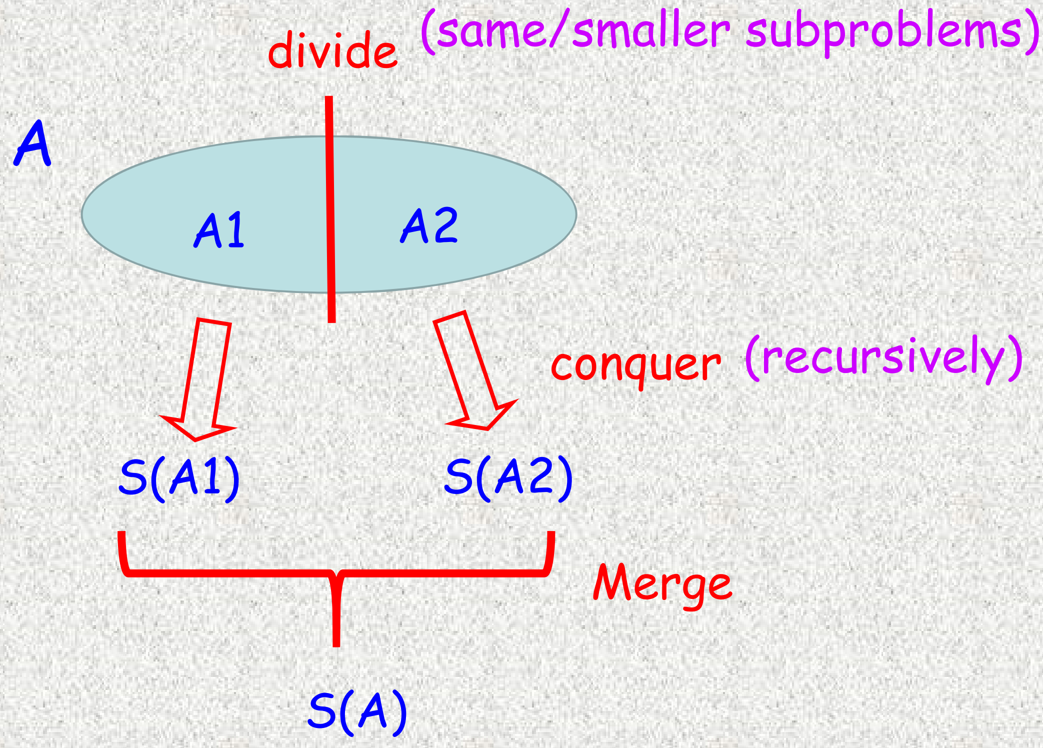
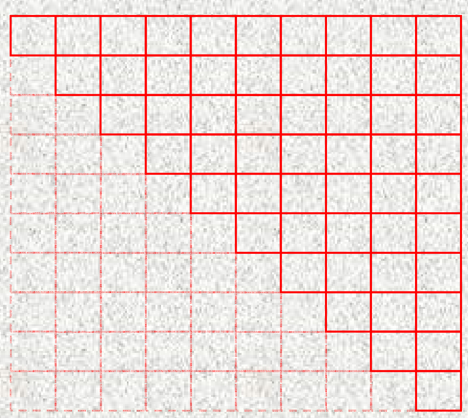
Divide-and-Conquer :



a table S



* $C_2^n = O(n^2)$ possible S[i, j]

* compute all S[i, j] in $O(n^3)$ time

i S[i, j] j

A	1	2	3	4	5	6	7	8	9	10	11	12
	-7	8	-5	20	-3	-8	-23	18	20	-7	12	-5

take $O(j - i + 1) = O(n)$ time

i $\quad$ S[i, j] $\quad$ j

	1	2	3	4	5	6	7	8	9	10	11	12
A	-7	8	-5	20	-3	-8	-23	18	20	-7	12	-5

take $O(j - i + 1)$ time $= O(n)$ time

len l	# of pairs
1	n
2	n-1
3	n-2
4	n-3
⋮	
n	1

$$
\begin{aligned}
T(n) &= \sum_{1 \le l \le n} (n - l + 1) \times l \\
&= (n+1) \sum l - \sum l^2 \\
&\cong (1/6)\, n^3 \\
&= O(n^3)
\end{aligned}
$$

(i_c, j_c, s_c)

A[2, 11] $\quad$ (32)

p = 1 $\quad$ q = 6 $\quad$ r = 12

	1	2	3	4	5	6	7	8	9	10	11	12
A	-7	8	-5	20	-3	-8	-23	18	20	-7	12	-5

A[2, 4]

(23)

$(i_1, j_1, s_1) = (2, 4, 23)$

A[8, 11]

(43)

$(i_2, j_2, s_2) = (8, 11, 43)$

⇨ A[8, 11] is the answer !

