CS381 Homework 1 Problem 3

Connor Couetil

January 29, 2026

1 Exercise 4.5.1

unique-sums(X, Y): Given two sets of integers, X and Y, each containing n distinct integers, and returns a set of unique sums from X and Y.

```
1. If $|X|$ or $|Y|$ is zero, return $\emptyset$
2. Let $A$ be an array of size $|X| \cdot |Y|$
3. for $x_i \in X : i = 1, ..., |X|$
   A. for $y_i \in Y : i = 1, ..., |Y|$
      1. $A[|Y| \cdot i + j] = x_i + y_j$
4. Return unique(merge-sort($A$))
```

unique(A): Given an array A of ordered integers, returns an array of only the unique integers in A

```
1. If $|A|$ is zero or 1, Return $A$
2. Let $n = |A|$
3. If $A[0] \neq A[1]$
   A. Return $A[0] +$ unique($A[1, ..., n]$)
4. Return unique($A[1, ..., n]$)
```

I'll begin by discussing the time complexity of unique(A). At each iteration, it reduces the size of the array A by one and terminates when the array's size is zero or one. Thus, its time complexity is proportional to the size of the array and is $O(n)$. It's proof of correctness is simple, if the beginning of a sorted array is not equal to the next element, it must be unique. If it wasn't unique, then a repeated element later in the array would necessarily be out of order with the next element, having returned to the initial value, contradicting its sorted property. The recurrence relation looks like

$$T(1) = 1$$
$$T(n) = T(n-1) + 1$$

Let's examine unique-sums(X, Y). We begin with a nested iteration of both arrays, producing an array of all possible sums in $O(n^2)$ time. We then sort the $n \times n$ array using merge-sort, taking $O(n^2\, log(n^2))$ time (it simplifies to $O(2 \cdot n^2\, log(n))$ then we will disregard the constant in our analysis) We then call unique, taking $O(n)$ time. There is no recursion in unique-sums, thus the $O(n^2\, log(n))$ dominates the complexity and is our worst-case runtime for the algorithm.

2 Exercise 4.5.2

unique-sums(X, Y): Given two sets of integers, X and Y, each containing n distinct integers whose values are between 0 and some fixed natural number M, returns a set of the unique sums from X and Y.

```
Let sets $A$ and $B$ be of size $n$
Let set $D$ be of size $2 \cdot M$
$A =$ merge-sort($X$)
$B =$ merge-sort($Y$)
Return something
```

I don't know

3 Exercise 4.5.3

I don't know