Week 5 Study Summary: Recursion, Merge Sort, Quicksort, and Master Theorem

Z5007 Programming and Data Structures · Nov 2–6, 2026 · Dr. Innocent Nyalala

1. Divide-and-Conquer Sorting and Quickselect

- **Merge Sort:** Splits evenly, recurses, and merges in $\Theta(n \log n)$ worst-case time and $O(n)$ auxiliary space. Counting `len(left) - i` whenever `right[j]` $<$ `left[i]` counts inversions in $O(n \log n)$.
- **Randomized Quicksort:** Partitions in-place around a random pivot in $\Theta(n \log n)$ expected time and $O(\log n)$ expected stack depth, avoiding the $O(n^2)$ sorted-input trap.
- **Quickselect:** Recurses on only one side of the partition, solving $T(n) = T(n/2) + \Theta(n) = \Theta(n)$ expected time for k-th order statistics.

2. The Master Theorem ($T(n) = a\,T(n/b) + f(n)$)

Compare $f(n)$ against the watershed $n^{\log_b a}$ ($a \geq 1, b > 1$):

Case	Condition	Asymptotic Bound
Case 1 (Leaves win)	$f(n) = O(n^{\log_b a - \epsilon})$	$T(n) = \Theta(n^{\log_b a})$
Case 2 (Balanced)	$f(n) = \Theta(n^{\log_b a} \log^k n)$	$T(n) = \Theta(n^{\log_b a} \log^{k+1} n)$
Case 3 (Root wins)	$f(n) = \Omega(n^{\log_b a + \epsilon})$ + regularity	$T(n) = \Theta(f(n))$