Addition of n-bit numbers

$$\begin{array}{r} \overbrace{1011010110011}^{n} \\ +0010100010110 \\ \hline 1 \\ 0 \\ 0 \\ 1 \\ \cdots \end{array}$$

⇨ O(n) time

Multiplication of n-bit numbers

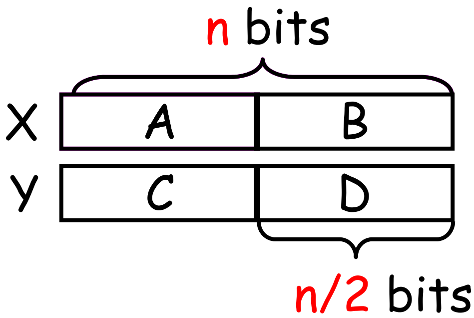
$$\begin{array}{r} \overbrace{1\,0\,1\,1}^{n} \\ \times\, 1\,0\,0\,1 \\ \hline \left.\begin{array}{r} 1\,0\,1\,1 \\ 0\,0\,0\,0 \\ 0\,0\,0\,0 \\ +\,1\,0\,1\,1 \end{array}\right\} n \\ \hline \text{sum of n numbers} \end{array}$$

⇨ $O(n^2)$ time

$$X = 123456$$
$$Y = 789555$$

$$\begin{aligned}
X \cdot Y &= (123\times10^3 + 456)\cdot(789\times10^3 + 555) \\
&= \underset{①}{123\times789}\cdot 10^6 \;\overset{①}{+}\; (\underset{②}{123\times555} \overset{②}{+} \underset{③}{456\times789})\cdot 10^3 \;\overset{③}{+}\; \underset{④}{456\times555} \\
&= 96678\cdot10^6 + (68265 + 359784)\cdot10^3 + 253080 \\
&= 96678\cdot10^6 + 428049\cdot10^3 + 253080 \\
&= 96678000000 + 428049000 + 253080 \\
&= 97475302080
\end{aligned}$$

One big * can be replaced by $\begin{cases}\text{4 smaller * and} \\ \text{3 + (and 2 shift)}\end{cases}$



(1) $X \cdot Y = (A \cdot 2^{n/2} + B) \cdot (C \cdot 2^{n/2} + D)$

$= AC \cdot 2^n + (AD + BC) \cdot 2^{n/2} + BD$

① ② ③ ④

$T(n) = 4\,T(n/2) + O(n) = O(n^2)$

(3+, 2 shift)

(2) Let $P = AC$, $Q = BD$, $R = (A + B)(C + D)$

$X \cdot Y = P \cdot 2^n + (R - P - Q) \cdot 2^{n/2} + Q$

① ② + ③ ④

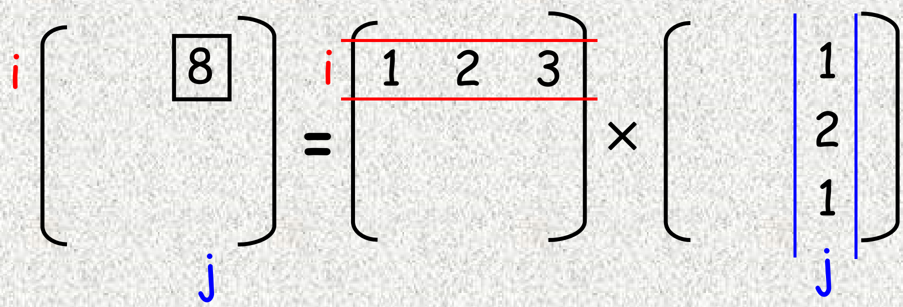
$T(n) = 3\,T(n/2) + O(n) = O(n^{\log_2 3})$

(6+, 2 shift)

