10.04M

SESSION 2010

Filière MP (groupe I)

Épreuve commune aux ENS de Paris, Lyon et Cachan

INFORMATIQUE

Durée : 4 heures

L'usage de calculatrices électroniques de poche n'est pas autorisé.

Ce sujet, composé de 30 questions, étudie une classe de programmes simples avec variables à valeurs dans $\mathbb{N}$. Les questions peuvent être résolues en admettant les résultats des questions précédentes. Les questions des parties II ou III sont indépendantes des questions des autres parties tandis que des questions de la partie IV dépendent de la partie III.

Les algorithmes peuvent être donnés dans la notation de votre choix du moment qu'elle soit suffisamment précise. Pour chaque algorithme proposé, il est indispensable de justifier sa correction et de donner sa complexité en temps dans le pire des cas en choisissant les paramètres pertinents. Comme d'habitude, il est demandé d'apporter le plus grand soin à la rédaction.

L'usage des calculatrices est interdit.

NOTATIONS

Dans la suite, $\mathbb{N}$ et $\mathbb{Z}$ représentent respectivement l'ensemble des entiers naturels et l'ensemble des entiers relatifs. Pour $l, l' \in \mathbb{Z}$, $[l, l']$ représente l'ensemble des entiers relatifs j tels que $l \leq j \leq l'$. Un élément $x \in \mathbb{Z}^n$ est dénoté par $\langle x(1), \ldots, x(n) \rangle$. Pour $x, y \in \mathbb{Z}^n$, on note $x + y$ l'élément de $\mathbb{Z}^n$ tel que pour $i \in [1, n]$, nous avons $(x + y)(i) \stackrel{\text{def}}{=} x(i) + y(i)$. Pour $I \subseteq [1, n]$, $\mathbb{Z}^I$ dénote l'ensemble des fonctions $I \to \mathbb{Z}$. Etant donnés $x \in \mathbb{Z}^n$ et $I \subseteq [1, n]$, on note $x_I \in \mathbb{Z}^I$, la restriction de x à I ($\mathbb{N}^I$ est défini de façon analogue). Ainsi, pour $i \in I$, nous avons $x_I(i) = x(i)$. Pour $n \geq 1$, l'ordre partiel $\preceq$ sur $\mathbb{Z}^n$ est défini ainsi : $x \preceq y \stackrel{\text{def}}{\Leftrightarrow}$ pour $i \in [1, n], x(i) \leq y(i)$. On note $x \prec y$ lorsque $x \preceq y$ et $x \neq y$. Le cardinal d'un ensemble X est dénoté par $\mathrm{card}(X)$. NB : dans la suite $\subset$ dénote l'inclusion ensembliste stricte tandis que $\subseteq$ dénote l'inclusion non stricte.

PARTIE I : SYSTÈMES D'ADDITION DE VECTEURS

Nous allons introduire une famille de programmes simples qui peuvent être vus comme des automates finis auxquels on adjoint la possibilité de mettre à jour des compteurs (variables à valeurs dans $\mathbb{N}$). Les valeurs des compteurs sont représentées par des vecteurs d'entiers naturels. Un *système d'addition de vecteurs avec états* (SAVE) est un triplet $\mathcal{S} = \langle S, T, n \rangle$ tel que

- $n \geq 1$ (la *dimension*).
- S est un ensemble fini non-vide (les *états de contrôle*).
- T est un sous-ensemble fini de $S \times \mathbb{Z}^n \times S$ (les *transitions*).

