta(\delta^{-1}(x)))) = h(\delta^{-1}(x)) = g(x)$$

contradicting the uniqueness of $\widehat{g}$ in this respect. Therefore, $\widehat{g}$ is a unique morphism such $\widehat{g}(\gamma(\delta(y))) = h(y)$ for all $y \in Y$, which proves that $A$ is free for $\mathbf{K}$ relative to $\gamma \circ \delta$. ■

Next we show that any two algebras that are free *in* $\mathbf{K}$ are essentially the same:

**Theorem 2.7.1** *Let $A_1$ and $A_2$ be two algebras that are free in* $\mathbf{K}$ *relative to* $\gamma_1 : X \to A_1$ *and* $\gamma_2 : X \to A_2$*, respectively. Then* $A_1 \cong A_2$.

**Proof**: Since $A_1$ is free for $\mathbf{K}$ relative to $\gamma_1$, $A_2 \in \mathbf{K}$, and $\gamma_2$ is a mapping from $X$ to $A_2$, there is a unique morphism $\hat{\gamma_2} : A_1 \to A_2$ such that $\gamma_2(x) = \hat{\gamma_2}(\gamma_1(x))$ for all $x \in X$. By the same token, since $A_2$ is free for $\mathbf{K}$ relative to $\gamma_2$, $A_1 \in \mathbf{K}$, and $\gamma_1$ is a mapping from $X$ to $A_1$, there is a unique morphism $\hat{\gamma_1} : A_2 \to A_1$ such that $\gamma_1(x) = \hat{\gamma_1}(\gamma_2(x))$ for all $x \in X$. Therefore, for all $x \in X$,

$$\gamma_2(x) = \hat{\gamma_2}(\hat{\gamma_1}(\gamma_2(x))) \text{ (i) and } \gamma_1(x) = \hat{\gamma_1}(\hat{\gamma_2}(\gamma_1(x))) \text{ (ii).} \tag{2.6}$$

Now because $A_2$ is free for $\mathbf{K}$ relative to $\gamma_2$, $A_2 \in \mathbf{K}$, and $\gamma_2$ is a mapping from $X$ to $A_2$, it follows that there is a unique morphism $g : A_2 \to A_2$ such that for all $x \in X$

$$\gamma_2(x) = g(\gamma_2(x)) \tag{2.7}$$

Since the composition of two morphisms is again a morphism (Lem. **??**), we conclude that $\hat{\gamma_2} \cdot \hat{\gamma_1} : A_2 \to A_2$ is a morphism. Moreover, (2.6i) tells us that $\hat{\gamma_2} \cdot \hat{\gamma_1}$ satisfies (2.7). Hence we conclude that the unique morphism $g$ of (2.7) is none other than $\hat{\gamma_2} \cdot \hat{\gamma_1}$. However, the identity function $\mathrm{id} : A_2 \to A_2$ is also a morphism and it also satisfies (2.7) for all $x \in X$. Thus we must also conclude that $g = \mathrm{id}_{A_2}$, and now by the uniqueness of $g$ it follows that $\hat{\gamma_2} \cdot \hat{\gamma_1} = \mathrm{id}_{A_2}$, i.e. $\hat{\gamma_2} \cdot \hat{\gamma_1}(a) = a$ for all $a \in A_2$.

A perfectly symmetrical argument will show that $\hat{\gamma_1} \cdot \hat{\gamma_2}$ is the identity function on $A_1$, from which we may infer that $\hat{\gamma_1}$ (and $\hat{\gamma_2}$) is bijective, and hence that $A_1$ and $A_2$ are isomorphic. ■

**Corollary 2.7.1** *Suppose that $A_1$ and $A_2$ are free in* $\mathbf{K}$ *relative to* $\gamma_1 : X_1 \to A_1$ *and* $\gamma_2 : X_2 \to A_2$*, respectively. If $X_1$ and $X_2$ are of the same cardinality then* $A_1 \cong A_2$.

**Proof**. If $X_1$ and $X_2$ have the same cardinality then there is a bijection $\delta : X_1 \to X_2$. Therefore, by Lem. 2.7.2, $A_2$ is free in $\mathbf{K}$ relative to $\gamma_2 \circ \delta : X_1 \to A_2$. Hence, by the preceding theorem, $A_1 \cong A_2$. ■

The special case that arises when $X$ is a subset of $A$ will be of particular interest. In that case the mapping $\gamma$ is the identity function and equation (2.5) becomes $h(x) = \hat{h}(x)$, thus the freedom condition is reduced to the existence of a unique morphism $\hat{h} : A \to B$ that agrees with $h$ on $X \subseteq A$, i.e., a unique morphism $\hat{h} : A \to B$ that extends $h$. When $X \subseteq A$ we say that $A$ is free for $\mathbf{K}$ **on** (or **over**) $X$. Observe now that even when $X$ is not a subset of $A$ *per se*, Lem. 2.7.1 allows us to think of it as such by virtue of the injective mapping $\gamma : X \to A$—we simply identify $X$ with $\{\gamma(x) \mid x \in X\} \subseteq A$. For this reason it suffices to focus our attention on algebras that are free for $\mathbf{K}$ over some subsets of their carriers, and from now on we will do so. If, in addition, $X$ happens to be a generating set for $A$, we say that the latter is **freely generated** by $X$. We then call $X$ a **generating system** or a **basis** for $A$.

The existence of free algebras for an arbitrary class of $\mathcal{F}$-algebras $\mathbf{K}$ will be established in Sec. **??**. The existence of free algebras *in* $\mathbf{K}$, a much more difficult issue, will be taken up in Sec. **??**.

