

6.001: Get Ready!

IAP 2004, By Ben Vandiver

Problems

Exercise 2

Write a procedure that when given a number, determines if it is the square of an integer without using the `sqrt` procedure. You'll most likely want to write a helper procedure that takes the number and "current integer", checks to see if the current integer squared is the number, and continues on based on the result of the check.

```
(is-square? 4)
;Value: 2
(is-square? 7)
;Value: #f
(is-square? 256)
;Value: 16
```

Exercise 3

Write a procedure that when given a number `n`, returns the `n`th prime. Assume that the 0th prime is 2. You'll also want to write a helper procedure for this exercise too.

```
(nth-prime 0)
;Value: 2
(nth-prime 1)
;Value: 3
(nth-prime 2)
;Value: 5
(nth-prime 12)
;Value: 41
```

Lists

- `(list ...)` - builds a list using its arguments
- `(cons a b)` - adds a to the list pointed to by b .
- `(list-ref lst k)` - returns k th element of lst .
- `(car lst)` - returns first element of lst .
- `(cdr lst)` - returns lst without the first element.

Sum-list

Build-list-of-numbers

Delete number

Higher Order Procedures

Example: integrate

```
(define square (lambda (x) (* x x)))  
  
(define rect  
  (lambda (fx0 fx1 dx)  
    (* fx1 dx)))
```

```
(define integrate
  (lambda (start end dx)
    (if (>= start end)
        0
        (+ (rect (square start) (square (+ start dx)) dx)
            (integrate (+ start dx) end dx))))))
```

Lines

Map

Feedback

Year:

Programming Experience:

Favorite Color:

1. How many of the exercises did you do?

- (a) All
- (b) Most
- (c) A couple
- (d) None

2. Were the exercises helpful? If not, what would make them better?

3. Comments on the class? Did the format work? Were there topics that should/should not have been covered?

4. Was the instructor

- (a) Too insane?
- (b) Not rowing with enough oars in the water?
- (c) In need of more marbles?
- (d) Just right?
- (e) Not looney enough?

Comments on teaching style / content delivery?

5. Do you feel more ready for 6.001?