Complexity and Performance

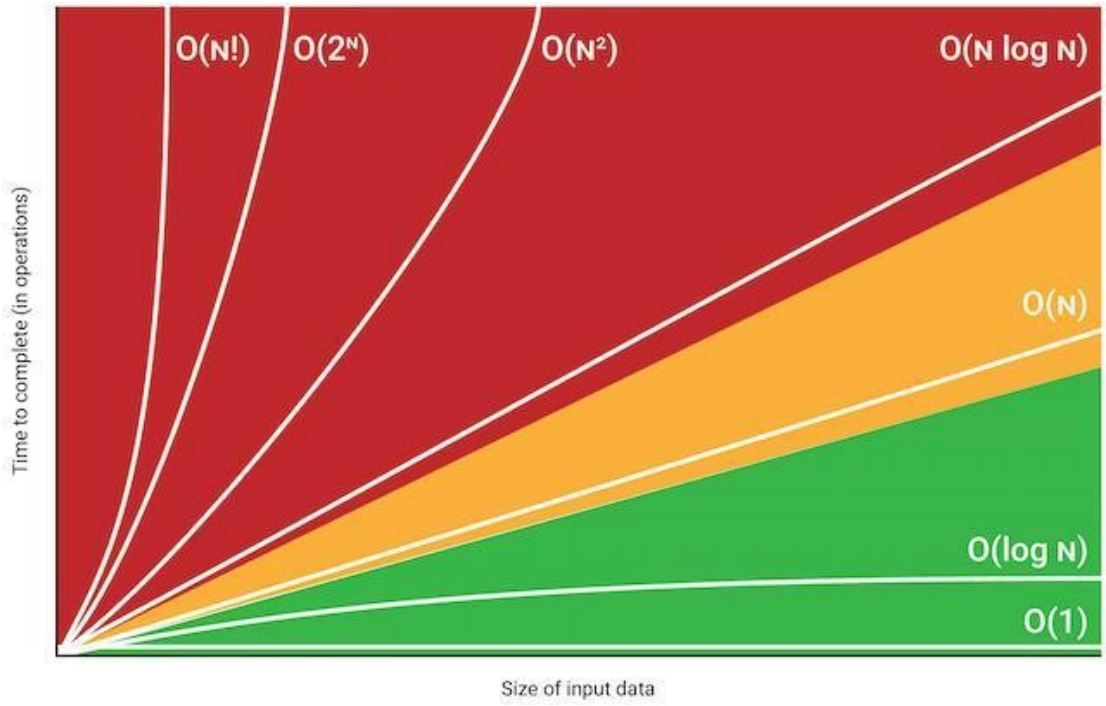
Alternative Big O notation:

O(1) = O(yeah)
O(log n) = O(nice)
O(n) = O(ok)
O(n^2) = O(my)
O(2^n) = O(no)
O(n!) = O(mg!)

8:10 PM · 06 Apr 19 · Twitter for Android

3,665 Retweets **9,265** Likes

You're far enough along in the course to have a decent understanding of why paying attention to performance is important. Just using the most straightforward approach can lead to long run times and extensive memory usage – even crashes!



1. Big O complexity is a way to describe both time and space (memory usage) performance in relation to input. Please read throught he following article before continuing:
https://www.interviewcake.com/article/c/big-o-notation-time-and-space-complexity

2. Look through some practice questions to make sure you understand algorithm complexity.
https://www.geeksforgeeks.org/practice-questions-time-complexity-analysis/

3. Consider the phone book problem. You're searching for a certain name in a very long list of names. There are quite a few approaches to this problem. As a group, discuss several different approaches and the time complexity for each. Is there any difference in space complexity? You can watch the clip from Harvard's CS50 course, below, for a visual if you'd like.
https://www.youtube.com/watch?v=o2LqhHoAXxI (a good visual from Harvard's CS50 course)

Other Resources
https://www.geeksforgeeks.org/analysis-algorithms-big-o-analysis/ (gets more mathy)