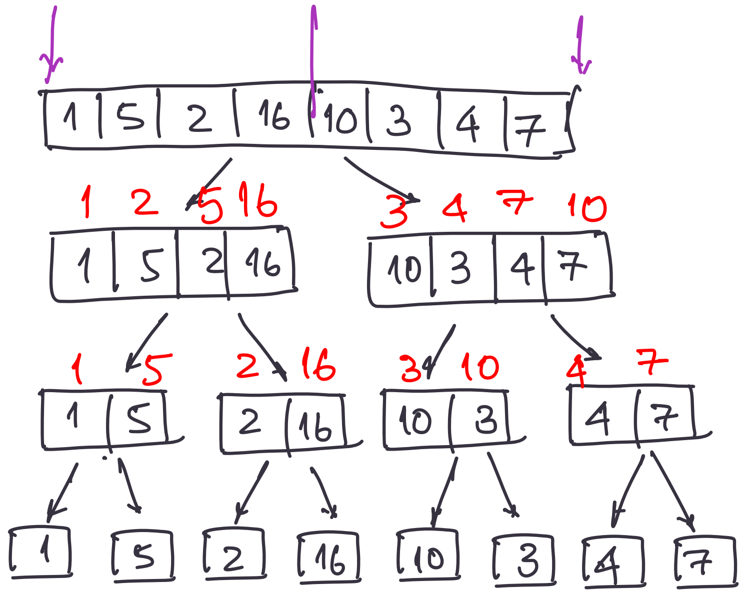
Divide and Conquer paradigm.

Ex: Merge sort



$T(n) = T\left(\lfloor \frac{n}{2} \rfloor\right) + T\left(\lceil \frac{n}{2} \rceil\right) + O(n).$

size of the array.

$O(n \log n).$

Integer multiplication.

Binary representation

$A = (a_0, a_1, a_2, \dots)$

$= a_0 + a_1 \cdot 2 + a_2 \cdot 2^2 + \dots$

$B = (b_0, b_1, \dots).$

$= b_0 + b_1 \cdot 2 + b_2 \cdot 2^2$

$AB = (a_0 + a_1 \cdot 2 + a_2 \cdot 2^2 + \dots)(b_0 + b_1 \cdot 2 + \dots)$

$$\left.\begin{aligned} A &= a_0 + a_1 \cdot 2 + \dots + a_n \cdot 2^n \\ B &= b_0 + b_1 \cdot 2 + \dots + b_n \cdot 2^n \end{aligned}\right\} O(n^2) \text{ naively.}$$

A_0

$A = a_0 + a_1 \cdot 2 + \dots + a_{\frac{n}{2}} \cdot 2^{n/2}$

$+ a_{\frac{n}{2}+1} \cdot 2^{n/2+1} + \dots + a_n \cdot 2^n$

$2^{n/2}\left(a_{\frac{n}{2}+1} \cdot 2 + a_{\frac{n}{2}+2} \cdot 2^2 + \dots + a_n \cdot 2^{n/2}\right)$

A_1

$A = A_0 + A_1 \cdot 2^{n/2}$

$B = B_0 + B_1 \cdot 2^{n/2}$

$A \cdot B = (A_0 + A_1 \cdot 2^{n/2})(B_0 + B_1 \cdot 2^{n/2})$

$= A_0 B_0 + (A_1 B_0 + B_1 A_0) \cdot 2^{n/2} + A_1 B_1 \cdot 2^n.$

$$T(n) = 4 \cdot T\left(\frac{n}{2}\right) + O(n) \Big\} \longrightarrow n^{\log_2 4} \sim O(n^2)$$

$$T(n) = a\, T\left(\frac{n}{b}\right) + O(n) \sim n^{\log_b a}$$

Karatsuba's method:

$$\left.\begin{aligned} C_0 &:= A_0 \cdot B_0 \\ C_1 &= A_1 B_1 \end{aligned}\right\} \quad \begin{aligned} C_2 &= (A_1 - A_0)(B_1 - B_0) \\ &= A_1 B_1 + A_0 B_0 - (A_0 B_1 + B_0 A_1) \end{aligned}$$

