CSE 311: Foundations of Computing I

QuickCheck: Structural Induction Solutions

0. SSttrruuccttuurraall Induction

Recall the recursive definition of a list:

$$\textbf{List} = [] \mid \text{Integer} :: \textbf{List}$$

And the definition of "len" on lists:

$$\begin{aligned} \text{len}([]) &= 0 \\ \text{len}(x :: L) &= 1 + \text{len}(L) \end{aligned}$$

Consider the following recursive definition:

$$\begin{aligned} \text{stutter}([]) &= [] \\ \text{stutter}(x :: L) &= x :: x :: \text{stutter}(L) \end{aligned}$$

Prove that $\text{len}(\text{stutter}(L)) = 2\text{len}(L)$ for all Lists L.

Solution:

We go by structural induction. Let L be a list.

Case $L = []$. Note that $\text{len}(\text{stutter}([])) = \text{len}([]) = 0 = 2\text{len}([])$.

Case $L = x :: L'$. Suppose that $\text{len}(\text{stutter}(L')) = 2\text{len}(L')$ for some list L'. Note that:

$$\begin{aligned} \text{len}(\text{stutter}(x :: L')) &= \text{len}(x :: x :: \text{stutter}(L')) && \text{[Definition of stutter]} \\ &= 1 + \text{len}(x :: \text{stutter}(L')) && \text{[Definition of len]} \\ &= 1 + 1 + \text{len}(\text{stutter}(L')) && \text{[Definition of len]} \\ &= 2 + 2\text{len}(L') && \text{[By IH]} \\ &= 2(1 + \text{len}(L')) && \text{[Distributivity]} \\ &= 2(\text{len}(x :: L')) && \text{[Definition of len]} \end{aligned}$$

Thus, the claim is true for all Lists by structural induction.