CS 135

F23 Midterm Help Session

Module 2

Stepping Rules for:
1. Built In Functions
2. User Defined Functions
3. Constants

What are the first, second, and final substitution steps?

```
(define (double-sus x) (* x 2))

(double-sus (+ 3 4))
```

Module 3

- Short Circuit Evaluation: `(and ...)`, `(or ...)`
- Stepping rules for `cond`

How can the following code be improved?

```
;; red-sus?: Sym Sym -> Bool
(define (red-sus? task1 task2)
  (cond
    [(symbol=? task1 'wires) (symbol=? task2 'cardswipe)]
    [(symbol=? task1 'power) (symbol=? task2 'oxygen)]
    [(symbol=? task1 'wires) (symbol=? task2 'asteroids)]
    [else false]))
```

What are the first, second, and final substitution steps?

```
(or (and (= 2 3) true) false)
```

Module 4

- Design Recipe Components
- Testing Code

Module 5

1. What should the function produce in the base case?
2. What should the function do to the first element of a non-empty list?
3. What should applying the function to the rest of the list produce?
4. How should the function combine 2 and 3 to produce the answer?

What are the first, second, and final substitution steps?

```
(define (dup x) (list x x))

(rest (rest (dup (list 'blue 'red))))
```

Module 6

- Data definitions and templates
  - How can I write a template according to the data definition?
- Natural numbers
- Count-down and Count-up

Module 7

- Insertion-sorting
  - How can I alter the insertion-sort functions to change the sorting order?
- ALs and Dictionaries
  - What are the similarities and differences between the two?
- Function that consumes two lists
  - Recursion on only one list
  - Lockstep
  - Working with two lists at different rates
  - A Nat and a List

Good Luck on the Midterm!