## 2.8 Algebraic closure systems and recursive set definitions

Very often a recursive definition of a set $S$ is presented as follows:

- One or more *initial clauses* are given, each stating, in a plain, non-recursive manner, that certain objects are elements of $S$. These are often called *initial* or *basis elements*.
- One or more *recursive clauses* are given, each of the form "If $x_1, \ldots x_n$ are elements of $S$ then so is $\phi(x_1, \ldots, x_n)$", where $x_1, \ldots, x_n$ are variables denoting arbitrary objects and $\phi(x_1, \ldots, x_n)$ is an object that might depend on $x_1, \ldots, x_n$.[3] The hypothesis "$x_1, \ldots, x_n$ are elements of $S$" is the *antecedent* of the clause, and the conclusion "$\phi(x_1, \ldots, x_n) \in S$" is the *consequent*.

Definitions of this form are usually called *inductive*. Oftentimes an additional clause is appended at the end stating that nothing else is an element of $S$, but this may be tacitly assumed as a convention.

**Example 2.8.1** The following is an inductive definition of the set of even natural numbers:

- 0 is an even number.
- If $n$ is an even number then so is $n + 2$.

Notice that this is essentially an English transcription of the following logic program:

```
even(0).
even(n+2) :- even(n).
```

We will see that algebraic closure systems provide a natural semantics for Horn clauses. ■

**Example 2.8.2** Consider the following definition of a simple language (set of strings) called **GA**, for "ground arithmetic":

- The symbol `0` is an element of **GA**.
- if $E$ is an element of **GA**, then so is `s(`$E$`)`.
- If $E_1$ and $E_2$ are elements of **GA**, then so is `(`$E_1$ `+` $E_2$`)`.
- If $E_1$ and $E_2$ are elements of **GA**, then so is `(`$E_1$ `*` $E_2$`)`.

It should be clear that **GA** is a context-free language. A perfectly equivalent way to define it is through a BNF grammar:

$$E ::= \texttt{0} \mid \texttt{s(}E\texttt{)} \mid \texttt{(}E \texttt{ + } E\texttt{)} \mid \texttt{(}E \texttt{ * } E\texttt{)}$$

which is, in fact, how we defined **GA** in Ex. 2.3.6. We will see that there is a close connection between context-free grammars, Horn clauses, and algebraic closure systems, and in particular between unambiguous grammars and freely generated algebras. ■

Two or more recursive clauses with the same antecedent are usually collapsed, for brevity, into one clause with a single antecedent and multiple conclusions. E.g. in the above definition the two clauses for `+` and `*` could be expressed as "If $E_1, E_2$ are elements of **GA** then so are `(`$E_1$ `+` $E_2$`)` and `(`$E_1$ `*` $E_2$`)`."

[3] Generally speaking recursive clauses can be more complex, but we may overlook this for our present purposes.

**Example 2.8.3** The formulas of propositional logic, called *propositional sentences*, are defined as follows:

- Every propositional atom $p_i, i \in N$, is a propositional sentence.
- If $A$ is a propositional sentence then so is $\neg A$.
- If $A_1, A_2$ are propositional sentences then so are $(A_1 \wedge A_2), (A_1 \vee A_2), (A_1 \Rightarrow A_2)$, and $(A_1 \Leftrightarrow A_2)$.

We mention in passing that this definition, too, can be given in the form of a BNF grammar if we are willing to view an atom $p_n$ as an abbreviation for the letter $p$ followed by $n$ occurences of some special "marker" symbol, such as the dollar sign \$:

$$\begin{array}{lcl} A & ::= & p \mid A\$ \\ S & ::= & A \mid \neg S \mid (S \wedge S) \mid (S \vee S) \mid (S \Rightarrow S) \mid (S \Leftrightarrow S) \quad \blacksquare \end{array}$$

Definitions such as these are sufficiently clear for purposes of exposition but lack a mathematical semantics, so a formal explication is necessary if we want to be sure that we understand exactly what—if anything—is being defined. One avenue for such an analysis is to recast the definition in an algebraic setting, as discussed below. Another alternative, which we will explore in the next section, is afforded by abstract deduction systems, although the two approaches—algebraic and deductive—are essentially equivalent.

The main observation is simple: the form of a recursive clause "If $x_1, \ldots, x_n$ are elements of $S$ then so is $\phi(x_1, \ldots, x_n)$" suggests that $\phi(x_1, \ldots, x_n)$ is a *function* of $x_1, \ldots, x_n$: whenever particular values for $x_1, \ldots, x_n$ are given as "input", $\phi(x_1, \ldots, x_n)$ assumes a particular value as "output". Accordingly, the idea is to consider an algebra whose carrier is the universe from which the elements of $S$ are drawn, and such that for each recursive clause of the above form there is an $n$-ary operation $\phi$ that can be applied to any given objects $a_1, \ldots, a_n$ in the universe. Such operations are usually called *constructors* to emphasize that they serve as building tools. With this machinery in place, $S$ can be formally defined as $[X]$, where $X$ is the set containing the initial elements. That is, $S$ is defined as the smallest subset of the universe that contains the basis elements and is closed under the constructors. Alternatively, we can consider an algebra in which the initial elements are taken as constants, and then $S$ can be defined as $[\emptyset]$. These two approaches are equivalent, but, for consistency, a fixed choice must be adopted through any given stretch of discourse. A few examples will help to clarify the idea.