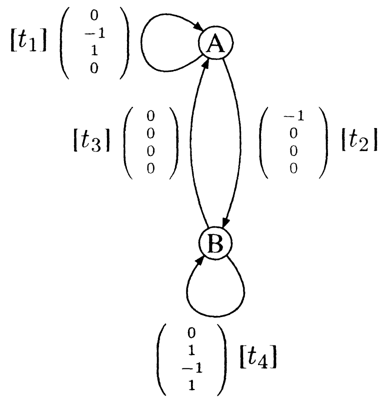
Ainsi, $\mathcal{S}$ peut être vu comme un automate fini dont l'alphabet est un sous-ensemble fini de $\mathbb{Z}^n$, automate cependant privé des états initiaux et finaux. Une transition $\langle s, u, s' \rangle$ de T est aussi notée $s \xrightarrow{u} s'$, et u est appelée la *mise à jour* de la transition. Une paire dans $S \times \mathbb{Z}^n$ est appelée une *configuration* de $\mathcal{S}$. Une configuration est dite *admissible* lorsqu'elle appartient à $S \times \mathbb{N}^n$. Soit $\vdash \subseteq (S \times \mathbb{Z}^n) \times (S \times \mathbb{Z}^n)$ la relation représentant un pas de calcul et définie ainsi : $\langle s, x \rangle \vdash \langle s', x' \rangle \stackrel{\text{def}}{\Leftrightarrow}$ il existe une transition $\langle s, u, s' \rangle$ dans T telle que $x' = x + u$. On note $\vdash_{adm}$ la restriction de $\vdash$ aux seules configurations admissibles. Un *pseudo-calcul* [resp. *calcul*] de $\mathcal{S}$ est une séquence de configurations $\langle s_0, x_0 \rangle \ldots \langle s_k, x_k \rangle$ telle que $j \in [0, k-1]$, $\langle s_j, x_j \rangle \vdash \langle s_{j+1}, x_{j+1} \rangle$ [resp. $\langle s_j, x_j \rangle \vdash_{adm} \langle s_{j+1}, x_{j+1} \rangle$]. Un calcul $\langle s_0, x_0 \rangle \ldots \langle s_k, x_k \rangle$ est donc une exécution du SAVE $\mathcal{S}$ vu comme un programme avec n compteurs, de la configuration initiale $\langle s_0, x_0 \rangle$ à la configuration $\langle s_k, x_k \rangle$. On écrit alors $\langle s_0, x_0 \rangle \stackrel{*}{\vdash}_{adm} \langle s_k, x_k \rangle$ lorsqu'il existe un calcul de la forme $\langle s_0, x_0 \rangle \ldots \langle s_k, x_k \rangle$. Les relations $\vdash$ et $\stackrel{*}{\vdash}_{adm}$ dépendent donc toujours d'un SAVE et dans la suite il sera toujours clair quel est le SAVE sous-jacent lorsque ces relations sont utilisées.

QUESTION 1. Ecrire en pseudo-code un algorithme qui prenne en entrée un SAVE $\mathcal{S} = \langle S, T, n \rangle$ et une séquence non vide $\langle s_0, x_0 \rangle \ldots \langle s_k, x_k \rangle$ de configurations et qui détermine si la séquence est un calcul de $\mathcal{S}$.

Les questions 2, 3 et 5 vont faire référence au SAVE $\mathcal{S}_{mult} = \langle \{\mathrm{A}, \mathrm{B}\}, \{t_1, \ldots, t_4\}, 4 \rangle$ tel que

$$t_1 \stackrel{\text{def}}{=} \mathrm{A} \xrightarrow{\begin{pmatrix} 0 \\ -1 \\ 1 \\ 0 \end{pmatrix}} \mathrm{A} \quad t_2 \stackrel{\text{def}}{=} \mathrm{A} \xrightarrow{\begin{pmatrix} -1 \\ 0 \\ 0 \\ 0 \end{pmatrix}} \mathrm{B} \quad t_3 \stackrel{\text{def}}{=} \mathrm{B} \xrightarrow{\begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}} \mathrm{A} \quad t_4 \stackrel{\text{def}}{=} \mathrm{B} \xrightarrow{\begin{pmatrix} 0 \\ 1 \\ -1 \\ 1 \end{pmatrix}} \mathrm{B}$$

La figure ci-dessous représente graphiquement $\mathcal{S}_{mult}$: chaque noeud correspond à un état de contrôle et chaque arc représente une transition, la mise à jour étiquetant l'arc. Dans les réponses, on pourra utiliser les noms de transition se trouvant entre crochets.



QUESTION 2. Calculer l'ensemble $\{\begin{pmatrix}a\\b\\c\\d\end{pmatrix} \in \mathbb{N}^4 : \langle \mathrm{A}, \begin{pmatrix}2\\1\\0\\0\end{pmatrix}\rangle \vdash^*_{adm} \langle \mathrm{A}, \begin{pmatrix}a\\b\\c\\d\end{pmatrix}\rangle\}$.

QUESTION 3. Montrer que

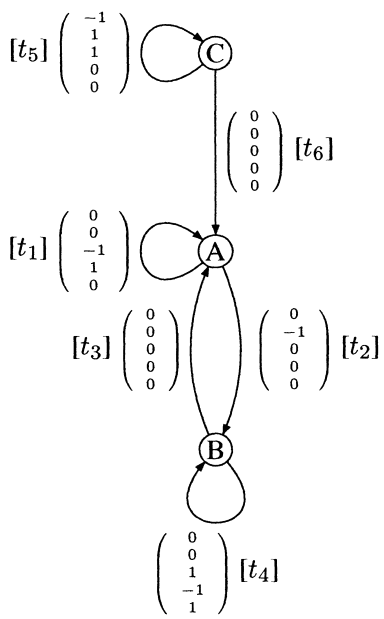
$$\{\begin{pmatrix}a\\b\\d\end{pmatrix} \in \mathbb{N}^3 : d \leq a \times b\} = \{\begin{pmatrix}a\\b\\d\end{pmatrix} \in \mathbb{N}^3 : \exists \begin{pmatrix}a'\\b'\\c'\end{pmatrix} \in \mathbb{N}^3, \langle \mathrm{A}, \begin{pmatrix}a\\b\\0\\0\end{pmatrix}\rangle \vdash^*_{adm} \langle \mathrm{A}, \begin{pmatrix}a'\\b'\\c'\\d\end{pmatrix}\rangle\}$$

On dit alors que $\mathcal{S}_{mult}$ *calcule faiblement* la multiplication. Les SAVE permettent de calculer faiblement de nombreuses autres fonctions usuelles.

