DS2030 Data Structures and Algorithms for Data Science
Lab 2 (In Person) Due on September 10, 5.00pm

Instructions

- You are to use Python as the programming language. Use may use Visual Studio Code (or any other editor you are comfortable with) as the IDE.
- You have to work individually for this lab.
- You are not allowed to share code with your classmates nor allowed to use code from the internet. You are encouraged engage in high level discussions with your classmates; however ensure to include their names in the report/code documentation. If you refer to any source on the Internet, include the corresponding citation in the report/code documentation. If we find that you have copied code from your classmate or from the Internet, you will get a straight fail grade in the course.
- The submission must be a zip file with the following naming convention - rollnumber.zip.
- Include appropriate comments to document the code. Include a `read me` file containing the instructions on for executing the code. The code should run on institute linux machines.
- Upload your submission to moodle by the due date and time. Do not email the submission to the instructor or the TA.

1 Introduction

The purpose of this lab is to implement an in-place HeapSort algorithm in Python. HeapSort is a comparison-based sorting algorithm that uses a binary heap data structure to sort elements. You will implement the algorithm on an array of integers, and you will also have the option to output each step of the sort.

2 Background

HeapSort first transforms an array into a binary heap (a complete binary tree) and then repeatedly extracts the largest (or smallest) element from the heap, placing it at the end of the array. It runs in $O(n \log n)$ time and is in-place, meaning it uses a constant amount of extra memory. The steps of HeapSort are:

1. Heapify the Array: Transform the array into a max-heap.
2. Sort the Array
   (a) Repeatedly swap the root of the heap (maximum element) with the last element of the array.
   (b) Decrease the size of the heap and heapify the root.

3 Input and Output

The input is an array of integers. If the `verbose` flag is set to `True`, the program will print each step of the sorting process:

- The heapification steps.

- Each swap during the sorting phase.
- The final sorted array.

4 References

1. M Goodrich, R Tamassia, and M. Goldwasser, "Data Structures and Algorithms in Python", 1^{st} edition, Wiley, 2013.