Matrix multiplication

C = A × B

$$c_{ij} = \sum_k \{ a_{ik} \times b_{kj} \}$$



Time: $O(n^3)$

Matrix addition

C = A + B

$$c_{ij} = a_{ik} + b_{kj}$$

Time: $O(n^2)$

Remark: RAM model (O(1) time for +, -, *, /, log, sin, cos, ...)
(not in bit complexity - ignore the size of each number)

$$C \quad = \quad A \quad \times \quad B$$

$$\left[\begin{array}{c|c|c} C_{11} & C_{12} & C_{13} \\ \hline C_{21} & C_{22} & C_{23} \\ \hline C_{31} & C_{k2} & C_{33} \end{array}\right] = \left[\begin{array}{c|c|c} A_{11} & A_{12} & A_{13} \\ \hline A_{21} & A_{22} & A_{23} \\ \hline A_{31} & A_{k2} & A_{33} \end{array}\right] \times \left[\begin{array}{c|c|c} B_{11} & B_{12} & B_{13} \\ \hline B_{21} & B_{22} & B_{23} \\ \hline B_{31} & B_{32} & B_{33} \end{array}\right]$$

$\frac{n}{3} * \frac{n}{3}$

$$C_{ij} = \sum_{1 \le s \le 3} A_{is} B_{sj}$$

$\Rightarrow T(n) = 3^3\, T(n/3) + O(n^2) = O(n^3)$

$\Rightarrow T(n) \overset{?}{=} q\, T(n/3) + O(n^2) \quad (q < 3^3)$

$\Rightarrow T(n) = O(n^{\log_3 q})$ by Master Thm.

4-9x

4-9a

$$C \quad = \quad A \quad \times \quad B$$

$$\left[\begin{array}{c|c|c|c} C_{11} & C_{12} & & C_{1k} \\ \hline C_{21} & C_{22} & & C_{2k} \\ \hline & & \cdots\cdots & \\ \hline C_{k1} & C_{k2} & & C_{kk} \end{array}\right] = \left[\begin{array}{c|c|c|c} A_{11} & A_{12} & & A_{1k} \\ \hline A_{21} & A_{22} & & A_{2k} \\ \hline & & \cdots\cdots & \\ \hline A_{k1} & A_{k2} & & A_{kk} \end{array}\right] \times \left[\begin{array}{c|c|c|c} B_{11} & B_{12} & & B_{1k} \\ \hline B_{21} & B_{22} & & B_{2k} \\ \hline & & \cdots\cdots & \\ \hline B_{k1} & B_{k2} & & B_{kk} \end{array}\right]$$

$\frac{n}{k} * \frac{n}{k}$

$$C_{ij} = \sum_{1 \le s \le k} A_{is} B_{sj}$$

$\Rightarrow T(n) = k^3\, T(n/k) + O(n^2)$

$\Rightarrow T(n) \overset{?}{=} q\, T(n/k) + O(n^2) \quad (q < k^3)$

$\Rightarrow T(n) = O(n^{\log_k q})$ by Master Thm.

"$\leq$" (upper bound)

* $f(n) = O(g(n))$ if we can find positive constants c and n_0

s.t. $0 \leq f(n) \leq cg(n)$ for all $n \geq n_0$

* Prove that $\underset{f}{3n^2 + 6n} = O(\underset{g}{n^2})$

Find c and n_0 such that

$\underset{f}{3n^2 + 6n} \leq \underset{cg}{cn^2}$ for all $n \geq n_0$

$\Rightarrow$ Choose $\begin{cases} c = 9 \\ n_0 = 1 \end{cases}$ or $\begin{cases} c = 4 \\ n_0 = 6 \end{cases}$

$T(n) = 2T(\lfloor n/2 \rfloor) + n$ (with $T(1) = 1$), find an upper bound

Claim: $\underset{f}{T(n)} = O(\underset{g}{n\lg n})$

($\exists$ c and n_0 such that $\underset{f}{T(n)} \leq c\underset{g}{n\lg n}$ for all $n \geq n_0$)

Claim: $T(n) = O(n\lg n)$ since $\boxed{T(n) \leq \underset{c}{3}n\lg n \text{ for all } n \geq \underset{n_0}{2}}$

Problem: How can we verify the solution of a recurrence?

⇨ By induction

Outline of Proof

Basis: ($n = n_0 = 2$) Show that $T(n) \leq 3n\lg n$ is true for $n = n_0 = 2$.

Induction: ($n > n_0$) Assume $T(x) \leq 3\, x \lg x$ for $x = 2, 3, \ldots, n - 1$.