QUESTION 4. Déterminer la fonction f telle que

$$\{\begin{pmatrix}a\\b\\e\end{pmatrix} \in \mathbb{N}^3 : e \leq f(a,b)\} = \{\begin{pmatrix}a\\b\\e\end{pmatrix} \in \mathbb{N}^3 : \exists \begin{pmatrix}a'\\b'\\c'\\d'\end{pmatrix} \in \mathbb{N}^4, \langle \mathrm{C}, \begin{pmatrix}a\\0\\b\\0\\0\end{pmatrix}\rangle \vdash^*_{adm} \langle \mathrm{A}, \begin{pmatrix}a'\\b'\\c'\\d'\\e\end{pmatrix}\rangle\}$$

pour le SAVE représenté ci-dessous



Dans la suite, nous cherchons à résoudre les problèmes ci-dessous.

(P1) Etant donnés un SAVE $\mathcal{S}$, une configuration admissible $\langle s, x\rangle$ et un état de contrôle (acceptant) s_{Acc}, existe-t-il $x' \in \mathbb{N}^n$ tel que $\langle s, x\rangle \overset{*}{\vdash}_{adm} \langle s_{Acc}, x'\rangle$? En d'autres termes, existe-t-il un calcul qui puisse atteindre un état de contrôle à partir d'une configuration admissible initiale ?

(P2) Etant donnés un SAVE $\mathcal{S}$ et une configuration admissible $\langle s, x\rangle$, l'ensemble des configurations $\langle s', x'\rangle$ telles que $\langle s, x\rangle \overset{*}{\vdash}_{adm} \langle s', x'\rangle$ est-il infini ?

QUESTION 5. Pour le SAVE $\mathcal{S}_{mult}$, existe-t-il $\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \in \mathbb{N}^4$ tel que l'ensemble

$$\left\{\begin{pmatrix} a' \\ b' \\ c' \\ d' \end{pmatrix} \in \mathbb{N}^4 : \langle \mathrm{A}, \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}\rangle \overset{*}{\vdash}_{adm} \langle \mathrm{A}, \begin{pmatrix} a' \\ b' \\ c' \\ d' \end{pmatrix}\rangle\right\}$$

soit infini ? On justifiera sa réponse.

QUESTION 6. Ecrire en pseudo-code un algorithme qui prenne en entrée un tuple $\begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \in \mathbb{N}^4$ et qui retourne l'ensemble des tuples $\begin{pmatrix} a' \\ b' \\ c' \\ d' \end{pmatrix} \in \mathbb{N}^4$ tels que $\langle \mathrm{A}, \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}\rangle \overset{*}{\vdash}_{adm} \langle \mathrm{A}, \begin{pmatrix} a' \\ b' \\ c' \\ d' \end{pmatrix}\rangle$. On évaluera le temps de calcul en fonction de $a + b + c + d$.

Par définition, une transition $s \xrightarrow{u} s'$ est une *auto-transition* lorsque $s = s'$.

QUESTION 7. Montrer que si l'on possède un algorithme pour résoudre les problèmes (P1) et (P2) restreints aux SAVE sans auto-transition, alors il existe un algorithme pour résoudre les problèmes (P1) et (P2) (sans aucune restriction).

SAV. Pour résoudre les problèmes sur les SAVE, nous introduisons une famille de systèmes un peu plus rudimentaires, qui sont des SAVE sans états de contrôle explicites. Un *système d'addition de vecteurs* (SAV) est la donnée d'un entier $n \geq 1$ (sa dimension) et d'un ensemble fini $T \subseteq \mathbb{Z}^n$ (dans la suite on identifiera le SAV avec T). Un élément $\mathsf{tr} \in T$ sera appelé une *transition*. Un *chemin* est une séquence finie de transitions. Le chemin c' est un *sous-chemin* du chemin $c = \mathsf{tr}_1 \dots \mathsf{tr}_k \overset{\text{def}}{\Leftrightarrow}$ il existe $1 \leq j_1 < j_2 \dots < j_{k'} \leq k$ tel que $c' = \mathsf{tr}_{j_1} \dots \mathsf{tr}_{j_{k'}}$, ce qui sera noté $c' \sqsubseteq c$.

