Post-Mortem

CS135 Winter 2024, Assignment 6

Question 1

Q1b: spread/acc

—— Many students seemed like they didn't know that they can use two accumulators for spread/acc. One to keep track of the max and one to keep track of the min. The base case is just returning the difference between the max and the min accumulators.

Q1c: smallest-first and smallest-first/acc

—— smallest-first/acc holds two accumulators; a min and the new list. As you traverse through the original list, update the min and list accordingly. Students had issues with the first element appearing as a duplicate in the list. The way to deal with is in smallest-first, you make the following call: `(smallest-first/acc (rest ne-lon) (first ne-lon) empty)`

Question 2

Q2c: build-inventory

—— Many students didn't include the requires as a part of their contract. That is, "Requires: All lists are of the same length"

Q2d: sort-evs

—— Use a sorting algorithm like insertion sort or a modified version of the selection sort algorithm from a previous question. We can compare two ev's using `compare-ev` from the previous question and sort accordingly.