CS3L: Introduction to Symbolic Programming

Lecture 19: HOF Problems

Summer 2008 Colleen Lewis colleenL@berkeley.edu

Announcements
- Midterm two Tuesday July 29th
  - This one will probably be harder than the first
- Colleen will be out of town Thurs-Monday
- Homework
  - Miniproject 3 started yesterday!
  - Due Friday July 25th at 11:59 pm
- Today is Gilbert’s 7,000th day! And we’re having a surprise party for him!

Today
- Problem: Successive-concatenations
- Which HOFs should I use?
- Work on the mini-project
  - Let me know if you don’t have a partner and want one

Successive Concatenations
(sc '(a b c d e))
⇒ (a ab abc abcd abcde)

(define (sc sent)
  (accumulate
   (lambda () …)
   sent))

Today
- Problem: Successive-concatenations
- Which HOFs should I use?
- Work on the mini-project
  - Let me know if you don’t have a partner and want one

Successive Concatenations
(accumulate (lambda () …) '(the big red barn)
⇒ (the thebig thebigred thebigredbarn)

(define (sc sent)
  (accumulate
   (lambda ??
       )
   sent)))

Successive Concatenations
(accumulate (lambda () …) '(the big red barn)
⇒ (the thebig thebigred thebigredbarn)

(define (sc sent)
  (accumulate
   (lambda ??
       )
   sent))

Successive Concatenations
(accumulate (lambda () …) '(the big red barn)
⇒ (the thebig thebigred thebigredbarn)

(define (sc sent)
  (accumulate
   (lambda ??
       )
   sent))

Successive Concatenations
(accumulate (lambda () …) '(the big red barn)
⇒ (the thebig thebigred thebigredbarn)

(define (sc sent)
  (accumulate
   (lambda ??
       )
   sent))
**Successive Concatenations**

\[(sc \quad \{a \ b \ c \ d \ e\})\]
\[
\Rightarrow (a \ ab \ abc \ abcd \ abcede)
\]

\[(sc \quad \{the \ big \ red \ barn\})\]
\[
\Rightarrow (the \ thebig \ thebigred \ thebigredbarn)
\]

\[(define \ (sc \ sent)\)
\[
(accumulate
\quad (lambda \ (new-wd \ so-far)
\quad (every
\quad (lambda \ (wd) \ (word \ new-wd \ wd))
\quad (se "" \ so-far))
\quad sent))
\]

---

**accumulate**

\[(accumulate \ procedure \ sent)\]

- **procedure**
  - a procedure that takes in two arguments
  - a procedure that combines things together
- **sent**
  - a sentence with 1 or more words
  - a word with 1 or more letters

---

**every**

\[(every \ procedure \ sent)\]

- **procedure**
  - a procedure that takes in one argument
  - a procedure that returns a word or a sentence
- **sent**
  - a sentence with 0 or more words
  - a word with 0 or more letters

---

**keep**

\[(keep \ procedure \ sent)\]

- **procedure**
  - a procedure that takes in one argument
  - a procedure that returns #t or #f
- **sent**
  - a sentence with 0 or more words
  - a word with 0 or more letters

---

**Which HOFs would you use?**

- Capitalize-proper-names
  \[(c-p-n \ \{mr. \ smith \ goes \ to \ washington\})\]
  \[
  \Rightarrow (mr. \ Smith \ goes \ to \ Washington)
  \]
- Every
- Keep
- Accumulate

---

**Which HOFs would you use?**

- Longest Word
  \[(longest-word \ \{have \ a \ great \ week\})\]
  \[
  \Rightarrow \text{great}
  \]
- Every
- Keep
- Accumulate
Which HOFs would you use?

- Count-if odd?
  \[(\text{count-if odd? '}(1 2 3 4 5)) \Rightarrow 3\]
  \[(\text{count-if odd? '}(1 2 3 4 2)) \Rightarrow 2\]

- Every
- Keep
- Accumulate

Which HOFs would you use?

- Count-vowels-in-each
  \[(\text{c-v-i-e '}(\text{good luck on tuesday})) \Rightarrow (2 1 1 3)\]

- Every
- Keep
- Accumulate

Which HOFs would you use?

- Squares-greater-than-100
  \[(\text{s-g-t-100 '}(2 9 13 16 9 45)) \Rightarrow (169 256 2025)\]

- Every
- Keep
- Accumulate