

TP 4 - Jeux sur les entiers

25 novembre 2011

Pour toutes questions, suggestions, remarques ou autres, n'hésitez pas à m'envoyer un mail à edbonnet@hotmail.com en mettant en objet [TP Caml].

1 Malin comme un sphinge

Un sphinx (qui pour les besoins de l'exercice prendra mieux la défaite que le sphinx d'Édipe) vous propose le jeu suivant. Il vous donne deux entiers x et y ainsi qu'une liste S d'opérations sur les entiers. Partant de l'entier 1, vous devez alors trouver une séquence minimale d'opérations de S pour atteindre x en restant tout le temps dans l'intervalle $[[1, y]]$. On note $\nu(S, x, y)$ la taille d'une telle séquence minimale et on fixe l'ensemble d'opérations $\mathcal{R} = \langle x \mapsto x + 1, x \mapsto 3x, x \mapsto 4x, x \mapsto x - 2 \rangle$.

Question 1 Combien valent $\nu(\mathcal{R}, 15, 99)$, $\nu(\mathcal{R}, 39, 99)$? Proposer un bon majorant pour $\nu(\mathcal{R}, 99, 99)$. Au fait, pourquoi $\forall x \leq y, \nu(\mathcal{R}, x, y) < \infty$?

Question 2 Trouver le plus petit triplet d'entiers (x, y, z) pour l'ordre lexicographique tel que $\nu(\mathcal{R}, x, z) < \nu(\mathcal{R}, x, y)$. Trouver le plus petit couple d'entiers (x, y) pour l'ordre lexicographique tel que $\nu(\mathcal{R}, x, y) < \nu(\mathcal{R} \setminus \{x \mapsto x - 2\}, x, y)$. À quel adage populaire cette situation correspond-elle?

Question 3 Écrire une fonction de type `(int -> int) list -> int -> int -> automate` qui prend en entrée un ensemble d'opérations S et un entier cible k et un entier maximum n et qui renvoie l'automate $A_{S,k,n}$ correspondant au jeu du sphinx. À quoi correspond $L(A_{S,k,n})$?

Question 4 À quoi correspond $\nu(S, k, n)$ pour l'automate $A_{S,n,k}$? En déduire une fonction de type `(int -> int) list -> int -> int -> int` qui prend en argument S, k, n et retourne $\nu(S, k, n)$.

Question 5 Calculer $\nu(\mathcal{R}, 99, 99)$. Aviez-vous vu juste à la première question?

Question 6 Que valent $\max_k \nu(\mathcal{R}, k, 99)$, $\max_k \nu(\mathcal{R}, k, 999)$, $\max_k \nu(\mathcal{R}, k, 9999)$?

2 Juniper Green

Le jeu de Juniper Green, aussi appelé jeu des diviseurs et des multiples, est un jeu à deux joueurs où chaque joueur joue à tour de rôle un entier entre 1 et N (ici on prendra $N=100$), diviseur ou multiple du nombre précédemment joué par l'adversaire. Chaque nombre ne peut être joué qu'une seule fois. Voici un exemple de début de partie 43-86-2-16-64-8-4-96-6-30...

Question 7 *Quelle règle est-on obligé d'ajouter sur le premier coup pour que le jeu présente un intérêt ?*

Question 8 *Écrire une fonction de type `int -> int list -> int list` qui prend en argument un entier et la liste des coups déjà joués et renvoie la liste des coups suivants possibles.*

Question 9 *Qui gagne si les deux joueurs adoptent la même stratégie de jouer le plus gros entier possible à chaque coup ? Quelle est alors la partie ?*

Question 10 *Qui gagne si les deux joueurs adoptent la même stratégie de jouer le plus petit entier parmi tous leurs coups possibles minimisant le nombre de réponses de l'adversaire ? Quelle est alors la partie ?*

Une stratégie est une fonction qui associe à un historique de partie (liste des coups de la partie en cours) le coup choisi. Une stratégie gagnante est une stratégie qui, appliquée, amène toujours à la victoire.

Question 11 *Expliquer pourquoi un des deux joueurs a une stratégie gagnante. Proposer un algorithme qui calcule une stratégie gagnante pour le joueur concerné. Quelle est sa complexité ?*

Question 12 *Au moyen d'heuristiques concevoir un petit programme qui joue le mieux possible. Faire s'affronter vos programmes.*

Le jeu de Juniper Green possède une version solitaire où le but est de trouver une séquence maximale respectant les règles du jeu.

Question 13 *Proposer un algorithme qui calcule une plus longue séquence. Quelle est sa complexité ?*

Question 14 *Essayer de trouver une plus longue séquence possible.*