**Example 2.8.4** For the explication of the recursive definition of the even numbers, in Ex. 2.8.1, we introduce the algebra $(N; \lambda n.n + 2)$. Then the defined set—the even numbers—is taken to be $[0]$, the smallest subset of $N$ that contains 0 and is closed under the constructor $\lambda n.n + 2$. Of course this is the top-down description of $[0]$. Because this is an algebraic closure system, we know that

$$[0] = \bigcup_{i \in N} Con^i(\{0\})$$

where, for any $Y \subseteq N$, $Con(Y) = Y \cup \{n + 2 \mid n \in Y\}$. Thus $[0]$ can also be characterized in a bottom-up way as

$$\{0\} \cup Con(\{0\}) \cup Con(Con(\{0\})) \cup \cdots$$

i.e. as

$$\{0\} \cup \{0,2\} \cup \{0,2,4\} \cup \cdots$$

which better captures the constructive spirit of the definition. ■

**Example 2.8.5** For the recursive definition of the language **GA**, in Ex. 2.8.2, we define an alphabet $\Sigma_{\mathbf{GA}} = \{\mathtt{0}, \mathtt{)}, \mathtt{(}, \mathtt{s}, \mathtt{+}, \mathtt{*}\}$ and an algebra $(\Sigma^*_{\mathbf{GA}}; \phi_{\mathtt{s}}, \phi_{\mathtt{+}}, \phi_{\mathtt{*}})$, where $\phi_{\mathtt{s}}$ is a unary constructor and $\phi_{\mathtt{+}}, \phi_{\mathtt{*}}$ are binary. These are defined, for any $u, v \in \Sigma^*_{\mathbf{GA}}$, as:

$$\begin{aligned} \phi_{\mathtt{s}}(u) &= \mathtt{s}(u) \\ \phi_{\mathtt{+}}(u,v) &= \mathtt{(}u \ \mathtt{+}\ v\mathtt{)} \\ \phi_{\mathtt{*}}(u,v) &= \mathtt{(}u \ \mathtt{*}\ v\mathtt{)}. \end{aligned}$$

Thus, for instance, $\phi_{\mathtt{+}}(\mathtt{ss*0}) = \mathtt{(ss*0 \ + \ ss*0)}$. Then **GA** is identified with $[\mathtt{0}]$, the smallest subset of the universe $\Sigma^*_{\mathbf{GA}}$ that contains $\mathtt{0}$ and is closed under $\phi_{\mathtt{s}}, \phi_{\mathtt{+}}$, and $\phi_{\mathtt{*}}$. Here we have

$$Con(Y) = Y \cup \{\mathtt{s}(u) \mid u \in Y\} \cup \{\mathtt{(}u \ \mathtt{+}\ v\mathtt{)} \mid u, v \in Y\} \cup \{\mathtt{(}u \ \mathtt{*}\ v\mathtt{)} \mid u, v \in Y\}$$

for any $Y \subseteq \Sigma^*_{\mathbf{GA}}$, so the bottom-up description of **GA** is

$$\{\mathtt{0}\} \cup \{\mathtt{0}, \mathtt{s(0)}, \mathtt{(0+0)}, \mathtt{(0*0)}\} \cup \{\mathtt{0}, \mathtt{s(0)}, \mathtt{(0+0)}, \mathtt{(0*0)}, \ldots, \mathtt{(s(0)+(0*0))}, \ldots\} \cup \cdots.$$

Note that $Con^i(Y)$ contains exactly those strings in **GA** whose parse trees (with respect to the grammar in Ex. 2.8.2) are of height $\leq i+1$. ■

**Example 2.8.6** An alternative explication of the same definition might take $\mathtt{0}$ as a constant and identify **GA** with $[\emptyset]$. In particular, we can consider the algebra $(\Sigma^*_{\mathbf{GA}}; \mathtt{0}; \phi_{\mathtt{s}}, \phi_{\mathtt{+}}, \phi_{\mathtt{*}})$ of Ex. 2.3.6, where $\phi_{\mathtt{s}}$, $\phi_{\mathtt{+}}$, and $\phi_{\mathtt{*}}$ are as in the preceding example. We can then take **GA** to be $[\emptyset]$, which is again the smallest subset of $\Sigma^*_{\mathbf{GA}}$ that contains the constant $\mathtt{0}$ and is closed under $\phi_{\mathtt{s}}$, $\phi_{\mathtt{+}}$, and $\phi_{\mathtt{*}}$. ■

The inductive definition of propositional sentences in Ex. 2.8.3 can be treated similarly.

The precision we gain by interpreting inductive definitions algebraically is welcome, but one might argue that it does not quite justify the associated effort. After all, an inductive definition such as that of the even numbers in Ex. 2.8.1 does a fine job by itself—we understand what it says well enough without having to bring in the extra algebraic machinery. Algebra pays dividends when we come to *structural induction* and *structural recursion*. We examine these in the next two subsections.

## Structural induction

The general induction principle for closure systems that we stated as Th. **??** assumes a special form when the closure system is algebraic—it goes by the name of "structural induction".

**Theorem 2.8.1 (Structural induction principle, algebraic version)** *Consider an algebra $(A, C, \Phi)$ and let $X \subseteq A$. To prove that a property $P$ holds for every element of $[X]$ it suffices to prove that*

(a) *$P$ holds for every $x \in X$;*

(b) *$P$ holds for every constant in $C$; and*

(c) *for all $\phi \in \Phi_n, n \in N_+$, if $P$ holds for any $a_1, \ldots, a_n \in [X]$ then it also holds for $\phi(a_1, \ldots, a_n)$.*

*More formally, for any $Y \subseteq [X]$, if $Y \supseteq X$ and $Y \leq A$ then $Y = [X]$.*

**Proof**. This follows directly from Th. **??**. We repeat the argument in the present context: if $Y \supseteq X$ and $Y \leq A$ then $Y$ is a subalgebra of $A$ that includes $X$, and since $[X]$ is the least such subalgebra, $[X] \subseteq Y$. But we also have $Y \subseteq [X]$, hence $Y = [X]$. ■

Proving parts (a) and (b) might be referred to as *the basis step*; establishing (c) is the *inductive step*. The antecedent of (c) is called "the inductive hypothesis".

