	Class 4: Lambda		
SI 413 - Programming Languages and Implementation			
Dr. Daniel S. Roche			
United States Naval Academy			
	Fall 2011		
Roche (USNA)	SI413 - Class 4	Fall 2011	1 / 8
Procedures are First-Class			

Functional languages generally give procedures *first-class status*:

- They can be given names.
- They can be arguments to procedures.
- They can be returned by procedures.
- They can be stored in data structures (e.g. lists).

Roche (USNA)

Roche (USNA)

SI413 - Class 4

Fall 2011 2 / 8

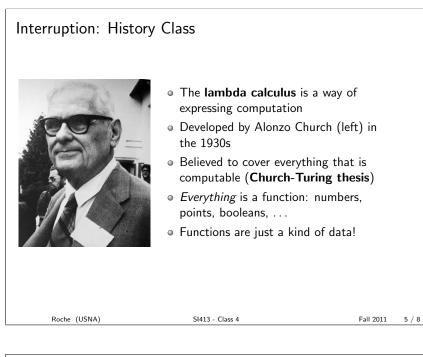
Fall 2011

3 / 8

```
Procedures returning procedures
Example: Get the Java division procedure for a sample input
  (define (java-divider sample)
    (if (inexact? sample) / quotient))
Useful when combined with higher-order procedures:
    (define (java-divide-all tops bottoms)
      (map (java-divider (car tops)) tops bottoms))
```

SI413 - Class 4

Storing procedures in a list Maybe we want to apply different functions to the same data: (define (apply-all alof alon) (if (null? alof) '() (cons ((car alof) alon) (apply-all (cdr alof) alon)))) Then we can get statistics on a list of numbers: (apply-all (list length mean stdev) (list 2.4 5 3.2 3 8)) Roche (USNA) SH13-Class 4 Fall 201 4/8



Anonymous functions in Scheme lambda is a special form in Scheme that creates a nameless function: (lambda (arg1 arg2 ...) expr-using-args)

SI413 - Class 4

Roche (USNA)

Fall 2011

6/8

Lambda with higher-order functions Remember the range function: (define (range a b) (if (> a b) null (cons a (range (+ a 1) b)))) Write the following functions without using recursion. (half L) divides each element in L by 2. (facsum n) gives the sum of all integers less than n that divide n. (g) (my-factorial n) computes n! (g) (my-length L) returns the length of the list L.

Roche (USNA)

SI413 - Class 4

Fall 2011 7 / 8

Behind the curtain
You have already been using lambda!

(define (f x1 x2 ... xn) exp-using-xs)
is the same as

Output: (x1 e1) (x2 e2) ... (xn en)) exp-using-xs)
is the same as
Rote (USM)
SH13-Clas 4
Fil 201 8/8