**Challenge Problems 4**
due 11/2/2022 at 11:59pm in Gradescope

**Instructions.** Limited collaboration is allowed while solving problems, but you must write solutions yourself. List collaborators on your submission.

You can choose which problems to complete, but must submit at least one problem per assignment. See the course page for information about how challenge problems are graded and contribute to your homework grade. Since you don’t need to complete every problem, you are encouraged to focus your efforts on producing high-quality solutions to the problems you feel confident about. There is no benefit to guessing or writing vague answers.

If you are asked to design an algorithm, please (a) give a precise description of your algorithm using either pseudocode or language, (b) explain the intuition of the algorithm, (c) justify the correctness of the algorithm; give a proof if needed, (d) state the running time of your algorithm, (e) justify the running-time analysis.

**Submissions.** Please submit a PDF file. You may submit a scanned handwritten document, but a typed submission is preferred. Please assign pages to questions in Gradescope.

1 **Challenge Problems**

**Problem 1.** Array medians. There are two sorted arrays $A$ and $B$ of $n$ numbers. You want to find the median of the overall set of $2n$ numbers, which we will define as the $n$th smallest value. Give an algorithm that finds the median in $O(\log n)$ time.

**Problem 2.** Butterflies (Again?). You have collected $n$ butterfly specimens of different species. You want to know if there is a single species that accounts for more than half of the specimens. You don’t know butterflies well enough to name the species of any single specimen, but you can carefully compare any two specimens and judge (correctly) if they are from the same species or not. Design an algorithm that returns “true” if there are more than $n/2$ specimens from one species, and returns “false” otherwise, using only $O(n \log n)$ pairwise comparisons.

**Problem 3.** Shelving books. Books numbered 1 to $n$ are to be shelved in order on consecutive bookshelves at the library. Book $i$ has height $h_i$. Each shelf can hold up to $L$ books (but you are allowed to place fewer) and its height can be adjusted to accommodate the tallest book placed on it. However, you want to keep the shelf heights as small as possible to use space efficiently — give an algorithm to find the smallest sum of shelf heights required to shelve all $n$ books.

**Problem 4.** Chicken Wings. The image in Figure 1 is a real restaurant menu. The goal of this problem is to find the cheapest way to buy $V$ chicken wings for some integer $V$ given a menu like this one. Assume the menu is given as a list of $n$ menu items $(v_1, w_1), (v_2, w_2), \ldots, (v_n, w_n)$, where $v_i$ is the price to buy $v_i$ wings and $v_i$ and $w_i$ are both integers. In the example, assuming costs are computed in cents, we would have:

\[ (v_1, w_1) = (4, 455) \]
\[ (v_2, w_2) = (5, 570) \]
\[ (v_3, w_3) = (6, 680) \]
\[ \ldots \]

You are allowed to choose any combination of orders whose quantities add up to $V$, including ordering the same quantity multiple times. You can assume there is always some combination of orders to buy exactly $V$ wings.

(a) There is a natural greedy algorithm where you first buy the largest quantity $v_i$ such that $v_i \leq V$, and
then repeat on the remaining $V - v_i$ wings. Show that this algorithm is not optimal for the menu shown in Figure 1.

(b) Write a dynamic programming algorithm to find the cost of the cheapest set of orders to buy exactly $V$ wings.

(c) Modify your algorithm to also return the set of orders you could make to achieve the smallest cost.