Lab W1D2 — Question 1: Comparing Algorithms (Third Largest)

Problem: Given an array A ($n \geq 3$), return the third largest value. Duplicates are allowed (e.g., 20,20,20 $\rightarrow$ 3rd = 20).

Algorithm 1 — Three Passes ($\Theta(n)$)
Idea: Pass 1 gets max1; pass 2 gets max2 ≤ max1; pass 3 gets max3 ≤ max2. Time $\Theta(n)$, Space O(1).

Algorithm 2 — One Pass (max, preMax, prePreMax) ($\Theta(n)$)
Idea: Maintain top-3 in one scan; use ≥ to handle duplicates. Time $\Theta(n)$, Space O(1).
Correctness: loop invariant ensures (max, preMax, prePreMax) are top-3 of the prefix.

Algorithm 3 — Ordered Dictionary (TreeMap) (O(n log n))
Idea: Build frequency map (sorted), then walk keys descending until 3rd element. Time O(n log n), Space O(u) ($u \leq n$).

Algorithm	Best	Average	Worst	Space
Three Passes	$\Theta(n)$	$\Theta(n)$	$\Theta(n)$	O(1)
One Pass	$\Theta(n)$	$\Theta(n)$	$\Theta(n)$	O(1)
TreeMap	O(n log n)	O(n log n)	O(n log n)	O(n)

Empirical Results: OnePass and ThreePasses scale linearly ($\Theta(n)$); TreeMap grows faster (O(n log n)).
As n increases, TreeMap visibly separates from linear curves, matching theory.

Recommendation: Use Algorithm 2 (one pass) for efficiency and simplicity. Algorithm 3 only if ranked queries are needed.

Chart: