

## Tail Recursion Problems.

---

All of the following functions should be implemented in a tail recursive way.

1. Write a function that takes a (flat) list as an argument and returns a count of the number of times the symbol 'spam' occurs.

2. Write a function which takes a list of numbers as an argument & returns a count of the number of odd numbers and the number of even numbers. Return them in a cons-pair.

```
(odd-fn '(5 2 7 4 1)) => (3 . 2)
```

3. The hilo function from the lecture trace found the highest & lowest values in a flat list of numbers. Check out the Lisp functions called "max" and "min" and write a more concise version of hilo.

4. Write a tail recursive version of sum-list calculate the sum of all numbers in a nested list/tree (you can assume that the tree only contains numbers – no symbols).

5. Write a version of hilo that works on a tree.