

// Strougsort:

```

• trie (t, g, d)
|
|  if t[g] < t[d]
|  |
|  |  swap (t[g], t[d])
|  |
|  |  if g-d+1 > 2
|  |  |
|  |  |  tiers = (g-d+1) // 3
|  |  |
|  |  |  trie (t, g, d-tiers)
|  |  |
|  |  |  trie (t, g+tiers, d)
|  |  |
|  |  |  trie (t, g, d-tiers)
|
|

```

trie parre-valon

complexité des tests

• Complexité:

$$T(n) = 3 * T(\frac{2}{3}n) + \Theta(1)$$

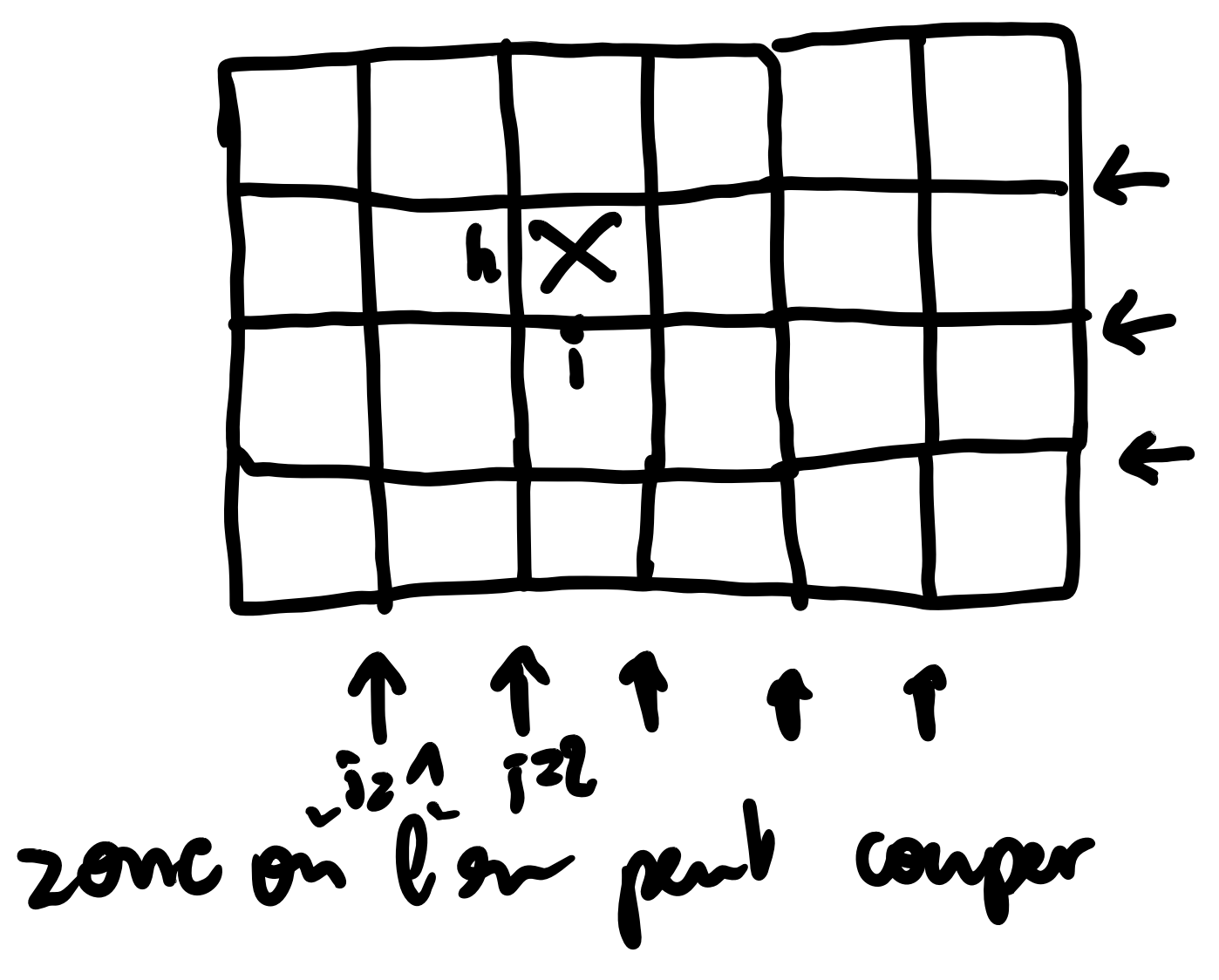
$$MT: a=3$$

$$b=\frac{3}{2}$$

Donc cas 1: $T(n) \in \Theta(n^{\log \frac{3}{2}})$

• on trie 2/3 ... à la fin trie
↳ termine.

