Partition of Qsort a_1 a_2 a_3 a_4 a_5 8-1a

$a_1 : a_5$

a_1 a_2 a_3 a_4 a_5 $\leq$ $>$ a_1 a_2 a_3 a_4 a_5

$a_2 : a_5$ $a_2 : a_5$

a_1 a_2 a_3 a_4 a_5 $\leq$ $>$ a_1 a_2 a_3 a_4 a_5 a_2 a_1 a_3 a_4 a_5 $\leq$ $>$ a_1 a_2 a_3 a_4 a_5

$a_3 : a_5$ $a_3 : a_5$ $a_3 : a_5$ $a_3 : a_5$

a_1 a_2 a_3 a_4 a_5 a_1 a_2 a_3 a_4 a_5 $\vdots$ $\vdots$ $\vdots$

$a_4 : a_5$ $a_4 : a_5$

$\leq$ $>$

$\vdots$ a_1 a_2 a_4 a_3 a_5 a_1 a_2 a_3 a_4 a_5

a_1 a_2 a_4 a_5 a_3 ------ a_1 a_2 a_5 a_4 a_3 ------------

$a_1 : a_4$ $a_1 : a_2$

$\vdots$ $\vdots$ **Qsort: recursive calls**

8-2a

$O(n^2)$ *O* - upper bound:

$O(n^{1.5})$ algorithm 的 品 質

$O(n \lg n)$

$\theta(n \lg n)$ $O = \Omega \implies$ optimal

$\Omega(n \lg n)$

$\Omega(n \lg \lg n)$

$\Omega(n \lg^* n)$ Ω - lower bound:

$\Omega(n)$ problem 的 困 難 度
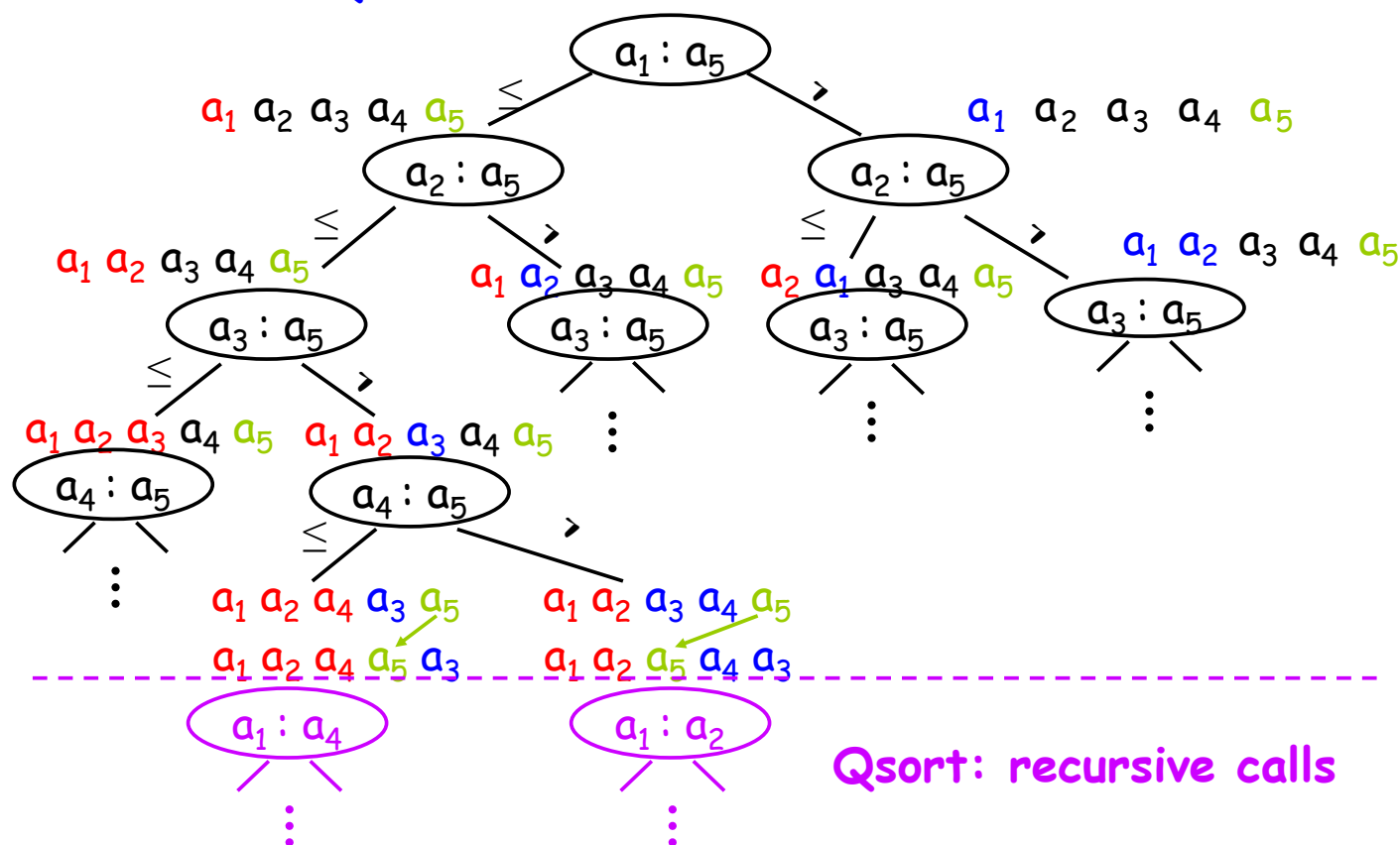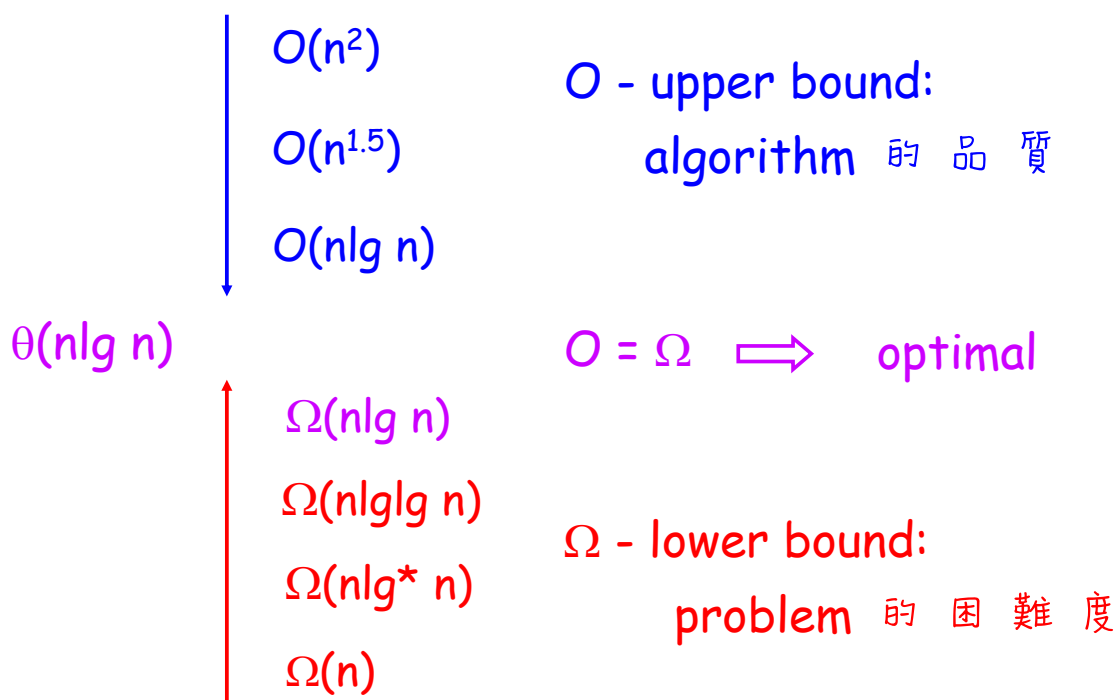
* Ω 很 重 要 , 但 也 很 困 難 , 所 以 好 的 結 果 不 多

