SESSION 2006

Filière : 2ème concours

INFORMATIQUE

Durée : 3 heures

Les calculatrices sont interdites.

Tournez la page S.V.P.

Autour de la notion de plus court chemin

Le sujet comporte deux parties. Il est recommandé de lire l'ensemble du sujet avant de commencer la rédaction. Il est également conseillé de traiter les questions dans l'ordre de l'énoncé. On pourra cependant aborder une question en admettant les résultats des questions précédentes. Les algorithmes pourront être écrits dans un langage au choix du candidat, en utilisant les structures de contrôle habituelles. Une attention particulière sera portée à l'analyse des algorithmes. En particulier, les coûts annoncés devront être clairement justifiés.

1 Recherche du meilleur itinéraire dans un réseau routier

Dans cette partie, nous nous intéressons à un problème courant de la vie quotidienne : la recherche dans un réseau routier d'un itinéraire minimisant plusieurs critères à la fois. On souhaite par exemple arriver le plus rapidement possible d'un point à un autre. Mais certaines routes sont également plus ou moins dangereuses. On souhaite donc également minimiser la probabilité qu'un accident survienne.

Par la suite, on notera V un ensemble de n villes. Ces villes sont reliées entre elles par un ensemble de routes R. Une route $r \in R$ de la ville u à la ville v est identifiée au couple (u, v) (R est donc une partie de $V \times V$). On notera que cette définition permet de modéliser aisément les routes à sens unique puisque si $(u, v) \in R$, (v, u) n'appartient pas nécessairement à R. On suppose disposer d'une fonction d'évaluation $\delta : R \mapsto \mathbb{R}^+$ qui indique la durée de parcours d'une route donnée et d'une fonction d'évaluation $\pi : R \mapsto [0,1]$ qui indique la probabilité qu'*aucun* accident ne survienne sur une route donnée.

Définition 1 (Réseau routier). Un *réseau routier* est un quadruplet $\mathcal{R} = (V, R, \delta, \pi)$ avec $R \subset V \times V$, $\delta : R \mapsto \mathbb{R}^+$ et $\pi : R \mapsto [0,1]$.

Définition 2 (Itinéraire). On appelle *itinéraire de u à v* une suite de routes (γ_0, γ_1), (γ_1, γ_2), $\ldots$, (γ_{p-1}, γ_p) telle que $\gamma_0 = u$ et $\gamma_p = v$. On notera $\Gamma_{u,v}$ l'*ensemble des itinéraires de u à v*.
La *longueur* d'un itinéraire est le nombre de routes qu'il emprunte (c'est-à-dire p dans la définition ci-dessus). On notera que si une route est empruntée plusieurs fois, elle compte autant de fois qu'elle apparaît dans l'itinéraire.
On étend δ aux itinéraires pour définir la *durée* d'un itinéraire γ par

$$\delta(\gamma) = \sum_{i=1}^{p} \delta(\gamma_{i-1}, \gamma_i).$$

De même, on étend π pour définir la *probabilité* qu'*aucun* accident ne survienne sur un itinéraire :

$$\pi(\gamma) = \prod_{i=1}^{p} \pi(\gamma_{i-1}, \gamma_i).$$

On note $d^{(k)}(u,v)$ la plus petite durée d'un itinéraire de u à v de longueur inférieure ou égale à k. Si un tel itinéraire n'existe pas, $d^{(k)}(u,v)$ vaut alors $+\infty$. On note $d^{(\infty)}(u,v)$ la plus petite durée d'un itinéraire de u à v (indépendamment de sa longueur).

▷ **Question 1** *Démontrer que pour tout $k \in \mathbb{N}$ et tout (u,v) dans $V \times V$, on a :*

$$d^{(k+1)}(u,v) = \min_{(w,v)\in R} d^{(k)}(u,w) + \delta(w,v).$$

▷ **Question 2** *Montrer que la suite $d^{(k)}$ est stationnaire à partir d'un certain rang, c'est-à-dire qu'il existe un $N \in \mathbb{N}$ tel que :*

$$\forall u, v \in V : \forall k \geqslant N : d^{(k)}(u,v) = d^{(N)}(u,v).$$

▷ **Question 3** *En déduire un algorithme qui étant donné un réseau $\mathcal{R}$ et deux villes u et v calcule $d^{(\infty)}(u,v)$ et un itinéraire réalisant cette durée. On donnera la complexité de l'algorithme en fonction de $|V|$ et $|R|$. Votre algorithme vous fournit-il d'autres informations que $d^{(\infty)}(u,v)$?*

▷ **Question 4** *Comment calculer l'itinéraire le plus fiable (c'est-à-dire celui qui minimise la probabilité d'avoir un accident) ?*

▷ **Question 5** *Proposer une méthode (il en existe plusieurs) permettant de trouver « un » itinéraire à la fois ni trop long ni trop dangereux en utilisant les mêmes techniques que précédemment.*

Lorsque le premier chemin de fer transcontinental fut construit aux États-Unis, la ligne fut construite en partant des deux extrémités à la fois. La ligne *Central Pacific* fut construite vers l'Est en partant de Sacramento et la ligne *Union Pacific* fut construite vers l'Ouest à partir de Omaha dans le Nebraska. La pose des rails se poursuivit rapidement jusqu'à ce que les deux lignes se rejoignent à *Promontory Summit* le 10 Mai 1869 et où l'évènement fut célébré en grandes pompes.

