**Piazza questions and answers from previous terms**

 list? vs. pair?

Recall that a pair is simply a container for two values; the simplest way to make one is to apply cons.

A list is a linked list of pairs.  Each pair except the last one is a reference to the next pair in the list; the cdr of the last pair must be null, otherwise the list is improper.

pair? is a constant-time procedure that simply asks, "is this value a reference to a pair?"

list? is a linear-time operation that asks, "is this value either null or a reference to the first pair of a proper list?

So efficiency is one basis to choose between the two tests.

I hope that the following transcript will help you better understand these procedures.

> (list? '())

#t

> (pair? '())

#f

> (list? '(a b c))

#t

> (pair? '(a b c))

#t

> (list? '(a b . c))

#f

> (pair? '(a b .c))

#t

> (pair? 'a)

#f

> (list? 'a)

#f

# duplication in lists

Can you ever do better than O(n^2) for determining the duplication of numbers in a list?

A novel approach might be for each index compare this value to all other index values.

Another way is to pre-process with a sorting algorithm with Big O lower than O(n^2). Then compare each value next to each other. This is O(n) + complexity of sort.

However sorting isn't an easy option in this case since we can't mutate anything yet.

So are we left to O(n^2) efficiency for now until we learn more about manipulation in scheme?

**the instructors' answer,**

*where instructors collectively construct a single answer*

[Actions](https://piazza.com/class/jl863803n0a6tl?cid=14)

If you don't use any auxiliary data structure, N^2 is the best you can do.  If you copy the numbers into a binary heap or balanced tree you can do O(N log N) worst case.  With a hash table you can have O(N) expected value, but O(N^2) worst case.

Finally, I don’t think Scheme’s sort procedure mutates anything.

# Operations with lists

I had no problem using - and + between a number and a list, but I do if I use it with two lists, even if there is only one element on them. Are there special operators for adding and subtracting the elements in a list? Or is it not possible to operate with them being in a list?

Thank you

[hw1](https://piazza.com/class/jl863803n0a6tl?cid=19)

This private post is only visible to Instructors and [2 others](https://piazza.com/class/jl863803n0a6tl?cid=19)

[**edit**](https://piazza.com/class/jl863803n0a6tl?cid=19)·[good question](https://piazza.com/class/jl863803n0a6tl?cid=19)0

Updated 2 months ago by

Ana Huerta

**the instructors' answer,**

*where instructors collectively construct a single answer*

[Actions](https://piazza.com/class/jl863803n0a6tl?cid=19)

Depending on what you are trying to do, **apply** may be what you need. Or (for things like A2-A5) you can use a loop or (for A1 where all lists have fixed length) you can just write two or three - or + expressions.

# Comparing numeric quantities

If x and y are known to be numbers, use (= x y) instead of (equal? x y) to compare them.