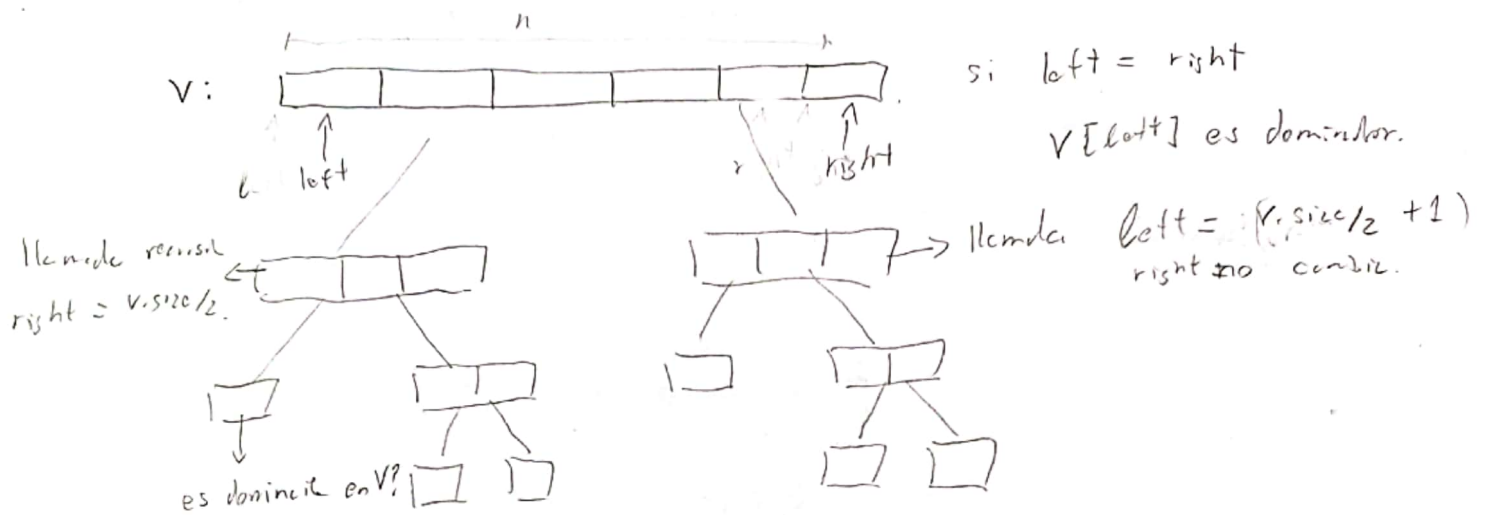


1.18/

n tarsets, única operac. t_1 equivalent t_2 .
 conjunt de tarsets equivalent dominon $t_f > \lceil n/2 \rceil$



tarset. tarseta - dominon (vector <tarsets> v , $left$, $right$) q.

si $left == right$
 devolvemos la tarseta $v[left]$

tarseta - dominon - $left = tarseta - dominon(v, left, v.size/2)$

contiene los vecs q aparece tarseta - $dom - left$ en v
 si $> \lceil n/2 \rceil$ lo devolvemos.

si $right = \dots$ ($v, (v.size/2 + 1), right$)
 contiene \dots tarseta - $dom - right$ en v
 si $> \lceil n/2 \rceil$ lo devolvemos.

si no hemos devuelto nada, entonces no hay elemento dominon

Entonces no hay elemento dominon: y para encontrar los vecs q aparece
 un tarseta cuesta $O(n)$.

Por tanto tenemos: $T(n) = 2T(n/2) + O(n)$

$T(n) = \Theta(n \log n)$

Maj tot \Rightarrow Maj meiotot.

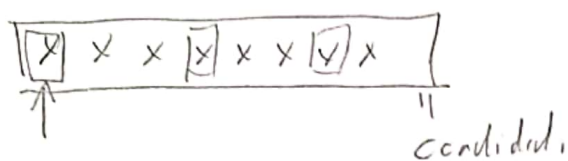
\nRightarrow Maj meiotot.



扫描全能王 创建

1.18

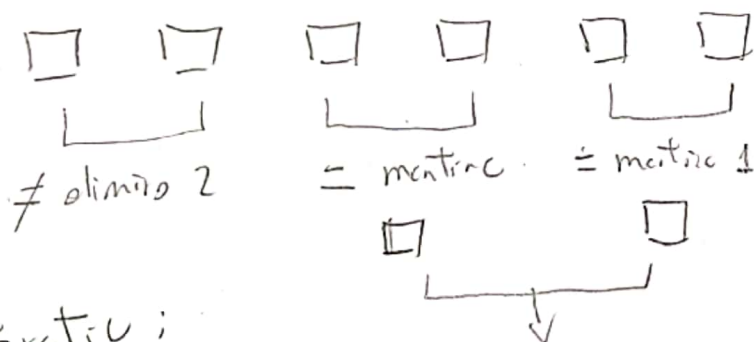
MJRTY - Boyer i Moore.



$$X_1 \neq X_2$$

-si elimino 2, el
majoritari segueix
sent majoritari.

Divide conquer:



Iterative:

dot a be n credit card

$O(n)$

count = 1
candidate = a[0]

for i = 1 to n

if count == 0

count = 1

candidate = a[i]

if not equal(candidate, a[i])

count = 1

else

count += 1

check if candidate appear more than $n/2$.



扫描全能王 创建