Show that $T(n) \leq 3n\lg n$ is true for n as well.

$T(n) = 2T(\lfloor n/2 \rfloor) + n$ (with $T(1) = 1$)

Claim:

$T(n) = O(n\lg n)$ since

$T(n) \le 3n\lg n$ for all $n \ge 2$

c $\qquad$ n_0

Basis: ($n = n_0 = 2$)

$T(n_0) = T(2) = 2T(1) + 2 = 4$

$3n_0\lg n_0 = 3\ 2\lg 2 = 6$

$T(2) = 4 \le 6$

⇨ OK!

Induction: ($n > 2$)

Assume $T(x) \le 3\, x \lg x$

for $x = 2, 3, \ldots, n - 1$.

$T(n)$ $\qquad$ $x = \lfloor n/2 \rfloor \le n - 1$

$= 2\underline{T(\lfloor n/2 \rfloor)} + n$

$\le 2\,(\underline{3\lfloor n/2 \rfloor \lg \lfloor n/2 \rfloor}) + n$

$\le 3\, n \lg (n/2) + n$

$\le 3\, n \lg n - 3\, n \lg 2 + n$

$\le 3\, n \lg n - 3\, n + n$

$\le 3\, n \lg n - 2\, n$

$\le 3\, n \lg n$ (goal !) $\qquad$ OK !

⇨ Done !

⇨ $T(n) = O(n\lg n)$

$A(1) = 1$, $A(2) = 1$, $A(n) = A(n-1) + A(n-2)$ for $n \geq 3$

Prove $A(n) \leq 2^{n-2}$ for $n \geq 2$

By induction

Basis: $n_0 = 2, 3$

$A(2) = 1 \leq 2^{2-2} \leq 1$ OK !

$A(3) = A(1) + A(2) = 2 \leq 2^{3-2} \leq 2$ OK !

multi-value !!!

Induction: ~~(for n > 2)~~ (for n > 3)

Assume $T(x) \leq 2^{x-2}$ for $x = 2, 3, \ldots, n-1$

$A(n) = A(n-1) + A(n-2)$ /* $n-1, n-2 \in 2, 3, \ldots, n-1$

$\leq 2^{n-3} + 2^{n-4}$

$\leq 2 \times 2^{n-3}$

$\leq 2^{n-2}$

Wrong for n = 3 !!!

Done!

Induction: ~~(n > 2)~~ (for n > 3)

Assume $T(x) \leq 3 x \lg x$
for $x = 2, 3, \ldots, n - 1$.

$x = \lfloor n/2 \rfloor \leq n - 1$

$\lfloor n/2 \rfloor \geq 2$

$T(n)$

$= 2T(\lfloor n/2 \rfloor) + n$

$\leq 2 (3 \lfloor n/2 \rfloor \lg \lfloor n/2 \rfloor) + n$

$\leq 3 n \lg (n/2) + n$

$\leq 3 n \lg n - 3 n \lg 2 + n$

$\leq 3 n \lg n - 3 n + n$

$\leq 3 n \lg n - 2 n$

$\leq 3 n \lg n$ (goal !) OK !

Done !

$T(n) = 2T(\lfloor n/2 \rfloor) + n$ (with $T(1) = 1$)

Claim:

$T(n) = O(n\lg n)$ since

$T(n) \le 3n\lg n$ for all $n \ge 2$

c n_0

Basis: ($\cancel{n = n_0 = 2}$) (n = 2, 3)

$T(n_0) = T(2) = 2T(1) + 2 = 4$

$3n_0\lg n_0 = 3\ 2\lg 2 = 6$

$T(2) = 4 \le 6$

⇨ OK!

$3\ 3\lg 3 \sim 13.5$

$T(3) = 5 \le 13.5$

⇨ OK!

$T(n) = 2T(\lfloor n/2 \rfloor) + n$ (with $T(1) = 1$) 4-10a

Claim:

$T(n) = O(n\lg n)$ since

$T(n) \le 3n\lg n$ for all $n \ge 2$

c n_0

Basis: ($\cancel{n = n_0 = 2}$) (n = 2, 3)

$T(n_0) = T(2) = 2T(1) + 2 = 4$

$3n_0\lg n_0 = 3\ 2\lg 2 = 6$

$T(2) = 4 \le 6$

⇨ OK!

$3\ 3\lg 3 \sim 13.5$

$T(3) = 5 \le 13.5$

⇨ OK!

