Monads

Samuel Mimram
samuel.mimram@lix.polytechnique.fr
http://lambdacat.mimram.fr

December 7, 2020

1 The exception monad

Given an adjunction $F \dashv G$ between categories $\mathcal{C}$ and $\mathcal{D}$, the composite $T = G \circ F$ is always equipped with a structure of a monad, and the goal of this question is to study an instance of this situation.

We write $\mathbf{Set}_*$ for the category whose objects are *pointed sets*, i.e. pairs (A, a) where A is a set and $a \in A$, and morphisms $f : (A, a) \to (B, b)$ are functions such that $f(a) = b$. Here, the distinguished element of the pointed set will be seen as a particular value indicating an error or an exception.

1. Describe the *forgetful functor* $U : \mathbf{Set}_* \to \mathbf{Set}$.
2. Construct a functor $F : \mathbf{Set} \to \mathbf{Set}_*$ which is such that the sets $\mathbf{Set}_*(FA, (B, b))$ and $\mathbf{Set}(A, U(B, b))$ are isomorphic. We will admit that F is left adjoint to U (what would remain to be shown?).
3. We recall that a *monad* consists of an endofunctor $T : \mathcal{C} \to \mathcal{C}$ together with two natural transformations $\mu : T \circ T \Rightarrow T$ and $\eta : \mathrm{id}_{\mathcal{C}} \Rightarrow T$ such that the following diagrams commute:

   Describe a structure of monad on $T = U \circ F$.
4. Explain how a function $A \to TB$ can be seen as "a function $A \to B$ which might raise an exception".
5. Given $f : A \to B$ an OCaml function which might raise an unique exception e and $g : B \to C$ a function which might raise an unique exception e', construct a function corresponding to the composite of f and g which might raise a unique exception e''.
6. Given an arbitrary monad T on a category $\mathcal{C}$, we write $\mathcal{C}_T$ for the category whose objects are the objects of $\mathcal{C}$ and morphisms $f : A \to B$ in $\mathcal{C}_T$ are morphisms $f : A \to TB$ in $\mathcal{C}$, called the *Kleisli category* associated to T. Define composition and identities and show that the axioms of categories are satisfied.
7. Give an explicit description of $\mathbf{Set}_T$ in the case of the above exception monad.

2 More monads

1. A *non-deterministic function* is a function that might return a set of values instead of a single value. How could we could we similarly define a category of non-deterministic functions by a Kleisli construction?
2. Recall the adjunctions defining a cartesian closed category. What is the associated monad?

3 Monads in Haskell

Here is an excerpt of `http://www.haskell.org/haskellwiki/Monad`:

```
Monads can be viewed as a standard programming interface to various data or control structures,
which is captured by the Monad class.  All common monads are members of it:

class Monad m where
  (>>=) :: m a -> (a -> m b) -> m b
  return :: a -> m a

In addition to implementing the class functions, all instances of Monad should obey the
following equations:

return a >>= k  =  k a
m >>= return  =  m
m >>= (\x -> k x >>= h)  =  (m >>= k) >>= h
```

1. What does the `Maybe` monad defined below do?

```
data Maybe a = Nothing | Just a

instance Monad Maybe where
    return         = Just
    Nothing  >>= f = Nothing
    (Just x) >>= f = f x
```

2. What does the `List` monad defined below do?

```
instance Monad [] where
    m >>= f  = concatMap f m
    return x = [x]
```

A *Kleisli triple* $(T, \eta, (-)^*)$ on a category $\mathcal{C}$ consists of

- a function $T : \mathrm{Ob}(\mathcal{C}) \to \mathrm{Ob}(\mathcal{C})$,
- a function $\eta_A : A \to TA$ for every object A of $\mathcal{C}$,
- a morphism $f^* : TA \to TB$ for every morphism $f : A \to TB$,

such that for every objects A, B, C and morphisms $f : A \to TB$ and $g : B \to TC$,

$$\eta_A^* = \mathrm{id}_{TA} \qquad f^* \circ \eta_A = f \qquad g^* \circ f^* = (g^* \circ f)^*$$

Our aim is to show that this data amounts to specify a monad on $\mathcal{C}$.

3. Construct the Kleisli category associated to a Kleisli triple.
4. Show that every Kleisli triple induces a monad.
5. Conversely show that every monad induces a Kleisli triple.

We admit that the two transformations are mutually inverse.

4 Monads in Rel

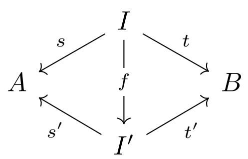
We define **Rel** as the 2-category whose 0-cells are sets, 1-cells $R : A \to B$ are relations $R \subseteq A \times B$, there is a unique 2-cell $\alpha : R \Rightarrow R' : A \to B$ whenever $R \subseteq R'$.

1. Recall both horizontal and vertical compositions in **Rel**.
2. Generalize the definition of adjunction and monad to any 2-category.
3. Show that a left adjoint in **Rel** is a function.
4. What is a monad in **Rel**?

5 Monads in Span

The 2-category of **Span** is the category where

- a 0-cell is a set
- a 1-cell from A to B is a *span*: $A \xleftarrow{s} I \xrightarrow{t} B$
- a 2-cell $f : (s,t) \to (s',t')$ is a function making the following diagram commute



Horizontal composition of 1-cells is given by pullback.

1. What is an endomorphism $A \to A$? A 2-cell between such endomorphisms?
2. Detail the compositions and identities of the 2-category.
3. Is it really a 2-category?
4. What is a monad in this "2-category"?