Promenade. Une configuration x du SAV T est un élément de $\mathbb{Z}^n$. Une configuration x est dite *admissible* lorsque $x \in \mathbb{N}^n$. Si $c = \mathsf{tr}_1 \dots \mathsf{tr}_k$ est un chemin, la *promenade* $\langle c, x\rangle$ est la suite de configurations $x_0 \cdots x_k$ telle que $x_0 = x$ et pour $i \in [1, k]$, $x_i = x_{i-1} + \mathsf{tr}_i$. La promenade $x_0 \cdots x_k$ est dite *induite* par le chemin c et de *longueur* $k+1$; c est de *longueur* k. La configuration x_0 est *initiale* dans la promenade $x_0 \cdots x_k$ et x_k est dite *finale*. De plus, x_k est dite *accessible* de x_0. La promenade $x_0 \cdots x_k$ est dite *admissible* lorsque toutes les configurations intermédiaires sont dans $\mathbb{N}^n$. Dans ce cas, la configuration x_k est dite *positivement accessible* de x_0. Ce qui nous intéresse vraiment, ce sont les promenades admissibles, ne serait-ce pour résoudre nos problèmes sur les SAVE qui font appel à des configurations admissibles : toutes les composantes sont positives. Cependant, la possibilité d'avoir des valeurs négatives pour certaines configurations d'une promenade d'un SAV, nous sera utile quelquefois.

Dans le reste du sujet, on s'intéresse aux deux problèmes suivants sur les SAV.

(P1′) Etant donnés un SAV T et deux configurations admissibles $x, y \in \mathbb{N}^n$, existe-t-il une configuration y' positivement accessible de x telle que $y \preceq y'$? Le triplet $\langle T, x, y\rangle$ est appelé une *instance du problème de couverture* et y' est appelée une *solution*.

(P2′) Etant donnés un SAV T et une configuration admissible $x \in \mathbb{N}^n$, existe-t-il un nombre infini de configurations positivement accessibles de x ?

QUESTION 8. Montrer que si l'on possède un algorithme pour résoudre les problèmes (P1′) et (P2′) sur les SAV, alors il existe un algorithme pour résoudre les problèmes (P1) et (P2) sur les SAVE.

PARTIE II : ATTEIGNABILITÉ D'UN ÉTAT ACCEPTANT

L'objectif de cette partie est de proposer un algorithme qui résolve le problème (P1′).

Tailles. Le *pic minimal* d'un élément $x \in \mathbb{Z}^n$, noté $\mathrm{pic}(x)$, est $\max(\{\max(0, -x(i)) : i \in [1,n]\})$. Le *maximum* d'un élément $x \in \mathbb{Z}^n$, dénoté $\max(x)$ est $\max(\{x(i) : i \in [1,n]\})$.

On note $\| \ T \ \|_{max}$ le maximum $\max(\{|\mathrm{tr}(i)| \ : \ \mathrm{tr} \in T, \ i \in [1,n]\})$. Le *pic minimal* d'un SAV T, dénoté par $\mathrm{pic}(T)$, est $\max\{\mathrm{pic}(\mathrm{tr}) : \mathrm{tr} \in T\}$. La *taille* de T, notée $\mathrm{taille}(T)$, est la valeur $n \times \mathrm{card}(T) \times (2 + \lceil log_2(1 + \| \ T \ \|_{max})\rceil)$. On peut remarquer que $\max(\mathrm{pic}(T), \mathrm{card}(T)) \leq 2^{\mathrm{taille}(T)}$ et que $(1 + \lceil log_2(N)\rceil)$ correspond à un nombre suffisant de bits pour représenter l'entier N en binaire (pour les entiers relatifs, un bit est aussi utilisé pour le signe).

QUESTION 9. Définir une instance $\langle T, x, y\rangle$ du problème de couverture ayant au moins une solution et dont chaque solution soit positivement accessible de x avec une promenade de longueur au moins égale à $n \times \max(y)$.

Dans la suite, on se fixe un SAV T et une configuration à couvrir y. Pour $I \subseteq [1,n]$, une promenade $x_0 \ldots x_k$ est dite *I-admissible* lorsque pour $i \in I$ et $j \in [0,k]$, nous avons $x_j(i) \geq 0$. Une promenade $x_0 \ldots x_k$ est dite *I-r-admissible* avec $r > 0$ si pour $i \in I$ et $j \in [0,k]$, nous avons $0 \leq x_j(i) < r$. Une promenade $x_0 \ldots x_k$ est dite *I-couvrante* si pour $i \in I$, nous avons $x_k(i) \geq y(i)$, c'est-à-dire $y_I \preceq (x_k)_I$.