This version is more usable than the general principle because we are dealing with a specific kind of closure system in which the definition of what constitutes a closed set is simple: a set is closed if applying any operation to any of its elements does not take us outside the set. For this reason the inductive step is more amenable to proof than in other types of closure systems, especially when the algebra contains only a small number of operations. The principle goes hand-in-hand with our interpretation of recursively defined sets as subalgebras generated by sets of initial elements.

**Example 2.8.7** Consider the algebra $(\Sigma^*_{\mathbf{GA}};\ \phi_{\mathtt{s}},\ \phi_{\mathtt{+}},\ \phi_{\mathtt{*}})$ of Ex. 2.8.5, and recall that the language **GA** is $[\mathtt{0}]$, the smallest subset of $\Sigma^*_{\mathbf{GA}}$ that contains $\mathtt{0}$ and is closed under the constructors $\phi_{\mathtt{s}}, \phi_{\mathtt{+}}$, and $\phi_{\mathtt{*}}$. Accordingly, whenever we want to prove that a certain property $P$ holds for every expression of **GA** we may be able to use the principle of structural induction. Here we will illustrate such a use by proving that every string in **GA** has an equal number of left and right parentheses. In symbols, if we write $\mathrm{LP}[u]$ and $\mathrm{RP}[u]$ to denote the number of left and right parentheses of a string $u$, respectively, we want to show that $\mathrm{LP}[u] = \mathrm{RP}[u]$ for every $u \in [\mathtt{0}]$. More formally, let $EP = \{u \in [\mathtt{0}] \mid \mathrm{LP}[u] = \mathrm{RP}[u]\}$. We want to prove that $EP = [\mathtt{0}]$.

The basis step is easy, as $\mathrm{LP}[\mathtt{0}] = \mathrm{RP}[\mathtt{0}] = 0$. Thus $EP \supseteq \{\mathtt{0}\}$. In this case there are no constants to consider so we continue with the inductive step, for which we need to show that $EP$ is closed under the three constructors $\phi_{\mathtt{s}}, \phi_{\mathtt{+}}$, and $\phi_{\mathtt{*}}$:

- Suppose $u \in EP$. Then $\phi_{\mathtt{s}}(u) = \mathtt{s}(u)$ is also an element of $EP$ because

$$\begin{aligned}
\mathrm{LP}[\mathtt{s}(u)] &= \mathrm{LP}[u] + 1 \\
&= \mathrm{RP}[u] + 1 \quad \text{(since } \mathrm{LP}[u] = \mathrm{RP}[u] \text{ by the inductive hypothesis)} \\
&= \mathrm{RP}[\mathtt{s}(u)].
\end{aligned}$$

  Thus $EP$ is closed under $\phi_{\mathtt{s}}$.

- Suppose $u, v \in EP$. Then, for $\circ \in \{\mathtt{+}, \mathtt{*}\}$, the string $\phi_\circ(u, v) = (u \circ v)$ is also in $EP$ as

$$\begin{aligned}
\mathrm{LP}[(u \ \circ \ v)] &= \mathrm{LP}[u] + \mathrm{LP}[\mathrm{v}] + 1 \\
&= \mathrm{RP}[u] + \mathrm{RP}[v] + 1 \quad \text{(by the inductive hypothesis)}
\end{aligned}$$

$$= \quad \mathrm{RP}[(u \circ v)].$$

Thus it follows from the principle of structural induction that $EP = [\mathtt{0}] = \mathbf{GA}$. That is, every string in $\mathbf{GA}$ has an equal number of left and right parentheses. ■

## Unique generation and structural recursion

Let $(A, C, \Phi)$ be an algebra, $X \subseteq A$. We say that the algebra $[X]$ is **uniquely generated** (or *uniquely constructed*) by $X$ in $(A, C, \Phi)$ (or simply "uniquely generated by $X$" when the reference to $(A, C, \Phi)$ is not necessary) iff for every $a \in [X]$ *exactly one* of the following three conditions obtains:

1. $a \in X$; or
2. $a \in C$; or else
3. there is a unique operation $\phi \in \Phi_n$ and a unique tuple of elements $(a_1, \ldots, a_n) \in [X]^n$ such that $a = \phi(a_1, \ldots, a_n)$.

An alternative characterization of unique generation is given in the exercises.

As a slight terminological abuse, the attribute "uniquely generated" will be ascribed both to the algebra $[X]$ and to the individual elements of $[X]$. Sometimes the latter are also said to be *uniquely readable*, and $[X]$ is said to satisfy "the unique readability condition".

**Example 2.8.8** Let us show that the algebra $\mathbf{GA} = [\mathtt{0}]$ of Ex. 2.8.5 is uniquely generated by $\mathtt{0}$. We must show that for every $u \in [\mathtt{0}]$ *exactly one* of the following is true:

1. $u = \mathtt{0}$; or
2. $u = \phi_{\mathtt{s}}(w) = \mathtt{s}(w)$ for a unique $w$; or
3. $u = \phi_{\circ}(w_1, w_2) = (w_1 \circ w_2)$ for a unique $\circ \in \{\mathtt{+}, \mathtt{*}\}$ and a unique pair of strings $(w_1, w_2) \in \mathbf{GA}^2$.

