L3-INFO Algorithmique et Modélisation

UGA UFR IM²AG Université Grenoble Alpes

Quick – 22 mars 2024 – durée 1 h

Sont interdits : les documents, les ordinateurs, les téléphones (incluant smartphone, tablettes,... tout ce qui contient un dispositif électronique).
Seuls les dictionnaires papier pour les personnes de langue étrangère sont autorisés.
Il sera tenu compte de la qualité de la rédaction et de la clarté de la présentation (2 pts).
En cas d'incompréhension du sujet, préciser les hypothèses de travail que vous faites et continuer.
Le barème **indicatif** : Exercice 1 : 7 pts ($25 \sim mn$) ; Exercice 2 : 4 pts ($5 \sim mn$), Exercice 3 : 7 pts ($25 \sim mn$), Relecture : ($5 \sim mn$). Les 3 exercices sont indépendants.

I Sur un des sommets de la montagne

On considère une grille $n \times n$ contenant des nombres distincts. Une case à l'intérieur de la grille possède 4 voisins, sur le bord 3 voisins et les 4 cases de coin ont 2 voisins.

	1	2	3	4	5	6	7
1	58	32	74	60	80	33	91
2	46	29	93	12	3	15	79
3	89	16	75	48	9	8	87
4	71	14	49	55	37	4	21
5	38	70	62	41	24	40	98
6	51	23	56	81	10	99	85
7	30	31	59	57	82	73	52

On appelle **sommet local** une case dont toutes les voisines sont plus petites. Dans la grille 7×7 ci-dessus la case $(1, 5)$ de valeur 80 est un sommet local car sa valeur est plus grande que les valeurs des cases voisines, la case $(5, 2)$ est également un sommet local.

I.1. Combien peut-on avoir au minimum de sommets locaux dans une grille $n \times n$? Quel est l'ordre de grandeur du nombre maximum de sommets locaux possibles ?

I.2. Écrire un algorithme naïf permettant de trouver un sommet local. Quelle est sa complexité ?

I.3. Dans cette question, on suppose que l'on cherche un sommet local sur une seule ligne. C'est à dire un sommet local dans un tableau de taille n, c'est à dire une grille $n \times 1$.
Proposer un algorithme de complexité $\mathcal{O}(\log n)$ qui détermine un sommet local dans le tableau.

En s'inspirant de l'algorithme de la question précédente, on propose de diviser la grille en quatre en sélectionnant la ligne du milieu et la colonne du milieu. On recherchera ensuite un sommet local dans un des quadrants bien choisi. On pourra supposer que n est de la forme $2^k - 1$ pour faciliter l'analyse et l'écriture de l'algorithme.

I.4. Écrire l'algorithme et expliquer, sans la formaliser, la preuve de cet algorithme.

I.5. Évaluer la complexité de cet algorithme et commenter votre résultat.

I.6. Comment adapter cet algorithme lorsque la valeur n n'est pas de la forme $2^k - 1$.

II Algorithme de Dijkstra

Étant donné un graphe fini quelconque $\mathcal{G} = (\mathcal{X}, \mathcal{A}, \mathcal{V})$, orienté dont les arêtes sont valuées par des nombres positifs, on considère l'algorithme de Dijkstra suivant :

Algorithme_Dijkstra($\mathcal{G}$,s)

Données : Un graphe $\mathcal{G}$ valué et un sommet s de $\mathcal{G}$

$d(s) = 0$

$S \leftarrow \{s\}$

while *il existe* $y \notin S$ *et* $x \in S$ *tels que* $(x, y) \in \mathcal{A}$

 `Choisir` $y \notin S$ minimisant

$$\pi(y) = \min_{x \in S, (x,y) \in \mathcal{A}} \{d(x) + v(x, y)\}$$

 $S \leftarrow S \cup \{y\}$

 $d(y) = \pi(y)$

Algorithme 1 : Version simplifiée de l'algorithme de Dijkstra

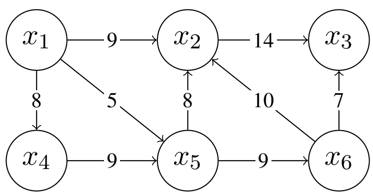
II.1. Quel est le résultat fourni par cet algorithme ? (2 lignes)

II.2. Montrer que l'algorithme ne fournit pas un résultat correct lorsque certaines valuations sont négatives.

II.3. Dans le cas où certaines valeurs sont négatives, on ajoute une même valeur v à toutes les valeurs des arcs de manière à ce que les nouvelles valeurs soient positives. On exécute ensuite l'algorithme de Dijkstra. Le résultat obtenu est-il correct ? (justifier)

III Ticket gratuit

On considère un réseau ferroviaire représenté par un graphe orienté $\mathcal{G} = (\mathcal{X}, \mathcal{A}, \mathcal{V})$, la valeur représentant le prix du ticket associé à un tronçon.



III.1. Calculer, sur cet exemple et en le justifiant les chemins de x_1 à chacun des autres sommets ayant un prix minimal.

L'entreprise ferroviaire décide de donner un ticket gratuit assurant la gratuité sur l'un des tronçons du chemin. C'est à dire que l'usager peut choisir un chemin et choisit un des tronçons (arcs) de ce chemin pour le ticket gratuit.

III.2. Proposer un algorithme permettant de calculer les chemins de prix minimal d'origine x_1.

III.3. Appliquer cet algorithme sur l'exemple ci-dessus.