- suite TD4 -



$$jeu(H, L, h, l) =$$

False si $H=L=h=l=1$

$$\neg (\bigwedge_{i=1}^{l-1} jeu(H, L-i, h, l-i))$$

↳ // couper à gauche
↳ // couper à droite

$$\wedge (\bigwedge_{i=l}^L jeu(H, i, h, l))$$

$$\wedge (\bigwedge_{i=1}^{H-1} jeu(H-i, L, h, l))$$

$$\wedge (\bigwedge_{i=h}^{H-1} jeu(i, L, h, l))$$

TDSV

Exercice 1

	0	1	2	3	4	5	6	7
0	0	1	2	1	2	3	2	3
1	X	0	1	0	1	2	3	4
2	X	X	0	1	1	2	3	4
3	X	X	X	0	0	1	2	3
4	X	X	X	X	0	1	2	3
5	X	X	X	X	X	0	1	2
6	X	X	X	X	X	X	0	1
7	X	X	X	X	X	X	X	0

$a \rightarrow a$ 0
 $ab \rightarrow aba$ 1
 $ac ec b \rightarrow bacecab$ 2

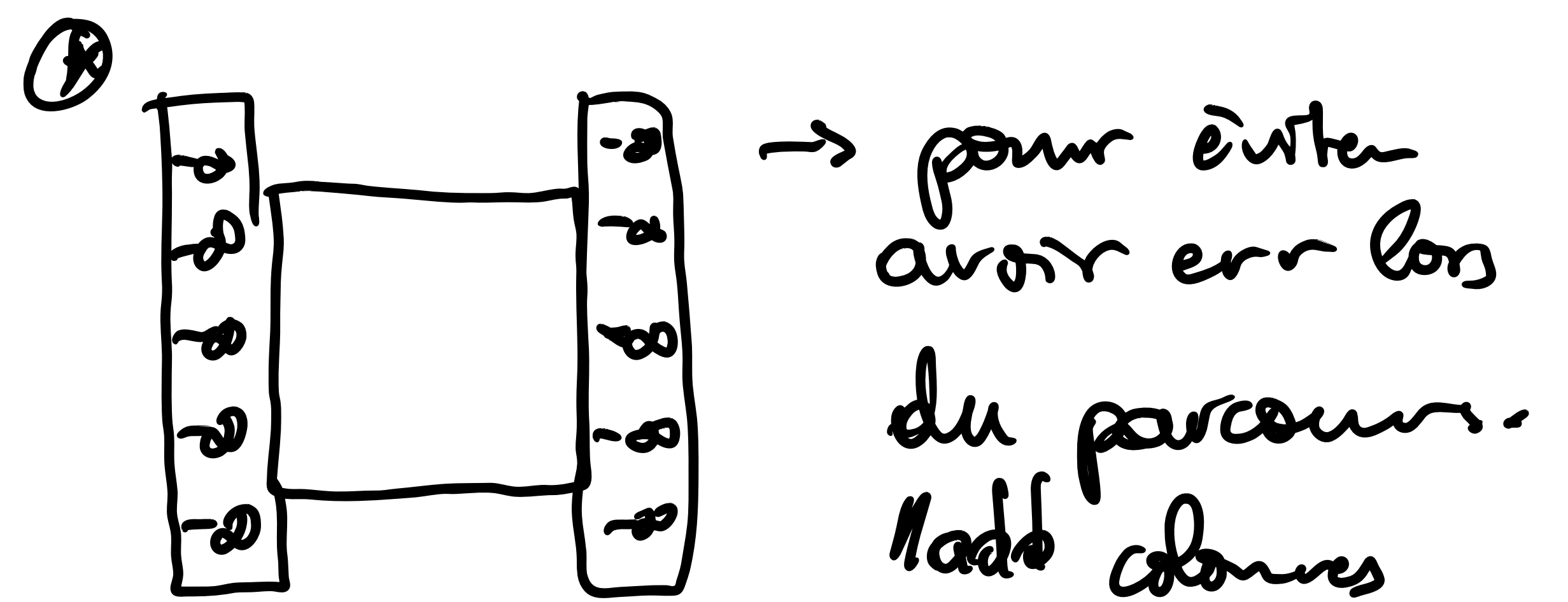
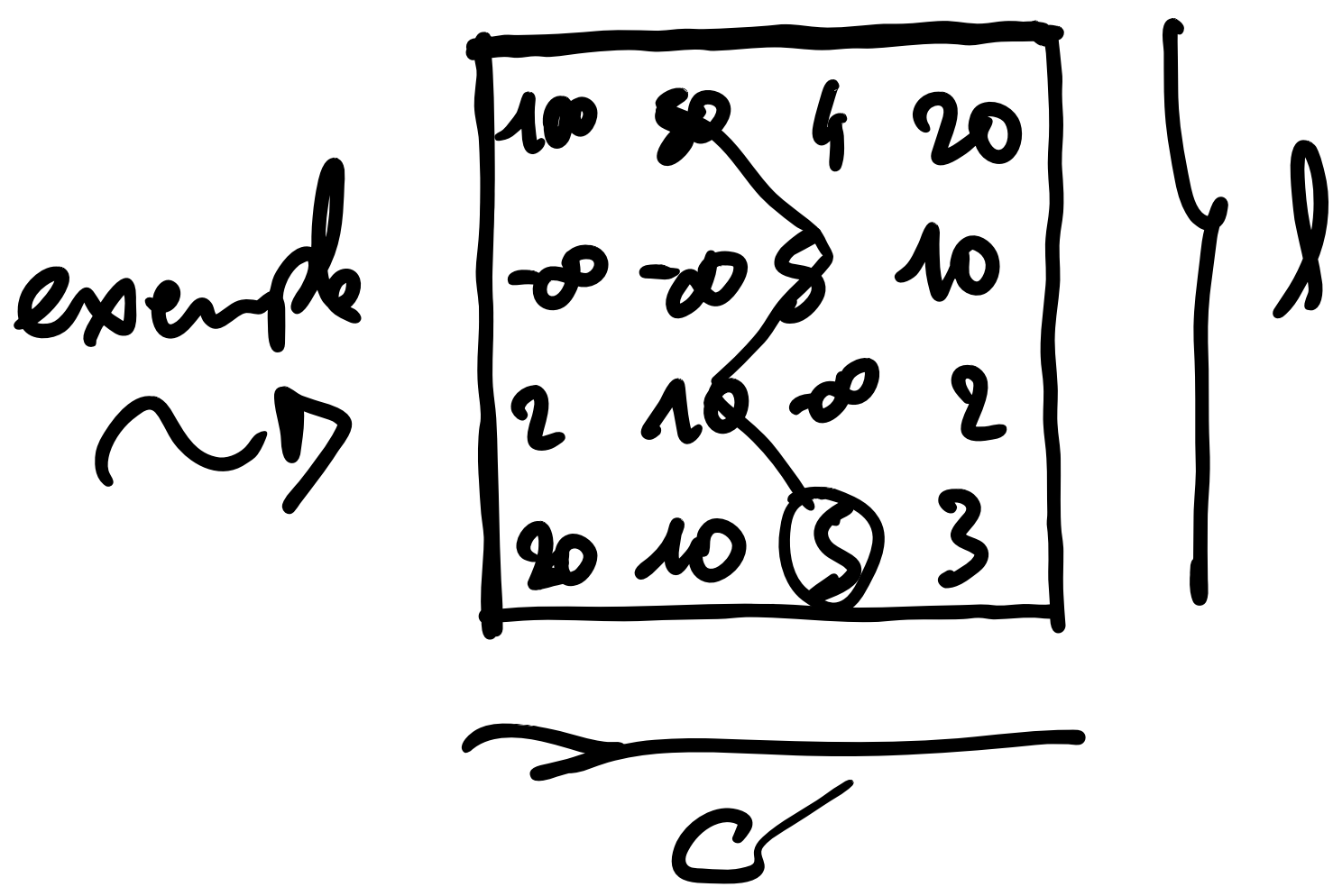
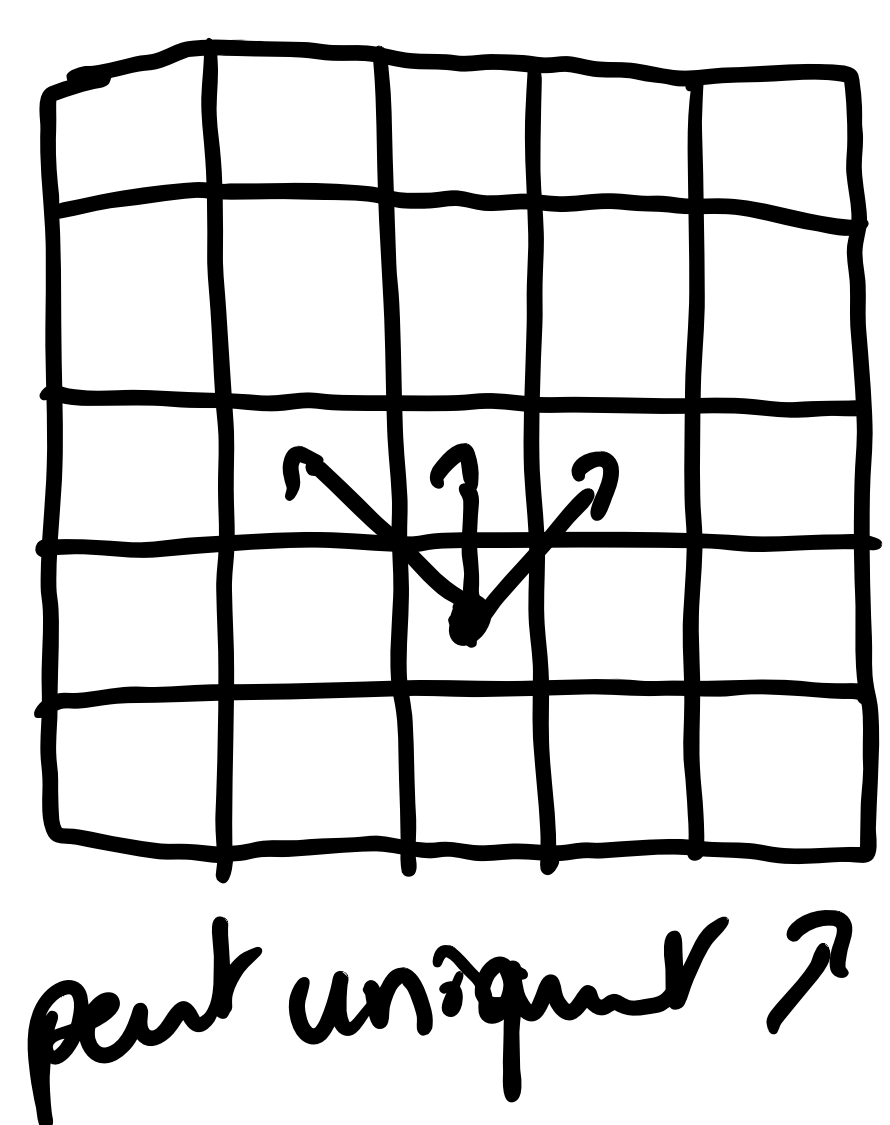
$$W = @ @ | @ cc | gab$$

↳ remplir en diagonale.

$$\min_{\text{pour } g \leq d} (w, g, d) = \begin{cases} \min(w, g+1, d-1) & \text{si } w[g] = w[d] \text{ et } g < d \\ \min(\min(w, g, d-1), \min(w, g+1, d)) + 1 & \text{si } w[g] \neq w[d] \end{cases}$$

7. n^2

Exercice 2:



$$\text{score}(i, j) = \begin{cases} \text{Bonus}[i][j] & , \text{si } i = l-1 \\ \max(\text{score}(i+1, j-1), \text{score}(i+1, j), \text{score}(i+1, j+1)) + \text{Bonus}[i][j] & \text{si } i < l-1 \\ -\infty & , \text{si } j = -1 \text{ ou } j = c \quad \textcircled{*} \end{cases}$$

Rq: ne pas oublier déterminer le point départ. (si possible)

Pour récupérer le chemin, il faut s'en rappeler en la stockant dans un array

$\begin{bmatrix} & 1 & & \\ & & -1 & \\ 1 & & & \\ & -1 & & \end{bmatrix}$	<p>// -1 aller à gauche</p> <p>// 1 aller à droite.</p>
--	---