

Algorithms & complexity 2014 EE10-C 1

①

$$a) \quad T(n) = 4T(n/2) + n^3$$

$$a = 4 \quad ; \quad b = 2 \quad d = 3.$$

$$\log_b a = \log_2 4 = 2 < d = 3.$$

$$\text{Hence } T(n) = O(n^3)$$

$$b) \quad T(n) = 17T(n/4) + n^2$$

$$a = 17 \quad b = 4 \quad d = 2$$

$$\log_b a = \log_4 17 > \log_4 16 = 2 > d$$

$$\text{Hence } T(n) = O\left(n^{\log_4 17}\right)$$

$$c) \quad T(n) = 9T(n/3) + n^2$$

$$a = 9 \quad b = 3 \quad d = 2$$

$$\log_b a = \log_3 9 = 2 = d = 2$$

$$\Rightarrow T(n) = O(n^2 \log n).$$

$$d) \quad T(n) = T(\sqrt{n}) + 1$$

$$\text{Let } n = 2^k$$

$$T(2^k) = T(2^{k/2}) + 1$$

$$= O(k) = O(\log n).$$

2

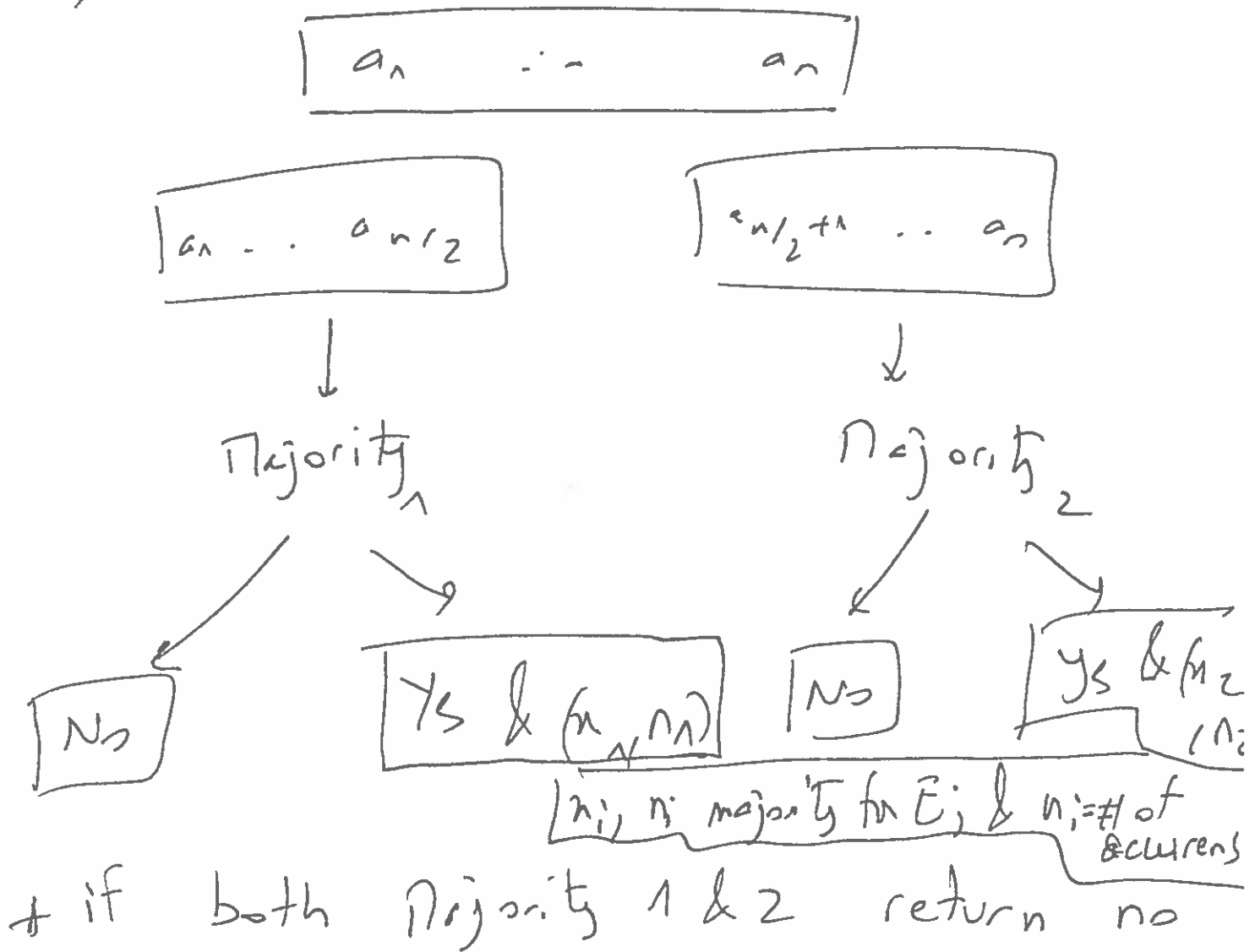
a) For each a_i the for loop on j counts the number of times a_i appears in the list.

