

Algorithmique IV (UE-41) - TD 7.

TD 7. Le tri par tas¹

EXERCICE 1. Décomposez les étapes du **tri par tas** de la liste $L = [11, 2, 1, 8, 6, 5, 9, 4, 7, 10, 3, 2, 8]$ en les matérialisant sur l'arbre binaire associé. Inutile de décomposer un tamisage, contentez-vous de surligner le chemin parcouru par la valeur du père dans l'arbre.

Solution. On transforme d'abord L en tas. Le 1er nœud à traiter est d'indice $k := \lfloor \frac{13}{2} \rfloor = 6$ puisque $\#L = 13$.

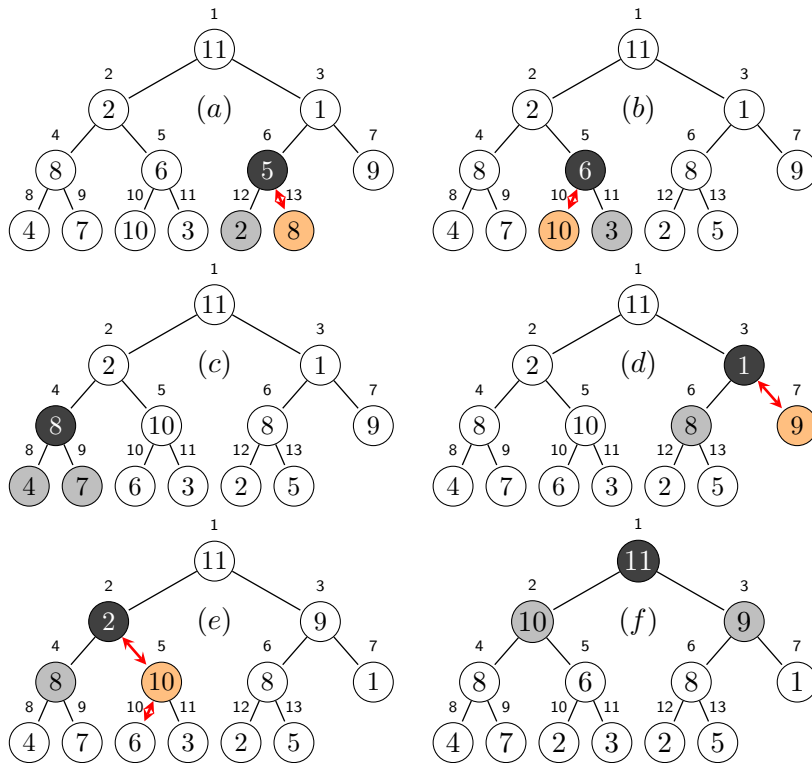


FIGURE 1. Transformation de L en tas.

Dans la figure 1, le père est noir et le fils avec lequel il y a échange orange (gris sinon). À l'étape (e), on a d'abord échangé 2 et 10 puis 2 et 6 lors du tamisage du nœud d'indice 2.

