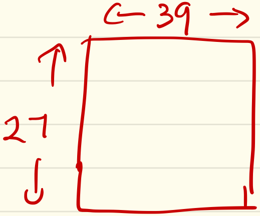
ALGORITHMS



XXXIX · XXVII

39 × 27

GRADE-SCHOOL MULTIPLICATION

$$\overbrace{(1011)_2}^{n} \times \overbrace{(1101)_2}^{n}$$

n colum

$$
\begin{array}{r}
1011 \\
00000 \\
101100 \\
1011000 \\
\hline
10001111
\end{array}
\quad n
$$

$O(n^2)$

RUSSIAN PEASANT ALGO

$O(n^2)$

$$\overbrace{10101\ldots1}^{n} \qquad \overbrace{1011111}^{n}$$

1	21	27	$\to 27\cdot 2^0$
0	~~10~~	~~54~~	
1	5	108	$\to 27\cdot 2^2$
0	~~2~~	~~216~~	
1	1	432	$\to 27\cdot 2^4$
		567	

$(10101)_2$, n

$$27\cdot 21 = 27\cdot(10101)_2$$

$$= 27\cdot 2^4 + 27\cdot 2^2 + 27\cdot 2^0 \to \Theta(n^2)$$

- Is it Correct ?? YES

→ What is the runtime??

→ measured as a function of input size

$T(n) =$ # of steps on inputs of size n.

→ $T(n) \leftarrow$ asymptotically

Grade school → $O(n^2)$

→ Can you do better??

$$\frac{(10111)_2}{2} \longrightarrow (1011)_2$$

$$\downarrow$$

$$(101)_2$$

$$\downarrow$$

$$(10)$$

$$\downarrow$$

$$1$$

DIVIDE AND CONQUER

$$\underbrace{\boxed{A_L \mid A_R}}_{A} \times \underbrace{\boxed{B_L \mid B_R}}_{B}$$

$\longleftarrow n \longrightarrow$ $\qquad$ $\longleftarrow n \longrightarrow$

$$A = \boxed{A_L \mid 0000} + \boxed{0000 \mid A_R}$$

$$(101111) = \boxed{101\ 000} + \boxed{000\ 111}$$

$$\boxed{A = A_L \cdot 2^{n/2} + A_R}$$

$$B = 2^{n/2} \cdot B_L + B_R$$

$$123456$$

$$= 123000 + 000456$$

$$= (10^3) \cdot 123 + 456$$

$$\underbrace{\boxed{A_L \mid A_R}}_{A} \times \underbrace{\boxed{B_L \mid B_R}}_{B}$$

$$\overleftrightarrow{A \cdot B}^{\,n} = \left(2^{n/2} A_L + A_R\right) \cdot \left(2^{n/2} B_L + B_R\right)$$

$$= 2^n \cdot \underbrace{(A_L \cdot B_L)}_{P1} + 2^{n/2} \cdot (\underbrace{A_L B_R}_{P2} + \underbrace{A_R B_L}_{P3}) + \underbrace{A_R B_R}_{P4}$$

MULTIPLY (A, B: n bit numbers)

$A = 2^{n/2} \cdot A_L + A_R \qquad B = 2^{n/2} B_L + B_R$

$\approx n$ bits

P1 ← MULTIPLY (A_L, B_L)

P2 ← MULTIPLY (A_L, B_R)