For each a_i it then checks whether it appears more than $n/2$ times.

The algorithm then either returns "there is no majority element" if none of the a_i appears more than $n/2$ times or returns the unique element that appears more than $n/2$ times.

The algorithm runs two for loops
 on $i \in \Delta$, from 1 to n so its
 complexity is $O(n^2)$.

b)



then there is no majority number since
 no say, that all a_i occur fewer than $n/4$

times in each of the lists \bar{e}_1 & \bar{e}_2 (5
& no fewer than $n/2$ times in total).

* if Majority 1 returns (n_1, n_1)
& Majority 2 returns no.

Here we need the number of occurrences
of n_1 in \bar{e}_2, n_2 , if $n_1 + n_2 > n/2$
then yes n_1 is a majority element
Otherwise no.

* Similarly if Majority 2 returns (n_2, n_2)
& Majority 1 returns no.

* ~~Finally~~ if Majority 1 & Majority
2 return (n, n) & (n, n) then
clearly n is the majority element.

* Finally if Majority 1 returns (u_1, n_1) (6)
& Majority 2 returns (u_2, n_2) $u_1 \neq u_2$

We need to look at the occurrence
of u_1 in $E_2 = m_{12}$

$$\& u_2 \text{ in } E_1 = m_{21}$$

then see whether $n_1 + m_{12} \geq$

$n_2 + m_{21}$ is larger than $n/2$.

Given a list and a number let

$\text{Num}(E, u) = \# \text{ appearances of } u \text{ in } E.$

Majority ($a_1 \dots a_n$)

(7)

Let $E_1 = a_1 \dots a_{n/2}$; $\Pi_{aj_1} = \text{Majority}(E_1)$
 $E_2 = a_{n/2+1} \dots a_n$; $\Pi_{aj_2} = \text{Majority}(E_2)$

If $\Pi_{aj_1} = (*, 0)$ and $\Pi_{aj_2} = (*, 0)$

then $(*, 0)$

else

If $\Pi_{aj_1} = (x_1, n_1)$ & $\Pi_{aj_2} = (*, 0)$.

then

if $n_1 \leq n_1 + \text{Non}(x_1, E_2) > n/2$

then (x_1, n_1)

else $(*, 0)$.

else
if

$\Pi_{aj_2} = (x_2, n_2)$ & $\Pi_{aj_1} = (*, 0)$

then if $n_2 \leq n_2 + \text{Non}(x_2, E_1) > n/2$

then (x_2, n_2)

else $(*, -)$

else

if $\Pi_{aj_1} = (x_1, n_1)$ & $\Pi_{aj_2} = (x_2, n_2)$

then if $x_1 = x_2$ then

$(x_1, n_1 + n_2)$.

else

if $n_1 \leftarrow n_1 + \text{Num}(x_1, \bar{E}_2) > n/2$
then (x_1, n_1)

else

if $n_2 \leftarrow n_2 + \text{Num}(x_2, \bar{E}_1) > n/2$
then (x_2, n_2)

else (x, ∞) .

REMARK: Students can either provide text or a pseudocode [Bonus (+2) if they provide both] [P]

Complexity:

$$C(n) = 2 \left[C(n/2) + n/2 \right]$$

Call for
 E_1, E_2

need to
count the number
of time an element
appears in one of
the subsets E_1 or E_2

$$C(n) = 2C(n/2) + n$$

Master theorem $\rightarrow C(n) = \Theta(n \log n)$.

(9)

(5)

a) We have to look at all possible subwords in both words $2 \cdot 2^h$.

b)

i) $a_i \neq b_j$ this means that a_i & b_j cannot belong to the same subword.

Hence either a_i belongs to the common subword & $p(i, j) = p(i-1, j-1)$

or b_j & $p(i, j) = p(i-1, j)$

ii) $a_i = b_j = x$ then either x

belongs to the common subword in which

(a) $p(i, j) = p(i-1, j-1) + 1$

or not & $p(i, j) = p(i-1, j-1)$.

iii) Putting all the pieces together

10

we have

$$p(i, j) = \max \left(p(i, j-1), p(i-1, j), p(i-1, j-1) + f(a_i, b_j) \right)$$

iv) either i or j decreases at each step
and we have at most mn steps yielding
a complexity of $O(mn)$.

c/

b	0	1	2	2	3	4	4
a	0	1	2	2	3	3	3
d	0	1	2	2	2	3	3
b	0	1	1	2	2	3	3
c	0	1	1	2	2	2	2
b	0	1	1	1	1	2	2
a	0	0	0	0	1	1	1
φ	0	0	0	0	0	0	0
	φ	b	d	c	a	<u>b</u>	a.

Hence b d a b