Now $u$ is either (a) $\mathtt{0}$, or (b) of the form $\mathtt{s}(w)$, or (c) of the form $(w_1 \circ w_2)$ for $\circ \in \{\mathtt{+}, \mathtt{*}\}$. If (a) is the case, then only (1) holds. If (b) is the case, then it is clear that only (2) holds. Finally, if (c) is the case, suppose we have $(w_1 \circ w_2) = (w_1' \circ' w_2')$. We will show that $w_1 = w_1', \circ = \circ', w_2 = w_2'$. Now either $w_1 = \mathtt{0}$ or not. If $w_1 = \mathtt{0}$ then we must also have $w_1' = \mathtt{0}$ (any other possibility for $w_1'$ would invalidate the equation $(w_1 \circ w_2) = (w_1' \circ' w_2)$), hence $\circ = \circ'$ and $w_2 = w_2'$ as well. On the other hand, if $w_1 \neq \mathtt{0}$ then $|w_1| > 1$, for $\mathtt{0}$ is the only string in $\mathbf{GA}$ of length 1. Trivially, either $w_1 \sqsubset w_1'$ or $w_1' \sqsubset w_1$, or $w_1 = w_1'$. Neither of the first two could hold because the condition $|w_1| > 1$ would then entail that either $w_1$ or $w_1'$ has more left than right parentheses (see Exer. **??**), which we proved impossible in Ex. 2.8.7. Thus $w_1 = w_1'$, and consequently, $\circ = \circ'$ and $w_2 = w_2'$ as well.

Note that this argument also proves that $\mathbf{GA}$ is uniquely generated by $\emptyset$ in the algebra of Ex. 2.8.6. ■

The importance of unique generation derives from the following result:

**Theorem 2.8.2** *Suppose that an algebra $A$ is uniquely generated by a subset $X \subseteq A$. Then for any translation $T$ from $A$ to an algebra $B$ and any function $h : X \to B$ there is a unique function $\widehat{h} : A \to B$ that extends $h$ and is a homomorphism with respect to $T$.*

This theorem is equivalent to Th. 2.8.3, whice we state and prove below. Once either of these two theorems has been established, the other can be easily derived. We chose to prove Th. 2.8.3 directly because the notation is more manageable in the setting of $\mathcal{F}$-algebras. We now show how to derive Th. 2.8.2 from Th. 2.8.3.

**Proof of Th. 2.8.2**: Let $\mathcal{A} = (A, C, \Phi)$ be the given algebra that is uniquely generated by $X \subseteq A$. Consider the signature $\mathcal{F} = C \cup \Phi$, where $\mathcal{F}_0 = C$ and $\mathcal{F} = \Phi$, with $\mathcal{F}_n = \Phi_n$ for all $n \in N_+$. Now $A$ is an $\mathcal{F}$-algebra with the identity functions on $\mathcal{F}_0$ and $\mathcal{F}_+$ as the realization assignments. Because these identity functions are bijections, it follows from Prop. 2.8.1 that $A$ is uniquely $\mathcal{F}$-generated by $X$. Let $T = (\tau_1, \tau_2)$ be a translation from $\mathcal{A}$ to an algebra $\mathcal{B} = (B, C', \Phi')$. Make $\mathcal{B}$ into an $\mathcal{F}$-algebra by using $\tau_1 : C = \mathcal{F}_0 \to C'$ and $\tau_2 : \Phi = \mathcal{F}_+ \to \Phi'$ as the realization assignments. This is possible because, be definition, $\tau_1$ and $\tau_2$ are surjective. Consider any function $h : X \to B$. By Th. 2.8.3, there is a unique $\mathcal{F}$-morphism $\widehat{h} : A \to B$ that extends $h$. But, by construction, an $\mathcal{F}$-morphism is precisely a morphism with respect to the translation $T$. ■

Incidentally, the other direction, deriving Th. 2.8.3 from Th. 2.8.2, is even easier. Because we are assuming that $A$ is normal, we can always get an $\mathcal{F}$-translation from $A$ to $B$ by mapping $c^A$ to $c^B$ and $f^A$ to $f^B$, for all $c \in \mathcal{F}_0, \mathcal{F}_+$. The existence of a unique morphism with respect to this translation that extends $h$ is immediately given by Th. 2.8.2. But a morphism with respect to this translation is, by definition, an $\mathcal{F}$-morphism, just as Th. 2.8.3 requires.

**Example 2.8.9** As we saw in Ex. 2.8.8, the algebra $(\mathbf{GA}; \mathtt{0}; \phi_{\mathtt{s}}, \phi_{\mathtt{+}}, \phi_{*})$ is uniquely generated by $\emptyset$. Now consider the algebra $\mathcal{PA} = (N; 0; \prime, +, \cdot)$ of Ex. 2.1.1 along with the translation $\{\mathtt{0} \mapsto 0, \phi_{\mathtt{s}} \mapsto \prime, \phi_{\mathtt{+}} \mapsto +, \phi_{*} \mapsto \cdot\}$. Th. 2.8.2 tells us that for any mapping $h : \emptyset \to N$—and in this case there is only one such mapping, namely $\emptyset$—there is a unique function $\widehat{h} : \mathbf{GA} \to N$ that extends $\emptyset$ (which is trivial in this case) and is a morphism with respect to this translation, so that

$$\begin{aligned}
\widehat{h}(\mathtt{0}) &= 0 \\
\widehat{h}(\mathtt{s}(E)) &= \widehat{h}(E) + 1 \\
\widehat{h}((E_1 \ \mathtt{+} \ E_2)) &= \widehat{h}(E_1) + \widehat{h}(E_2) \\
\widehat{h}((E_1 \ \mathtt{*} \ E_2)) &= \widehat{h}(E_1) \cdot \widehat{h}(E_2)
\end{aligned}$$

We now note that $\widehat{h}$ is identical to the "meaning function" $\mathcal{M}$ that we defined in Ex. 2.3.6. Thus in this case we may regard the application of Th. 2.8.2 as formal validation of the existence of $\mathcal{M}$. ■

At first one might question the utility of such "formal validation". Applying Th. 2.8.2 seems like a rather roundabout and complex way of proving something obvious. After all, we can immediately see from the defining equations of $\mathcal{M}$ in Ex. 2.3.6 that the definition is legitimate; it is a simple well-founded recursion. And if we really need to be formal then it seems that a straightforward application of the principle of well-founded recursion will prove the existence of $\mathcal{M}$ much more easily, with no need to bring in unique generation and homomorhisms.

