UNIT 3 FINAL

1. (4 points) Object-oriented Programming

Create a class cs61as that has the following characteristics:

- CS61AS is initialized with the semester season and year
- A method request-discussion that takes in a lesson number. You may assume there are only lessons 1 through 14. A discussion may only be led once it has been requested.
- A method lead-discussion that takes in the name of a TA and a lesson number.
  - If the discussion has not been requested yet or if it is not a TA requesting to lead a discussion, return an error message saying “you are not a TA or No one is here for your discussion :(
  - Otherwise, return the sentence '(Discussion has been led!)
  - Once a discussion has been led, it cannot be led again until someone requests it a second time.
  - Someone who is a TA for one semester is a TA for ANY SEMESTER and is eligible to lead a discussion.
- Two methods, add-readers and add-TA, that will allow you to add Readers and TAs to the object.
  - If a TA has TAed in a previous semester, their name should only be on the TA list one time.
  - TAs should be added one at a time.
  - Readers should be added all at once (input is a list).

Hint: for request-discussion and lead-discussion you’ll need to keep track of which discussions have been requested and which ones have been led.

STk> (define sp14 (instantiate cs61as 'spring 2014))
STk> (define fa14 (instantiate cs61as 'fa 2014))
STk> (ask fa14 'add-readers '(Chenning Judy Reia Yuan))
STk> (ask sp14 'add-readers '(Chenning Judy Reia Alicia))
STk> (ask sp14 'add-TA 'Marion)
STk> (ask sp14 'add-TA 'Sam)
STk> (ask sp14 'add-TA 'Jisoo)
STk> (ask sp14 'add-TA 'Rohin)
STk> (ask fa14 'add-TA 'Vivek)
STk> (ask fa14 'add-TA 'Alicia)
STk> (ask sp14 'add-TA 'Sam)
STk> (ask fa14 'readers)
(chenning judy reia yuan)
STk> (ask sp14 'TAs)
(alicia vivek marion jisoo sam rohin)
STk> (ask sp14 'lead-discussion 'Sam 14)
Error: You are not a TA or no one is here for your discussion
STk> (ask sp14 'request-discussion 13)
STk> (ask sp14 'lead-discussion 'yuan 13)
Error: You are not a TA or no one is here for your discussion
STk> (ask sp14 'lead-discussion 'rohin 13)
(Discussion has been led!)
STk> (ask sp14 'lead-discussion 'rohin 13)
Error: You are not a TA or no one is here for your discussion

Write your code on the next page
2. (4 points) Merging vectors

We're going to write a procedure \texttt{merge}, which takes in two sorted vectors of numbers as arguments and returns a sorted vector whose elements are the same as the argument vectors. The argument vectors can have different lengths.

Do NOT use \texttt{list->vector} or \texttt{vector->list}

```
STk> (merge (vector 1 3 5) (vector 2 4 6))
(1 2 3 4 5 6)
STk> (merge (vector 1) (vector 2 3 4))
(1 2 3 4)
STk> (merge (vector 2 3 4) (vector 1))
(1 2 3 4)
```
3. (4 points) Saving the environment

Draw the environment diagram for the following lines of code, and write what the last line of code returns.

```
(define mystery
  (let ((b 5))
    (define (foo a)
      (* a b))
    (lambda (b)
      (foo 4))))
STk> (mystery 6)

---------
```