

-TD10-

Exo 2:

Rm 1: 8h45 BE, ES, DK

" 2: 13h11 FR, GR

" 3: 17h22 LU, DE, IT, PL

" 4: 22h02 PT

Algo:

Dans les ministres qui n'ont pas encore assisté à une réunion on prend celui qui part le plus tôt appelé x , on fait la réunion à $d_x - 1h$ et on y met tous les ministres présents qui n'ont pas encore assisté à une réunion
prétraitement: trier les ministres par heure de départ Complexité: $O(n \log n)$

Optimalité:

On prend une sol^o optimale I_2 de la sol^o "gloutonne" et on montre qu'on peut la transformer vers la sol^o "gloutonne"

sol^o optimale: $x_1 x_2 \dots x_i x_{i+1}$

sol^o gloutonne: $x_1 x_2 \dots x_i x'_{i+1}$ avec $x_{i+1} \neq x'_{i+1}$

Est-ce qu'on peut remplacer x_{i+1} par x'_{i+1} dans la sol^o opt ?
on sait $x_{i+1} < x'_{i+1}$ Donc, réponse oui.

Exo paranthèse:

Algo:

on se donne un cpt d , initialisé 0

on lit de $G \rightarrow D$

Si ($\Rightarrow d++$

Si non) $\Rightarrow d--$

et à la fin d doit être 0 et ne doit jamais < 0 .

) (() (() (

trouver un algo glouba qui resolv. ce prob.

↳ faire parcours $G \rightarrow D$ puis $D \rightarrow G$

↳ si passe sous barre neg, ne pas perdre la parenthèse.

) (() (() (

⇒ ① de gauche → droite on efface tous les) qui ont cpt 1
 ② droite → gauche : synchron.

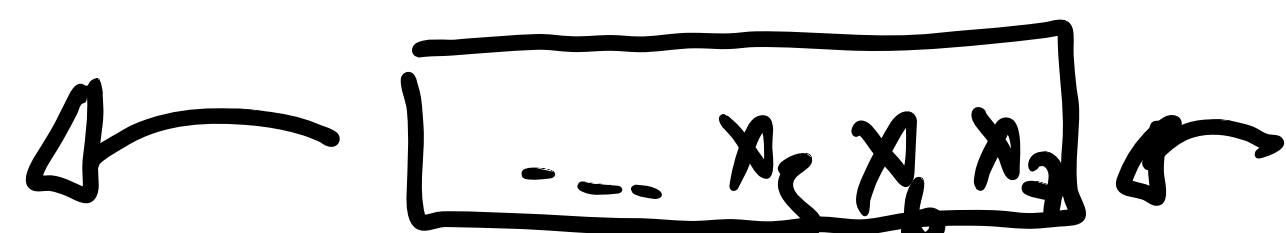
→ TD 11 -

Exo 1:

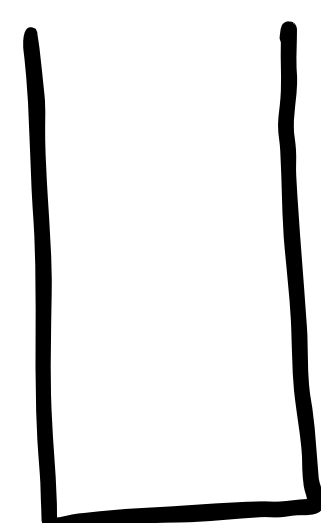
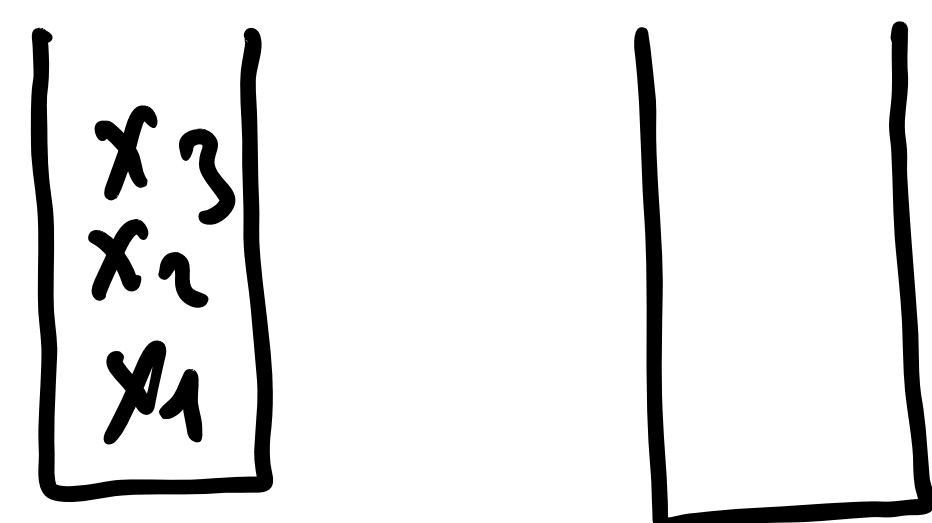
Push(x_1) Push(x_2) Push(x_3) Push(x_4) ... Push(x_8) SSSS



