



Concours national d'informatique  
Épreuve écrite d'algorithmique  
Paris III

Samedi 17 mars 2012

# THE FINAL PROBLEM



## 1 Préambule

Bienvenue à **Prologin**. Ce sujet est l'épreuve écrite d'algorithmique et constitue la première des trois parties de votre épreuve régionale. Sa durée est de 3 heures. Par la suite, vous passerez un entretien (20 minutes) et une épreuve de programmation sur machine (4 heures).

### Conseils

- Lisez bien tout le sujet avant de commencer.
- **Soignez la présentation** de votre copie.
- N'hésitez pas à poser des questions.
- Si vous avez fini en avance, relisez bien, ou préparez votre présentation pour l'entretien.
- N'oubliez pas de passer une bonne journée.

### Remarques

- Le barème est donné à titre indicatif uniquement.
- Indiquez lisiblement vos nom et prénom, la ville où vous passez l'épreuve et la date en haut de votre copie.
- Tous les langages sont autorisés, veuillez néanmoins préciser celui que vous utilisez.
- Ce sont des humains qui lisent vos copies : laissez une marge, aérez votre code, ajoutez des commentaires (**seulement** lorsqu'ils sont nécessaires) et évitez au maximum les fautes d'orthographe, sinon ça va barder.
- Le barème récompense les algorithmes les plus efficaces : écrivez des fonctions qui trouvent la solution le plus rapidement possible.
- Si vous trouvez le sujet trop simple, relisez-le, réfléchissez bien, puis dites-le-nous, nous pouvons ajouter des questions plus difficiles.



## Question 1 (1 point)

Alors ? Êtes-vous sélectionné en finale ?

## Question 2 (2 points)

Proposez une ou des structures de données pour représenter en mémoire les grilles. Cette question est importante car vous utiliserez votre structure pour écrire les fonctions des questions suivantes. Relisez donc entièrement le sujet ainsi que les questions qui suivent avant de répondre.

## Question 3 (3 points)

Écrivez une fonction qui prend en argument l'identifiant d'un candidat (ils sont numérotés de 0 à  $n - 1$  selon leur ordre en haut de la grille) et détermine s'il est sélectionné.

## Question 4 (3 points)

Écrivez une fonction qui retourne le nombre de candidats qui seront sélectionnés.

## Question 5 (4 points)

Écrivez une fonction qui affiche la liste de tous les finalistes.

## Question 6 (4 points)

Les grilles sont automatiquement générées sur le serveur de Prologin et maintenues secrètes. Mais le hacker malicieux que vous êtes a découvert le mot de passe `root`<sup>2</sup>. Ayant à présent accès au serveur, vous pouvez modifier la grille à votre convenance.

Partant d'une grille vide et d'un ensemble d'identifiants correspondant aux candidats que vous aimeriez voir passer en finale (les autres finalistes vous importent peu), écrivez une fonction qui affiche le nombre minimal de barreaux à insérer.

## Question 7 (4 points)

Ça ne va pas. Cette grille manque de barreaux de manière trop flagrante, vous allez vous faire griller. Vous décidez alors de complexifier son apparence, en ajoutant des contraintes.

Écrivez une fonction qui, à partir d'un ordre des candidats à l'arrivée (un tableau représentant les positions de tous les candidats en bas de la grille, par exemple :  $\{1, 4, 2, 3, 0\}$ ), détermine un ensemble de barreaux permettant d'atteindre cette configuration.

## Question 8 (4 points)

Lorsqu'une grille utilise le moins de barreaux possible pour aboutir à un ordre des candidats donné, on la dit *optimale*.

Votre algorithme à la question précédente retourne-t-il toujours une grille optimale ? Si oui, prouvez-le, sinon, effacez-le et recommencez.

---

2. C'était un nombre à deux chiffres.

### Question 9

(3 points)

On souhaite à présent obtenir une grille de hauteur minimale. La fausse raison, c'est qu'elle utilisera moins de place, la vraie raison, c'est qu'elle permettra de savoir à quel point on peut paralléliser l'exécution des barreaux.

Écrivez une fonction qui, à partir d'un ensemble de barreaux, détermine la hauteur minimale nécessaire pour les représenter sans changer l'ordre des candidats à l'arrivée.

### Question 10

(4 points)

Une grille de hauteur minimale est-elle forcément optimale ? Si oui, prouvez-le, sinon, donnez un contre-exemple.

### Question 11

(5 points)

Écrivez une fonction qui à partir d'un ordre des candidats à l'arrivée retourne la hauteur minimale d'une grille permettant de le réaliser.

Vous ne pouvez attaquer les questions suivantes que si vous avez tracé toute la grille de la page 2<sup>3</sup>.

### Question bonus 12

(4 points)

Partant d'une grille, on effectue les opérations suivantes, jusqu'à ce qu'on ne puisse plus le faire.

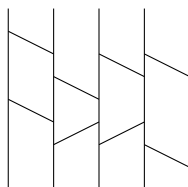


Montrer que ces opérations ne changent pas l'ordre des candidats à l'arrivée<sup>4</sup> et que la grille ainsi générée est optimale<sup>5</sup>.

### Question bonus 13

(2 points)

Les barreaux sont à présent inclinés, et on ne peut pas remonter un barreau, uniquement le descendre (cf. figure suivante). Comment se comportent vos algorithmes à présent ?



Le sujet est sur 37 points, et les questions bonus rapportent au total 6 points, plus 1 point de présentation.

3. Non, ce n'est pas vrai. De toute façon, il n'y a pas de page 8.

4. Facile.

5. Moins facile.