Claim:

$T(n) = O(n\lg n)$ ($\exists$ c and n_0 s.t.

$T(n) \le cn\lg n$ for all $n \ge n_0$)

Basis: ($n = n_0$)

$n_0 = 1$? $T(1) = 1 \le c\ 1\lg 1$? (×)

$n_0 = 2$? $T(2) = 4 \le c\ 2\lg 2$?

OK for $c \ge 4/(2\lg 2) = 2$

$n_0 = 3$? $T(3) = 5 \le c\ 3\lg 3$?

OK for $c \ge 5/(3\lg 3)$

$T(n_0) \le cn_0\lg n_0$

⇨ OK for $\begin{cases} n_0 \ge 2 \\ c \ge T(n_0)/n_0\lg n_0 \end{cases}$ 1

Induction: ~~($n > n_0$)~~ (for $n > 3$)

Assume $T(x) \le 3\, x \lg x$

for $x = 2, 3, \ldots, n-1$.

$x = \lfloor n/2 \rfloor \le n-1$

$\lfloor n/2 \rfloor \ge 2$

$$\begin{aligned}
&T(n)\\
&= 2T(\lfloor n/2 \rfloor) + n\\
&\le 2\,(3 \lfloor n/2 \rfloor \lg \lfloor n/2 \rfloor) + n\\
&\le 3\, n \lg (n/2) + n\\
&\le 3\, n \lg n - 3\, n \lg 2 + n\\
&\le 3\, n \lg n - 3\, n + n\\
&\le 3\, n \lg n - 2\, n\\
&\le 3\, n \lg n \quad \text{(goal !)} \qquad \text{OK !}
\end{aligned}$$

$\Rightarrow$ Done !

Induction: ($n > n_0$)

Assume $T(x) \le c\, x \lg x$

for $x = n_0, n_0+1, \ldots, n-1$.

⓪ $n_0 \le \lfloor n/2 \rfloor \le n-1$

$$\begin{aligned}
&T(n)\\
&= 2T(\lfloor n/2 \rfloor) + n\\
&\le 2\,(c \lfloor n/2 \rfloor \lg \lfloor n/2 \rfloor) + n\\
&\le c\, n \lg (n/2) + n\\
&\le c\, n \lg n - c\, n \lg 2 + n\\
&\le c\, n \lg n - c\, n + n\\
&\le c\, n \lg n - (c-1)\, n\\
&\le c\, n \lg n \quad \text{(goal !)}
\end{aligned}$$

$\Rightarrow$ OK for $c \ge 1$ ②

⓪ $\cap$ ① $\cap$ ② $\ne \varnothing$ $\Rightarrow$ Done !

Basis: ($n = n_0$)

$n_0 = 1$? $T(1) = 1 \le c\, 1 \lg 1$? (×)

$n_0 = 2$? $T(2) = 4 \le c\, 2 \lg 2$?

OK for $c \ge 4/(2\lg 2) = 2$

$n_0 = 3$? $T(3) = 5 \le c\, 3 \lg 3$?

OK for $c \ge 5/(3\lg 3)$

$T(n_0) \le c n_0 \lg n_0$

$\Rightarrow$ OK for $\begin{cases} n_0 \ge 2 \\ c \ge T(n_0)/n_0 \lg n_0 \end{cases}$ ①

Induction: ($n > n_0$)

$\Rightarrow$ OK for $c \ge 1$ ②

⓪ $n_0 \le \lfloor n/2 \rfloor \le n-1$

⓪ $n_0 = 2$ $\Rightarrow$ $n \ge 4$ $\Rightarrow$ basis: 2, 3

① $\begin{cases} 2 \text{ ok for } c \ge 2 \\ 3 \text{ ok for } c \ge 5/(3\lg 3) \sim 1.1 \end{cases}$ $\Rightarrow$ $c \ge 2$

⓪ $\cap$ ① $\cap$ ② $\Rightarrow$ $n_0 = 2$, $c = 2$

⓪ $n_0 = 3$ $\Rightarrow$ $n \ge 6$ $\Rightarrow$ basis: 3,4,5

① $\begin{cases} 3 \text{ ok for } c \ge 1.1 \\ 4 \text{ ok for } c \ge 1.08 \\ 5 \text{ ok for } c \ge 1.07 \end{cases}$ $\Rightarrow$ $c \ge 1.1$

⓪ $\cap$ ① $\cap$ ② $\Rightarrow$ $n_0 = 3$, $c = 1.5$

⓪ $n_0 = 4$ $\Rightarrow$ $n \ge 8$ $\Rightarrow$ basis: 4,5,6,7

$T(n) = T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 1$ (with $T(1) = 1$)

Prove $T(n) = 2n - 1$ (By induction)

Basis: $n = 1$

$T(1) = 1 = 2 * 1 - 1$ OK !

