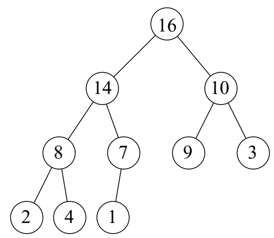
Tri par tas

La structure de tas



1	2	3	4	5	6	7	8	9	10
16	14	10	8	7	9	3	2	4	1

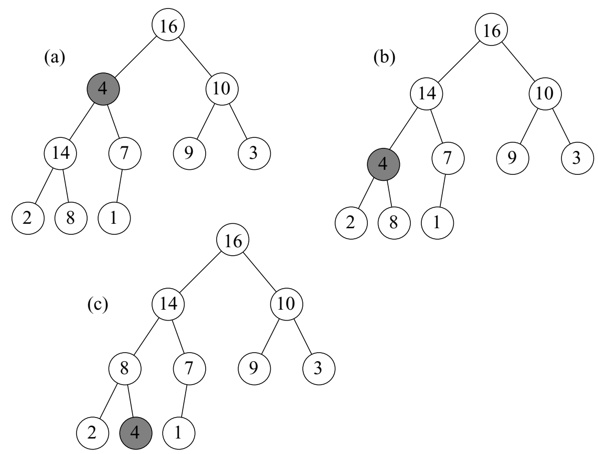
PARENT (i)
  retourner $\lfloor i/2 \rfloor$

GAUCHE (i)
  retourner 2i

DROITE (i)
  retourner 2i + 1

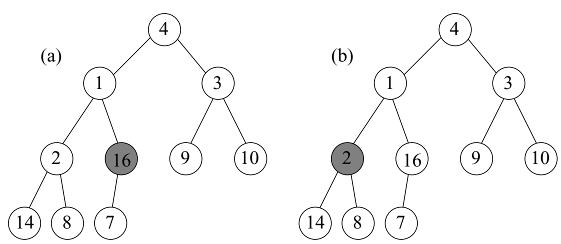
Procédure ENTASSER-MAX

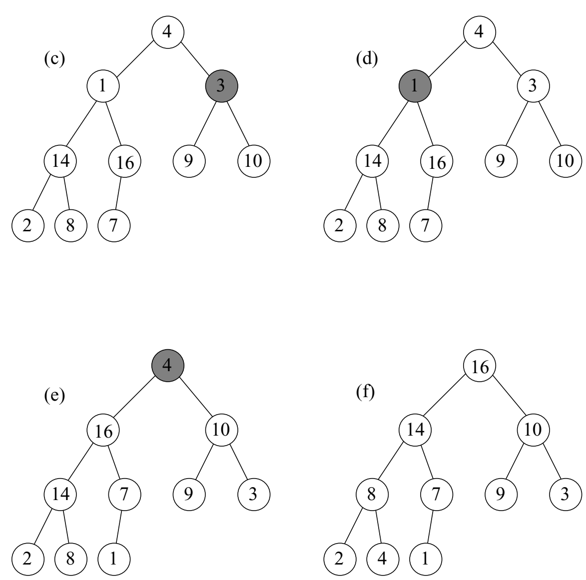
```
ENTASSER-MAX (A, i)
g ← GAUCHE (i)
d ← DROITE (i)
si l<= taille[A] et A[g] > A[i]
        alors max ← g
        sinon max ← i
si l<= taille[A] et A[d] > A[max]
        alors max ← d
si max ≠ i
        alors   échanger A[i] ↔ A[max]
                ENTASSER-MAX (A, max)
```




```
CONSTRUIRE-TAS-MAX (A)
taille [A] ← longueur [A]
pour i ← ⌊(longueur[A])/2⌋ jusqu'à 1
        faire ENTASSER-MAX (A, i)
```

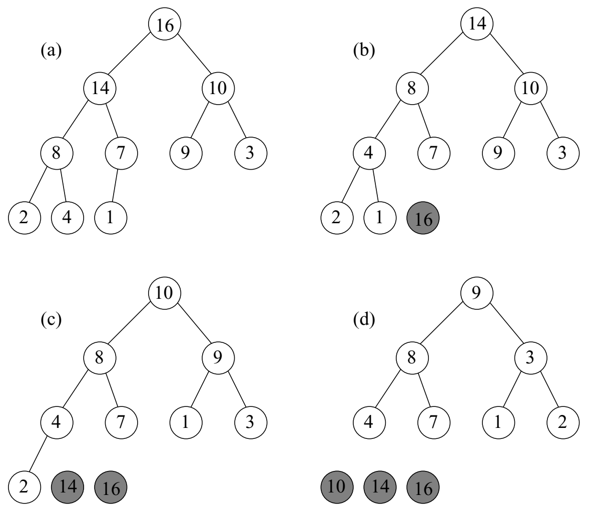
1	2	3	4	5	6	7	8	9	10
4	1	3	2	16	9	10	14	8	7

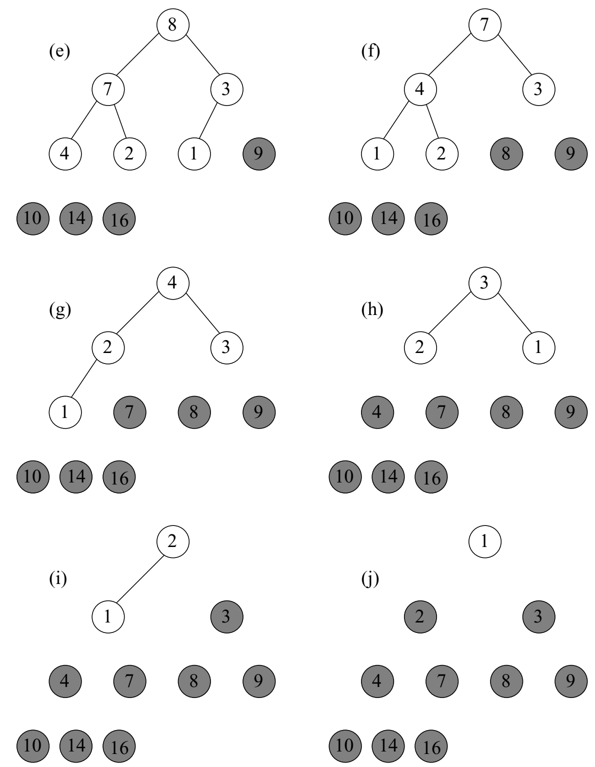




TRI-PAR-TAS (A)

CONSTRUIRE-TAS-MAX (A)
pour i ← longueur [A] **jusqu'à** 2
    faire échanger A[1] ↔ A[i]
        taille [A] ← taille [A] - 1
        ENTASSER-MAX (A, 1)





Problèmes.

Problème 1: SUPPRIMER(S, x) d'un tas en temps O(log n)

Problème 2: DECRÉMENTER-CLÉ(S, x, k) en temps O(log n)

Problème 3: Fusionner k listes triées en temps O(n log k) (n = nombre total d'éléments)