But things are not as simple as they seem. The definition of $\mathcal{M}$ in Ex. 2.3.6, although left unquestioned there, is really a definition by cases, which might be rephrased as follows:

$$\mathcal{M}[E] = \begin{cases} 0 & \text{if } E = \texttt{0} \\ \mathcal{M}[E_1] + 1 & \text{if } E = \texttt{s(}E_1\texttt{)} \\ \mathcal{M}[E_1] + \mathcal{M}[E_2] & \text{if } E = \texttt{(}E_1 \texttt{ + } E_2\texttt{)} \\ \mathcal{M}[E_1] \cdot \mathcal{M}[E_2] & \text{if } E = \texttt{(}E_1 \texttt{ * } E_2\texttt{)} \end{cases}$$

The potential problem here is not with the recursion, which is admittedly patently innocuous, but with the *cases* of the definition. We know that a function definition by cases is legitimate only if the cases are mutually exclusive and jointly exhaustive. This is where the issue of unique generation enters the picture. In this particular example the cases *are* mutually exclusive and jointlye exhaustive (precisely because **GA** is uniquely generated), but in other cases this might not hold, and the problem might not be obvious because the recursive equations will still look innocent enough to be taken for granted. Consider, for instance, the following recursive definition of a language $L$:

- $\texttt{0} \in L$;
- if $E$ is in $L$ then so is $\texttt{s(}E\texttt{)}$; and
- if $E_1, E_2$ are in $L$ then so are $E_1 \texttt{ + } E_2$ and $E_1 \texttt{ * } E_2$.

Now we might go ahead and haphazardly define a similar meaning function $\mathcal{M}_L : L \to N$ as

$$\begin{aligned} \mathcal{M}_L[\texttt{0}] &= 0 \\ \mathcal{M}_L[\texttt{s(}E\texttt{)}] &= \mathcal{M}_L[E] + 1 \\ \mathcal{M}_L[E_1 \texttt{ + } E_2] &= \mathcal{M}_L[E_1] + \mathcal{M}_L[E_2] \\ \mathcal{M}_L[E_1 \texttt{ * } E_2] &= \mathcal{M}_L[E_1] \cdot \mathcal{M}_L[E_2]. \end{aligned}$$

These equations again look reasonable enough that one might take the existence of $\mathcal{M}_L$ for granted. But in fact the equations fail to define a function. This can be seen by recasting the equations in a definition-by-cases format:

$$\mathcal{M}_L[E] = \begin{cases} 0 & \text{if } E = \texttt{0} \\ \mathcal{M}_L[E_1] + 1 & \text{if } E = \texttt{s(}E_1\texttt{)} \\ \mathcal{M}_L[E_1] + \mathcal{M}_L[E_2] & \text{if } E = E_1 \texttt{ + } E_2 \\ \mathcal{M}_L[E_1] \cdot \mathcal{M}_L[E_2] & \text{if } E = E_1 \texttt{ * } E_2 \end{cases} \tag{2.8}$$

Again the recursion is well-founded, but the cases are *not* mutually exclusive. Consider, for instance, the string $E = \texttt{s(0) + 0 * s(s(0))}$. For this string *both* of the last two clauses of Eq. 2.8 are applicable, because $E$ can be seen as being of the form$E_1$+$E_2$—take $E_1 = \texttt{s(0)}$ and $E_2 = \texttt{0 * s(s(0))}$—as well as of the form $E_1$*$E_2$—take $E_1 = \texttt{s(0) + 0}$ and $E_2 = \texttt{s(s(0))}$. Accordingly, if we follow this definition we are forced to conclude that

$$\mathcal{M}_L[\texttt{s(0) + 0 * s(s(0))}] = 1 \quad \text{(by the third clause of 2.8)}$$

and also that

$$\mathcal{M}_L[\texttt{s(0) + 0 * s(s(0))}] = 2 \quad \text{(by the fourth clause of 2.8).}$$

Clearly, this is no way for a function to behave. The underlying problem is that our definition of $L$ is ambiguous, which will be apparent if we express it as a context-free grammar (e.g. the string `0 + 0 * 0` will then be seen to have two distinct parse trees).

The idea of unique generation is a convenient technical tool for making the connection between unambiguous grammars and the validity of recursive function definitions such as that of $\mathcal{M}$ in Ex. 2.3.6. In the case of $L$, if we formalized the definition algebraically we would see that we would not be able to prove unique generation, and hence we would not be able to apply Th. 2.8.2 to prove that the definition of $\mathcal{M}_L$ is legitimate—and rightly so, as we just explained. Let us see where things would go wrong. For an algebraic formalization of the definition of $L$ we may introduce the algebra $(\Sigma^*_{\mathbf{GA}}; \texttt{0}; \phi_{\texttt{s}}, \phi_{\texttt{+}}, \phi_{\texttt{*}})$, where everything is as before except $\phi_{\texttt{+}}$ and $\phi_{\texttt{*}}$, which are now defined as $\phi_{\texttt{+}}(E_1, E_2) = E_1\texttt{+}E_2$ and $\phi_{\texttt{*}}(E_1, E_2) = E_1\texttt{*}E_2$. The difference, of course, is that the parentheses have now been dropped for $\phi_{\texttt{+}}$ and $\phi_{\texttt{*}}$, which is ultimately what causes the ambiguity. Again we identify $L$ with $[\emptyset]$, the smallest subset of $\Sigma^*_{\mathbf{GA}}$ that contains `0` and is closed under $\phi_{\texttt{s}}$, $\phi_{\texttt{+}}$ and $\phi_{\texttt{*}}$. This is a correct formalization of the verbal definition. But is $[\emptyset]$ uniquely generated by $\emptyset$ in this algebra? No. The string $E =$ `s(0) + 0 * s(s(0))` is generated by two different constructor applications: we have

$$E = \phi_{\texttt{+}}(\texttt{s(0)}, \texttt{0 * s(s(0))})$$

and also

$$E = \phi_{\texttt{*}}(\texttt{s(0) + 0}, \texttt{s(s(0))}).$$

Hence unique generation fails and Th. 2.8.2 does not apply.

