Announcements
- Midterm grades are posted on UCWISE
- I need to leave 5 minutes early today – Sorry!

Today
- Why we need Recursion
- Factorial
- Define count with recursion

Recursion
- We want to be able to work with arbitrary length sentences
  - Add up a sentence of numbers
  - Find every odd number in a list
  - Find the number of times a word appears in a sentence

Factorial
- \(10! = 10 \times 9 \times 8 \times 7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1\)
- \(0! = 1! = 1\) (base case)
- \(x! = x \times (x-1)!\) (recursive case)
- (define (factorial0) 1)
- (define (factorial1) 1)
- (define (factorial2) 2)
- (define (factorial3) 6)
- (define (factorial4) 24)
- (define (factorial5) 120)
- (define (factorial6) 720)
- (define (factorial7) 5040)
- (define (factorial8) 40320)
- (define (factorial9) 362880)
- (define (factorial10) 3628800)

The Leap of Faith…
- Dude this seems like a hard problem!
  I’ll do this small piece and hope that someone can do the rest.
- When we say \(85! = 85 \times 84!\)
- Write a definition for (factorial85).
  (define (factorial85) (* 85 (factorial84)))

Factorial REAL Recursion
- \(10! = 10 \times 9!\) (recursive case)
- \(1! = 1\) (base case)
- \(0! = 1\) (base case)

(define (factorial x)
  (if (< x 2)
      1
      (* x (factorial (- x 1)))))

(define (factorial85) (* 85 (factorial84)))
What is (factorial 4)?

(define (factorial x)
  (if (< x 2)
      1
      (* x (factorial (- x 1)))))

(define (count sent)
  (if (empty? sent) ; no more?
      0 ; base case: return 0
      (+ 1
         (count (bf sent))) ; recurse on the rest of sent
  ))

> (count '(a b c))
  (+ 1
     (+ 1
        (+ 1
           (+ 1
              0))))

All Recursive Procedures Need

1. Base Case(s)
   - Where the problem is simple enough to be solved directly
2. Recursive Cases(s)
   1. Divide the Problem (Make the problem Smaller!)
      - into one or more smaller problems
   2. Invoke the function
      - Have it call itself recursively on each smaller part
   3. Combine the solutions
      - Combine each subpart into a solution for the whole