CS 473 Fall 2023

Algorithms

Instructor: Sariel Har-Peled

Ed Stem / Gradescope

grades
- ~ 20% hw
- ~ 25% -2 midterms
- ~ 30% Final

$P = NP$?

$P \neq NP$

SETH = strong exponential time hypothesis

F boolean formula

$\bigwedge_{i=1}^{n} C_i \qquad C_i = x_i \vee \overline{x_j} \vee x_k \cdots \overline{x_k}$

2^n ⟵ Seth: Any alg to solve kSAT takes 2^n time for k sufficiently large

$n^{O(1)}$

$O(n^2)$

$O(n^c)$ — $2^{\sqrt{n}}$ — 2^n — $2^{O(\log^2 n)} = n^{O(\log n)}$ quasi-polynomial

$2^{\log^{O(1)} n}$

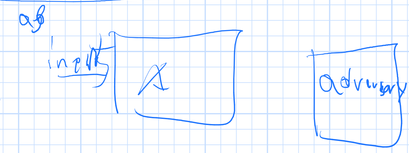
$2^{c_k n}$

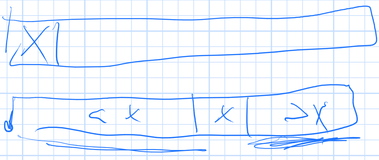
$c_k \to 1$ as $k \to \infty$

---

randomized alg

- approximation algorithms



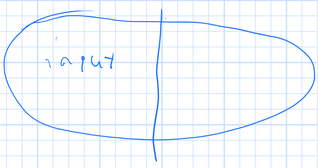


"quicksort"

$T(n) = O(n) + T(n-1) = \Theta(n^2)$

---

divide and conquer



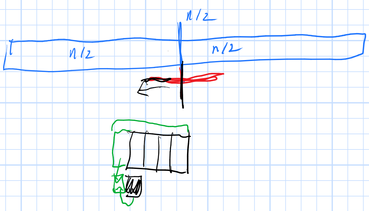
A(input):
- $I_1, I_2 \leftarrow$ break input to two pieces
- $S_1 \leftarrow A(I_1)$
- $S_2 \leftarrow A(I_2)$
- combine(S_1, S_2) // conquer
  - ↑ solution for whole input

$X[1 \ldots n]$: array of numbers

$s(i,j) = \sum_{k=i}^{j} X[k]$

$\max_{i \le j} s(i,j)$

Max subarray problem



$T(n) = O(n) + 2T(n/2) = O(n \log n)$