QUESTION 10. Soient $I \subseteq [1,n]$ et $x_0 \ldots x_k$ une promenade I-admissible et I-couvrante.

1. Pour chaque $\Delta \in \mathbb{Z}^n$ tel que $\Delta_I \in \mathbb{N}^I$, montrer que la séquence $(x_0 + \Delta) \ldots (x_k + \Delta)$ est une promenade I-admissible et I-couvrante.
2. Supposons qu'il existe $0 \leq \alpha < \beta \leq k$ tels que $(x_\alpha)_I = (x_\beta)_I$ avec $\Delta = x_\alpha - x_\beta$. Montrer que la séquence $x_0 \ldots x_{\alpha-1}(x_\beta + \Delta) \ldots (x_k + \Delta)$ est une promenade I-admissible et I-couvrante.

QUESTION 11. Soit $x_0 \ldots x_k$ une promenade I-r-admissible induite par le chemin c avec $I \subseteq [1,n]$ et $r > 0$. Montrer qu'il existe un chemin $c' \sqsubseteq c$ de longueur strictement inférieure à $r^{\mathrm{card}(I)}$ tel que $\langle c', x_0\rangle = y_0 \cdots y_l$ soit I-r-admissible et $(y_l)_I = (x_k)_I$. Si de plus, $x_0 \ldots x_k$ est I-couvrante, alors $y_0 \cdots y_l$ est aussi I-couvrante.

QUESTION 12. Ecrire en pseudo-code un algorithme qui prenne en entrée une instance $\langle T, x, y\rangle$ et un entier $k \in \mathbb{N}$ et détermine s'il existe une configuration y' positivement accessible de x avec un chemin de longueur au plus k telle que $y \preceq y'$.

Pour $x \in \mathbb{Z}^n$ et $I \subseteq [1,n]$, on note $M(I,x)$ la plus petite longueur d'une promenade I-admissible et I-couvrante dont la configuration initiale est x. Si une telle promenade n'existe pas, $M(I,x)$ prend la valeur zéro par défaut.

Dans la suite, on verra que $\{M(I,x) : x \in \mathbb{Z}^n\}$ possède un maximum. Soit $f(I) \stackrel{\text{def}}{=} \max(\{M(I,x) : x \in \mathbb{Z}^n\})$. On remarque que f dépend implicitement de T et de y. De plus, pour $\emptyset \subset I \subseteq [1,n]$, $f_{\subset}(I) \stackrel{\text{def}}{=} \max(\{f(I') : I' \subset I\})$ et pour $m \in [0,n]$, $F(m) \stackrel{\text{def}}{=} \max(\{f(I) : I \subseteq [1,n],\ \mathrm{card}(I) \leq m\})$.

QUESTION 13. Calculer $F(0)$.

QUESTION 14. Ecrire en pseudo-code un algorithme qui prenne en entrée une instance $\langle T, x, y\rangle$ telle que $\mathrm{pic}(T) \times \max(y) = 0$ et qui détermine s'il existe une configuration y' positivement accessible de x telle que $y \preceq y'$. De plus, lorsque $\mathrm{pic}(T) \times \max(y) = 0$, majorer $F(n)$.

Dans la suite, on suppose que $\mathrm{pic}(T) \times \max(y) \geq 1$. On dit que $I \subseteq [1,n]$ satisfait la propriété $(\star)$ dans le cas suivant : pour toute promenade $\langle c, x\rangle$ I-admissible et I-couvrante, il existe un chemin c' de longueur au plus $(f(I) - 1)$ tel que $c' \sqsubseteq c$ et $\langle c', x\rangle$ soit I-admissible et I-couvrante.

QUESTION 15. Soient $I' \subset I \subseteq [1,n]$ et $\langle c, x_0\rangle = x_0 \dots x_l \dots x_k$ une promenade I-admissible et I-couvrante avec $0 \leq l < k$ tels que

- I' satisfait la propriété $(\star)$,
- $x_0 \dots x_{l-1}$ est I-r-admissible avec $\mathsf{r} = \mathrm{pic}(T) f(I') + \max(y)$ et $(x_l)_I \notin [0, \mathsf{r} - 1]^I$,
- pour $i \in I$, nous avons $i \in (I \setminus I')$ si et seulement si $x_l(i) \geq \mathsf{r}$.

Montrer qu'il existe un chemin c' de longueur au plus $\mathsf{r}^{\mathrm{card}(I)} + f(I') - 2$ tel que $c' \sqsubseteq c$ et $\langle c', x_0\rangle$ est I-admissible et I-couvrante.