Induction:

Assume $T(x) = 2x - 1$ for $x < n$.

$T(n)$

$= \underline{T(\lfloor n/2 \rfloor)} + \underline{T(\lceil n/2 \rceil)} + 1$

$= (\underline{2 \times \lfloor n/2 \rfloor - 1}) + (\underline{2 \times \lceil n/2 \rceil - 1}) + 1$

$= 2n - 1$ (goal !) OK !

⇨ Done !

Prove $T(n) \le 2n$

Basis: $n = 1$

$T(1) = 1 \le 2 * 1$ OK !

Induction:

Assume $T(x) \le 2x$ for $x < n$.

$T(n)$

$= \underline{T(\lfloor n/2 \rfloor)} + \underline{T(\lceil n/2 \rceil)} + 1$

$\le \underline{2 \times \lfloor n/2 \rfloor} + \underline{2 \times \lceil n/2 \rceil} + 1$

$\le 2n + 1$

$\le 2n$ (goal !)

⇨ Fail ! Why???

$T(n) = T(n-1) + 1$ (with $T(1) = 1$) * Answr: $T(n) = n$

Prove $T(n) = 1$ (By induction)

Basis: $n = 1$

$T(1) = 1$ OK !

Induction:

Assume $T(x) = 1$ for $x = 1, 2, \ldots, n - 1$.

$T(n)$

$= \underline{T(n-1) +} 1$

$= 1 + 1$

$= 2$

⇨ We can not prove $T(n) = 1$.
But, we conclude that $T(n) = 2$. ???!!!

Find upper bound of $T(n) = T(n-1) + 2n$ (with $T(1) = 1$)

$T(x) = 2x + T(x-1)$

$$
\begin{aligned}
T(n) &= 2n + \underline{T(n-1)} \\
&\le 2n + 2(n-1) + \underline{T(n-2)} \\
&\le 2n + 2(n-1) + 2(n-2) + \underline{T(n-3)} \\
&\quad \vdots \\
&\le 2n + 2(n-1) + 2(n-2) + \ldots + 2(2) + T(1) \\
&\le 2(2 + 3 + \ldots + n) + 1 \\
&\le 2(1 + 2 + 3 + \ldots + n) - 1 \\
&\le n(n+1) - 1 \\
&= O(n^2)
\end{aligned}
$$

Find upper bound of $T(n) = 3T(\lfloor n/4 \rfloor) + n$ (with $T(1/0) = \theta(1)$)

$T(x) = x + 3T(\lfloor x/4 \rfloor)$

$$
\begin{aligned}
T(n) &= n + 3\underline{T(\lfloor n/4 \rfloor)} \\
&\le n + 3\{\lfloor n/4 \rfloor + 3T(\lfloor \lfloor n/4 \rfloor /4 \rfloor)\} \\
&\le n + 3\lfloor n/4 \rfloor + 3^2\underline{T(\lfloor n/4^2 \rfloor)} \\
&\le n + 3\lfloor n/4 \rfloor + 3^2\{\lfloor n/4^2 \rfloor + 3T(\lfloor \lfloor n/4^2 \rfloor /4 \rfloor)\} \\
&\le n + 3\lfloor n/4 \rfloor + 3^2\lfloor n/4^2 \rfloor + 3^3\underline{T(\lfloor n/4^3 \rfloor)} \\
&\quad \vdots \\
&\le n + 3\lfloor n/4 \rfloor + 3^2\lfloor n/4^2 \rfloor + 3^3\lfloor n/4^3 \rfloor + \ldots + 3^k\underline{T(\lfloor n/4^k \rfloor)}
\end{aligned}
$$

$\frac{n}{4^k} \le 1 \Rightarrow k \ge \lg_4 n$

T(1) or T(0)