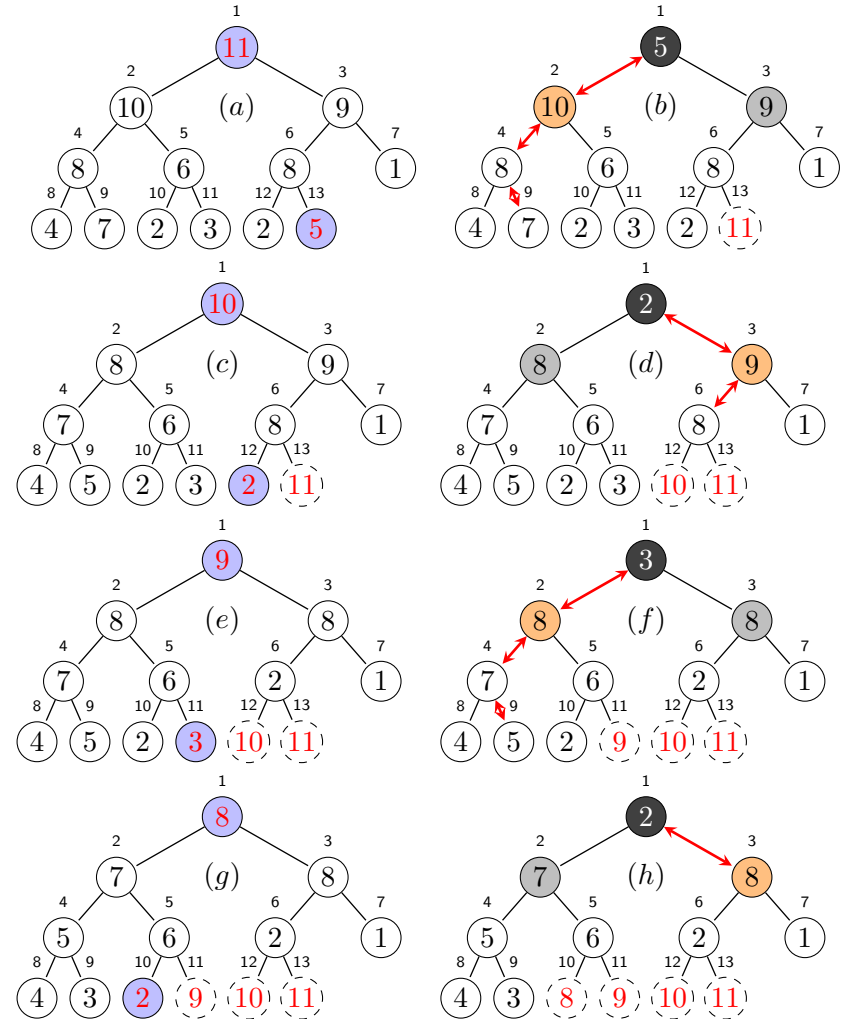


FIGURE 2. Tri du tas.

Reste à trier le tas $L = [11, 10, 9, 8, 6, 8, 1, 4, 7, 2, 3, 2, 5]$. Les échanges entre la tête et la fin de liste sont matérialisés en bleu et les valeurs définitivement placées sont entourées en traitillés dans la figure 2. Nous n'illustrons que les 4 premiers échanges et les tamisages qui suivent, à ce stade, $L = [8, 7, 2, 5, 6, 2, 1, 4, 3, \underline{8}, \underline{9}, \underline{10}, \underline{11}]$.

EXERCICE 2. Soit $A = (X, U)$ un **arbre binaire enraciné** non-vide.

- (1) Exprimez en logique des prédicats que tout nœud différent de la racine de l'arbre admet un unique prédécesseur.
- (2) Démontrez cette proposition.

Solution. (1) On note r la racine de cet arbre. Voici deux propositions possibles (entre autres) :

- (a) $\forall x \in X \setminus \{r\} \exists ! p \in X (p, x) \in U$
- (b) $\forall x \in X (x \neq r) \Rightarrow (|\Gamma^{-1}(x)| = 1)$

(2) Si X est réduit à la racine ou ne contient que la racine et un nœud, c'est évident. Supposons que $|X| \geq 3$. D'après la définition d'un **arbre enraciné**, tous les nœuds différents de la racine admettent (au moins) un prédécesseur. Montrons qu'il ne peut y en avoir qu'un par l'absurde. Supposons qu'un nœud interne x admette deux prédécesseurs a_0 et b_0 , alors on construit inductivement et alternativement un prédécesseur a_{i+1} de a_i et b_{j+1} de b_j sauf si $a_i = r$ ou $b_j = r$, auquel cas on continue la construction de l'autre suite dont les nœuds sont distincts de tous ceux construits précédemment, sans quoi on aurait un cycle contredisant la définition d'un arbre (cf. fig. 3). Mais alors l'arbre ainsi construit serait infini ce qui encore une fois contradictoire.

EXERCICE 3. Soit T un tas non-vide.

- (1) Exprimez en logique des prédicats que la première valeur d'un tas est la valeur maximale.
- (2) Démontrez cette proposition.

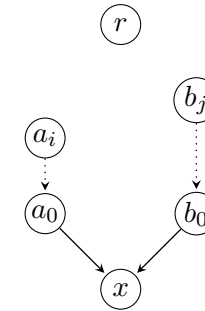


FIGURE 3. Construction d'un cycle.

Solution. (1) En définissant $n := \#T$, voici deux propositions possibles :

- (a) $T[1] = \max\{T[i] \mid i \in \llbracket 1, n \rrbracket\}$
- (b) $\forall k \in \llbracket 1, n \rrbracket T[1] \geq T[k]$

(2) Soit $k \in \llbracket 1, n \rrbracket$. On sait que dans un arbre il existe un unique chemin qui relie deux nœuds. Notons $i_1 i_2 \dots i_r$ celui qui relie la racine $i_1 = 1$ au nœud $i_r = k$. La propriété d'ordre partiel de l'arbre associé au tas T assure que $\forall j \in \llbracket 1, r-1 \rrbracket T[i_j] \geq T[i_{j+1}]$ et la transitivité de l'inégalité nous donne le résultat :

$$\forall k \in \llbracket 1, n \rrbracket T[1] \geq T[k].$$

EXERCICE 4. Démontrez que lors du tri d'un tas de longueur $n \geq 4$, l'algorithme **Tamiser** réalise au moins $n - 3$ échanges. On ne compte donc pas les $n - 1$ échanges réalisés par l'algorithme pour placer la valeur maximale en fin de liste avant l'appel à **Tamiser**. On suppose que les listes sont des permutations de $\llbracket 1, n \rrbracket$.

Indication : démontrez qu'à chaque appel de l'algorithme **Tamiser**, celui-ci réalise nécessairement un échange entre la valuation de la racine et celle de l'un de ses deux fils si le tas contient au moins quatre valeurs.

