

INSTRUCTIONS

- You have 10 minutes to complete this quiz.
- The exam is closed book, closed notes, closed computer, closed calculator.
- The final score for this quiz will be assigned based on **effort** rather than correctness.
- Mark your answers **on the exam itself**. We will *not* grade answers written on scratch paper.
- For multiple choice questions,
 - means mark **all options** that apply
 - means mark a **single choice**

Last name	
First name	
Student ID number	
CalCentral email (_@berkeley.edu)	
Teaching Assistant	<input type="radio"/> Alex Stennet <input type="radio"/> Kelly Chen <input type="radio"/> Angela Kwon <input type="radio"/> Michael Gibbes <input type="radio"/> Ashley Chien <input type="radio"/> Michelle Hwang <input type="radio"/> Joyce Luong <input type="radio"/> Mitas Ray <input type="radio"/> Karthik Bharathala <input type="radio"/> Rocky Duan <input type="radio"/> Kavi Gupta <input type="radio"/> Samantha Wong
Name of the person to your left	
Name of the person to your right	
<i>All the work on this exam is my own.</i> (please sign)	

1. (5 points) Interpretive Dance

(a) (2 pt) Assume the following definition has been loaded into the Scheme interpreter.

```
(define (sum-of-squares x y z)
  (+ (* x x) (* y y) (* z z)))
```

Given the following Scheme expressions, mark the number of calls made to `scheme_eval` and `scheme_apply`:

i. `(+ 5 (* 3 7 3))`

`scheme_eval` 3 4 5 6 7 8 9

`scheme_apply` 1 2 3 4 5 6

ii. `(sum-of-squares 3 4 5)`

`scheme_eval` 4 5 8 10 14 19 24 25

`scheme_apply` 1 2 3 4 5 6

(b) (3 pt) For each of the following Scheme expressions, mark the choice corresponding to the correct Pair representation that `scheme_read` would create.

i. `(func (4 5) 3)`

`Pair('func', Pair(4, Pair(5, Pair(3))))`

`Pair('func', Pair(4, Pair(5, Pair(3, nil))))`

`Pair('func', Pair(Pair(4, 5), Pair(3, nil)))`

`Pair('func', Pair(Pair(4, Pair(5, nil)), Pair(3, nil)))`

Attempting to `scheme_read` the above expression would result in a syntax error.

ii. `'(1 2 (3))`

`Pair(1, Pair(2, Pair(3, nil)))`

`Pair(1, Pair(2, Pair(Pair(3, nil), nil)))`

`Pair('quote', Pair(1, Pair(2, Pair(Pair(3, nil), nil))))`

`Pair('quote', Pair(Pair(1, Pair(2, Pair(Pair(3, nil), nil))), nil))`

Attempting to `scheme_read` the above expression would result in a syntax error.

iii. `(cdr () 'cdr)`

`Pair('cdr', Pair(nil, Pair('cdr', nil)))`

`Pair('cdr', Pair(Pair(nil, Pair('cdr', nil))))`

`Pair('cdr', Pair(nil, Pair(Pair('quote', Pair('cdr', nil)), nil)))`

`Pair('cdr', Pair(Pair(nil, Pair(Pair('quote', Pair('cdr', nil))))))`

Attempting to `scheme_read` the above expression would result in a syntax error.