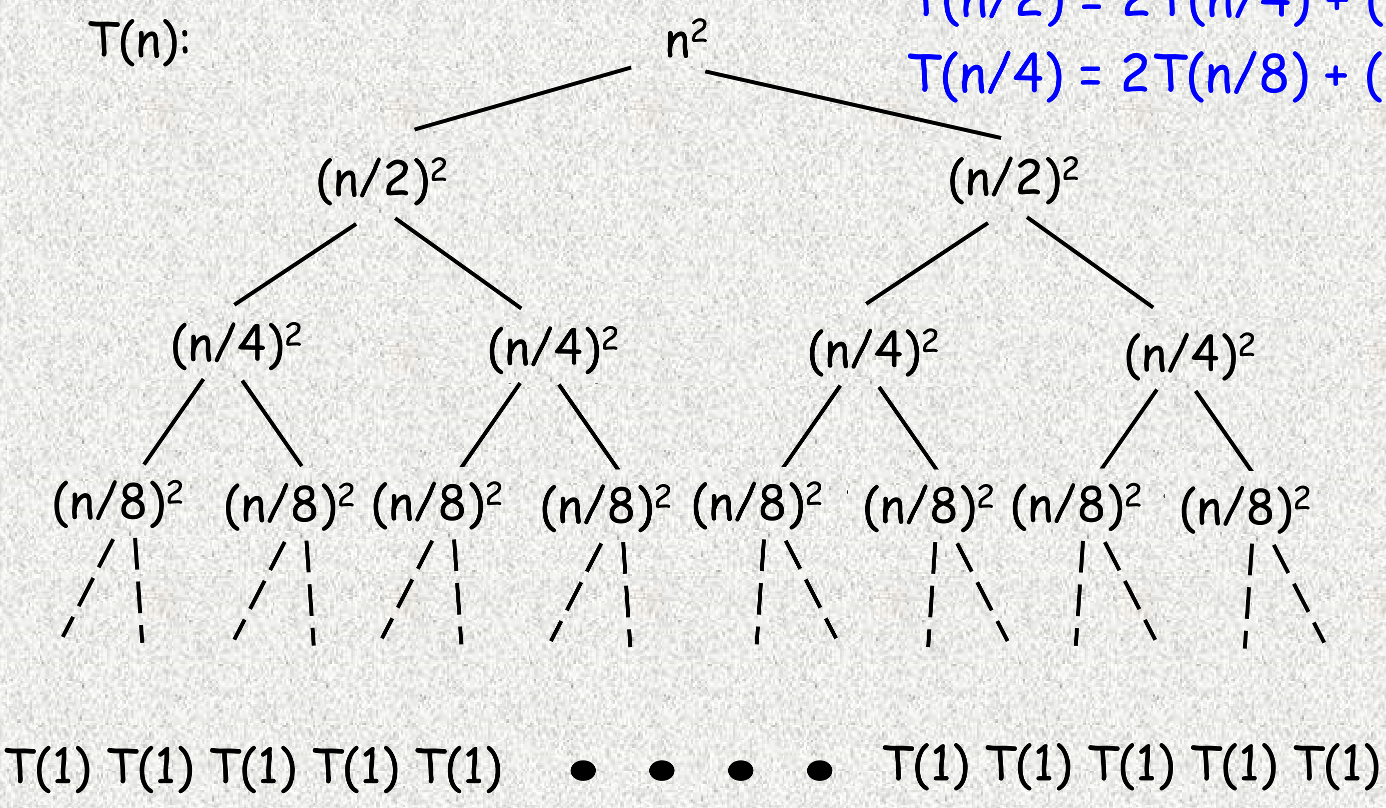
$$
\begin{aligned}
&\le n + 3n/4 + 3^2 n/4^2 + 3^3 n/4^3 + \ldots + 3^k \theta(1) \\
&\le O(n) + 3^k \theta(1)
\end{aligned}
$$

等 比 會 收 斂 ！

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

$T(n/2) = 2T(n/4) + (n/2)^2$

$T(n/4) = 2T(n/8) + (n/4)^2$



Assume that we do not take ceiling for both.

* cost(internal-nodes) of a level is at most n.
(Prove that $n/3 + 2n/3 \le n$ for $\lfloor\,\rfloor\lfloor\,\rfloor$, $\lfloor\,\rfloor\lceil\,\rceil$, and $\lceil\,\rceil\lfloor\,\rfloor$.)