Une des questions que le congrès des État-Unis et les dirigeants des compagnies de chemins de fer auraient pu se poser est la suivante :

▷ **Question 6** *Étant donnés deux points de départ, les différentes routes et étapes possibles ainsi que le temps nécessaire à la pose des rails le long de ces différentes routes en fonction des différentes difficultés qui peuvent être rencontrées (plaines, collines, marais, montagnes, etc.), quel est l'itinéraire permettant aux deux lignes de se rejoindre en le moins de temps possible ?*

Proposer une modélisation et une résolution de ce problème (on n'oubliera pas de préciser la complexité des algorithmes).

On suppose maintenant que la fonction δ est à valeurs entières. Notre objectif est maintenant de proposer un algorithme qui, étant donnés un réseau $\mathcal{R}$, deux villes et une durée entière d, détermine un itinéraire entre ces deux villes de durée *exactement* d s'il en existe un. Pour ce faire on pourra s'intéresser au calcul de la fonction A définie de la façon suivante :

$$A(u, v, d) = \begin{cases} 1 & \text{s'il existe un itinéraire de } x \text{ à } y \text{ de durée exactement } d, \\ 0 & \text{sinon.} \end{cases}$$

▷ **Question 7** *Proposer un algorithme qui détermine un itinéraire entre u et v de durée* exactement *d s'il en existe un. On donnera la complexité de l'algorithme en fonction de* $|V|$, $|R|$ *et* d.

2 Jeux de jarres

On s'intéresse dans cette partie au petit problème suivant :

> *Deux hommes disposent d'une jarre de 8 pintes pleine de vin qu'ils souhaitent se diviser équitablement. Ils n'ont à leur disposition que deux jarres vides non graduées de capacités respectives 3 et 5 pintes. Comment procéder au partage ?*

Dans la suite on notera X, Y et Z le volume de chacune des jarres (on suppose $X > Y > Z$). Dans notre exemple on a donc $X = 8$, $Y = 5$ et $Z = 3$.

L'état de nos jarres peut se représenter à l'aide d'un triplet (x, y, z) où x, y et z dénotent les volumes de vin présents dans chacune des jarres. Dans notre exemple, nous cherchons donc un moyen de passer de l'état $(8, 0, 0)$ à l'état $(4, 4, 0)$.

On supposera qu'aucun des deux hommes ne va boire de vin au cours de l'opération et qu'ils feront bien attention à ne pas en renverser à côté. On suppose également que comme aucun instrument de mesure n'est disponible, les seules opérations possibles consistent à transvaser du vin d'une jarre à l'autre. Le transvasement s'arrête dès que la jarre de départ est vide *ou* dès que la jarre d'arrivée est pleine. Le tableau suivant illustre une séquence de transvasements valides :

Étape	Jarre 1	Jarre 2	Jarre 3
0	8	0	0
1	3	5	0
2	3	2	3
3	6	2	0

L'objectif de cette partie est de répondre aux deux questions naturelles suivantes :

1. Existe-t-il un algorithme pour trouver une solution à un tel problème ?
2. Comment déterminer la *meilleure* solution, c'est-à-dire par exemple celle qui requiert le moins d'opérations de transvasement ?

▷ **Question 8** *Donner des contraintes devant être respectées par* x, y *et* z *pour que* (x, y, z) *soit un état accessible à partir de* $(X, 0, 0)$.

▷ **Question 9** *Dessiner l'ensemble des états accessibles à partir de* $(8, 0, 0)$ *(pour* $X = 8$, $Y = 5$ *et* $Z = 3$*) et les relations entre les différents états. On vérifiera qu'il n'existe en tout que 16 états possibles. Combien de relations comptez-vous ?*

▷ **Question 10** *Montrer à partir de la question précédente qu'il existe plusieurs façons d'accéder à l'état* $(4, 4, 0)$ *à partir de l'état* $(8, 0, 0)$ *(sans repasser deux fois par le même état).*

Dans la question suivante, on va s'intéresser à la conception d'un algorithme permettant d'énumérer les différents états atteignables à partir de $(X, 0, 0)$. Pour ce faire on pourra s'intéresser au calcul de la fonction A définie de la façon suivante :

$$A(x,y,z) = \begin{cases} 1 & \text{s'il existe un moyen d'arriver à l'état } (x,y,z) \text{ en partant de } (X,0,0), \\ 0 & \text{sinon.} \end{cases}$$

▷ **Question 11** *Écrire une relation de récurrence sur* A *et en déduire un algorithme permettant d'énumérer les différents états valides.*

▷ **Question 12** *En remarquant que* y *et* z *suffisent à représenter l'état* (x, y, z) *et en s'aidant des contraintes nécessaires sur* x, y *et* z *pour que* (x, y, z) *soit accessible à partir de* $(X, 0, 0)$*, proposer une autre façon d'énumérer les états valides (une représentation graphique des couples* (y, z) *pourra être utile). Démontrer également qu'il existe au plus* $2(Y + Z)$ *états valides à partir du moment où* $X \geqslant Y + Z$*. Proposer une borne plus fine que* $2(Y + Z)$ *sur le nombre d'états quand cette condition n'est pas vérifiée.*

▷ **Question 13** *En s'inspirant de la partie 1, proposer un algorithme qui calcule le nombre minimal d'étapes nécessaires pour atteindre* $(X/2, X/2, 0)$ *ainsi que la façon de procéder. Quelle est la complexité de cet algorithme en fonction de* X, Y *et* Z *?*

▷ **Question 14** *On suppose maintenant que le coût d'une étape est proportionnel au poids du volume transvasé. Comment calculer la séquence d'étapes la moins fatigante ? Quelle est la complexité de cet algorithme en fonction de* X, Y *et* Z *?*