CSCI 230 Fall 2023
Instructor: Michael Levet

Homework 7

Name .... **Oscar Jiang**
Student ID **20231398**
Collaborators ....

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 OAKS page** only. 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 with grading. Even if you do not attempt every problem, please submit this document with no fewer pages than the blank template.
- 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 the Honor Code)

Problem 1. On my honor, my submission reflects the following:

- My submission is in my own words and reflects my understanding of the syllabus.
- 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.

In the specified region below, clearly indicate that you have upheld the Honor Code. Then type your name.

Honor Code Agreement. Oscar Jiang □

3 Problem 1: One Trivial Method

Problem 2.

The trivial method I will explain is the add method. This method will add an element to the minHeap iteratively using an algorithm. In the beginning of the method, it checks whether or not the size of the internal list is the same as the size variable. If it is, it will resize and double its space. Either way, it passes to the next segment in which a condition will be checked. If the first space, being index 1 in this list, is not empty, then that means the element to be added will have something to compare to. Thus, starting the process of adding it to the list. For this process, 2 variables are instantiated: the index elem will be placed in(rootSpot) and a temporary variable(temp) used for swapping later. Then, the loop starts after elem is placed in rootSpot first– just in case there are no swaps. The loop will run $log_2(rootSpot)$ times with each iteration comparing elem to whatever is at index rootSpot/2. Integer division here should allow both elements at 2i and 2i+1 compare itself to i. If elem is smaller than the element at i, then they will swap, utilizing the temp variable. Should this happen, rootSpot will be divided by 2, allowing the next iteration to compare elem to the next parent and so forth. However, If elem is greater than the element at i, and assuming the minHeap property is respected(which should always be the case), then the loop will end before $log_2(rootSpot)$ iterations. Else, the first index of the list will be set to elem. Finally, size will be increment by 1.