Assignment 2 – Heaps (Pair 4)

Student (B): Shakalova Diana
Partner (A): Nurzhaina Kuralbay
Group: SE-2431
Course: Algorithms and Data Structures
Date: 05 October 2025

1. Purpose

The purpose of this assignment was to analyze and evaluate my partner’s algorithm — MinHeap. Each student in the pair implemented one heap (MinHeap or MaxHeap), tested both, and compared theoretical and practical results.

2. Tools and Environment

Programming Language: Java SE 21
IDE: IntelliJ IDEA 2024.3.1
Build Tool: Maven
OS: Windows 10
Terminal: PowerShell
Version Control: Git + GitHub
Report Generation: Pandoc + Chrome (headless)
Performance Analysis: BenchmarkRunner + PerformanceTracker

3. Description of Partner’s Algorithm (MinHeap)

MinHeap is a binary heap where each node satisfies $a[\text{parent}(i)] \le a[i]$, which means the smallest element is always at the root (index 0). The algorithm supports operations such as insert, peekMin, extractMin, decreaseKey, and buildFrom. Theoretical complexities: peek O(1), insert/extract O(log n), buildFrom Θ(n).

4. Work Process

1. Cloned the GitHub repository and checked structure (src, docs, pom.xml).
2. Reviewed partner’s MinHeap.java for heap logic and correctness.
3. Built the project with Maven using 'mvn -q -DskipTests=true package'.
4. Ran benchmark for MinHeap with data sizes 100–100000 and exported results to docs/performance-plots/heaps.csv.
5. Generated performance graphs and analyzed execution times.

5. Results

n = 100 → 3.8×10■ ns
n = 1,000 → 5.7×10■ ns
n = 10,000 → 9.2×10■ ns
n = 100,000 → 1.24×10^{1}■ ns
The time increased proportionally to n·log(n), confirming theoretical behavior.

6. Observations

The algorithm correctly maintains the heap property, with performance results matching theoretical expectations. The PerformanceTracker accurately measures operations. Possible improvements include adding an index map for faster decreaseKey(), adding helper methods like clear() and contains(), and extending documentation.

7. Conclusion

The MinHeap algorithm implemented by Nurzhaina Kuralbay is correct, stable, and efficient. Theoretical and experimental results match perfectly. The project was built, tested, and documented successfully.