n integers, $k = n^d$

d lg n bits

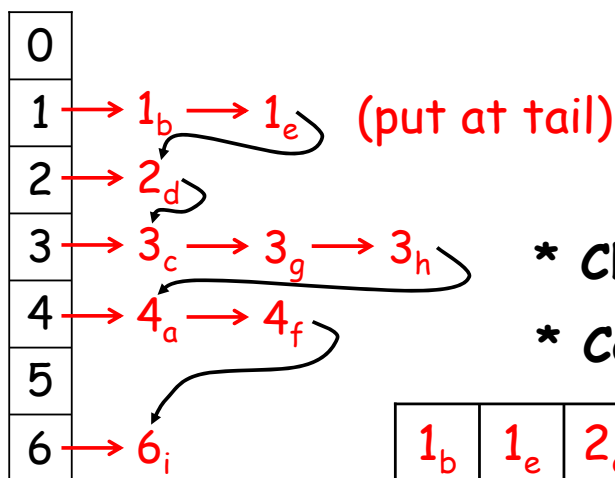| 1	1	1	0	1	1	0	0	1	0	1	1	1	1	1	0	1	1	0	0	1	0	1	1

lg n bits

$k = 2^{lg\ n} = n$
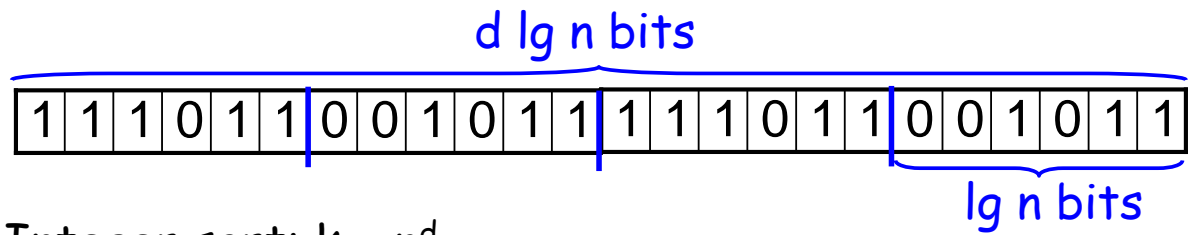
Integer sort: $k = n^d$

$O(n + k) = O(n + n^d)$

Radix sort: d, $k = n$

$O(d(n + k)) = O(dn)$

If d is a constant, it needs linear time !

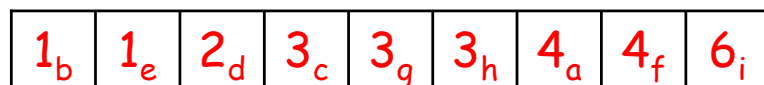4_a 1_b 3_c 2_d 1_e 4_f 3_g 3_h 6_i

* put all items into buckets O(n)

| 0 |
| 1 | → 1_b → 1_e (put at tail)
| 2 | → 2_d
| 3 | → 3_c → 3_g → 3_h
| 4 | → 4_a → 4_f
| 5 |
| 6 | → 6_i

* Chain all lists into one O(k)

* Convert into array O(n)

O(k)

1_b	1_e	2_d	3_c	3_g	3_h	4_a	4_f	6_i

Time: O(n + k)