* $n/3 + 2n/3 \le n$ for $\lfloor\,\rfloor\lfloor\,\rfloor$, $\lfloor\,\rfloor\lceil\,\rceil$, and $\lceil\,\rceil\lfloor\,\rfloor$

Proof for $\lfloor\,\rfloor\lceil\,\rceil$

Case 1. $n = 3k$
$\lfloor n/3 \rfloor + \lceil 2n/3 \rceil = k + 2k = 3k = n$

Case 2. $n = 3k + 1$
$\lfloor n/3 \rfloor + \lceil 2n/3 \rceil = k + \lceil 2k + 2/3 \rceil = k + 2k + 1 = 3k + 1 = n$

Case 3. $n = 3k + 2$
$\lfloor n/3 \rfloor + \lceil 2n/3 \rceil = k + \lceil 2k + 1 + 1/3 \rceil = k + 2k + 2 = 3k + 2 = n$

4-15x

4-15a

Assume that we do not take ceiling for both.

* cost(internal-nodes) of a level is at most n.
(Prove that $n/3 + 2n/3 \le n$ for $\lfloor\,\rfloor\lfloor\,\rfloor$, $\lfloor\,\rfloor\lceil\,\rceil$, and $\lceil\,\rceil\lfloor\,\rfloor$.)

How to compute the number of leaves L?

$2^{\lg_{(3/2)} n} = n^{\lg_{(3/2)} 2} = O(n^{1.xxx})$

* From $L \le 2^{\lg_{(3/2)} n}$, we have $L = \omega(n\lg n)$, which is larger than $O(n\lg n)$. （太 悲 觀）

* Avoid the computation of L. (text book)
Prove $O(n\lg n)$ is correct by substitution method.

* Prove by induction that $L \le n$.
(Try it!)

* If we take ceiling for both, $\lceil n/3 \rceil + \lceil 2n/3 \rceil$ may be larger than n. For example, 11 is partitioned into 4 and 8. Then,

(a) cost(internal-nodes) of a level may be larger than n.

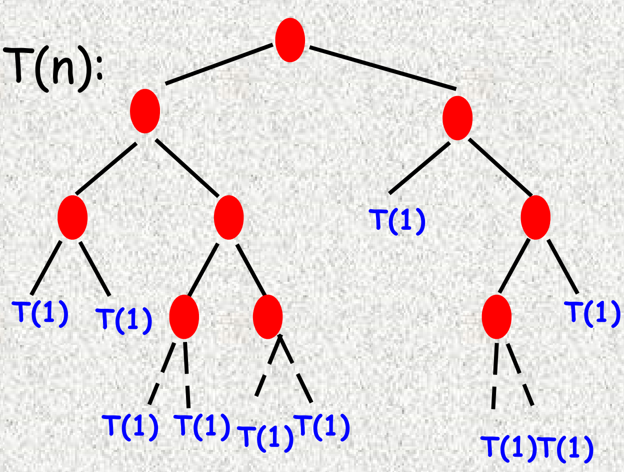
(b) The number of leaves, L, is unknown.

(c) Using substitution method, T(n) = O(n lg n) still can be proved. (with some effort!)

* Usually, cost(leaves) can be ignored; however, we should consider it in a formal proof.

Why cost(leaves) can be ignored usually ?

T(n):



* V: internal nodes L: leaves
* for binary tree |V| = |L| + 1
* usually,
  - T(1) = 1 and cost(L) = |L|
  - $t(v) \geq 1$ and $cost(V) = \sum_v t(v) \geq |V| \geq |L| + 1$
  - that is, $cost(V) \geq cost(L)$

$o(n^{\log_b a})$ $\qquad$ $\omega(n^{\log_b a})$

case 1	case 2	case 3
$O(n^{\log_b a - \varepsilon})$	$\Theta(n^{\log_b a})$	$\Omega(n^{\log_b a + \varepsilon})$

f(n)

Ex.

$T(n) = 9T(n/3) + f(n)$

$\log_b a = \log_3 9 = 2$

$O(n^{2-\varepsilon}) = O(n^2/n^\varepsilon)$	$\Theta(n^2)$	$\Omega(n^{2+\varepsilon}) = \Omega(n^2 n^\varepsilon)$

$f(n) = n^2$ -> case 2

$f(n) = n^{1.9}$ -> case 1

$f(n) = n^{2.2}$ -> case 3

$$\underbrace{n^2/\lg n \qquad\qquad n^2 \lg n}_{\text{cannot apply M.T.}}$$