

Lecture 29

CSE 331

Nov 5, 2014

Online OH tonight @8:45pm

 note ☆ [stop following](#) **11** views

Online OH#7 tomorrow at 8:45pm

Note the slightly different time than usual: 8:45-9:45pm tomorrow. Please tag all your questions with 'onlineoh7'

[onlineoh7](#)

[edit](#) - good note | 0 2 hours ago by Abri Rudra

Mini project report due **TONIGHT**

note ☆

stop following 1 views

Mini Project Report due Nov 5

A gentle reminder about the upcoming deadline of 11:59pm to email me your group's project report. For more details see:

<http://www.cse.buffalo.edu/~atri/courses/331/handouts/mini-project.pdf>

(The link is also available from the "Resources" tab.)

#pin

mini_project

edit good note 0

Just now by Atri Rubra

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

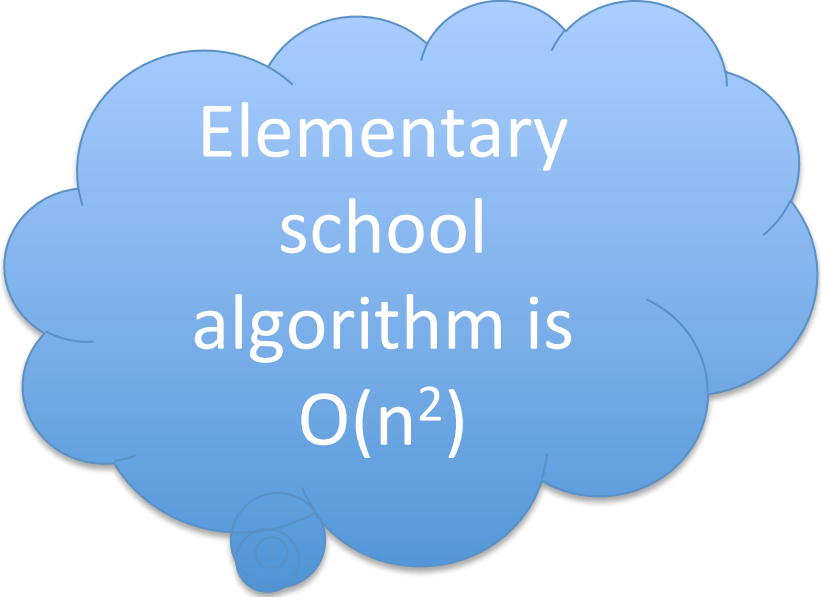
(Typical) Divide and Conquer: $O(n^2)$ \rightarrow asymptotically smaller running time

Multiplying two numbers

Given two numbers a and b in binary

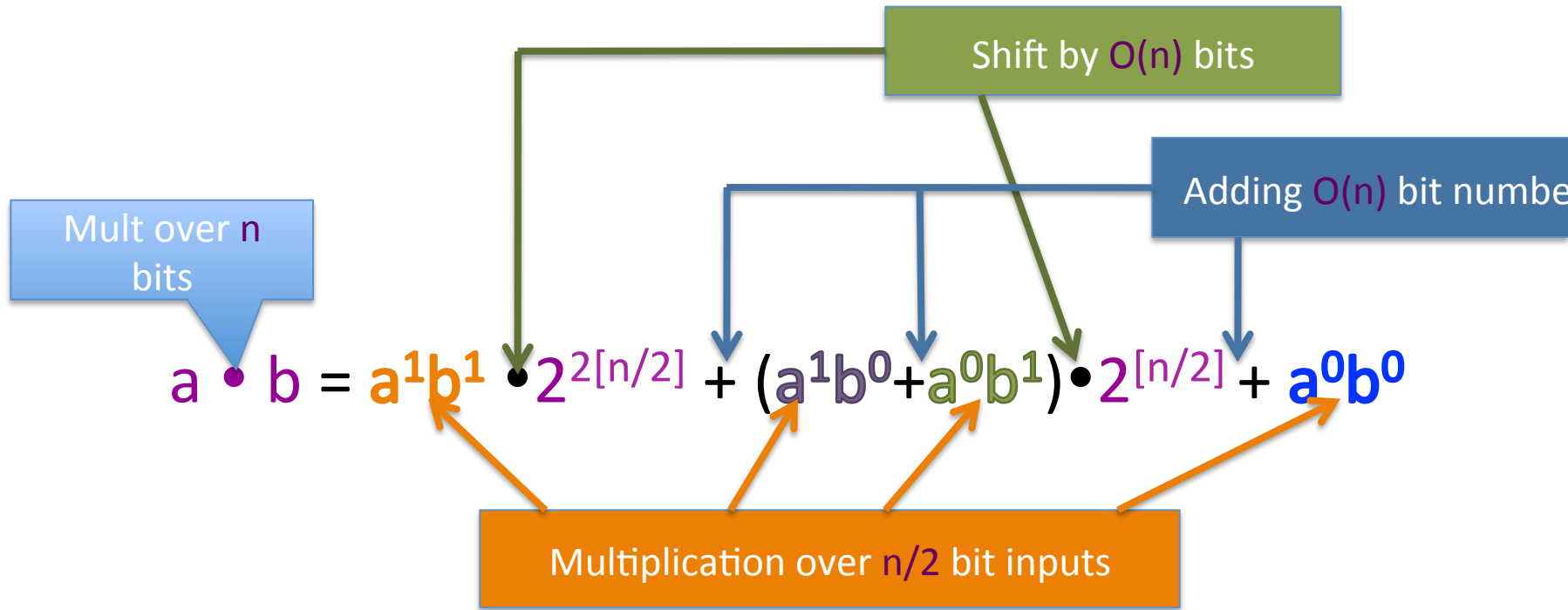
$$a = (a_{n-1}, \dots, a_0) \text{ and } b = (b_{n-1}, \dots, b_0)$$

Compute $c = a \times b$



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

The current algorithm scheme



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

$$T(1) \leq c$$

$T(n)$ 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 $a_0 b_0$

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

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 $D \cdot 2^{2\lfloor n/2 \rfloor} + F \cdot 2^{\lfloor n/2 \rfloor} + E$

$$T(1) \leq c$$

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

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

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

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