Monads

d.morgan1

11 July 2022

1 Monads

The infix operator `>>=` is called bind. It is used to extract contents from a monad, e.g. we can reach inside `Just x` to operate on `x`.

```
instance Monad Maybe where
    -- (>>=) :: Maybe a -> (a -> Maybe b) -> Maybe b
    (Just x) >>= k      = k x
    Nothing  >>= _      = Nothing
    return              = pure
```

The following are equivalent:

```
madd mx my =
    mx >>= \x ->
    my >>= \y ->
    return $ x + y
```

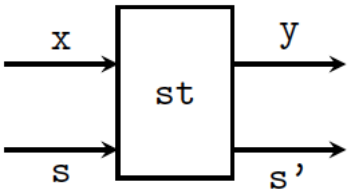
```
madd mx my = do
    x <- mx
    y <- my
    return $ x + y
```

Monad Laws

1. `return x >>= f = f x` *Left identity*
2. `mx >>= return = m x` *Right identity*
3. `(mx >>= f) >>= g = mx >>= (\x -> f x >>= g)` *Associativity*

2 The State Monad

A *state transformer* takes an input state as its argument and produces an output state as its result.




```
instant Monad ST where
    -- (>>=) :: ST a -> (a -> ST b) -> ST b
    st >>= f = S(\ S ->
        let
            (x, s') = app st s
        in
            app (f x) s')
```