CSSE 304 Day 2

Basics of Lambda

Turn in your Assignment 0 hand-in sheet

Day 2 estimated schedule

- Announcements & roll call (5 min)
- Your non-Scheme questions (5 minutes)
- Slides, mainly about lambda (5 minutes)
- Running the A1 test code offline (5 minutes)
- Scheme live code demo (25 minutes)
- Your questions about Scheme (10 minutes)
- I set an alarm for 10 minutes before the end of class to make sure that we have that time.
- If you run out of questions, back to live coding.

Questions about the syllabus and course policies?

- Or ask on Piazza, so everyone in all three sections can see the answers.
- Questions about installation, tool use, etc.?

Some scheme questions from beginning of a previous term's class

1. Why is the operator called "cons"?
2. Are Scheme forms that are defined as syntax instead of as procedure unable to be implemented as procedures?
3. Can we use "interval-contains" when we write "interval-intersects"?

Running my test code on your own computer

- I will demonstrate this in SWL (now) and Emacs (when we do the live coding).

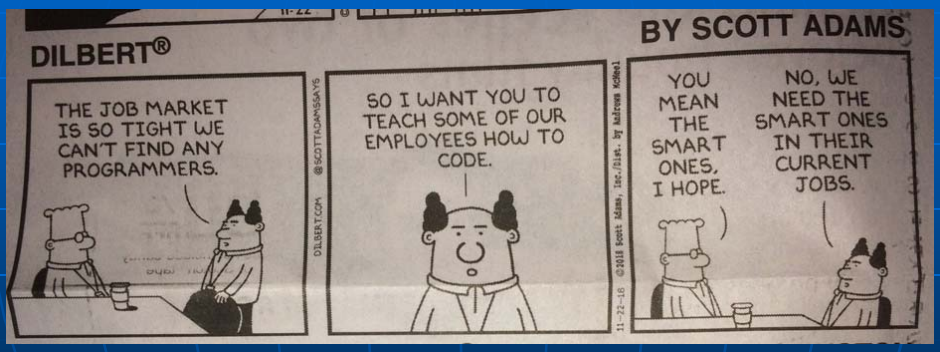
lambda the magnificent

- The title comes from ***The Little Schemer*** by Friedman and Felleisen (a great little book about recursive programming in Scheme).
- **lambda** is **not** a procedure. It is a "special form", syntax that is used to create a procedure.
  - `(define add1`
    `(lambda (n)`
    `(+ n 1)))`
  - `(add1 7)` ➔ 8 ;; here we apply the new procedure
- Note that creating the procedure and giving it a name are two separate operations.
- Maple: `n -> n + 1;`
- Python: `lambda n: n + 1`
- Java 8: `(n)-> n + 1`
- JavaScript: `n => n + 1`
- Alternate Scheme syntax: (use with caution!)
  - `(define (add1 n)`
    `(+ n 1))`

Writing recursive procedures

- A lot like mathematical induction:
  - Start with the base case.
  - Tell how to get from a simpler case to a more complex case.
- `(define list-sum ; sum of elements`
  `(lambda (ls) …`
  - base case? answer for the base case?
  - Other case? how to relate to simpler case?
  - Write the code

Interlude



Interlude

Tom Swifties (most from Wikipedia):

"Who left the toilet seat down?" Tom asked peevishly.

"I'll never again put my arm in a lion's mouth," Tom said off-handedly.

"Can I go looking for the Holy Grail again?" Tom requested.

"I unclogged the drain with a vacuum cleaner," Tom said succinctly.

"We just struck oil!" Tom gushed.

"They had to amputate them both at the ankles," Tom said defeatedly.

"Who discovered radium?" asked Marie curiously.

"Hurry up and get to the back of the ship," Tom said sternly.

"Who put the moss in the bog again?" asked Tom repeatedly.

"A word that contains all five vowels? And I suppose you want those vowels to appear in alphabetical order!?" asked Tom facetiously.

"The robber is coming down the stairs", Tom said condescendingly.

"Nnnn", Tom murmured forensically.

Live demo

- Later… your questions about Scheme

- As many of these as time permits:
  - **cond** vs. **case** vs. **nested if**
  - **vector-sum**
  - **largest-in-list**
  - **make-adder** (not about snake creation!)
  - **positives**
  - **count-reflexive-pairs** (two versions)

Back to the live demo

- Suggestions for after the Live Demo…*:
- Write procedures to
  - Find the median of three numbers
  - Find the largest element in a list.
  - Given a list of numbers ls and a number n, return a new list of numbers where each element is n more than the corresponding element of ls.
  - Count how many times the number n occurs in the list ls.

Answers to your Scheme questions and any other questions

- Simple Scheme questions.
  - At this point, I don't want to intimidate the Scheme newbies (most of the class) by discussing **deep** Scheme questions in class.
  - Ask any question, but if I think it is too advanced for most students, I will defer it.