Solution. Considérons un APO contenant au moins 4 nœuds. La profondeur p de cet arbre est donc supérieure ou égale à $2 = \lfloor \log_2(4) \rfloor$, par conséquent la valuation de la racine sera échangée avec celle d'une feuille à cette profondeur. La propriété APO impose que la valuation de cette feuille est nécessairement strictement inférieure à la valuation de chacun

des fils de la racine puisqu'elle est située en dessous de ces deux nœuds, l'algorithme **Tamiser** réalisera donc nécessairement un échange entre la nouvelle valuation de la racine et celle de l'un des deux fils. L'algorithme du tri opère sur un tas dont la taille décroît d'une unité après chaque tamisage, il y aura donc un échange tant que la taille du tas est supérieure ou égal à 4.

EXERCICE 5. Soit $A = (X, U)$ un arbre binaire enraciné de racine r dont les sous-arbres gauche et droit sont des APO. Démontrez que l'arbre obtenu après l'application de l'algorithme **Tamiser** à r est un APO.

Solution. Si la valeur en r est supérieure à ses deux fils, l'algorithme s'arrête et l'arbre A est un APO. Dans le cas contraire, l'échange ne se faisant que sur l'un des deux sous-arbres, la propriété APO reste satisfaite pour l'autre. Le raisonnement s'applique inductivement à ce sous-arbre jusqu'à ce qu'une feuille soit atteinte.

EXERCICE 6. Trouvez un contre-exemple qui prouve que le tri par tas n'est pas stable.

Solution. Il suffit de trier la liste $T = [1_a, 1_b]$. C'est trivialement un tas et la première opération de tri commence par l'échange entre la tête de la liste et la fin de la liste, soit $T = [1_b, 1_a]$.

EXERCICE 7. On considère un tableau de taille n indexé de 1 à n et l'arbre binaire équilibré associé de profondeur p .

(1) Soit $k \in \llbracket 0, p-1 \rrbracket$. Démontrez que le nombre de nœuds à la profondeur k est égal à 2^k . Indication : faites une **récurrence finie**.

(2) En déduire que le nœud d'index i est à la profondeur $\lfloor \log_2(i) \rfloor$. Exprimez la profondeur p de l'arbre en fonction de la taille n du tableau.

(3) Définissez en logique des prédicats le plus grand indice \hat{i} des nœuds internes (i.e. qui ne sont pas des feuilles) puis démontrez que $\hat{i} = \lfloor \frac{n}{2} \rfloor$. On admettra que le fils gauche (resp. droit) d'un nœud d'indice i a pour indice $2i$ (resp. $2i+1$).

Solution. (1) On considère un arbre binaire équilibré A de profondeur p et le prédicat $P(k)$ sur l'intervalle $\llbracket 0, p-1 \rrbracket$ suivant : le nombre de nœuds à la profondeur k est égal à 2^k . La propriété $P(0)$ est vraie, en effet, à la profondeur 0 il n'y a que la racine. Par construction d'un arbre binaire équilibré, chaque nouveau niveau de l'arbre, sauf le dernier à la profondeur p , est constitué de tous les fils des nœuds du niveau précédent, il y a en a donc deux fois plus, soit $2 \times 2^k = 2^{k+1}$ d'après l'hypothèse de récurrence, prouvant ainsi que $P(k+1)$ est vrai. On en conclut que pour tout entier $k \in \llbracket 0, p-1 \rrbracket$, le nombre de nœuds à la profondeur k est égal à 2^k .

(2) On considère le prédicat $P(k)$ sur l'intervalle $\llbracket 0, p-1 \rrbracket$ suivant : les 2^k index des nœuds à la profondeur k sont les entiers de l'intervalle $I_k := \llbracket 2^k, 2^{k+1} - 1 \rrbracket$. La propriété $P(0)$ est vraie puisque la racine a pour index 1. Supposons que la propriété soit vraie au rang k . Le premier indice à la profondeur $k+1$ est donc égal à $2^{k+1} - 1 + 1 = 2^{k+1}$ et comme il y a 2^{k+1} nœuds à la profondeur $k+1$, les index des nœuds à cette profondeur appartiennent cette fois à l'intervalle I_{k+1} prouvant ainsi $P(k+1)$. On vérifie aisément que les intervalles I_k forment une partition de \mathbb{N}^* , ainsi pour tout indice $i \in \mathbb{N}^*$, il existe un unique $k \in \mathbb{N}$ tel que $i \in I_k$ et donc

$$2^k \leq i < 2^{k+1}.$$

La fonction logarithme étant croissante, on en déduit que

$$k \leq \log_2(i) < k+1.$$

On rappelle que la partie entière d'un réel x , notée $\lfloor x \rfloor$, est l'unique entier k qui satisfait

$$k \leq x < k+1.$$

On a donc $k = \lfloor \log_2(i) \rfloor$ ce qui nous fournit la profondeur de l'arbre

$$p = \lfloor \log_2(n) \rfloor.$$

(3) Les feuilles sont des nœuds qui n'ont pas de fils, l'indice i d'une feuille doit donc satisfaire $2i < n$, on en déduit :

$$\hat{i} := \max\{i \in \llbracket 1, n \rrbracket \mid 2i < n\}. \quad (1)$$

On conclut que \hat{i} est le plus grand entier i qui satisfait l'inégalité $i < \frac{n}{2}$, soit $\hat{i} = \lfloor \frac{n}{2} \rfloor$.