Post-Mortem

CS135 Winter 2025, Midterm

Question 1

—— The most common error was answering `true` for part (e).

Question 2

—— A common error was using list abbreviations, as they are disallowed for this question.

—— For part (a), a common error was answering `(cons 'green empty)`.

Question 3

—— One common error was not naming the function correctly.

—— Another was trying a recursive approach when only a mathematical expression is needed.

Question 4

—— Some students did not correctly implement checking if a number is odd, such as thinking `remainder` only consumes one argument.

—— Some solutions were missing the checks for being odd or being positive.

—— Some implementations used `odd?`, which is not an allowed function.

—— Some solutions did not correctly use short-circuit evaluation.

Question 5

—— A common error was using `list` or `append` instead of `cons`.

—— Some solutions had an incorrect base case.

—— Some solutions did not correctly check for primary colors, e.g., using `=` instead of `symbol=?`.

Question 6

—— Many solutions had incorrect base cases.

—— Some solutions incorrectly utilized `append`.

—— Some solutions produced a reversed answer.

Question 7

—— The most common errors were getting parts (d) and (e) wrong.

Question 8

—— Many solutions had an incorrect base case.

—— Some solutions had incorrect recursive calls.

—— Some solutions did not use the provided helper function, `in?`.

Question 9

—— A common error was having an additional `cons` before the call to `zipper`, producing a list of lists.

—— Some solutions incorrectly used `cons` or `append`.

—— Some solutions added items to the zippered list backwards.

—— Some solutions made incorrect recursive calls, leading to non-terminating programs.

Question 10

—— Many solutions incorrectly called `min` and `max` on lists.

—— Some solutions incorrectly initialized accumulators with static values like `0` instead of `(first lst)` or such.

—— Some solutions used `append` instead of `cons`.

—— Some solutions only checked between the first two elements of a given list for the minimum and maximum elements.