

Oraux blancs MPI

type CCINP

Florian Bourse

Exercice A.

Dans cet exercice, on considère des graphes non-orientés $G = (S, A)$.

Une *clique* est un sous-graphe complet, c'est-à-dire un ensemble $X \subset S$ tel que : $\forall (s, t) \in X^2, \{s, t\} \in A$.

Une *couverture par sommets* est un ensemble de sommets $X \subset S$ qui touchent toutes les arêtes : $\forall \{x, y\} \in A, x \in X \vee y \in X$.

On considère le problème d'optimisation CLIQUE défini comme suit :

Instance : un graphe $G = (S, A)$.

Solution : une clique $X \subset S$.

Optimisation : maximiser $|X|$.

On considère également le problème d'approximation VERTEX-COVER défini comme suit :

Instance : un graphe $G = (S, A)$.

Solution : une couverture par sommets $X \subset S$.

Optimisation : minimiser $|X|$.

1. Énoncer le problème de seuil k -CLIQUE associé au problème CLIQUE et le problème de seuil k -VERTEX-COVER associé au problème VERTEX-COVER.
2. On admet que k -CLIQUE est NP-complet. Montrer que k -VERTEX-COVER est NP-complet.
3. Proposer un algorithme permettant de calculer un couplage maximal (au sens de l'inclusion) en temps polynomial.
4. On considère l'algorithme suivant prenant en entrée un graphe $G = (S, A)$:

Calculer un couplage M maximal

Renvoyer l'ensemble $\cup_{a \in M} a$ des extrémités des arêtes de M

Montrer qu'il s'agit d'une 2-approximation pour VERTEX-COVER.

Exercice B. L'exercice suivant est à traiter dans le langage OCaml

On s'intéresse aux arbres binaires localement complets non vides dont les nœuds sont étiquetés par des éléments comparables, représentés en OCaml par le type suivant :

```
type 'a arbre = F of 'a | N of 'a arbre * 'a * 'a arbre;;
```

1. Écrire une fonction **h** qui calcule la hauteur d'un arbre.
2. Écrire une fonction **deepest** qui prend en entrée un arbre a et renvoie l'étiquette d'une feuille de profondeur maximale dans a . La solution proposée doit être de complexité dans le pire des cas linéaire en la taille de l'arbre.
Sur l'exemple de la figure 1, la fonction doit renvoyer 13 ou 15.

3. Écrire une fonction **mystere** qui prend en entrée un arbre a et qui calcule la quantité

$$\sum_{f \in F(a)} 2^{-p(f,a)},$$

où $F(a)$ est l'ensemble des feuilles de a , et $p(s, a)$ est la profondeur du nœud s dans l'arbre a .

4. Quelle valeur très particulière est calculée par **mystere**? le démontrer.
5. Écrire une fonction **highest** qui prend en entrée un arbre a et qui renvoie l'étiquette d'une feuille de profondeur minimale dans a . On cherchera une solution dont la complexité dépend uniquement de la profondeur de ce nœud.
Sur l'exemple de la figure 1, la fonction doit renvoyer 9.

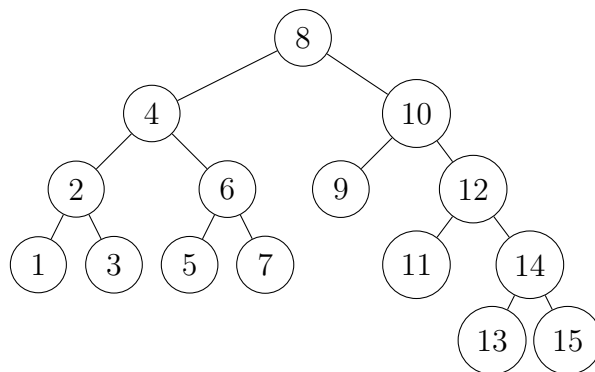


FIGURE 1 – Exemple d'arbre binaire localement complet non vide.