Problem Set 2

January 20, 2022

Homework is due midnight before next Thursday's class.
Note that you may submit your solutions jointly with a partner on Canvas.

Problem 1

Awodey Section 1.9, Exercise 11, part b[1].

. . . .

Problem 2

This problem is inspired by Scott [1980].

Software contracts [Findler and Felleisen, 2013] are used for runtime assertions in untyped programming languages or when a type system is not sophisticated enough to describe the desired invariant.

A simple[2] model of a contract in a category is an *idempotent*. An idempotent in a category C is simply an endomorphism:

$$c : A \to A$$

That is equal to its composition with itself

$$c \circ c = c$$

We think of the idempotent c as a (partial) function that coerces everything to conform to some property. It is idempotent because once something is coerced, it already has the desired property.

For instance the values of a simple first-order dynamically typed language supporting integers and booleans might be modeled by a universal set of tagged values:

$$D = \{(0, b) | b \in \{\text{true}, \text{false}\}\} \cup \{(1, n) | n \in \mathbb{Z}\}$$

[1] Part a is recommended as well, but will not be graded since the solution is in the back of the book

[2] in practice, a bit too general, but a good starting point

and functions in the language as *partial* functions, with undefinedness used to model errors and infinite loops.

Then a contract $\mathrm{int} : D \rightharpoonup D$ for integers could be defined as

$$\mathrm{int}((1, n)) = (1, n)$$

and undefined on other values. Clearly this is idempotent.

We would like to think of an idempotent as presenting an object itself, the "invariants" of the idempotent, i.e., those that "satisfy the contract". In Set we could define the invariants of an idempotent c on A as $\{x \in A | c(x) = x\}$. We generalize this idea to a general category as follows.

We say an idempotent c on A is *split* if there exists an object I and a *section-retraction* pair

$$s : I \to A$$

$$r : A \to I$$

that is, $r \circ s = \mathrm{id}$, that splits the idempotent in that $s \circ r = c$.

Note that for any section/retraction pair s, r, the composite $s \circ r$ is an idempotent (exercise!). We can then ask if in a particular category every idempotent splits. For instance, in Set every idempotent splits, using the set of invariants of the idempotent. However, clearly not every category has this property.

For any category C, we can construct a category $K(C)$, called the *Karoubi envelope* (sometimes called the *Cauchy completion*) that extends C with the splittings of all idempotents.

Define the Karoubi envelope $K(C)$ as follows.

1. Objects are pairs (A, c) of an object in C and an idempotent c on A.
2. A morphism from (A, c) to (B, d) is a morphism $f : A \to B$ satisfying

$$d \circ f \circ c = f$$

If we think of an idempotent as a contract, this provides us with a category where "types are contracts".

First, complete the definition of the category $K(C)$:

1. Show that composition in $K(C)$ can be given by composition in C, i.e., that the composition interacts properly with the idempotents[3].
2. Define the identity morphisms and show they are identities with respect to composition in $K(C)$.

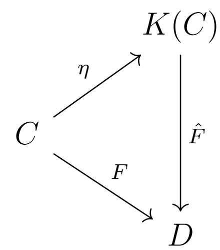
Show that $K(C)$ is the "idempotent splitting completion" of C:

1. Show that every idempotent in $K(C)$ splits.

[3]Hint: It may be helpful to prove that the condition $d \circ f \circ c = f$ is equivalent to $d \circ f = f = f \circ c$ first.

2. Define a functor $\eta : C \to K(C)$.

3. Show that for any category D where every idempotent splits, any functor $F : C \to D$ can be extended to a functor $\hat{F} : K(C) \to D$ satisfying:



You do not need to show this functor is unique[4].

For the logically-inclined, note that constructing $\hat{F}$ will use the axiom of choice.

. . . .

References

Robert Bruce Findler and Matthias Felleisen. Icfp 2002: Contracts for higher-order functions. *SIGPLAN Not.*, 48(4S):34–45, jul 2013. ISSN 0362-1340. doi: 10.1145/2502508.2502521. URL `https://doi.org/10.1145/2502508.2502521`.

Dana S Scott. Relating theories of the lambda calculus. *To HB Curry: Essays on combinatory logic, lambda calculus and formalism*, pages 403–450, 1980. URL `http://maxsnew.com/docs/scott80.pdf`.

[4] because it is only “unique up to isomorphism”, something we have not yet defined