The purpose of this worksheet is to give you practice programming in Scheme. Please put all of your code in a file **Worksheet2-*yourInitials.rk***t.

1. Write a function that increments each element of LIS by one, and return that as a new list
2. Write a function that sums the numbers from 1 to n
3. Write quicksort
4. Write a function that sums all the elements in a list
5. Write a function that counts all the atoms in a list where the list can have nested lists (e.g., '(1 (2 (3)) (5)) has 4 atoms in it, so that would be the return value)
6. Write a function that takes two lists and returns -1 if both lists are empty and returns the length of the longest list otherwise.
7. Write a function that swaps the first two elements in a list.
8. Then try to make each function tail-recursive and prove it to yourself with trace. If you can't make something tail recursive, why not?