CSCI 3104 Spring 2023
Instructors: Ryan Layer and Chandra Kanth Nagesh

Homework 18

Due Date ..... March 23, 2023
Name ..... **Blake Raphael**
Student ID **109752312**
Collaborators ..... **Alex Barry, Brody Cyphers, and Ben Kohav**

Contents

1 Instructions

- The solutions **should be typed**, using proper mathematical notation. We cannot accept hand-written solutions. Here's a short intro to LaTeX.
- You should submit your work through the **class Gradescope page** only (linked from Canvas). Please submit one PDF file, compiled using this LaTeX template.
- You may not need a full page for your solutions; pagebreaks are there to help Gradescope automatically find where each problem is. Even if you do not attempt every problem, please submit this document with no fewer pages than the blank template (or Gradescope has issues with it).
- You are welcome and encouraged to collaborate with your classmates, as well as consult outside resources. You must **cite your sources in this document. Copying from any source is an Honor Code violation. Furthermore, all submissions must be in your own words and reflect your understanding of the material.** If there is any confusion about this policy, it is your responsibility to clarify before the due date.
- Posting to **any** service including, but not limited to Chegg, Reddit, StackExchange, etc., for help on an assignment is a violation of the Honor Code.
- You **must** virtually sign the Honor Code (see Section 2). Failure to do so will result in your assignment not being graded.

2 Honor Code (Make Sure to Virtually Sign)

- My submission is in my own words and reflects my understanding of the material.
- Any collaborations and external sources have been clearly cited in this document.
- I have not posted to external services including, but not limited to Chegg, Reddit, StackExchange, etc.
- I have neither copied nor provided others solutions they can copy.

Agreed (I agree to the above, Blake Raphael). □

3 Standard 18 - Divide and Conquer: Counterexample (4 points)

Problem 1. Consider the following problem:

> Min Pair Difference
> *Input:* A list L of integers
> *Output:* An integer that corresponds to the minimum difference among all differences between adjacent pairs (i.e. $L[i] - L[i+1]$ for $i \in 1, \ldots, len(L) - 1$) in the list.

(Note the list here is 1-indexed, so the problem simply does not consider the last element, as it has nothing to pair it with.)

Consider the algorithm below that attempts to solve this problem. **Give an instance** of input (A list of length at most 6) for which it fails to output the correct value for the above problem, **explain why it fails** and **draw the recurrence tree that the problem generates**.

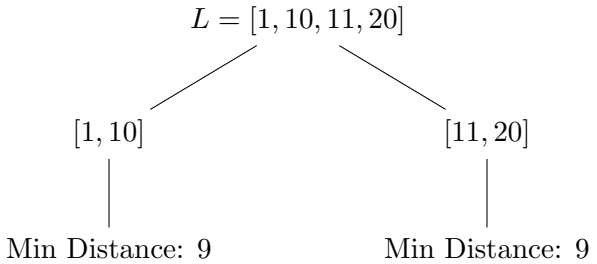
Algorithm 1 Proposed divide-and-conquer algorithm for the Min Pair Difference problem

```
1: procedure MinPairDiff(List L) n ← len(L)
2:     if n ≤ 1 then return ;
3:     if n = 2 then return (|L[1] − L[2]|);
4:     (diff1) ← MinPairDiff(L[1..⌊n/2⌋]);
5:     (diff2) ← MinPairDiff(L[⌊n/2⌋ + 1..n]);
6:     if diff1 ≤ diff2 then
7:         return (diff1);
8:     else
9:         return (diff2);
```

Proof. Consider the following list: $L = [1, 10, 11, 20]$

This is an input that fails because the algorithm will return 9 instead of the real minimum distance of 1, which is $11 - 10$.

The following tree shows the recurrence tree that this problem generates:



□