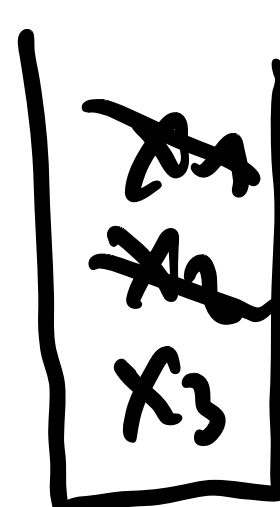
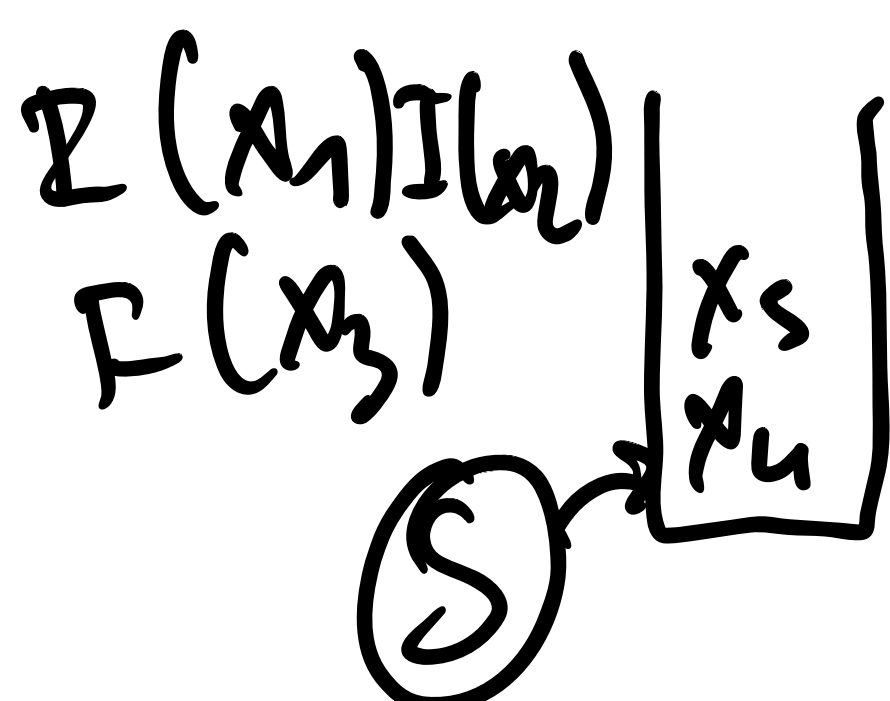
Pile : I(x_1) ... I(x_8) SSSS



On implémente une file avec deux piles P_c P_s



insérer :
push sur P_c



Supprimer :

