

8INF433 Algorithmique

Devoir 2

(À remettre au plus tard: jeudi le 28 février 2019 **avant le cours**)

Justifiez toutes vos réponses.

Vous serez évalué pour la rectitude et la qualité de vos réponses.

Travaillez en équipe de trois ou quatre étudiants.

Remettre dans mon casier une copie papier brochée par équipe.

1. Le théorème sur les récurrences que nous avons vu en classe possède une version plus simple dans le cas où la fonction $f(n)$ est un polynôme. Lorsque la récurrence est de la forme:

$$T(n) = aT(n/b) + cn^k$$

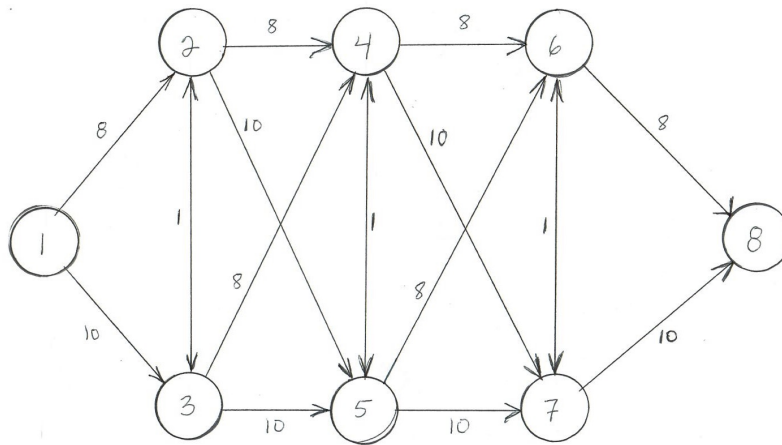
où c et k sont des constantes réelles positives, alors la solution est donnée par:

$$T(n) = \begin{cases} \Theta(n^{\log_b a}) & \text{si } a > b^k \\ \Theta(n^k \lg n) & \text{si } a = b^k \\ \Theta(n^k) & \text{si } a < b^k \end{cases} \quad (1)$$

Nous avons vu en classe la démonstration du premier cas (où $a > b^k$). Utilisez le théorème sur les récurrences vu en classe pour démontrer les deux autres cas.

2. L'algorithme de Dijkstra que nous avons vu en classe, nous permet d'obtenir la longueur du plus court chemin entre un noeud source et tous les autres noeuds mais il ne donne pas les chemins eux-mêmes. Comment modifieriez-vous cet algorithme afin d'obtenir cette information? Notez que la modification est mineure si vous choisissez la bonne façon de représenter les chemins.

3. Considérez le graphe dirigé de la figure suivante:



Montrez le contenu du tableau D , de l'ensemble S ainsi que la valeur de la variable v après chacune des itérations de l'algorithme. Décrivez toutes les étapes de l'algorithme Dijkstra. Utilisez le noeud 1 comme source.

4. Résoudre les récurrences suivantes.

- (a) $T(n) = 2T(n/2) + 1$
- (b) $T(n) = T(9n/10) + n$
- (c) $T(n) = 16T(n/4) + n^2$
- (d) $T(n) = 7T(n/3) + n^2$
- (e) $T(n) = 7T(n/2) + n^2$
- (f) $T(n) = 2T(n/4) + \sqrt{n}$
- (g) $T(n) = 3T(n/2) + 5n$
- (h) $T(n) = T(n/2 + 5) + 1$

5. Considérez la matrice suivante

$$F = \begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}.$$

Soit i et j , deux entiers. Quel est le produit du vecteur (i, j) et de la matrice F ? Qu'arrive-t-il si i et j sont deux nombres consécutifs de la suite de Fibonacci? Utilisez cette idée pour concevoir un algorithme diviser-pour-régner capable de calculer le n ième nombre de Fibonacci en temps $\Theta(\log n)$ en considérant (faussement) que les multiplications prennent un temps constant. Expliquez le lien entre votre algorithme et l'algorithme fib3 du devoir 1.

6. Soit $X[1 \cdots n]$ et $Y[1 \cdots n]$ deux tableaux, chacun contenant n nombres déjà triés. Donnez un algorithme diviser-pour-régner pour trouver le médian des $2n$ éléments présents dans les tableaux X et Y . Le temps d'exécution de votre algorithme doit être dans $\Theta(\log n)$ en pire cas.