

Exercice 1 On considère des arbres binaires de recherche munis d'une opération $\hat{\text{ÉVASER}}$ décrite ci-dessous. On note $\text{clé}(x)$ la clé d'un sommet x et $\text{clé}(S)$ l'ensemble des clés d'un arbre S , supposées 2 à 2 distinctes. On s'intéresse aux opérations suivantes.

Nom	Description
$\text{APPARTIENTA}(i, S)$	Teste si la clé i figure dans $\text{clé}(S)$.
$\text{INSÉRER}(i, S)$	Insère la clé i dans l'arbre S , si elle n'y figure pas déjà.
$\text{SUPPRIMER}(i, S)$	Supprime la clé i de S , si elle y figure.
$\text{CONCATÉNER}(S_1, S_2)$	Construit un arbre contenant l'union des clés de S_1 et de S_2 , en supposant : $\forall x_1 \in S_1, \forall x_2 \in S_2 \text{ clé}(x_1) < \text{clé}(x_2)$.
$\text{SCINDER}(i, S)$	Scinde l'arbre S en deux arbres S_1, S_2 tels que <ul style="list-style-type: none"> – $\text{clé}(S) = \text{clé}(S_1) \cup \text{clé}(S_2)$, – $\text{clé}(S_1) \cap \text{clé}(S_2) = \emptyset$, et – $\forall x_1 \in S_1, \forall x_2 \in S_2 \text{ clé}(x_1) \leq i \leq \text{clé}(x_2)$

Pour réaliser ces opérations, on utilise une nouvelle opération, $\hat{\text{ÉVASER}}(i, S)$, qui réorganise l'arbre en conservant son ensemble de clés de telle sorte que

- si $i \in \text{clé}(S)$, la clé i apparaît à la racine de $\hat{\text{ÉVASER}}(i, S)$,
- sinon, la clé de la racine de $\hat{\text{ÉVASER}}(i, S)$ est soit $\max\{k \in \text{clé}(S) \mid k < i\}$, soit $\min\{k \in \text{clé}(S) \mid k > i\}$.

1. Montrer que chaque opération précédemment décrite se réalise à l'aide d'un nombre constant d'appels à $\hat{\text{ÉVASER}}$, en plus d'un nombre constant d'opérations de coût $O(1)$. On pourra supposer que l'ensemble des clés contient un élément $+\infty$, supérieur à tous les autres, qui n'apparaît pas comme clé d'un sommet (mais qui peut être utilisé comme premier argument d' $\hat{\text{ÉVASER}}$).

Pour réaliser l'opération $\hat{\text{ÉVASER}}(i, S)$, on utilise des rotations pour amener le sommet x portant la clé i à la racine :

- a. si le père y de x est la racine, on effectue une rotation simple autour de (y, x) ;
- b. sinon, soit y le père de x et z le père de y (cf. Fig. 1).
 - (i) Si x est fils gauche et y est fils gauche, on effectue une rotation autour de (z, y) suivie d'une rotation autour de (y, x) .
 - (ii) Si x est fils droit et y est fils gauche, on réalise la double rotation gauche-droite correspondante.

Les autres cas sont symétriques. On observe que la profondeur de x diminue, une séquence de telles rotations amène donc x à la racine.

2. Quelle est la hauteur maximale d'un arbre à n sommets obtenu à partir de l'arbre vide par insertions et suppressions réalisées à l'aide de l'opération $\hat{\text{ÉVASER}}$?
3. On considère l'arbre S sur les clés $\{0, \dots, 9\}$, où 9 est (la clé de) la racine et $i - 1$ est fils gauche de i , pour $i > 0$. Quel est le résultat de $\hat{\text{ÉVASER}}(0, S)$? De $\text{SUPPRIMER}(0, S)$?

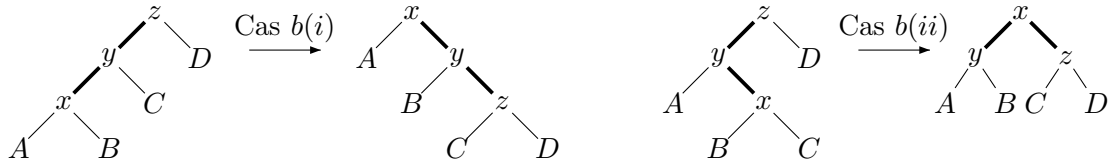


FIG. 1 – *Doubles rotations pour ramener x à la racine*

On appelle *rang de S* la valeur $r(S) = \lfloor \log |S| \rfloor$ (où $|S|$ est le nombre de sommets de S), et *rang de x* la valeur $r(S(x))$, où $S(x)$ est le sous-arbre de S enraciné en x . Pour alléger les notations, on écrit $r(x)$ au lieu de $r(S(x))$.

