Complexité en espace

Exercice 1 — Complexité du langage de Dick

On considère le langage de Dyck D_n^* sur n paires de paranthèses.

1. Montrer qu'il peut être décidé en temps linéaire si un mot appartient à D_n^*.

2. Montrer qu'il peut être décidé en espace logarithmique si un mot appartient à D_n^*.

Exercice 2 — Jeu de Nim

Le jeu de Nim est joué avec une collection de tas d'allumettes. À tour de rôle, chaque joueur retire un nombre non nul arbitraire d'allumettes d'un des tas. Le joueur qui retire la dernière allumette a gagné. Une configuration du jeu de Nim avec k tas est $\langle s_1, \ldots, s_k\rangle$ (le nombre d'allumettes dans chaque tas). Une position est équilibrée si, quand on écrit chaque nombre s_i en binaire, chaque nombre binaire étant écrit sur une ligne différente, aligné avec les autres selon le bit de poids faible, chaque colonne contient un nombre pair de 1.

1. Montrer qu'à partir d'une position non-équilibrée, en un coup, on peut atteindre une position équilibrée.

2. Montrer qu'à partir d'une position équilibrée, on ne peut atteindre, en un coup, que des positions non-équilibrées.

Soit Nim $= \{\langle s_1, \ldots, s_k\rangle \mid$ chaque s_i est un nombre binaire et le joueur qui commence a une stratégie gagnante$\}$.

3. Montrer que Nim$\in L$.

Exercice 3 — Réduction en espace logarithmique

On dit qu'un langage L se réduit au langage L' s'il existe une fonction R de A^* vers A^* calculable par une MT qui utilise un espace $O(\log n)$, telle que $x \in L$ ssi $R(x) \in L'$. On ne compte que les bandes de travail de la machine et pas la bande d'entrée et la bande de sortie.

1. Montrer que si R est une réduction calculée par une MT M, alors pour toute entrée x, M s'arrête après un nombre polynomial d'étapes.

2. Montrer que si R est une réduction de L_1 vers L_2 et R' est une réduction de L_2 vers L_3, alors $R' \circ R$ peut être calculée en espace $O(\log n)$.

Accessibilité$= \{\langle G, s, t\rangle \mid G$ est un graphe orienté qui contient un chamin de s vers $t\}$.

3. Montrer que le problème Accessibilité est NL-complet.

Exercice 4 **Théorème d'Immermann-Szelepcsényi**

Le théorème d'Immermann-Szelepcsényi affirme que pour toute fonction $f(n) \geq \log(n)$, NSPACE$(f(n))$ = co-NSPACE$(f(n))$.

1. Étant donné un graphe G et un sommet x, on suppose que l'on connaît r_i le nombre de sommets atteignables en au plus i étapes à partir d'un sommet s. Donner alors un algorithme non déterministe qui accepte en espace logarithmique t si t n'est pas atteignable en au plus i étapes à partir de s.

2. Donner un algorithme qui calcule r_{i+1} en fonction de r_i en espace logarithmique.

3. En déduire que le problème Accessibilité est co-NL.

4. En déduire que si $f(n) \geq \log n$, alors NSPACE$(f(n))$=co-NSPACE$(f(n))$ et donc NL=co-NL.

Exercice 5 **2-SAT est NL-complet**

Le problème 2-SAT, qui consiste à décider si une formule booléenne où chaque clause contient au plus deux littéraux, est décidable en temps polynomial. On va montrer qu'il est en outre NL-complet. Soit ϕ une instance de 2-SAT. On définit le graphe $G(\phi)$ de la manière suivante : les sommets de $G(\phi)$ sont les variables de ϕ avec leur négation et il existe un arc (α, β) si et seulement si ϕ a une clause $(\neg\alpha \vee \beta)$ ou $(\beta \vee \neg\alpha)$. Le graphe $G(\phi)$ a une curieuse symétrie : si (α, β) est un arc, alors $(\neg\beta, \neg\alpha)$ aussi.

1. Montrer qu'il est aussi possible de le décider en utilisant un espace au plus logarithmique avec une machine non-déterministe. (Montrer que ϕ n'est pas satisfaisable si et seulement s'il y a une variable x telle que il existe un chemin de x à $\neg x$ et de $\neg x$ à x dans $G(\phi)$.)

2. Montrer que le problème est NL-complet. (On supposera que Accessibilité est toujours NL-complet même si le graphe est acyclique.)