

4: Structure de données

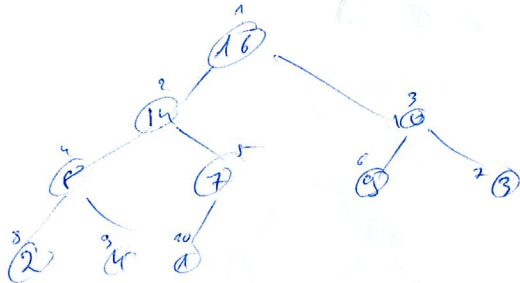
Exemple: Tas (max)

La structure de tas est un tableau qui peut être vu comme un arbre binaire A presque parfait complet tel que pour tout $i > 1$,



$$A(\text{Père}(i)) \geq A(i)$$

Exemple:



→

16	14	10	3	7	9	3	2	4	1
----	----	----	---	---	---	---	---	---	---

$$\begin{aligned} \text{Fils Gauche}(i) &= 2i \\ \Rightarrow \text{Fils Droite}(i) &= 2i+1 \end{aligned}$$

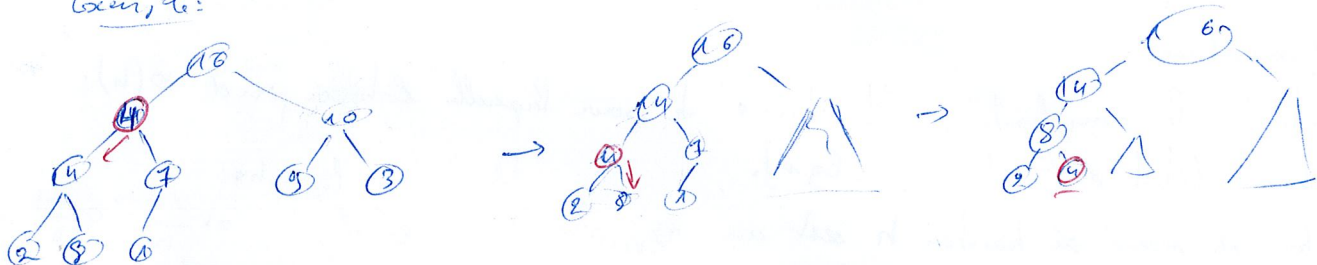
Elle permet l'utilisation d'un tri en place en temps $O(n \ln n)$, le tri par tas.

- les procédures Insérer et Extraire-max en temps $O(\ln n)$

Construction du tas.

La construction se base sur la fonction Entasser (A, i) qui rétablit la propriété des tas sur un sous-arbre dont la racine pourrait être plus petite que ses enfants:

Exemple:



Pseudo-code:

Entasser (A, i)

Entrée: un tableau A tel que les sous-arbres gauche et droite enracinés en $2i$ et $2i+1$ sont des tas.

Sortie: Tasse binaire enracinée en i

Trouver l'indice k de la plus ^{grande} valeur parmi $A[i]$, $A[2i]$, $A[2i+1]$.

si $k \neq i$

échange $A[k]$ et $A[i]$

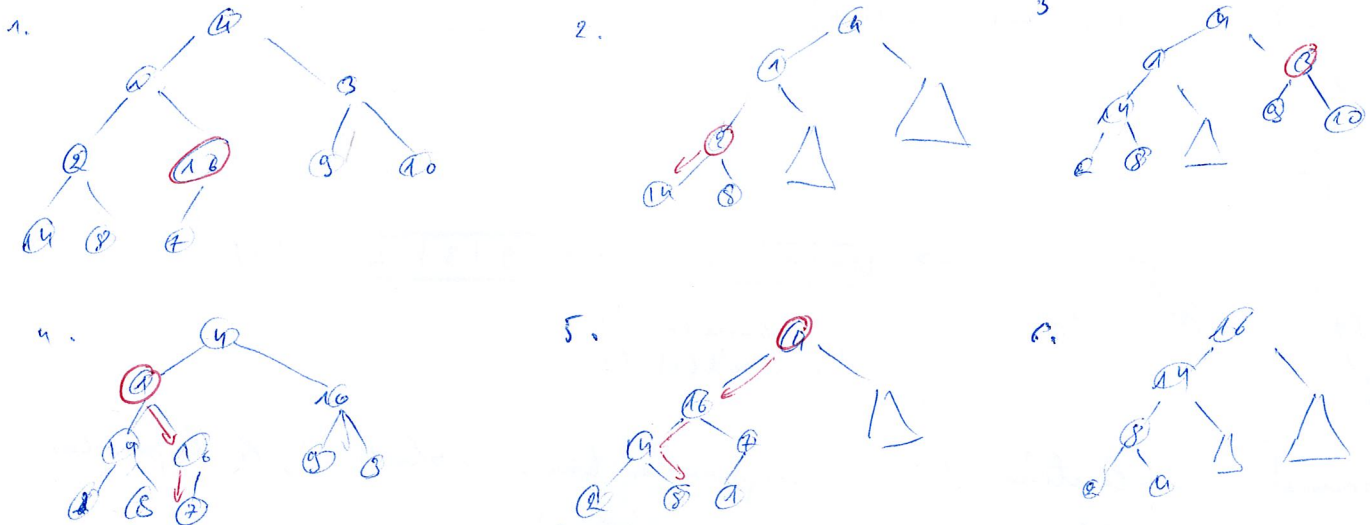
Entasse (A, k)

Le temps d'exécution est donné par la hauteur h mesuré en i donc $O(\ln n)$.

On construit alors un tas en appliquant entasser sur chaque noeud qui n'est pas une feuille à partir du dernier.

Exemple:

A: [4 | 1 | 3 | 2 | 16 | 9 | 10 | 14 | 8 | 7]



Pseudo-code

Construire-Tas (A)

Tableau A

Tableau A Tasse

pour i de $\lfloor \text{taille}(A)/2 \rfloor$ à 1:

Entasser (A, i).

Temps d'exécution:

On calcule la complexité par "hauteurs" le pour laquelle Entasser prend $O(h)$.

La hauteur totale de l'arbre est $\lceil \log n \rceil$.

Le nombre de noeud de hauteur h est au plus 2^{h-1} $\leq 2^{\lceil \log n \rceil - h} = \frac{2^n}{2^h} = O\left(\frac{n}{2^h}\right)$

La complexité est donc $\sum_{h=1}^{\lceil \log n \rceil} O\left(\frac{n}{2^h}\right) O(h)$

$$\sum_{h=1}^{\lceil \log n \rceil} O\left(\frac{n}{2^h}\right) O(h) = O\left(n \sum_{h=1}^{\lceil \log n \rceil} \left(\frac{h}{2^h}\right)\right)$$

$$\text{Or } \sum_{h=1}^{\infty} \frac{h}{2^h} = O(1)$$

Donc complexité = $O(n)$.

Exercices:

1. Entasser A: [5 | 3 | 17 | 10 | 19 | 6 | 22 | 9]

2. Coder. Construire Tas et Tri par Tas. + Fonctions insérer et extraire Max