QUESTION 16. Soit $I \subseteq [1,n]$ tel que pour chaque $I' \subset I$, l'ensemble de composantes I' vérifie la propriété $(\star)$. Montrer que si $\langle c, x\rangle$ est une promenade I-admissible et I-couvrante alors il existe un chemin $c' \sqsubseteq c$ tel que $\langle c', x\rangle$ est I-admissible et I-couvrante, et de longueur inférieure à $(\mathrm{pic}(T) f_{\subset}(I) + \max(y))^{\mathrm{card}(I)} + f_{\subset}(I) - 1$.

QUESTION 17. Lorsque le produit $\mathrm{pic}(T)\max(y)$ vaut au moins 1, majorer $F(n)$ en fonction de n, $\mathrm{pic}(T)$ et $\max(y)$, par exemple par $((\mathrm{pic}(T) + 2)\max(y))^{(n+1)!}$.

QUESTION 18. Ecrire en pseudo-code un algorithme qui prenne en entrée une instance $\langle T, x, y\rangle$ et qui retourne une configuration y' positivement accessible de x telle que $y \preceq y'$ si elle existe, sinon false. On justifiera la correction de l'algorithme et on évaluera le temps de calcul.

PARTIE III : CHEMINS

Cette partie met en relation les graphes orientés finis et des systèmes d'inéquations, ce qui sera utile dans la partie IV dédiée à la résolution du problème (P2′).

Graphe orienté. Un *graphe orienté fini* $G = \langle S, A\rangle$ est une paire composée d'un ensemble fini S de *sommets* et d'un ensemble d'*arcs* $A \subseteq S \times S$. Pour tout arc $a = \langle s, s'\rangle \in A$, on note $or(a)$ l'*origine* s de a et $ex(a)$ l'*extrémité* s'. Un *chemin* $c = a_1 \dots a_k$ est une séquence d'arcs de A telle que pour $i \in [1, k-1]$, $ex(a_i) = or(a_{i+1})$. L'origine du chemin c, notée $or(c)$, est $or(a_1)$ et son extrémité, notée $ex(c)$, est $ex(a_k)$. On dit alors que c est un chemin de $or(c)$ vers $ex(c)$. On admet aussi des chemins vides de longueur 0, un pour chaque sommet de graphe. Deux chemins c, c' sont dits *consécutifs* si $ex(c) = or(c')$. Lorsque c et c' sont *consécutifs*, on note cc' le chemin obtenu en concaténant c avec c'.

L'*image* d'un chemin $c = a_1 \dots a_k$ est la fonction $\mathcal{I}_c : A \to \mathbb{N}$ qui compte combien de fois chaque arc est utilisé, c'est-à-dire pour $a \in A$, nous avons $\mathcal{I}_c(a) \stackrel{\text{def}}{=} \mathrm{card}(\{j \in [1,k] : a_j = a\})$. Etant donnée une fonction $\mathcal{I} : A \to \mathbb{N}$, on note $G_{|\mathcal{I}} = \langle S', A'\rangle$ le graphe tel que

- $A' = \{a \in A : \mathcal{I}(a) > 0\}$.
- S' est l'ensemble de sommets $s \in S$ pour lequels il existe un arc $a \in A'$ tel que $s = or(a)$ ou $s = ex(a)$.

Un graphe $G = \langle S, A\rangle$ est dit *connexe* lorsque pour s, s' de S, il existe un chemin de s vers s' dans le graphe $\langle S, A \cup \{\langle t', t\rangle : \langle t, t'\rangle \in A\}\rangle$.

QUESTION 19. Construire un graphe $G = \langle S, A\rangle$ dont on puisse montrer l'existence des images ci-dessous. On donnera des justifications.

1. Il existe des images de chemin $\mathcal{I}, \mathcal{I}' : A \to \mathbb{N}$ telles que $\mathcal{I} + \mathcal{I}'$ ne soit pas l'image d'un chemin de G. $(\mathcal{I} + \mathcal{I}')(a)$ est défini par $\mathcal{I}(a) + \mathcal{I}'(a)$ pour chaque arc a.
2. Il existe des fonctions $\mathcal{I}, \mathcal{I}' : A \to \mathbb{N}$ qui ne soient images d'aucun chemin de G et dont $\mathcal{I} + \mathcal{I}'$ est l'image d'un chemin.
3. Il existe une fonction $\mathcal{I} : A \to \mathbb{N}$ qui soit image de deux chemins distincts de G.

QUESTION 20. Soient c_1 et c_2 deux chemins ayant un sommet en commun et dont $or(c_2) = ex(c_2)$. Montrer qu'il existe un chemin c tel que $\mathcal{I}_c = \mathcal{I}_{c_1} + \mathcal{I}_{c_2}$, $or(c) = or(c_1)$ et $ex(c) = ex(c_1)$.

