

## UE: ALGORITHMIQUE APPLIQUÉE - 4TIN903U

(test – durée 1h)

– *Documents (cours et notes personnelles) autorisés* –

---

### Solveur surpuissant

On dispose d'un solveur surpuissant capable de résoudre n'importe quel problème NP de décision sur les graphes en 1h de temps, à condition que le graphe soit simple et non orienté, et surtout que l'instance se code sur au plus 1 000 bits.

**Question 1.** *Donner un exemple de problème de décision qui soit NP-complet pour les graphes simples et non-orientés.*

**Question 2.** *Quel est le nombre maximum de sommets  $n$  du graphe pour lequel on est sûr de pouvoir résoudre un tel problème en 1h de temps avec un tel solveur ?*

**Question 3.** *Quelle serait la meilleure approche entre un algorithme naïf hypothétique de complexité  $2^n$  exécuté sur un ordinateur à 1 GHz ou le solveur sur 1 000 bits ?*

### Vertex Cover

**Question 4.** *Donner un algorithme efficace qui résout Vertex Cover de taille  $k$  pour les graphes de degré maximum au plus 1. Donner sa complexité en fonction du nombre de sommets  $n$  et d'arêtes  $m$  du graphe.*

**Question 5.** *Proposer un algorithme pour Vertex Cover dans le cas général qui soit récursif et basé sur le principe que soit  $u$  est dans la solution, soit tous ses voisins le sont, avec comme cas terminal le cas où le degré maximum est au plus 1.*

**Question 6.** *Donner l'équation de récurrence que doit vérifier la complexité de votre algorithme. Justifiez.*

**Question 7.** *En déduire sa complexité, éventuellement un majorant. Justifiez.*

## Max Fun

On dira qu'un problème est un problème de « type sommet » s'il s'agit de trouver dans un graphe un sous-ensemble de sommets vérifiant une certaine propriété, l'objectif étant d'en trouver un de taille minimum, maximum, ou de taille  $k$  donné suivant la variante. La variante « valué » d'un problème de type sommet est le même problème mais dans un graphe où les sommets possèdent des poids (non négatifs). Il s'agit alors de trouver un ensemble de sommets solution de poids minimum, maximum, ou de poids  $k$  donné.

Par exemple, ENSEMBLE DOMINANT VALUÉ MINIMUM est le problème de trouver un ensemble dominant dont la somme des poids de l'ensemble est minimum.

On considère le problème décrit informellement comme suit. Vous voulez organiser une fête avec potentiellement tous vos collègues d'une grande entreprise. Chaque collègue  $x$  a un « taux de fun » individuel bien déterminé, noté  $\text{fun}(x)$ , mais aussi a un chef de service, un chef direct noté  $\text{boss}(x)$ , sauf peut-être le PDG de l'entreprise. Inviter un collègue  $x$  et son chef  $\text{boss}(x)$  n'est pas fun du tout. Vous voulez absolument l'éviter. Votre objectif est donc d'inviter certains de vos collègues, sans leurs chefs, de façon à maximiser le fun total. On supposera valide une solution où vous n'êtes pas invité.

Vous noterez que la relation  $x \mapsto \text{boss}(x)$  définit un arbre  $T$  dont la racine est le PDG. Pour tout sommet  $x$ , on notera  $T[x]$  le sous-arbre composé de  $x$  et de tous ses descendants.

**Question 8.** *Donnez un problème de type sommets bien connu permettant de résoudre MAX FUN.*

On se propose de résoudre le problème MAX FUN par programmation dynamique, en partant des feuilles. On introduit les deux variables suivantes, définies pour chaque sommet  $x$  de  $T$ .

- $A[x]$  est le fun total maximum du sous-arbre  $T[x]$ ; et
- $B[x]$  est le fun total maximum de  $T[x]$  sans que  $x$  soit invité.

**Question 9.** *Soit  $x$  un sommet de  $T$  ayant pour fils  $x_1, \dots, x_t$ . Donner les formules de récurrence entre  $A[x]$ ,  $B[x]$  et les fils de  $x$ .*

**Question 10.** *Détailler le code permettant de résoudre MAX FUN et donnez sa complexité en fonction du nombre de sommets de  $T$ . Justifiez.*

**Question 11.** *Comment définiriez-vous les variables  $A$  et  $B$  si, à juste titre, vous ne voulez retenir que la solution où vous faites partie des invités ?*

**FIN.**