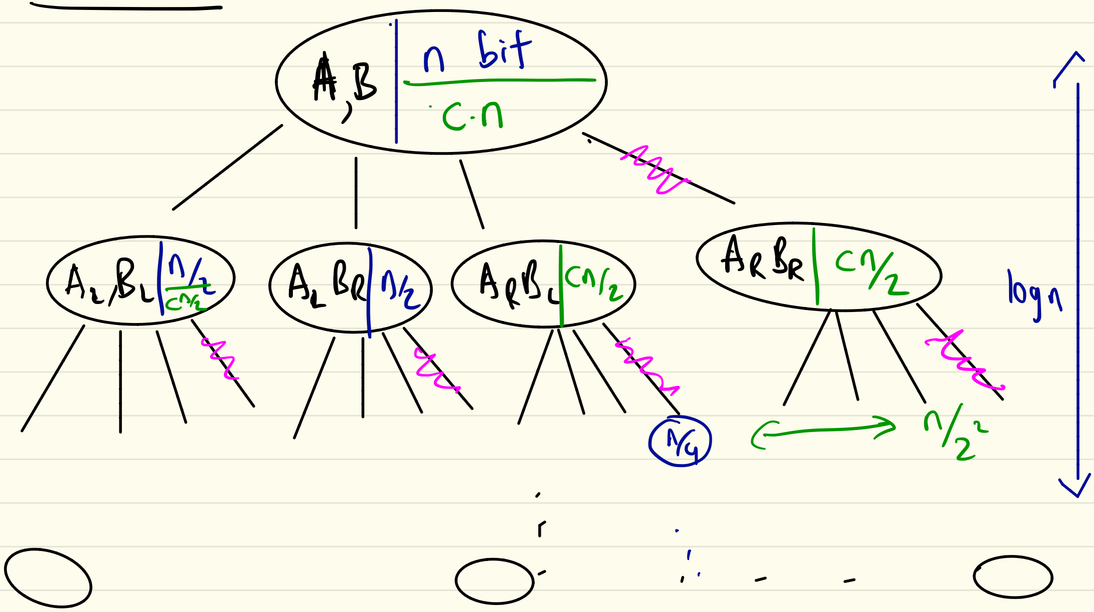
P3 ← MULTIPLY (A_R, B_L)

P4 ← MULTIPLY (A_R, B_R)

RETURN: $\underline{2^n \cdot P1 + 2^{n/2}(P2 + P3) + P4}$ ← $O(n)$ step $= c \cdot n$ steps

$\boxed{\text{Base Case} \quad n = 1}$

EXECUTION:



Cost per Node	Number	
$c\cdot n$	1	1
$c\cdot(n/2)$	3	4
$c\cdot n/2^2$	3^2	4^2
⋮	3^3	4^3

$$= 1\cdot(cn) + \overset{3}{4}\cdot\left(\frac{cn}{2}\right) + \overset{3}{4^2}\left(\frac{cn}{2^2}\right) + \cdots\cdots + \overset{3}{4}^{\log n}\cdot\left(\frac{cn}{2^{\log n}}\right)$$

$$= cn + cn\cdot\left(4/2\right) + cn\cdot\left(4^2/2^2\right) \cdots \left(\frac{1}{2}\right) \qquad O(n^2)$$

$$= O\left(n\cdot\left(\frac{4}{2}\right)^{\log n}\right) = O\left(4^{\log n}\cdot\frac{n}{2^{\log n}}\right) = O\left(\frac{4^{\log n}}{3^{\log n}}\right) = n^{\log 4} \quad n^{\log 3} \approx n^{1.6}$$

FACT: In an increasing geometric progression

sum $\approx O$(last term)

$$\boxed{2^{\log n} = n}$$

$$\boxed{a^{\log b} = b^{\log a}}$$

A_L	A_R

A

$\times$

B_L	B_R

B

n

$$\underbrace{A \cdot B}_{n} = \left(2^{n/2} A_L + A_R\right) \cdot \left(2^{n/2} B_L + B_R\right)$$

$$A \cdot B = 2^n \cdot \underbrace{(A_L \cdot B_L)}_{P2} + 2^{n/2} \cdot \underbrace{(A_L B_R + A_R B_L)}_{P1 - P2 - P3} + \underbrace{A_R B_R}_{P3}$$

$$P1 = \boxed{(A_L + A_R)}^{\,n/2+1} \boxed{(B_L + B_R)}^{\,n/2+1}$$

$$P2 = A_L B_L$$

$$P3 = A_R B_R$$

$n^{1.57} \leftarrow$

MULTIPLY $(A, B$: n bit numbers$)$

$A = 2^{n/2} \cdot A_L + A_R \quad B = 2^{n/2} B_L + B_R$

$P1 \leftarrow$ MULTIPLY $(A_L + A_R,\ B_L + B_R)$

$P2 \leftarrow$ MULTIPLY $(A_L,\ B_L)$

$P3 \leftarrow$ MULTIPLY $(A_R,\ B_R)$

RETURN: $2^n \cdot P2 + 2^{n/2}(P1 - P2 - P3) + P3$