ALGORITHMS PROBLEM SESSION

2025-01-24

$S : \mathbb{N} \to \mathbb{N}$

5.15 (*)
- $S(n) > 0$
- $S(n)$ ↗
- $S(n) \le 3\,S\left(\frac{n}{2}\right) + O(n)$ for $n = 2^k$, $k \in \mathbb{N}$

$\Rightarrow S(n) = O(n^\alpha)$ $\quad \alpha = \log_2 3 \approx 1.58$

$\not\Rightarrow S(n) = o(n^\alpha)$

Karatsuba

LOGIC $\quad (\forall S\ (*))(S(n) = o(n^\alpha))$

disprove it: counterexample

$S(n) = n^\alpha$ for $n = 2^k$

$2^{k-1} < n \le 2^k$ $\quad S(n) := \left(2^k\right)^\alpha = 3^k$

5.18

(*) $T(n) \le 7\,T\left(\lceil \frac{n}{2} \rceil\right) + \underbrace{O(n^2)}_{Cn^2}$ $\Rightarrow T(n) = O(n^\beta)$ $\Rightarrow T(n) = O(n^{\beta})$

best β ?

How do we guess β ?

$T'(n) = 7 \cdot T'\left(\frac{n}{2}\right)$ $\quad \therefore \beta = \log_2 7$

$n = 2^k \Rightarrow T'(n) = 7^k$

Find $g(1) \ge T(1)$

$n \ge 2$ $\quad g(n) \ge 7\,g\left(\lceil \frac{n}{2} \rceil\right) + Cn^2$ $\Rightarrow \forall n = 2^k$ $g(n) \ge T(n)$

guess $g(n) := A \cdot n^\beta + B \cdot n^2$ ←

NEED $g(1) = \boxed{A + B \overset{?}{\ge} T(1)}$

$An^\beta + Bn^2 \overset{?}{\ge} 7 \cdot \left(A\left(\frac{n}{2}\right)^\beta + B\left(\frac{n}{2}\right)^2\right) + Cn^2$

$\div n^2$

$B \overset{?}{\ge} \frac{7}{4}B + C$

$-\frac{3}{4}B \overset{?}{\ge} C$

$\boxed{B \overset{?}{\le} -\frac{4}{3}C}$

$B := -\frac{4}{3}C$

$A := T(1) - B$

$= T(1) + \frac{4}{3}C$

Conclusion so far:

$$\textcircled{*} \Rightarrow (\exists C')\left((\forall k)\left(T(2^k) \le C'(2^k)^\beta\right)\right)$$

Claim: $(\exists C'')(\forall n \; T(n) \le C'' n^\beta)$

Pf $\quad 2^{k-1} < n \le 2^k < 2n$

$$T(n) \le T(2^k) \le C' \cdot (2^k)^\beta$$

$$\le C'(2n)^\beta = \underbrace{7 \cdot C'}_{C'' := 7C'} n^\beta$$

n^β best possible: $o(n^\beta)$ does not follow

---

for induction, we need

$O(n) \longrightarrow Cn$

(p4)

1 Claim $2^n = O(n)$

2 Base case $2^1 = 2 = O(1)$

3 Inductive step:

4 Assume $2^n = O(n)$

5 Desired concl $2^{n+1} = O(n+1)$

6 $2^{n+1} = 2 \cdot 2^n = 2 \cdot O(n)$

7 $= O(n) = O(n+1)$

8

✓

$f(n) = O(g(n))$

$\Rightarrow 7 \cdot f(n) = O(g(n))$ ✓

5.22

$b = b(n)$ # bits of $n \in \mathbb{N}$

$$2^{b-1} \leq n < 2^b$$

$$2^{b-1} < n+1 \leq 2^b$$

$$b-1 < \log_2(n+1) \leq b$$

$$\therefore \quad b = \lceil \log_2(n+1) \rceil$$

✓

5.28 $a_n, b_n > 0$ s.t., a_n bounded

$$a_n = \Theta(b_n)$$

but $\neq \lim \frac{a_n}{b_n}$

$a_n = 1$

$b_n \begin{cases} 1 & n \text{ odd} \\ 2 & n \text{ even} \end{cases}$

Comm complexity of identity

$X \stackrel{?}{=} Y$ n-bit numbers

Alice Bob

$X \bmod q \neq Y \bmod q$ $\quad X \not\equiv Y \bmod q$

q k-bit number

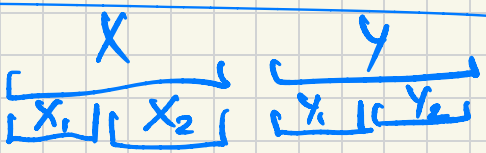
Q: find i s.t. $X_i \neq Y_i$

$(k+1)\lceil \log_2 n \rceil$ bits of comm

Lemma

If $X_1 \equiv Y_1 \ (q)$ •
$X_2 \equiv Y_2 \ (q)$ •

$\Rightarrow X \equiv Y \ (q)$



$X = 2^l \cdot X_1 + X_2$

$Y = 2^l \cdot Y_1 + Y_2$

$\therefore$ Alice → Bob: $(X_1 \bmod q)$

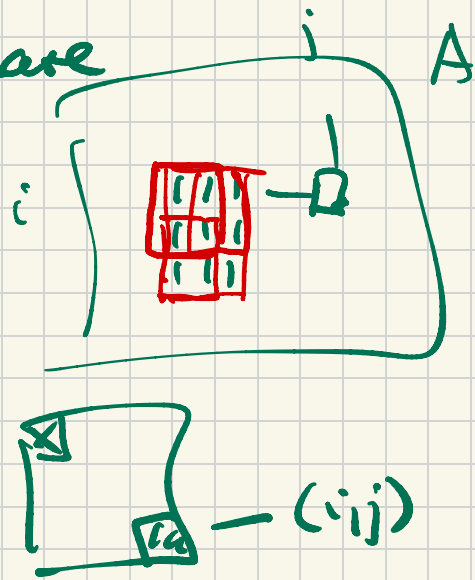
Bob → Alice: 1 if $X_1 \not\equiv Y \ (q)$
0 o/w

5.18 $T(n) \le 7 \cdot T(\lceil \frac{n}{2} \rceil) + O(n^2)$

① $\lceil \frac{n}{2} \rceil \le \frac{n+1}{2}$ ←

② use reverse ineq.

