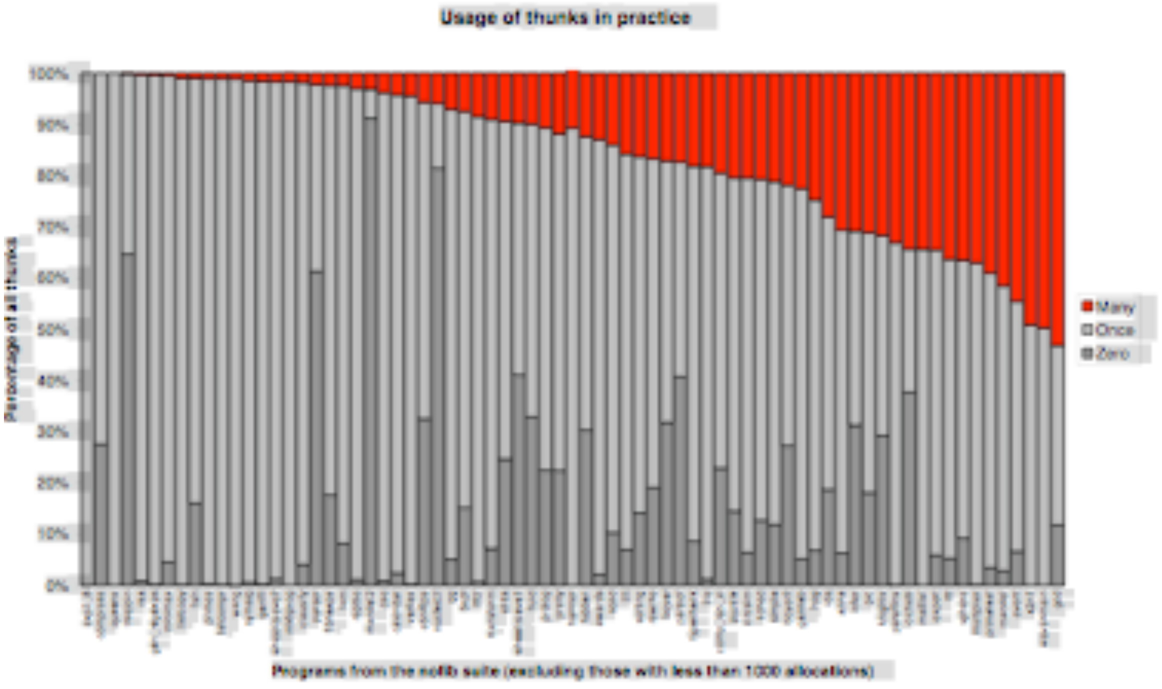
Call-by-need is not as nice as call-by-name…

Memoization is a hidden effect

- Implicit sharing behavior makes space reasoning hard

Memoization is costly

- Execution time overhead
- Complexity in RTS
- Complexity in GC



…so who really needs call-by-need?

```
e ::= x | n | e e | p e e | if0 e e e
      | λ!x . e | λ~x . e
      | let !x = e in e | let ~x = e in e
      | cons e e | nil | hd e | tl e
      | scons e e | snil | shd e | stl e

myif = λ!x λ~t λ~f . if0 x t f
  myif n (+ 2 2) (/ 3 n)
mydouble = λ~x. let !x’ = x in + x’ x’
  mydouble (fact 1000000)
```

- Please **don’t** complain about types, polymorphism, …
- Please **do** explain why this is a terrible idea…
- Please **do** explain how all this has been tried before…