



CS3L: Introduction to Symbolic Programming

Lecture 14:
Number Spelling Mini-project

Summer 2008

Colleen Lewis
colleenL@berkeley.edu



Administration

- Number Spelling Mini-project 2
 - DUE: Thursday at 11:59 pm
- Compress (Hwk12) occurs-in? (Hwk13)
 - DUE: Tuesday at 11:59 pm
- Design your project before you start



Today

- `item` instead of `cond`
- Testing Framework
- Tail vs. Embedded Recursion



Guideline: Use `item` instead of `cond`

```
(define (get-day-name num)
  (cond
    ((equal? num 1) 'Sunday)
    ((equal? num 2) 'Monday)
    ((equal? num 3) 'Tuesday)
    ((equal? num 4) 'Wednesday)
    ((equal? num 5) 'Thursday)
    ((equal? num 6) 'Friday)
    ((equal? num 7) 'Saturday)))

(define (get-day-name num)
  (item num '(Sunday Monday Tuesday Wednesday
             Thursday Friday Saturday)))
```



Testing Framework

- `(add-test-case`
 `test-name`
 `expected-return-value`
 `scheme-expression)`
- `(run-test-cases)`
- `(clear-test-cases)`



Random Administration

- DO NOT CODE BEFORE YOU HAVE A PLAN!
- Put yours and your partners names and logins on the top
- You should be able to use `scheme->sendBuffer` without errors
- 40 is spelled Forty
- Please no "and"s

Tail versus “embedded” recursions



- A tail recursion has no combiner, so it can end as soon as a base case is reached
 - Compilers can do this efficiently
- An embedded recursion needs to combine up all the recursive steps to form the answer
 - The poor compiler has to keep track everything

Patterns in recursion (1-3 of 6)



- Mapping
 - does something to every part of the input sentence
- Counting
 - Counts the number of elements that satisfy a predicate
- Finding
 - Return the first element that satisfies predicate (or, return rest of sentence)

Patterns in recursion (4-6 of 6)



- Filtering
 - Keep or discard elements of input sentence
- Testing
 - A predicate that checks that *every* or *any* element of input satisfies a test
- Combining
 - Combines the elements in some way...