

[Back to Chapter](#)

Resources

# A Cheatsheets

[Report Issue \(https://github.com/LeetCode-Feedback/LeetCode-Feedback/issues\)](https://github.com/LeetCode-Feedback/LeetCode-Feedback/issues)☐ A Code templates☐ A Stages of an interview  
You will find:☒ A Cheatsheets

- Time complexity (Big O) cheat sheet
- General DS/A flowchart (when to use each DS/A)
- Stages of an interview cheat sheet

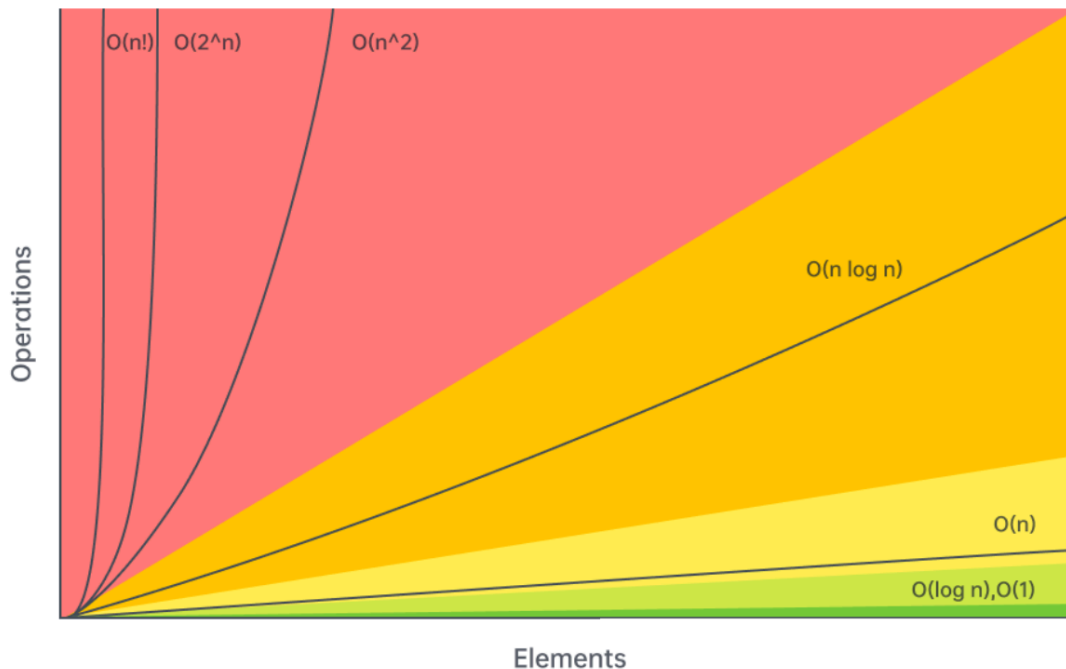
[Discuss](#)

22 topics

[\(/discuss/explore/cheatsheets\)](#)

## Time complexity (Big O) cheat sheet

### Big-O Complexity Chart



First, let's talk about the time complexity of common operations, split by data structure/algorithm. Then, we'll talk about reasonable complexities given input sizes.

#### Arrays (dynamic array/list)

Given `n = arr.length`,

- Add or remove element at the end:  $O(1)$  amortized (<https://stackoverflow.com/questions/33044883/why-is-the-time-complexity-of-pythons-list-append-method-o1>)
- Add or remove element from arbitrary index:  $O(n)$
- Access or modify element at arbitrary index:  $O(1)$
- Check if element exists:  $O(n)$
- Two pointers:  $O(n \cdot k)$ , where  $k$  is the work done at each iteration, includes sliding window
- Building a prefix sum:  $O(n)$
- Finding the sum of a subarray given a prefix sum:  $O(1)$













