Centrale Informatique MP 2009 — Corrigé

Ce corrigé est proposé par Antoine Taveneaux (ENS Lyon) et Vincent Puyhaubert (Professeur en CPGE) ; il a été relu par Olivier Levillain (École Polytechnique) et Vincent Danjean (Enseignant-chercheur en école d'ingénieur).

Le sujet de Centrale de cette année comporte, comme souvent, deux problèmes touchant à des parties bien distinctes du programme d'informatique de MP.

- Le premier est un problème d'algorithmique classique qui se place dans un contexte assez général de « matching », c'est-à-dire d'appariement. Il s'agit de réaliser des couplages entre des éléments d'un ensemble tout en respectant certaines contraintes, comme on chercherait à le faire sur un site de rencontres par exemple. Dans le cadre de cet énoncé, les éléments à coupler sont des points d'un cercle ; la contrainte est que les cordes ne doivent pas se chevaucher.

  Les questions sont très classiques : un peu de combinatoire pour s'échauffer, écriture de méthodes naïves puis écriture et preuve de codes plus efficaces. On répond finalement à deux questions : comment tester si un couplage est correct, et comment obtenir un tel couplage de manière automatique ?
- Le second problème est un sujet un peu fourre-tout sur les automates (reconnaissance, automate minimal, déterminisation). Il est composé de trois exercices complètement indépendants les uns des autres. Le deuxième est le plus délicat, en grande partie à cause de notations compliquées. On avait intérêt le jour du concours à le sauter pour s'intéresser au dernier, beaucoup plus classique et largement abordable.

Au final, tout ceci donne une épreuve assez longue, variée et plutôt intéressante. Si elle n'offre pas la satisfaction de traiter un sujet difficile dans sa globalité, elle a le mérite de permettre de s'exercer en une ou deux heures sur une partie ciblée du programme, quitte à ne pas traiter immédiatement le problème en totalité.

Indications

Partie I

I.A.1 Raisonner sur le nombre de façons de construire une stratégie : combien y a-t-il de candidats susceptible d'être relié à P_6 ? Une fois ce candidat choisi, de combien de façons peut-on relier les autres ?

I.A.2 Raisonner comme précédemment en respectant cette fois le fait que les cordes ne sont plus autorisées à se croiser. Remarquer par exemple que cela impose que $|c[i] - i|$ soit impaire pour tout $i \in [\![1\,;\,6]\!]$.

I.A.3 Généraliser le raisonnement effectué en I.A.1 et conclure par récurrence.

I.B.1 Utiliser le fait que la position exacte des points ne compte pas pour placer les 4 points à des endroits simples. Conclure à l'aide de dessins exhibant les différents ordres possibles de ces points sur le cercle.

I.B.3 Tester naïvement si les segments $[P_i P_{c[i]}]$ et $[P_j P_{c[j]}]$ se croisent pour tout $(i, j) \in [1, 2n]^2$.

I.C.2 Justifier que pour tous $i, j \in [\![1\,;\,2n]\!]$,

```
evalue(i,j) = evalue(i+1,c[i]-1) && evalue(c[i]+1,j)
```

I.C.3 Calculer récursivement `evalue(1,2n)` grâce aux deux questions précédentes.

I.C.4 Raisonner par récurrence sur la quantité $j - i$ pour majorer la complexité de `evalue(i,j)`.

I.D.1 Utiliser l'application φ qui à un entier k compris entre 1 et $2n$ associe le nombre de chasseurs parmi $\{P_1, \ldots, P_k\}$ moins le nombre de fantômes de ce même ensemble.

I.D.2 Décrire une fonction récursive en utilisant les questions précédentes.

Partie II

II.A.1.a Montrer que le langage complémentaire de L (ie $A^* \smallsetminus L$) est l'ensemble des mots de longueur paire.

II.A.1.b Raisonner de préférence sur le complémentaire du langage L. Ce type de question se résout souvent avec le lemme de l'étoile ou avec une démonstration directe proche de la démonstration de ce théorème.

II.A.1.c Les résultats sont identiques à ceux qui précèdent : il suffit d'adapter les preuves.

II.B.a Déterminer l'écriture en base 2 des entiers $0, 2, 3, 4, 5$. Utiliser ces décompositions pour écrire les « lignes » des représentations des n-uplets.

II.B.b Remarquer que les représentations des éléments de *Eg* ne comportent que les lettres $(0, 0)$ ou $(1, 1)$. Les éléments de *Inf*, de leur coté, comportent nécessairement la lettre $(0, 1)$ suivie uniquement de lettres $(0, 0)$ ou $(1, 1)$.

