UNIVERSITE de VERSAILLES - Master 1 Info.

Module **Recherche opérationnelle et algorithmes probabilistes**
Janvier 2018

- Durée : 2h
- Tous documents autorisés
- Téléphones et ordinateurs interdits

1 Dépasser la médiane (4 points)

Nous avons un tableau T de taille 1000001 contenant des valeurs numériques toutes différentes. Rappelons que la médiane est l'élément qui possède autant de valeurs inférieures à lui que de valeurs supérieures à lui (donc ici 500000 valeurs plus petites et 500000 valeurs plus grandes). Le problème est de trouver une entrée $T[i]$ strictement plus grande que la médiane. Mais notons aussi qu'à moins de la calculer, la médiane n'est pas connue (il faudrait alors certainement trier tous les éléments et prendre le 500001 ème)

Pour répondre à ce problème, une approche peut consister à déterminer le maximum dans tout le tableau. Ceci nécessite $n - 1$ (donc 1000000) de comparaisons mais la valeur obtenue est bien sûr correcte, c'est à dire supérieure à la médiane.

1. Ce n'est toutefois pas nécessaire de déterminer le maximum de tout le tableau. De combien d'éléments peut-on se contenter pour être certain que leur maximum est bien supérieur à la médiane ? Et donc combien de comparaisons ?

2. Comme alternative, on propose l'algorithme probabiliste suivant :

   Upper-Half(T)
   n = 1000001;
   Pour i allant de 1 à 10 faire

   - j = Random(1,n);
   - $B[i] = T[j]$;

   Retourner Maximum(B[1..10])

   S'agit-il d'un algorithme de Monte-Carlo, de Las-Vegas ou d'Atlantic-City ? Justifier votre réponse.

3. Quelle est la probabilité que cet algorithme fournisse une réponse erronnée ? Combien de comparaisons effectue cet algorithme ?

2 Du est alité et ne pourra pas venir (7 points)

Soit le Programme Linéaire (P) suivant:

Minimize Obj = $10\ x_1 + 15\ x_2 + 20\ x_3 + 25\ x_4$

Subject To:
c1: $x_1 \geq 50$
c2: $x_1 + x_2 + x_3 + x_4 \geq 100$
c3: $3\ x_1 + 5\ x_2 + 8\ x_3 + 10\ x_4 \geq 310$
c4: $5\ x_1 + 7\ x_2 + 12\ x_3 + 15\ x_4 \geq 400$

Bounds :
$0 \leq x_1,\ 0 \leq x_2,\ 0 \leq x_3,\ 0 \leq x_4$

1. Ecrire le programme dual D.
2. La solution optimale de D, obtenue par un solveur, est fournie ci-dessous. Donnez la précisément: la valeur économique optimale, la valeur des variables, les contraintes saturées et la marge restante sur les autres contraintes. Indiquez également quelles sont les variables de base pour la solution optimale.
3. En déduire la solution optimale du problème initial P que de la même façon que pour celle du dual vous détaillerez précisément.
4. Si le coefficient économique de la première variable x_1 devient 8 (à la place du 10 actuel), peut-on dire des choses sur l'évolution des solutions optimales de P et de D sans relancer la résolution d'un nouveau simplexe ?
5. Si le second terme de la troisième contrainte (c3) devient 400 (à la place du 310 actuel), peut-on dire des choses sur l'évolution des solutions optimales de P et de D sans relancer la résolution d'un nouveau simplexe ?

Optimal Solution :
Problem Name : D
Problem Direction : MAX
Objective function value : 1020.00
Number of iterations : 4

Activities :

Num	Name		Level	Shadow Cost	Lower Obj	Objective	Upper Obj
1	y1	Z	0.00	-48.00	-INFINITY	50.00	98.00
2	y2	A	4.00	0.00	70.00	100.00	103.33
3	y3	A	2.00	0.00	300.00	310.00	550.00
4	y4	Z	0.00	-114.00	-INFINITY	400.00	514.00

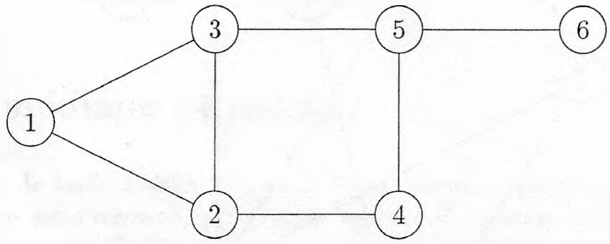
Constraints :

Num	Name		Slack	Shadow Price	Lower Lim	Limit	Upper Lim
1	C1	L	0.00	-98.00	7.50	10.00	11.66
2	C2	L	1.00	0.00	14.00	15.00	+INFINITY
3	C3	L	0.00	-2.00	10.00	20.00	20.66
4	C4	L	1.00	0.00	24.00	25.00	+INFINITY

3 Il faut surveiller toutes les arêtes (9 points)

Dans un graphe non orienté $G = (V, E)$ avec V les sommets et E les arêtes, le problème de couverture minimum par sommets (Vertex Cover en anglais) est un problème algorithmique classique qui consiste à trouver un ensemble minimum de sommets pour couvrir toutes les arêtes. Si les arêtes sont les rues d'une agglomération et les carrefours les sommets, il faut trouver sur quels carrefours placer des caméras pour être en mesure de surveiller toutes les rues et minimiser le nombre de caméras.