Si P_s est vide

on copie tout P_c vers P_s

(l'ancien est inversé) et on pop de P_s

I(x_4)
I(x_5)

sinon : on pop P_s

Méthode aggrégation:

on a T opérations: $T = T_1 + T_2$
insérer supprimer

coût des T_1 insérations: T_1

coût des T_2 suppressions: T_2

$2T_1$ chaque élément insérer et copier une fois
par pop en P_S

Marking des suppr insér

pas P_S mais P_S

coût amorti: $\frac{\text{coût total}}{\# \text{opérations}} = \frac{T_1 + T_2 + 2T_1}{T_1 + T_2} \leq \frac{T_1 + T_2 + 2T_1 + 2T_2}{T_1 + T_2} = \frac{3(T_1 + T_2)}{T_1 + T_2} = 3$

Méthode copiables

on donne un coût de 3 à l'insertion: 1 pour le push de l'insert et un coût de 2 pr recopier plus tard. On donne un coût de 1 à la suppression

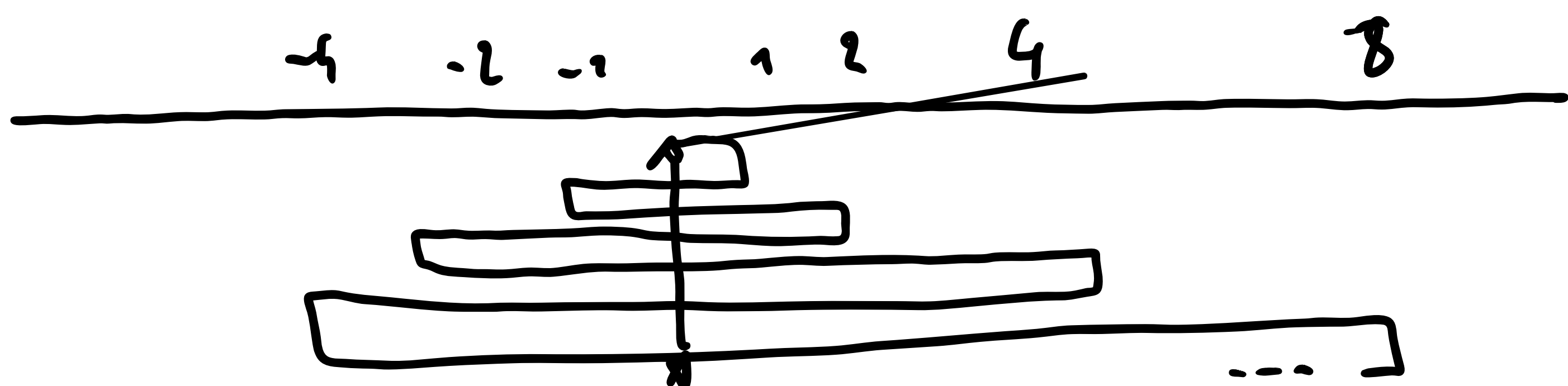
Méthode du potentiel
pile entrée
pile sortie

$$\Phi(P_c, P_s) = 2|P_c|$$

donner coût pas opérations mais structure de données.

Opérations	différence de potentiel	coût amorti	coût amorti
insérer	1	2	3
supprimer si pile vide	$2 P_c + 1$	$2 P_c $	1
suppr si P_S pas vide	1	0	0

Exercice 2:



Algo: on commence avec $d=1$

La vache va à droite de distance d

Elle retourne et parcourt distance $2d$

Elle retourne et parcourt distance d

On recommence avec $2d, 4d, 8d, \dots$

{ Quel est la distance parcourue avant de trouver l'ouverture?

Si l'ouverture est à gauche c'est pire qu'à droite

Supposons que l'ouverture est à $(-k, 1)$

$$2^j < k \leq 2^{j+1}$$

$$\text{distance parcourue} = \sum_{i=0}^j 4 \cdot 2^i + 2 \cdot 2^{j+1} + k$$

$$= 4 \sum_{i=0}^j 2^i + 2 \cdot 2^{j+1} + k$$

$$= 4(2^{j+1} - 1) + 2 \cdot 2^{j+1} + k \leq 4 \cdot 2 \cdot 2^j + 4 \cdot 2^j + k$$

$$\leq 8k + 4k + k \leq$$

$$\leq 13k$$