University of Waterloo
CS240, Spring 2021
Assignment 2 Post Mortem

In general, many students left out details regarding justification of run-time, justification or correctness, etc. on Problems 1, 2, 3, and 5.

Problem 1 [4+6=10 marks]

- For part b), some students used a linear search on the ancestors of the node at index i.

Problem 2 [3+3+5+4=15 marks]

- For part b), students incorrectly interpreted the partition as running in $O(n)$ time, resulting in a final runtime of $O(n^2)$.

Problem 3 [8 marks]

- Many students didn't provide the sorting algorithm or prove its correctness. Some others didn't find a way to sort the almost-sorted array in $O(n)$ using swaps.

Problem 4 [3+3+5=11 marks]

- For a), many students had $O(1)$ for the best-case running time, which doesn't specify a lower bound (and vice versa for $\Omega(1)$).

Problem 5 [6 marks]

- Some students did not define their R when conducting radix sort, claiming that $O(mnR) \in O(n)$ (for MSD-radix-sort).

- Students used MSD-radix-sort, which has a slower run-time than LSD-radix-sort for this problem.

- We didn't take off points for this, but some students claimed that in base n, our areas could have up to 3 digits. The upper bound for the areas is $(n-1)^2$, not n^2

Problem 6 [6 marks]

- Many students forgot to put a ceiling function around the final run-time. This is required for an exact lower bound.

- Some students did not use a decision tree but came up with an algorithm to find i and j by iterating through the loop. Unfortunately, this gets a lower bound of O(n) comparisons, which is not efficient enough.