9 · ~~compete~~

# Two multiply 2 big integer numbers :

10

→ **Basic Algo**

| Basic Algo | Time complexity |
|---|---|
| 11   4167   $n$<br>$\times$ 9832   $m$<br>8334<br>12   125010<br>. . . 600<br>13 . . . . . . 00 | ~~$(n \times m$~~ .<br>~~Per digit = 1 mult / 1 additⁿ~~<br>multiplicatⁿ $= n^2$<br>additⁿ = ($n$ rows) ($n$ colums)<br>$= n^2$<br>Net $= O(n^2)$ |

14

→ **Divide and Conquer.**

$$x = \boxed{x_L}\ \boxed{x_R} = 2^{n/2} x_L + x_R$$

16

$$y = \boxed{y_L}\ \boxed{y_R} = 2^{n/2} y_L + y_R$$

$n/2$ bits     $n/2$ bits



$$xy = (2^{n/2} x_L + x_R) \cdot (2^{n/2} y_L + y_R)$$
$$= 2^n x_L y_L + 2^{n/2}(x_L y_R + x_R y_L) + x_R y_R$$

$T(n) = T(n/2) \quad T(n/2) \quad T(n/2) \quad T(n/2)$

$T(n)$ ↓ Product of $n$ bit No.s

$T(n/2)$ ↓ Product of 2 numbers of size $n/2$ bits

$T(n) \le 4T(n/2) + O(n)$

$\downarrow$

$2^n \times$ Product
shifting n bits

Recurrence Equat$^n$: time complexity of recursive algo shown with recursive equat$^n$

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

$T(n) \le 4\{4T(n/4) + cn/2\} + cn$

$\le 4^2 T(n/2^2) + 3C \cdot n$

$\vdots$

At step $x \le 4^x T(n/2^x)$ + ~~$Cn/2^x$~~

$+ (2^x - 1) C \cdot n$

At some $x$ : $n/2^x = 1$

$x = \log_2(n)$

$T(n) = 4^{\log_2(n)} T(1) + (2^{\log_2 n} - 1) Cn$

$= n^2 + O(\cancel{n\log(n)})\ (n-1) C \cdot n$

$= \cancel{O(n^2)}$

$= O(n^2)$ ⟨No gain⟩

Appointments

$$xy = 2^{n} x_L y_L + 2^{n/2}(x_L y_R + x_R y_L) + x_R y_R .$$

$$x_L y_R + x_R y_L = (x_L + x_R)(y_L + y_R) - x_L y_L - x_R y_R$$

function multiply (x, y)
  input: n-bit positive integer numbers x and y.
  output: their product

  if (n==1): return xy
  $x_L, x_R$ = leftmost $\lceil n/2 \rceil$, rightmost $\lfloor n/2 \rfloor$ of x.
  $y_L, y_R$ = leftmost $\lceil n/2 \rceil$, rightmost $\lfloor n/2 \rfloor$ of y.

Solving Recurrence Equat$^n$ with Recursion Tree

Master Theorem

if $T(n) = aT(\lceil n/b \rceil) + O(n^d)$
for some constant $a > 0$, $b > 1$
and $d \geq 0$ then

$$T(n) = \begin{cases} O(n^d) & \text{if } d > \log_b a \\ O(n^d \log n) & \text{if } d = \log_b a \\ O(n^{\log_b a}) & \text{if } d < \log_b a \end{cases}$$

Proof:

Assumptions: $n$ is power of $b$
$n = b^m$

Total work done at level depth $K$
$= a^K \times O(n/b^K)^d$
$= O(n^d) \times (a/b^d)^K$

$$T(n) = \sum_{K=0}^{\log_b n} O(n^d) \times (a/b^d)^K$$

Case 1:

$a/b^d > 1$
$b^d < a$
$d < \log_b a$

Appointments

Case 1: if ratio < 1

Then the series is decreasing and it's sum is just given by the first term $O(n^d)$

$$P(n) = a_k n^k + a_{k-1} n^{k-1} + \dots a_0$$
$$P(n) = O(n^k)$$

$$a/b^d < 1$$
$$d > \log_b a$$

Case 2: if ratio = 1

$$a/b^d = 1$$
$$d = \log_b a$$

$$T(n) = O(n^d) \times \log_b n$$
$$= O(n^d \log_b n)$$

Case 3: ratio > 1.

$$a/b^d > 1$$
$$d < \log_b a$$

The series is increasing and it's sum is given by the last term

$$O(n^d) \times (a/b^d)^{\log_b n}$$

$$\text{Last term} = cn^d \times \left(\frac{a}{b^d}\right)^{\log_b n}$$

$$= cn^d \times \frac{a^{\log_b n}}{b^{d\log_b n}}$$

$$= cn^d \times \frac{a^{\log_b n}}{n^d}$$

$$= c\, a^{\log_b n}$$

$$\cancel{= c\, a}$$

$$\left[ \log_b n = \frac{\log_a n}{\log_a b} \right]$$

$$= c \cdot a^{(\log_a n) \cdot (\log_b a)}$$

$$= c\, n^{(\log_b a)}$$

$$= O\left(n^{\log_b a}\right)$$

Merge Sort

$$T(n) = T(n/2) + T(n/2) + \text{Running time for merge}$$

Combination step should be $O(n^d)$ for divide and conquer algo's.

$$= 2T(n/2) + O(n)$$

$$= O(n^d \log n)$$

$$O(n \log n)$$

Appointments

9 Prove that for all $n \geq 0$

$$\sum_{i=1}^{n} \frac{i(i+1)}{2} = \frac{n(n+1)(n+2)}{6}$$

10

The proof is by inductⁿ on $n$. (the uper limit)

11 1. Base case:

$n = 0$

12 In this case both side of the equatⁿ is 0

2. Induction step:

13 For $n \geq 0$, assume that

$$\sum_{i=0}^{k} \frac{i(i+1)}{2} = \frac{k(k+1)(k+2)}{6}$$ holds for $k < n$

14

15

$$\sum_{i=0}^{n-1} \frac{i(i+1)}{2} = \frac{(n-1)(n)(n+1)}{6}$$

16

$$\sum_{i=0}^{n} \frac{i(i+1)}{2} = \sum_{i=0}^{n-1} \frac{i(i+1)}{2} + \frac{n(n+1)}{2}$$

The Day Maldives Embraced Islam (Maldives)

Sunday 9

$$= \frac{(n-1)(n)(n+1)}{6} + \frac{n(n+1)}{2}$$

$$= \frac{n(n+1)}{2}\left(\frac{n-1+3}{3}\right)$$

$$= \frac{n(n+1)(n+2)}{6}$$