Motivation

Given: p, q two n-digit decimal numbers.
Goal: compute the product $p \cdot q$ efficiently

p, q are stored as arrays $P = [1, \ldots, n]$ and $Q = [1, \ldots, n]$, with p_1, q_1 as 'most significant digit'.

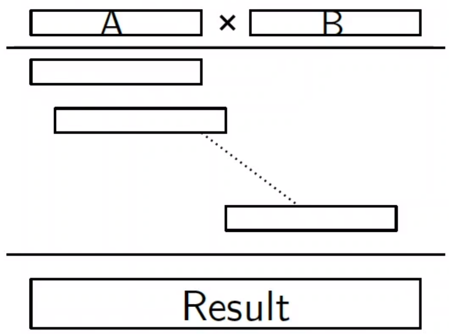
$$p = \sum_{i=1}^{n} P[i] \cdot 10^{n-i}$$

$$q = \sum_{i=1}^{n} Q[i] \cdot 10^{n-i}$$

-

Methods

Method 1: School method



- n^2 multiplications (plus additions) $\Rightarrow \mathcal{O}(n^2)$ time.

Method 2: Divide & Conquer

Divide p into a and b: $p = a \,||\, b = a \cdot 10^{n/2} + b$

Divide q into c and d: $q = c \,||\, d = c \cdot 10^{n/2} + d$

$$p \cdot q = a \cdot c \cdot 10^{n} + (a \cdot d + c \cdot b) \cdot 10^{n/2} + b \cdot d$$

- 4 multiplications with $n/2$ digits (10^x only needs a shift-operation)

Method 3: Improved Divide & Conquer

Calculate $u = ac, v = bd, w = (a+b)(c+d)$. This are three multiplications with $n/2$ digits. We also need $ad + bc$.

$$\begin{aligned}(ad+bc) &= ad + ac + bd + bc - ac - bd \\ &= (a+b)(c+d) - ac - bd \\ &= w - u - v\end{aligned}$$

So no more multiplications are required. Additions can be done in $\mathcal{O}(n)$ time.

- $$\begin{aligned}T(n) &= 3T(n/2) + \mathcal{O}(n) \\ &= 3[3T(n/4) + \mathcal{O}(n/2)] + \mathcal{O}(n) \\ &= 3^2T(n/2^2) + 3^1\mathcal{O}(n/2^1) + 3^0\mathcal{O}(n/2^0) = \cdots = \\ &= 3^kT(n/2^k) + \sum_{i=0}^{k-1} 3^i\mathcal{O}(n/2^i) \\ &= \mathcal{O}(3^{ld(n)}) + \mathcal{O}(n)\sum_{i=0}^{ldn-1}(3/2)^i \\ &= \mathcal{O}(n^{ld(3)}) \sim \mathcal{O}(n^{1.59})\end{aligned}$$
-