Functional Data Structures

Mattox Beckman

Illinois Institute of Technology
Computer Science

Who we are / October 6, 2010
Who is Mattox?

Name  Mattox Beckman

History PhD, Fall 2003, University of Illinois at Urbana-Champaign

Area Programming Languages

Specialty Partial Evaluation, Functional Programming

Professional Interests Teaching; Partial Evaluation; Interpreters; Functional Programming; Semantics and Types; Continuations

Personal Interests Home-brewing; Theology; Investing; Plants; Tarantulas; Evolution; Travel; Korean Culture... and many many more ...

Teaching philosophy is available http://dijkstra.cs.iit.edu/media/mattox/teaching-philosophy.pdf.
Models of Computation

- Why do programming languages look the way they do?
  - Imperative Languages: control the machine
    \[ a := a + 1 \]
  - OO Languages: model the agents
    \[ a.increment() \]
  - Functional Languages: express the mathematics
    \[ a + 1 \]
  - Logic Language: express the constraints
    \[ \text{inc}(a,b) : - b \text{ is } a+1 \]
Why Functional Programming

- Economy of Expression
  
  \[
  \text{guess \{} = \{}
  \]
  
  \[
  \text{guess \{} (x:xs) = \text{guess \{} [y \mid y \leftarrow xs, y < x] ++ [x] ++ \text{guess \{} [y \mid y \leftarrow xs, y \geq x]\}
  \]

- Lazy Evaluation

- No Assignment!!
  - May use more memory
  - Easier to verify
  - Easy to parallelize
What if we run \( t2 = \text{add}(t1, 35) \) on this?
What if we run $t3 = add(t2, 55)$ on this?
Shared References and Mathematical Purity

\[
\begin{align*}
\ast &\quad 10 \quad 30 \\
+ &\quad 10 \quad 30
\end{align*}
\]
How to do this?

\[ \ast \quad + \]

\[ \ast \quad + \]

\[ 10 \quad 30 \quad 11 \quad 31 \]
And not this?

\[
\begin{align*}
\ast & \quad \ast \\
+ & \quad + \\
10 & \quad 10 \\
30 & \quad 31 \\
\end{align*}
\]
Some Solutions

- Hard code the functions. (Yuck)
- Use a “trace”. (Works, but cumbersome.)
- Build a new language construct!