II.B.c Un automate à deux états suffit : la lecture d'un mot représentant un triplet (x, y, z) simule le calcul de $x + y$ bit à bit, avec une retenue éventuelle.

II.B.d Pour reconnaître les représentations des éléments de $\mathcal{Q}$, il suffit « d'effacer » la dernière composante de chacune des lettres apparaissant dans les transitions d'un automate reconnaissant les représentations d'un élément de R.

II.C.2 Construire un automate avec un unique état final, qui n'est l'origine d'aucune transition et qui ne peut être atteint que si l'on a lu la lettre a puis N lettres quelconques.

II.C.3 Considérer les états visités par une exécution réussie sur $a^{\mathrm{N}+1}$ pour montrer qu'il y a au moins $\mathrm{N}+2$ états distincts. Ensuite, considérer les états visités par une exécution qui échoue sur $a^{\mathrm{N}}b$ et montrer que l'on accède ainsi à un nouvel état.

II.C.4 Appliquer directement les méthodes de déterminisation du cours, en éliminant les états non accessibles.

II.C.5 Considérer tous les états auxquels on accède par des mots de longueur $\mathrm{N}+1$ et montrer qu'ils sont deux à deux distincts.

II.C.6 Considérer les états visités par un chemin réussi d'étiquette $a^{\mathrm{N}+1}$ et montrer qu'ils sont deux à deux distincts.

Les conseils du jury

D'après le rapport du jury, cette épreuve consistait à évaluer les candidats sur leurs capacités à concevoir et prouver de petits algorithmes et sur ce qu'ils « comprennent des automates ». La production de programmes et d'automates (qu'il faut dessiner !) a été jugée globalement satisfaisante. En revanche, les questions plutôt mathématiques (validité d'un algorithme, non-reconnaissabilité d'un langage) ont posé plus de difficultés.

I. La chasse aux fantômes

I.A.1 Il s'agit de compter le nombre de façons de partitionner 6 éléments en groupes de 2. Soit c une stratégie. L'entier $c[6]$ est un élément de $\{1, \ldots, 5\}$. Il y a donc 5 possibilités pour le choisir. Ceci étant fait, il ne reste plus qu'à apparier 4 éléments deux à deux, ce qui ramène au cas $n = 2$.

Lorsque $n = 2$, il y a 3 choix pour $c[4]$ et les 2 éléments restants sont nécessairement associés. Chacun de ces choix produit une stratégie différente et toutes les stratégies peuvent être décrites par une suite de tels choix. Ainsi, il y a 3 choix de stratégie pour $n = 2$. Finalement,

$$\boxed{\text{Il y a exactement 15 stratégies (admissibles ou non).}}$$

I.A.2 Remarquons dans un premier temps que si c est une stratégie admissible, alors pour tout entier i la quantité $|c[i] - i|$ est nécessairement impaire. Dans le cas contraire, l'intervalle $\mathrm{I} = [\![\, i\,;\, c[i]\,]\!]$ (resp. $[\![\, c[i]\,;\, i\,]\!]$ si $i > c[i]$) contiendrait un nombre impair d'entiers. Il existerait ainsi nécessairement un entier j de cet ensemble tel que $c[j]$ ne soit pas élément de I, et les cordes $[\mathrm{P}_i\mathrm{P}_{c[i]}]$ et $[\mathrm{P}_j\mathrm{P}_{c[j]}]$ seraient sécantes. Cette condition nécessaire d'admissibilité permet d'éliminer de nombreux cas.

On considère maintenant une stratégie admissible c de taille 3. D'après ce qui précède, $c[1]$ est un élément de $\{2, 4, 6\}$.

- Si $c[1] = 2$, la condition nécessaire impose $c[3] \neq 5$. Il ne reste que les stratégies pour lesquelles $c[3] = 4$ et $c[5] = 6$ ou $c[3] = 6$ et $c[4] = 5$, qui sont toutes deux admissibles.
- Si $c[1] = 4$, on a $c[2] = 3$ donc $c[2] = 3$ et $c[5] = 6$.
- Si $c[1] = 6$, on est dans le cas symétrique du cas $c[1] = 2$ d'où deux possibilités : $c[2] = 3$ et $c[4] = 5$ ou $c[2] = 5$ et $c[3] = 4$.

Finalement, $\boxed{\text{Il y a 5 stratégies admissibles de taille 3.}}$

Les graphes qui suivent représentent ces 5 stratégies, que l'on résume également dans le tableau ci-dessous :

i	1	2	3	4	5	6
$c[i]$	2	1	4	3	6	5
	2	1	6	5	4	3
	4	3	2	1	6	5
	6	3	2	5	4	1
	6	5	4	3	2	1