On effectue sur un arbre, via l'opération ÉVASER, les opérations précédemment décrites (insertions, suppressions,...). Lors des transformations de l'arbre, chaque sommet reçoit des crédits et on veut maintenir l'invariant : *chaque sommet x porte au moins $r(x)$ crédits*.

On considère dans les questions 4–8 une rotation (ou rotation double) particulière correspondant à l'un des cas a , $b(i)$ ou $b(ii)$. On note r et r' les valeurs du rang avant et après cette rotation.

4. On suppose l'invariant vérifié, et on considère x dont le père y est la racine (on est donc dans le cas a). Exprimer en fonction de $r(x)$, $r(y)$, $r'(x)$ et $r'(y)$ le coût nécessaire pour maintenir l'invariant.
5. En déduire que le coût pour effectuer la rotation et maintenir l'invariant est inférieur à $3(r'(x) - r(x)) + 1$.
6. On se place maintenant dans le cas $b(i)$. Exprimer le coût pour maintenir l'invariant en fonction des valeurs du rang sur x, y, z avant et après la double rotation.
7. Toujours dans le cas $b(i)$, montrer que le coût pour maintenir l'invariant et effectuer les rotations est au plus $3(r'(x) - r(x))$. On pourra distinguer le cas $r'(x) > r(x)$ du cas $r'(x) = r(x)$.
8. Montrer de même que dans le cas $b(ii)$, le coût pour maintenir l'invariant et effectuer les rotations est au plus $3(r'(x) - r(x))$.
9. En déduire que le coût d'une opération ÉVASER(i, S) est au plus $3(r(S) - r(x)) + 1$ pour l'ensemble des rotations et le maintien de l'invariant (où x est le sommet de clé i).
10. Montrer qu'une séquence de m opérations partant de l'arbre vide et contenant n insertions a un coût $O(m \log n)$.

Exercice 2 (Reconnaissance de motifs) Soit x un mot sur un alphabet A et soit $\mathcal{A}(x)$ l'automate minimal du langage A^*x .

1. Calculer $\mathcal{A}(ababa)$ sur $A = \{a, b, c\}$.
2. Montrer que pour tout $u \in A^*$, on a $u^{-1}(A^*x) = v^{-1}(A^*x)$, où v est le plus long suffixe de u qui est aussi préfixe de x .
3. Montrer que l'automate minimal $\mathcal{A}(x)$ de A^*x possède exactement $|x| + 1$ états, qu'on identifie désormais aux $|x| + 1$ préfixes de x .
4. On note $\text{bmax}(u)$ le bord maximal d'un mot $u \in A^*$. Montrer que la fonction de transition de $\mathcal{A}(x)$ peut se calculer par la formule suivante, où y est un préfixe de x :

$$\delta(y, a) = \begin{cases} ya & \text{si } ya \text{ est préfixe de } x, \\ \delta(\text{bmax}(y), a) & \text{sinon.} \end{cases}$$

5. En déduire que l'on peut calculer $\mathcal{A}(x)$ en temps $O(|A|(|x| + 1))$.
6. Comment améliorer la formule de la question 4 en utilisant les bords disjoints?
7. On appelle *flèche arrière* de $\mathcal{A}(x)$ une transition (y, a, z) , avec $z \neq \varepsilon$ et z préfixe de y . On considère deux flèches arrières distinctes (y, a, z) et (u, b, v) . Montrer que ya et ub ont des périodes différentes.
8. Majorer le nombre de flèches arrières de l'automate, et en déduire le coût en espace de la construction de $\mathcal{A}(x)$.
9. On utilise l'automate minimal de A^*x pour rechercher le motif x dans un texte t . Suivre une flèche arrière correspond à décaler le motif. Comparer les décalages réalisés par l'automate minimal de A^*x et l'algorithme de Knuth, Morris et Pratt. Donner un exemple où les décalages sont différents dans les deux algorithmes.
10. Soit $x = a_1 \cdots a_n$, $a_i \in A$. On considère l'automate $\mathcal{B}(x)$ suivant :

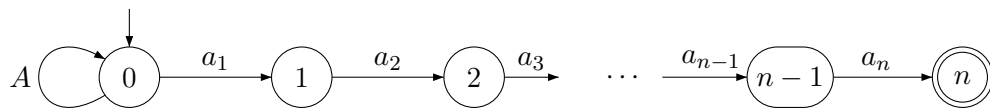


FIG. 2 – Automate non-déterministe de A^*x

Quel est le nombre d'états de l'automate déterminisé accessible de $\mathcal{B}(x)$?