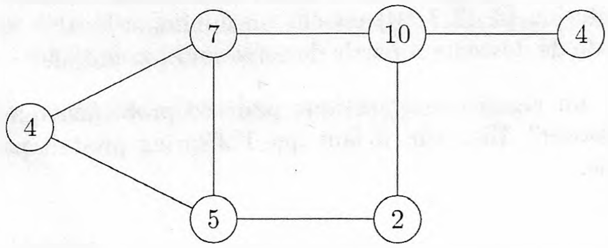
1. Sur le petit graphe ci-dessous, la valeur optimale est 3.



   Donner un exemple de 3 sommets qui permettent de couvrir toutes les arêtes.

2. Dans un graphe non orienté pondéré $G = (V, E, w)$ avec V les sommets, E les arêtes et w une fonction de poids sur les sommets, le problème de couverture minimum pondérée (Weighted Vertex Cover en anglais) consiste à trouver un ensemble de sommets de poids total minimum qui permet de couvrir toutes les arêtes.

   Sur le même graphe qu'à la question précédente, le numéro de sommet a été remplacé par son poids:



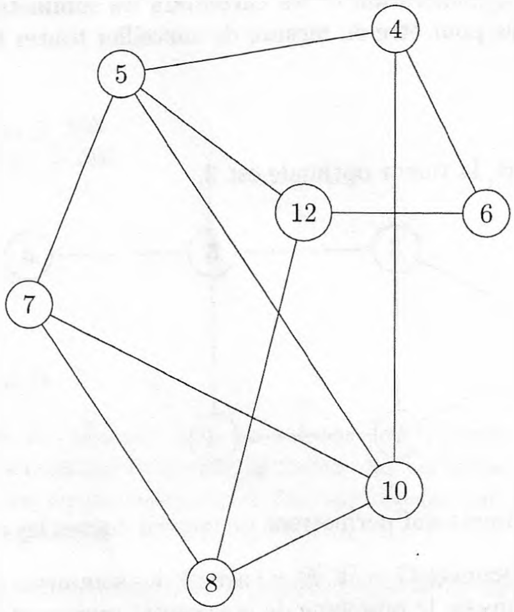
   Expliquez pourquoi la solution que vous avez trouvée à la première question (avec 3 sommets) est de valeur 19 (voire 22). Mais ce n'est pas la solution optimale du problème pondéré. Quelle est selon vous la solution optimale ?

3. La modélisation mathématique du problème est la suivante:
   Minimiser $\sum_{i \in V} w_i x_i$
   $\forall (i, j) \in E,\ x_i + x_j \geq 1$
   $\forall i \in V,\ x_i \in \{0, 1\}$

   Expliquer cette modélisation. Pourquoi n'a t'on pas affaire à un Programme Linéaire ? Que pourrions nous obtenir en appliquant la méthode du simplexe sur cette modélisation ?
   Le simplexe sur le problème de la question précédente donne comme valeur optimale 16. A quelle solution correspond-elle ?

4. Proposer une heuristique gloutonne pour le problème de couverture minimum pondérée. Ecrire clairement l'algorithme correspondant.

5. On prend dorénavant l'exemple suivant où comme précédemment ce sont les poids des sommets qui sont indiqués dans les cercles.



Appliquer votre heuristique gloutonne sur cet exemple.

6. Nous allons maintenant considérer une méthode de descente. Un mouvement élémentaire consiste à supprimer un sommet de la solution courante et à ajouter tous ses voisins qui ne figurent pas déjà dans la solution. Indiquez quelles sont les solutions voisines de $\{4, 12, 7, 10\}$ et leurs valeurs. Cette solution $\{4, 12, 7, 10\}$ est-elle un minimum local ? Si ce n'est pas le cas, appliquez une méthode de descente à partir de cette solution initiale.

7. Si on voulait développer un algorithme génétique pour ce problème, que prendriez-vous comme opérateur de crossover? Bien sûr, il faut que l'offspring produit par cet opérateur soit une solution réalisable.

4 La question bonus (1 point)

Quel est l'age de votre sympathique enseignant ? 1 point si vous trouvez son age, 0.5 point si votre erreur est inférieure ou égale à 3 ans. Aucun point négatif ne sera donné pour cette question même pour ceux qui vieilliront sensiblement l'intéressé. Par contre, aucun point bonus ne sera donné à ceux qui le flatteront en le rajeunissant beaucoup.