

EE 200 Lecture 18: Algorithm analysis and sorting

Steven Bell

14 November 2023



Big-O notation

What is the Big-O?

Find the largest element of an array of length N

$O(N)$

For sorted

$O(1)$

What is the Big-O?

length N

Find the two numbers in an array that are closest to each other

Brute force $O(N^2)$

$[0, 1000, 1001, 5000, 9000]$

Sort $O(N \log(N))$

Search $O(N)$

for big N

$$O(N \log(N) + N) \rightarrow O(N (\log(N) + 1)) \xrightarrow{\sim} O(N \log(N))$$

What is the Big-O?

Find the two numbers that are furthest to each other

$N \log(N)$ sort + take first + last

$O(N)$ find largest + smallest

What is the Big-O?

All of the English words that can be made with a set of letters
(e.g., playing Scrabble)

$$A \ C \ T \quad N = 3 \quad O(N!) + \cancel{(N-1)!} + \cancel{(N-2)!}$$

$$26^{10} \quad 1.5 \times 10^{14} \text{ bits} \approx 2 \times 10^{13} \text{ bytes} = 20 \text{ TB}$$

$$O(w) \quad w \text{ is \# of words}$$

What is an intractable problem?

Anything with exponential time (or worse) that can't be solved exactly for meaningfully large N .

These are "NP-hard" (non-polynomial)

Sorting algorithms

Bubble sort

Insertion sort

Mergesort

Quicksort

(We'll come back to heapsort)