

Lecture 25

CSE 331

Apr 3, 2020

Logistics

- Homework 6 is out today
- Deadline for regrading requests is Mon (Apr 6)
- Mid-semester temp grades will be out on Tue (Apr 7)
 - Based on 4 hws, midterms, quiz 1 (nothing dropped)
- Video Project (remember?)
 - Due April 20
 - See mini project website for details

- New S/U policy: It's about you, not me 😊
 - I'll assign letter grades as usual;
 - YOU choose to convert your letter grade to S/U
 - Chance to prevent any possible damage to your GPA
 - C and above: S
 - C- and below: U

- **Most importantly: Take care of yourself!**
 - I mean mentally!
 - **Go easy on yourself!**

Divide and Conquer

Divide up the problem into at least two sub-problems

Recursively solve the sub-problems

“Patch up” the solutions to the sub-problems for the final solution

Improvements on a smaller scale

Greedy algorithms: exponential \rightarrow poly time

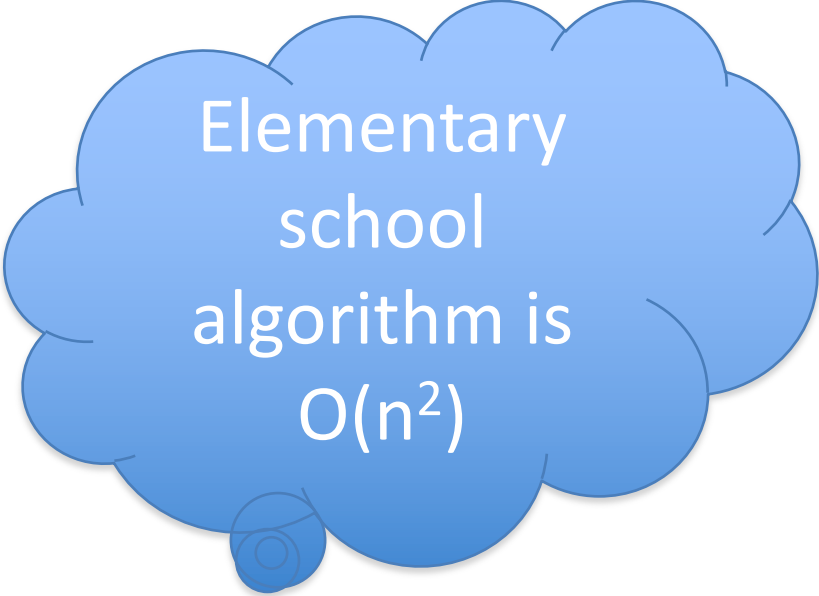
(Typical) Divide and Conquer: $O(n^2)$ \rightarrow asymptotically smaller running time

Multiplying two numbers

Given two numbers a and b in binary

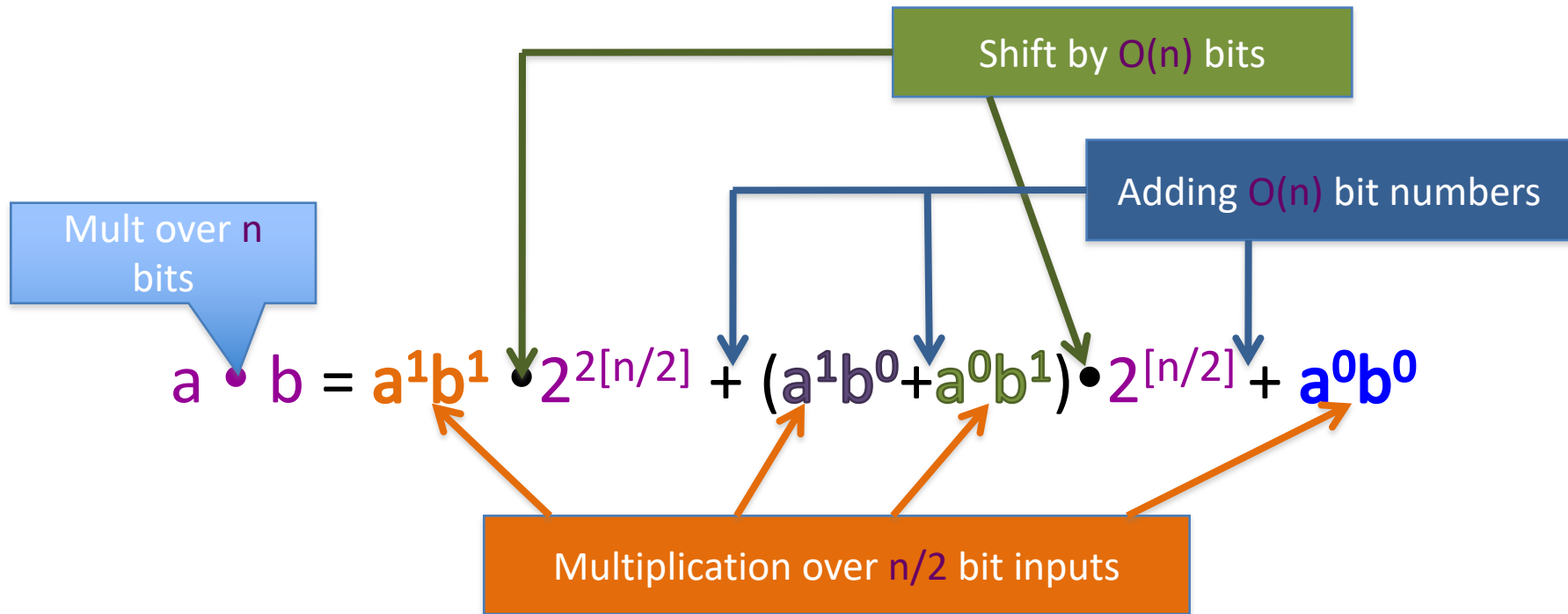
$$a = (a_{n-1}, \dots, a_0) \text{ and } b = (b_{n-1}, \dots, b_0)$$

Compute $c = a \times b$



Elementary
school
algorithm is
 $O(n^2)$

The current algorithm scheme



$$T(n) \leq 4T(n/2) + cn$$

$$T(1) \leq c$$

$T(n)$ is $O(n^2)$

The key identity

$$a^1b^0 + a^0b^1 = (a^1 + a^0)(b^1 + b^0) - a^1b^1 - a^0b^0$$

The final algorithm

Input: $a = (a_{n-1}, \dots, a_0)$ and $b = (b_{n-1}, \dots, b_0)$

Mult (a, b)

If $n = 1$ return a_0b_0

$a^1 = a_{n-1}, \dots, a_{\lceil n/2 \rceil}$ and $a^0 = a_{\lceil n/2 \rceil - 1}, \dots, a_0$

Compute b^1 and b^0 from b

$x = a^1 + a^0$ and $y = b^1 + b^0$

Let $p = \text{Mult}(x, y)$, $D = \text{Mult}(a^1, b^1)$, $E = \text{Mult}(a^0, b^0)$

$F = p - D - E$

return $D \cdot 2^{2\lceil n/2 \rceil} + F \cdot 2^{\lceil n/2 \rceil} + E$

$$T(1) \leq c$$

$$T(n) \leq 3T(n/2) + cn$$

$O(n^{\log_2 3}) = O(n^{1.59})$
run time

All **green** operations
are $O(n)$ time

$$a \cdot b = a^1b^1 \cdot 2^{2\lceil n/2 \rceil} + ((a^1+a^0)(b^1+b^0) - a^1b^1 - a^0b^0) \cdot 2^{\lceil n/2 \rceil} + a^0b^0$$