Category Theory and Functional Programming Algebras and Recursive Data Types

Andrés Sicard-Ramírez

Universidad EAFIT

Semester 2022-2

### Preliminaries

Convention

The number assigned to chapters, examples, exercises, figures, pages, sections, and theorems on these slides correspond to the numbers assigned in the textbook [Abramsky and Tzevelekos 2011].

### Outline

Introduction

F-Algebras

References

# Introduction

The relation between recursive (inductive) data types and initial algebras is described by [Fong, Milewski and Spivak 2020, p. 98] as:

The string of a recursive data type is a functor, and the knot is its initial algebra.

#### Example (Natural numbers)

We define the recursive data type for natural numbers by

data Nat = Zero | Succ Nat

and we define the non-recursive associated functor by

```
data NatF a = ZeroF | SuccF a
instance Functor NatF where
fmap :: (a -> b) -> NatF a -> NatF b
fmap _ ZeroF = ZeroF
fmap f (SuccF n) = SuccF (f n)
```

### Functors Associated with Recursive Data Types

### Example (Lists)

We define the recursive data type for lists by

```
data List c = Nil | Cons c (List c)
```

and we define the non-recursive associated functor by

```
data ListF c a = NilF | ConsF c a
```

```
instance Functor (ListF c) where
fmap :: (a -> b) -> ListF c a -> ListF c b
fmap _ NilF = NilF
fmap f (ConsF x y) = ConsF x (f y)
```

# F-Algebras

## F-Algebras

Definition

Let  $F : C \to C$  be an endofunctor. An *F*-algebra  $(A, \alpha)$  is [Fong, Milewski and Spivak 2020, p. 104]

(i) an object A in  $Obj(\mathcal{C})$  (the carrier),

(ii) an arrow  $\alpha : F_0 A \to A$  (the structure map).

## Algebra Arrows

Definition

Let  $F : \mathcal{C} \to \mathcal{C}$  be an endofunctor and let  $(A, \alpha)$  and  $(B, \beta)$  be two *F*-algebras. An **algebra arrow**  $f : (A, \alpha) \to (B, \beta)$  is an arrow  $f : A \to B$  in  $\mathcal{C}$  such that the following diagram commutes [Fong, Milewski and Spivak 2020, p. 104]:



$$\left(f\circ\alpha=\beta\circ(F_1f)\right)$$

## References

### References

- Abramsky, S. and Tzevelekos, N. (2011). Introduction to Categories and Categorical Logic. In: New Structures for Physics. Ed. by Coecke, B. Vol. 813. Lecture Notes in Physics. Springer, pp. 3–94. DOI: 10.1007/978-3-642-12821-9\_1 (cit. on p. 2).
- Fong, B., Milewski, B. and Spivak, D. I. (2020). Programming with Categories (DRAFT). Last update: October 6, 2020. URL: http://brendanfong.com/programmingcats.html (cit. on pp. 5, 9, 10).