HASKELL WRAP-UP

Curt Clifton
Rose-Hulman Institute of Technology

Open *EddieEval.hs* from last time

RECALL…

CAN WE BE JUST A LITTLE BIT IMPURE?

- How are we getting side effects if Haskell is a pure language?

- Solution: Pass along an object to be "mutated"

- Original: $f :: Tree \rightarrow Int$

- New: $f :: (Tree, State) \rightarrow (Int, State)$

Original State

"Mutated" State

Monads automate this pattern

THE MONAD TYPECLASS

Sequences two expressions that have Monad results

Sequences two Monad expressions binding result of first for use in second

```
class Monad m where
  (>>) :: m a -> m b -> m b
  (>>=) :: m a -> (a -> m b) -> m b
  return :: a -> m a
  fail :: String -> m a
```
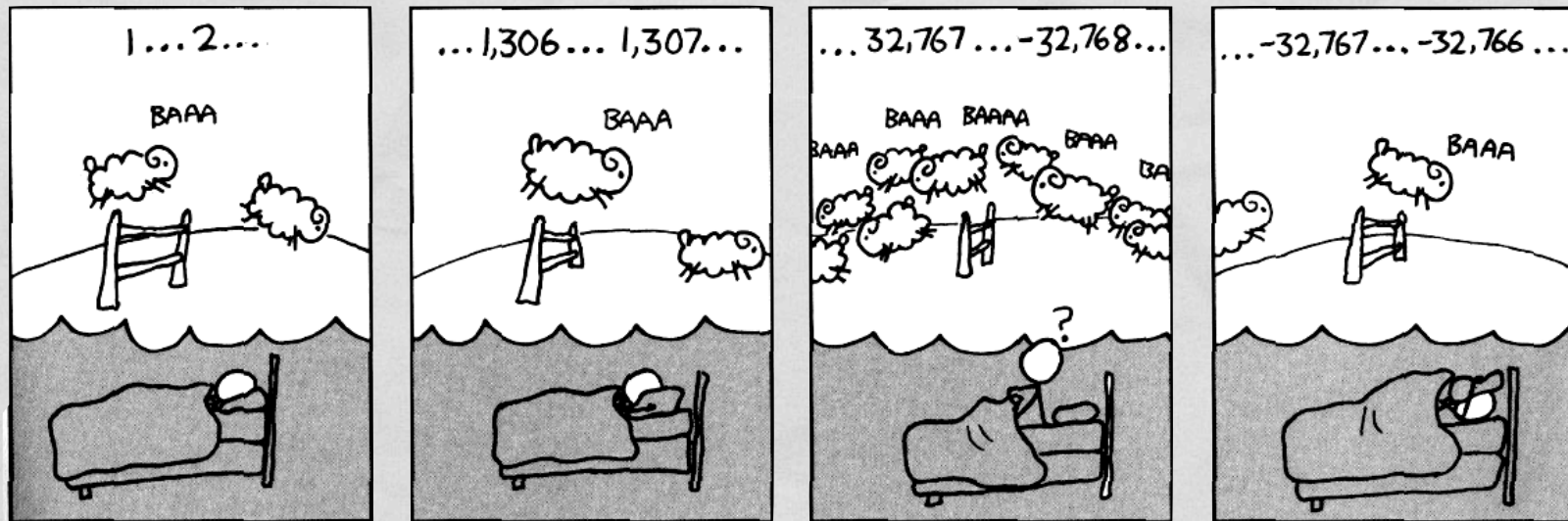
Wrap pure value in Monad

DA DO DO DO

The *do* expression in Haskell is just a sugar for Monad sequencing

Inside *do*	Monad notation
`e1` `e2`	`e1 >>= \_ -> e2` or `e1 >> e2`
`x <- e1` `e2`	`e1 >>= \x -> e2`
`return e1`	`return e1`

IMPLEMENTING AN INTERPRETER USING MONADS

CARTOON OF THE DAY

If androids someday do dream of electronic sheep
don't forget to declare *sheepCount* as a long int.

HASKELL WRAP-UP

- Characteristics of projects that would…

  - make Haskell a reasonable choice

  - make Haskell a poor choice

- Jot down a few ideas, then we'll brainstorm

Q1,2