QUESTION 21. Soient $G = \langle S, A\rangle$ un graphe orienté fini et $c = a_1 \ldots a_k$ un chemin de s à s' avec pour image $\mathcal{I}_c : A \to \mathbb{N}$. Montrer les propriétés ci-dessous.

(I) $G_{|\mathcal{I}_c}$ est connexe.

(II) Si $s = s'$ alors pour $t \in S$, $\sum_{a \in A \text{ tq } ex(a)=t} \mathcal{I}_c(a) \; - \sum_{a \in A \text{ tq } or(a)=t} \mathcal{I}_c(a) = 0$.

(III) Si $s \neq s'$ alors
- pour $t \in S \setminus \{s, s'\}$, $\displaystyle\sum_{a \in A \text{ tq } ex(a)=t} \mathcal{I}_c(a) \; - \sum_{a \in A \text{ tq } or(a)=t} \mathcal{I}_c(a) = 0$.
- $\displaystyle\sum_{a \in A \text{ tq } ex(a)=s} \mathcal{I}_c(a) \; - \sum_{a \in A \text{ tq } or(a)=s} \mathcal{I}_c(a) = -1$.
- $\displaystyle\sum_{a \in A \text{ tq } ex(a)=s'} \mathcal{I}_c(a) \; - \sum_{a \in A \text{ tq } or(a)=s'} \mathcal{I}_c(a) = 1$.

QUESTION 22. Soient $G = \langle S, A\rangle$ un graphe orienté fini et $\mathcal{I} : A \to \mathbb{N}$ une fonction. Montrer que $\mathcal{I}$ est l'image d'un chemin dans G si et seulement s'il existe $s, s' \in S$ tels que les conditions (I), (II) et (III) de la question 21 soient vérifiées.

Système d'inéquations. Soient N et $\alpha \leq \beta \in \mathbb{N}$ trois entiers naturels, $\mathcal{B} = (b_{i,j}) \in [-\beta, \beta]^{N \times \alpha}$ une matrice avec N lignes et α colonnes et $b \in [-\beta, \beta]^N$. Posons $sol^+(\mathcal{B}, b) = \{x \in \mathbb{N}^\alpha : \mathcal{B}x \geq b\}$, l'ensemble des solutions entières positives. Borosh et Treybis ont montré qu'il existe une constante $C > 1$ indépendante de α, β, N, $\mathcal{B}$ et b telle que si $sol^+(\mathcal{B}, b)$ est non vide alors $sol^+(\mathcal{B}, b) \cap [0, \beta^{C \times N} - 1]^\alpha$ est aussi non vide. Dans la suite on admettra cette propriété et on supposera connue la constante C. On note $\mathrm{SOL}^+(\mathcal{B}, b)$ une fonction qui retourne false si $sol^+(\mathcal{B}, b)$ est vide, sinon un élément de $sol^+(\mathcal{B}, b) \cap [0, \beta^{C \times N} - 1]^\alpha$.

QUESTION 23. Soient $\mathcal{B}_0, \mathcal{B}_1, \mathcal{B}_2 \in \mathbb{Z}^{N \times \alpha}$ et $b_0, b_1, b_2 \in \mathbb{Z}^N$. Définir un algorithme pour déterminer s'il existe $y \in \mathbb{N}^\alpha$ tel que $\mathcal{B}_0\, y \geq b_0$, $\mathcal{B}_1\, y > b_1$ et $\mathcal{B}_2\, y = b_2$ en utilisant la fonction SOL^+.

QUESTION 24. Soit $G = \langle S, A\rangle$ et $G' = \langle S', A'\rangle$ deux graphes orientés finis tels que $S' \subseteq S$, $\emptyset \subset A' \subseteq A$ et G' est connexe (G' est un sous-graphe de G). Soient $\mathcal{B} = (b_{i,j}) \in \mathbb{Z}^{\alpha \times \mathrm{card}(A')}$ une matrice avec $\alpha > 0$ et $b \in \mathbb{Z}^\alpha$. A l'aide de la procédure SOL^+, définir un algorithme qui détermine s'il existe un chemin c dans G tel que

- $G_{\mathcal{I}_c} = G'$,
- $\mathcal{B}x \geq b$ avec $x \in \mathbb{N}^{\mathrm{card}(A')}$ et pour $i \in \{1, \ldots, \mathrm{card}(A')\}$, $x(i) = \mathcal{I}_c(\rho(i))$, étant donnée une bijection arbitraire $\rho : \{1, \ldots, \mathrm{card}(A')\} \rightarrow A'$ (contrainte sur l'image de c).

PARTIE IV : PROBLÈME DE FINITUDE

Cette partie est dédiée à la conception d'un algorithme pour résoudre le problème (P2′).

