CS 603: Programming Language Organization

Lecture 12

Spring 2005

Department of Computer Science

University of Alabama

Joel Jones

Outline

- Questions
- μ-Scheme (cont.)
- Reading for next time

Let's play at the board again (No Books!)

- Property lists–a-list where attribute is an attribute list
- Examples
  - `(set fruits '((apple ((texture crunch)))`
    `(banana ((color yellow)))))`
  - `(getprop 'apple 'texture fruits)`
    `crunchy`
- Write `(getprop x p plist)`, where x is the individual, p is the property and plist is the property list

Let's play at the board again (No Books!)

- `(putprop x p y plist)` give individual x value y for property p in plist
  - `(set fruits (putprop 'apple 'color 'red fruits))`
    `->((apple ((texture crunchy)(color red)))(banana`
    `((color yellow))))`

μ-Scheme

- Closures
  - Pair of lambda (function value) and environment
    - «`(lambda (y) …)`, $\{x \mapsto l\}$»
  - Environment maps name to mutable location

```
->(val counter-from
       (lambda (n)
           (lambda () (set n (+ n 1)))))
->(val ten (counter-from 10))
<procedure>
->(ten)
11
->(ten)
12
```

μ-Scheme (cont.)

- Closures

Pair Up:

• Write a function (make-withdraw) that models a bank account balance, where only withdrawals are allowed

```
->(val make-withdraw (lambda (balance) …))
->(val W1 (make-withdraw 100))
->(W1 10)
90
```