Unique generation is slightly different for $\mathcal{F}$-algebras due to the naming issue, i.e., due to the presence of the realization assignments. In particular, let $A$ be an $\mathcal{F}$-algebra and let $X \subseteq A$. We say that the algebra $[X]$ is **uniquely $\mathcal{F}$-generated by** $[X]$ iff for every $a \in [X]$ exactly one of the following holds:

(1) $a \in X$; or

(2) $a = c^A$ for a unique $c \in \mathcal{F}_0$; or

(3) $a = f^A(a_1, \ldots, a_n)$ for a unique $f \in \mathcal{F}_n, n \in N_+$, and a unique tuple $(a_1, \ldots, a_n) \in [X]^n$.

Note that uniqueness here is demanded of the *names* $c$ and $f$, not just of the objects $c^A$ and $f^A$. An alternative characterization of unique $\mathcal{F}$-generation is the following:

**Proposition 2.8.1** *Let $((A, C, \Phi), \sigma, \tau)$ be an $\mathcal{F}$-algebra and let $X \subseteq A$. The subalgebra $[X]$ is uniquely $\mathcal{F}$-generated by $X$ iff it is uniquely generated by $X$ in $(A, C, \Phi)$ and the realization assignments $\sigma$ and $\tau$ are bijective.*

The proof is straightforward and omitted.

We can prove the following important result:

**Theorem 2.8.3** *Let $A$ be an $\mathcal{F}$-algebra that is uniquely $\mathcal{F}$-generated by a subset $X \subseteq A$. Then $A$ is free in* $\mathrm{Alg}(\mathcal{F})$ *over $X$. That is, for any $\mathcal{F}$-algebra $B$ and any function $h : X \to B$ there is a unique morphisn $\hat{h} : A \to B$ extending $h$, so that*

$$\begin{aligned}\hat{h}(x) &= h(x)\\ \hat{h}(c^A) &= c^B\\ \hat{h}(f^A(a_1,\ldots,a_n)) &= f^B(\hat{h}(a_1),\ldots,\hat{h}(a_n))\end{aligned}$$

*for all* $x \in X, c \in \mathcal{F}_0, f \in \mathcal{F}_n, n \in N_+$, *and* $a_1,\ldots,a_n \in A$.

**Proof**: Consider the product $\mathcal{F}$-algebra $A \times B$, with

$$\begin{aligned}c^{A\times B} &= (c^A, c^B)\\ f^{A\times B}((a_1,b_1),\ldots,(a_n,b_n)) &= (f^A(a_1,\ldots,a_n), f^B(b_1,\ldots,b_n))\end{aligned}$$

for every $c \in \mathcal{F}_0, f \in \mathcal{F}_n, (a_j,b_j) \in A \times B$. Set

$$\hat{h} = [h]_{Sub\{A\,\times\,B\}} = [\{(x,h(x)) \mid x \in X\}]_{Sub\{A\,\times\,B\}}$$

so that $\hat{h}$ is the smallest subalgebra of $A \times B$ that contains $h$, as well as every constant $c^{A\times B}$, and is closed under every operation $f^{A\times B}, f \in \mathcal{F}_+$. Accordingly,

$$\hat{h} = \bigcup_{i\in N} Con^i_{A\times B}(h)$$

where, for any $Y \subseteq A \times B$,

$$Con_{A\times B}(Y) \;=\; Y \cup \{c^{A\times B} \mid c \in \mathcal{F}_0\} \cup \{f^{A\times B}((a_1,b_1),\ldots,(a_n,b_n)) \mid f \in \mathcal{F}_n, (a_j,b_j) \in Y\} \;=$$
$$Y \cup \{(c^A,c^B) \mid c \in \mathcal{F}_0\} \cup \{(f^A(a_1,\ldots,a_n), f^B(b_1,\ldots,b_n)) \mid f \in \mathcal{F}_n, (a_j,b_j) \in Y\}.$$

We will show by structural induction that for every $a \in A = [X]$ there is exactly one $b \in B$ such that $(a,b) \in \hat{h}$, thus proving that $\hat{h}$ is a function. For the basis step, suppose that $a \in X$. Then, by definition, $(a,h(a)) \in \hat{h}$, so the existence part is readily established. For uniqueness, we show that for all $i \in N$, if $(a,b) \in Con^i(h)$ then $b = h(a)$ (we will be writing $Con$ rather than $Con_{A\times B}$). We use induction on $i$. When $i = 0$ the claim follows directly since $Con^0(h) = h$ and $h$ is a function by supposition. For the inductive step we have

$$Con^{i+1}(h) = Con(Con^i(h)) = Con^i(h) \cup \{(c^A,c^B) \mid c \in \mathcal{F}_0\} \cup New_i \tag{2.9}$$

where

$$New_i = \{(f^A(a_1,\ldots,a_n), f^B(b_1,\ldots,b^n)) \mid f \in \mathcal{F}_n, (a_j,b_j) \in Con^i(h)\}.$$

Thus, if $(a,b) \in Con^{i+1}(h)$ then either $(a,b) \in Con^i(h)$, or $(a,b) = (c^A,c^B)$ for some $c \in \mathcal{F}_0$, or else $(a,b) \in New_i$. In the first case we have $b = h(a)$ from the inductive hypothesis. The second and third cases are impossible because they would respectively entail that $a = c^A$ or $a = f^A(a_1,\ldots,a_n)$ for some constant $c^A$ in $A$, or $n$-ary operation $f^A$ on $A$ and $a_1,\ldots,a_n \in A$, which would contradict the unique generation of $A = [X]$ since we are currently assuming that $a \in X$. Hence in these two cases the claim follows vacuously by virtue of a false antecedent. This completes the case analysis and the inductive step.

