Divide & Conquer — Core (Quick Guide)

1) Count Inversions (Merge Sort)

- Split, count in left + right + cross inversions during merge.
- Time: O(n log n)

2) Maximum Subarray (Divide & Conquer)

- Combine four quantities: total sum, best prefix, best suffix, best overall.
- Conquer: best = max(left.best, right.best, left.bestSuffix + right.bestPrefix)
- Time: O(n)

3) Majority Element (Boyer–Moore)

- Single pass candidate/vote.
- Time: O(n), Space: O(1)
- If majority may not exist, verify candidate count at the end.

Tips

- Prefer divide & conquer when merging results from subproblems is efficient.
- Track minimal sufficient state (sum/pref/suf/best) to combine in O(1).
- For inversions, the key is counting how many left elements remain when picking a right element.