6.15 All-ones square



$m(i,j)$ = max size of all-1-sq ending at (i,j)

♡ $m(0,j) = m(i,0) = 0$

if $A[i,j] = 0$ then $m(i,j) = 0$

else $m(i,j) = 1 + \min\{m(i-1,j), m[i,j-1], m[i-1,j-1]\}$

$\le \quad = \quad =$

$*\begin{cases} a \equiv b \pmod m \\ x \equiv y \pmod m \end{cases}$ Assumptions

$m \mid a-b$

$m \mid x-y$

$\Rightarrow ax \equiv by \pmod m$

DC $m \mid ax-by$

Lemma $r \equiv s \pmod m$, $c \in \mathbb{Z}$ $\Rightarrow cr \equiv cs \pmod m$ ✓

Pf: Assn $m \mid r-s$

DC $m \mid cr-cs$

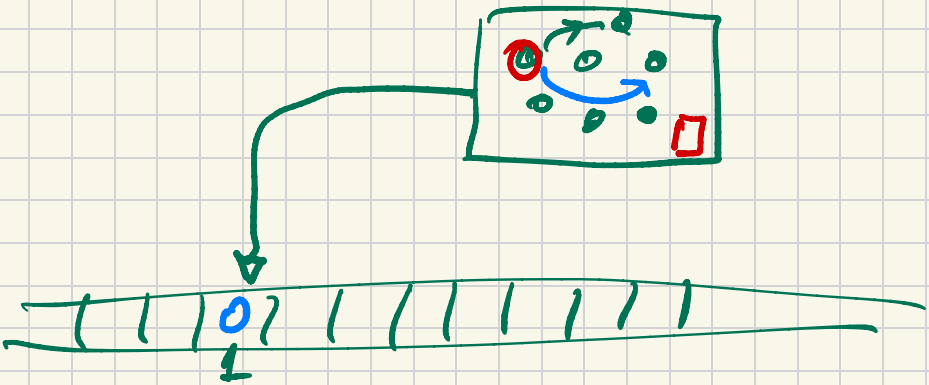
$m \mid c(r-s)$

$m \mid r-s \mid c(r-s)$

$a \equiv b \pmod m$ $\quad x \equiv y \pmod m$

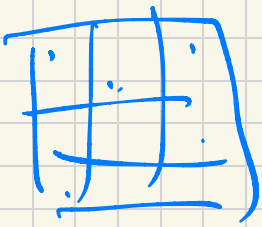
$ax \equiv bx \pmod m$ $\quad bx \equiv by \pmod m$

"$\equiv \bmod m$" is transitive ✓

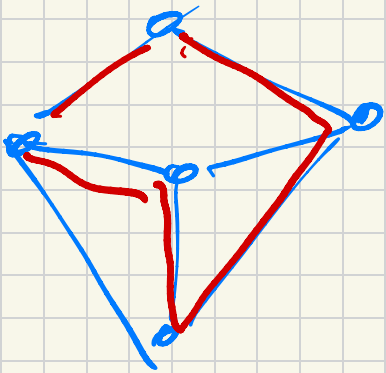


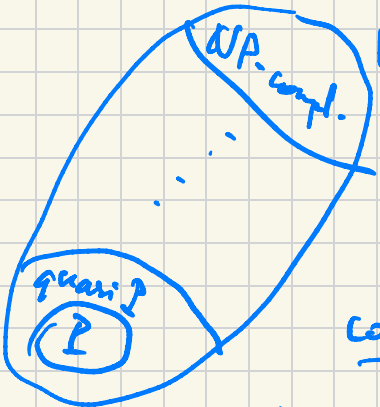
P vs NP

5.15



$n \log a = O(n^2)$





$NP \neq coNP$

poly time: $O(n^c)$

$\exists \leftrightarrow \forall$

$\exists\forall \leftrightarrow \forall\exists$

Σ_2 Π_2

quasipoly time: $O(\exp((\log n)^c)) = t(n)$

exp. time: $\exp(n^c) = T(n)$

$\exp(O(n^c))$ $\exp(\sqrt{n})$

$\log(t(n)) = (\log n)^c$

$\log(T(n)) = n^c$

THM If GI is NP-complete then $\Sigma_2 = \Pi_2$

via "interactive proofs"

Interactive proofs (B, Goldwasser-Micali-Rackoff 1985)

Arthur - Merlin games

IP = PSPACE (Lund-Fortnow-Karloff-Nisan, Shamir 1989)

in polynomial time, an all-powerful prover (Merlin) can convince a polynomial-time, randomizing verifier (Arthur) that in a given configuration in generalized chess, Black has a winning strategy

much more can be done with 2 provers

MIP = NEXPTIME (B-Fortnow-Lund, UChicago 1990)

→ "probabilistically checkable proofs"

Thm If $P \neq NP$ then one cannot approximate the CLIQUE NUMBER in polynomial time

within a constant factor

within $\sqrt{n}$

within $n^{1-\varepsilon}$ → can't tell these graphs apart

JOHAN HÅSTAD