Next suppose that $a$ is a constant $d^A$ for some $d \in \mathcal{F}_0$. Then $(d^A, d^B) \in Con^{i+1}(h)$ for all $i \in N$, thus existence is proved. For uniqueness, we show by induction that for all $i \in N$, if $(a, b) \in Con^i(h)$ then $b = d^B$. For $i = 0$ this holds by virtue of a false antecedent: we cannot have $(a, b) \in Con^0(h)$ since this would entail that $a = c^A \in X$, contradicting the unique generation of $[X]$. For the inductive step, Eq. 2.9 holds, so if $(a, b) \in Con^{i+1}(h)$ then either $(a, b) \in Con^i(h)$, or $(a, b) = (c^A, c^B)$ for some $c \in \mathcal{F}_0$, or else $(a, b) = (f^A(a_1, \ldots, a_n), f^B(b_1, \ldots, b_n))$ for some $f \in \mathcal{F}_n, (a_j, b_j) \in Con^i(h)$. In the first case the claim follows from the inductive hypothesis. In the second case we have $a = c^A, b = c^B$, and because $A$ is uniquely $\mathcal{F}$-generated, $c = d$, hence $b = c^B = d^B$. In the third case the claim holds vacuously since the antecedent, if it were to be true, would imply $a = f^A(a_1, \ldots, a_n)$ for some $n$-ary operation $f^A, n \in N_+$, and $a_1, \ldots, a_n \in A$, contradicting the unique generation of $a$. This completes the case analysis, as well as the inductive step.

Finally, for the inductive step, suppose that $a = g^A(a'_1, \ldots, a'_m)$ for some $g \in \mathcal{F}_m, m \in N_+$, and $a'_1, \ldots, a'_m \in A$. From the inductive hypothesis, there are unique elements $b'_1, \ldots, b'_m$ in $B$ such that $(a'_j, b'_j) \in \hat{h}$ for $j = 1, \ldots, m$. Therefore, since $\hat{h}$ is closed under $g^{A \times B}$, we have

$$g^{A \times B}((a'_1, b'_1), \ldots, (a'_m, b'_m)) = (g^A(a'_1, \ldots, a'_m), g^B(b'_1, \ldots, b'_m)) = (a, g^B(b'_1, \ldots, b'_m)) \in \hat{h}$$

and the existence part is established. For uniqueness, we claim that for all $i \in N$, if $(a, b) \in Con^i(h)$ then $b = g^B(b'_1, \ldots, b'_m)$. We wil prove this by induction on $i$. When $i = 0$ the claim holds vacuously because the antecedent is false: there is no $b \in B$ such that $(a, b) \in Con^0(h) = h$. This is because otherwise we would have $a \in X$, which would contradict the unique generation of $A$ since we are currently assuming that $a = g^A(a'_1, \ldots, a'_n)$. For the inductive step, Eq. 2.9 holds again, so if $(a, b) \in Con^{i+1}(h)$ there are three cases: $(a, b) \in Con^i(h)$, or $(a, b) \in (c^A, c^B)$ for some $c \in \mathcal{F}_0$, or else $(a, b) \in New_i$. In the first case it follows immediately from the inductive hypothesis that $b = g^B(b'_1, \ldots, b'_m)$. The second case is impossible because it would contradict the unique generation of $a$, since we are currently assuming that $a = g^A(a'_1, \ldots, a'_m)$. In the third case we have

$$(a, b) = (f^A(a_1, \ldots, a_n), f^B(b_1, \ldots, b_n))$$

for some $f \in \mathcal{F}_n, n \in N_+$, and $(a_1, b_1), \ldots, (a_n, b_n) \in Con^i(h)$, hence $a = f^A(a_1, \ldots, a_n), b = f^B(b_1, \ldots, b_n)$. Since $A$ is uniquely $\mathcal{F}$-generated, $f = g, n = m$, and $a_1 = a'_1, \ldots, a_n = a'_n$. Thus, from the outer inductive hypothesis and the assumption $(a_1, b_1), \ldots, (a_n, b_n) \in Con^i(h)$ it follows that $b_1 = b'_1, \ldots, b_n = b'_n$, hence $b = f^B(b_1, \ldots, b_n) = g^B(b'_1, \ldots, b'_m)$ and the induction is complete.

The above inductive argument also serves to show that $\hat{h}$ is a homomorphism. The uniqueness of $\hat{h}$ follows from Prop. 2.6.1. ■

# Exercises

1. Prove or disprove: the algebra $(\mathcal{P}(A); \quad, \cup, \cap)$ is generated by the set of all singletons over $A$ (one-element subsets of $A$). Can you find necessary and sufficient conditions for this to hold?
   **Solution**. This does not hold in general. Let $S \subseteq \mathcal{P}(A)$ consist of all one-element subsets of $A$. Using theorem 2.2.1 it is easy to see that $[S] = Con^\infty(S)$ will contain only finite sets. It is not hard to prove that the claim will hold iff $A$ is finite.