$$A_0 B_1 + B_0 A_1 = C_1 + C_0 - C_2$$

$$A \cdot B = C_0 + (C_1 + C_0 - C_2) \cdot 2^{n/2} + C_1 \cdot 2^n.$$

$$T(n) = 3 \cdot T\left(\frac{n}{2}\right) + O(n) \longrightarrow O\left(n^{\log_2 3}\right)$$

$$\begin{aligned} A &= A_0 + A_1 \cdot 2^{n/3} + A_2 \cdot 2^{2n/3} \\ B &= B_0 + B_1 \cdot 2^{n/3} + B_2 \cdot 2^{2n/3} \end{aligned}$$

naively.

$$T(n) = 9\, T\left(\frac{n}{3}\right) + O(n)$$

$$n^{\log_3 9}$$

Matrix Mult.

$$A = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right]_{\underline{n \times n}} \quad B = \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} \quad C = \begin{bmatrix} A_{11} B_{11} + A_{12} B_{21} & A_{11} B_{12} + A_{12} B_{22} \\ A_{21} B_{11} + A_{22} B_{21} & A_{21} B_{12} + A_{22} B_{22} \end{bmatrix}$$

$$(A_{ij})_{\frac{n}{2} \times \frac{n}{2}}$$

$$T(n) = 8 \cdot T\left(\frac{n}{2}\right) + O(n^2) \qquad n^{\log_2 8}$$

$$\sim O(n^3).$$

$$T(n) = \underbrace{7 \cdot T\left(\frac{n}{2}\right)}_{\hookrightarrow\, n^{\log_2 7}} + O(n^2)$$

$M_1 = (A_{11}+A_{22})(B_{11}+B_{12})$

$M_2 = (A_{21}+A_{22})\,B_{11}$

$M_3 = A_{11}\,(B_{12}-B_{22})$

$M_4 = A_{22}\,(B_{21}-B_{11})$

$M_5 = (A_{11}+A_{12})\cdot B_{22}$

$M_6 = (A_{21}-A_{11})\cdot(B_{11}+B_{12})$

$M_7 = (A_{12}-A_{22})\,(B_{21}+B_{22})$

$$C = \begin{bmatrix} M_1+M_4-M_5+M_7 & M_3+M_5 \\ M_2+M_4 & M_1-M_2+M_3+M_6 \end{bmatrix}$$

$n^{\log_2 7}$

$\underbrace{n^{2.32\cancel{88}}}_{\text{Laser method}}$

Laser method

(Coppersmith-Winograd)

Tensor.

Discrete Fourier Transform.

DFT matrix

$\downarrow$

$A_{n\times n}$

$\vec{a} = (a_0, a_1, \dots, a_{n-1})$

$(A)_{i,j} = \omega^{ij}$ → n^{th} primitive root of unity.

$0 \le i,j \le n-1$

$(\vec{b})^T = A\cdot(\vec{a})^T.$

$b_i = \sum_{k=0}^{n-1} A_{ik}\cdot a_k$

$O(n^2)$ time

$$\begin{bmatrix} b_0 \\ b_1 \\ \vdots \\ b_{n-1} \end{bmatrix} = \underbrace{\begin{bmatrix} & & \\ & \omega^{ij} & \\ & & \end{bmatrix}}_{A_{n\times n}} \begin{bmatrix} a_0 \\ \vdots \\ a_{n-1} \end{bmatrix}$$

→ n^{th} root of unity

↳ Roots of polynomial $x^n - 1 = 0$.

↳ $\underbrace{e^{\frac{2\pi i}{n} \cdot k}}$ $\quad k \in [0, n-1]$

$\underbrace{1, w, w^2, \cdots, w^{n-1}.}$ $\qquad \sum_{i=0}^{n-1} w^i = 0$.

$w_1, w_2 \cdots w_{n-1}$

$w^n = 1$ and $\forall\, k < n$, $w^k \neq n$.