Etant donné un SAV T, une promenade $x_0 \ldots x_k$ (pas nécessairement admissible) est dite *auto-couvrante* lorsqu'il existe $l < k$ tel que $x_l \prec x_k$. Etant donnés un SAV T et une configuration admissible x, Karp et Miller ont montré l'équivalence des propositions ci-dessous :

- il existe un nombre infini de configurations positivement accessibles de x,
- il existe une promenade admissible auto-couvrante de configuration initiale x.

Nous admettrons cette équivalence dans la suite.

Pour $x \in \mathbb{Z}^n$ et $I \subseteq [1, n]$, on note $N(I, x)$ la plus petite longueur d'une promenade I-admissible et auto-couvrante dont la configuration initiale est x. Si une telle promenade n'existe pas, $N(I, x)$ prend la valeur zéro par défaut. On verra que $\{N(I, x) : x \in \mathbb{Z}^n\}$ possède un maximum. Soit $g(I) \stackrel{\text{def}}{=} \max(\{N(I, x) : x \in \mathbb{Z}^n\})$. On remarque que g dépend de T seulement. De plus, pour $\emptyset \subset I \subseteq [1, n]$, $g_{\subset}(I) \stackrel{\text{def}}{=} \max(\{g(I') : I' \subset I\})$ et pour $m \in [0, n]$, $G(m) \stackrel{\text{def}}{=} \max(\{g(I) : I \subseteq [1, n],\ \mathrm{card}(I) \leq m\})$.

Dans la suite, on se fixe un SAV T de dimension n.

QUESTION 25. Soit $x_0 \ldots x_k$ une promenade auto-couvrante. Montrer que pour $\Delta \in \mathbb{Z}^n$, $(x_0 + \Delta) \ldots (x_k + \Delta)$ est aussi auto-couvrante.

QUESTION 26. Soient $r > 0, I \subseteq [1, n]$ et $\mathcal{S} = \langle S, T', n \rangle$ un SAVE tels que

- $S \subseteq [0, r-1]^{\mathrm{card}(I)}$ et $T' \subseteq S \times T \times S$,
- Pour $s, s' \in S$ et $\mathrm{tr} \in T$, nous avons $s \xrightarrow{\mathrm{tr}} s' \in T' \stackrel{\text{def}}{\Leftrightarrow} s + \mathrm{tr}_I = s'$.
- $\langle S, \{\langle s, s' \rangle \in S \times S : \exists\, \mathrm{tr} \in T,\ s \xrightarrow{\mathrm{tr}} s'\} \rangle$ est connexe.

Montrer que s'il existe un pseudo-calcul $\langle s_0, x_0 \rangle \ldots \langle s_k, x_k \rangle$ dans $\mathcal{S}$ tel que $\{s_0, \ldots, s_k\} = S$, $(x_0)_I = s_0$ et $x_0 \prec x_k$, alors il existe un pseudo-calcul $\langle s'_0, x'_0 \rangle \ldots \langle s'_{k'}, x'_{k'} \rangle$ dans $\mathcal{S}$ tel que

3. $s'_0 = s_0, x'_0 = x_0, s'_{k'} = s_k$ et $x'_0 \prec x'_{k'}$,
4. $k' \leq \mathcal{F}(n, r, \mathrm{taille}(T))$ où $\mathcal{F}(n, r, \mathrm{taille}(T))$ est une expression à définir, construite à partir de fonction polynômes, exponentielles. et de constantes (indépendantes des arguments).

On pourra par exemple chercher à majorer $\mathcal{F}(n, r, \mathrm{taille}(T))$ par $(r^{2n} \times 2^{\mathrm{taille}(T)})^{C \times (2 \times r^{2n} + n + 1)}$.

QUESTION 27. Soient $I \subseteq [1, n]$ et $r > 1$ tels qu'il existe une promenade $x_0 \ldots x_k$ I-r-admissible et auto-couvrante. Montrer qu'il existe alors une telle promenade commençant en x_0 de longueur au plus $r^{\mathrm{card}(I)} + \mathcal{F}(n, r, \mathrm{taille}(T))$.

QUESTION 28. Majorer $G(0)$ en fonction de n et $\mathrm{taille}(T)$.

QUESTION 29. Pour $\emptyset \subset I \subseteq [1, n]$, majorer $g(I)$ en fonction de $g_{\subset}(I)$, $\mathrm{pic}(T)$, $\mathrm{taille}(T)$ et n. On pourra par exemple montrer $g(I) \leq (\mathrm{pic}(T) g_{\subset}(I))^{\mathrm{card}(I)} + \mathcal{F}(n, \mathrm{pic}(T) g_{\subset}(I), \mathrm{taille}(T))$.

QUESTION 30. Définir un algorithme qui prenne en entrée un SAV T et une configuration admissible et qui détermine si l'ensemble des configurations positivement accessibles de x est infini.

FIN DU SUJET