Abstract

Some exercises from `http://brendanfong.com/programmingcats_files/ps1.pdf`

Exercise 2

a) $Ob(\mathcal{C}) = 1, 2$; The sets of morphisms: $\mathcal{C}(1,1), \mathcal{C}(1,2), \mathcal{C}(2,1), \mathcal{C}(2,2)$. The composition rule $f \circ id_1 = id_2 \circ f : 1 \to 2$. The identity morphisms: id_1 and id_2.

b)Unit: $f \circ id_1 = f$ and $id_2 \circ f = f$. Associative: $f \circ id_1 = id_2 \circ f$.

Exercise 3

If there are no identity morphisms id_c and id_d, then we can't affirm an isomorphism with $g = f^{-1}$, given that we can't prove $g \circ f = id_c$ and $f \circ g = id_d$?

Exercise 4

a)Following associative, but not unity law:

$$\begin{tikzcd} 1 \arrow[r, bend left, "f"] & 2 \arrow[l, bend left, "g"] \end{tikzcd}$$

Here if $h = f \circ g$, we have $(f \circ g) \circ h = f \circ (g \circ h) = f$, but since there aren't any identity morphisms unity is not followed.

b)Following unit, but not associative law: Can build free categories where each object has an identity morphism?

Exercise 5

a) The carrier set is $\mathbb{N}$, operation or function is $+$ and the identity element 0. Unity is followed as any $n \in \mathbb{N} = n$, when $e * n$ or $n * e$, given that $n + 0 = n$. Associativity follows given that addition is commutative.

b) Here $\text{List}_{\{0,1\}}$ is the carrier set, concatenation the operation, and empty string [] the identity. Unity is followed given that concatenating an empty string to an initial string s will output the s. Associativity isn't followed? Order of concatenation matters?

c) This is because a monoid is a category that only has one homset. Consider the $\mathbb{N}$ carrier set in a). All morphisms in this monoid are of the type $\mathbb{N} \to \mathbb{N}$, with the corresponding monoid operation and identity element. This is the case for any monoid independent of the triple $(M, e, \diamond)$.

Exercise 6

a) $id_{12} = 1$ given that for $x * 12 = 12$ only $x = 1$ suffices.

b) If $x : a \to b$ and $y : b \to c$ then $y \circ x : a \to c$ implies that if b is divisible by a and c is divisible by b, then c is divisible by a, which follows given that division is transitive.

c) If $\mathcal{P}$ would be given by all $\mathbb{N}$, then the previous arguments wouldn't work because division by 0 is undefined.

Exercise 7

Let

$$\text{True} = \lambda x.(\lambda y.x)$$

$$\text{False} = \lambda x.(\lambda y.y)$$

$$\text{AND} = \lambda p.(\lambda q.(pq)p)$$

$$\text{OR} = \lambda p.(\lambda q.(pp)q)$$

If we have (AND True)False, this will evaluate to

$$\begin{aligned}(\text{AND True})\text{False} &= \lambda p.(\lambda q.(pq)p)(\text{True})(\text{False}) \\ &= \text{True False True} \\ &= \text{False}\end{aligned}$$

For (OR False)True

$$\begin{aligned}(\text{OR False})\text{True} &= \lambda p.(\lambda q.(pp)q)(\text{False})(\text{True}) \\ &= \text{False False True} \\ &= \text{True}\end{aligned}$$

Exercise 8

The Y combinator is defined as follows

$$Y = \lambda f.((\lambda x.f(xx))(\lambda x.f(xx)))$$

Yg will then be computed as

$$
\begin{aligned}
Yg &= \lambda f.((\lambda x.f(xx))(\lambda x.f(xx)))g \\
&= \lambda f.(f((\lambda x.f(xx))(\lambda x.f(xx)))g \\
&= g((\lambda x.g(xx))(\lambda x.g(xx))) \\
&= g(\underbrace{g((\lambda x.g(xx))(\lambda x.g(xx)))}_{\text{Y g}})
\end{aligned}
$$

$$Yg = g(Yg) = g(g(Yg)) = \ldots$$