



Concours National d'Informatique

**Algorithmique**  
**Demi-finale Paris / Bordeaux**

Samedi 14 Février 2004

# LAISSE FAIRE : LE RENDEZ-VOUS



## 1 Préambule

Bienvenue à **Prologin**. Ce sujet est l'épreuve écrite d'algorithmique et constitue la première des trois parties de votre demi-finale. Sa durée est de 3 heures. Vous passerez également un entretien (15-20 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.
- 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 la demi-finale et la date en haut de votre copie.
- Tous les langages sont autorisés, veuillez néanmoins préciser celui que vous utilisez.
- Vous pouvez considérer que vous avez à votre disposition une bibliothèque d'algorithmes classiques (tri, tables de hachage, arbres équilibrés, ...).
- 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.



Bien sûr, les Sphères sont parfaitement adaptées à ce type de terrain. Elles se déplacent d'une manière très particulière, mais qui convient tout à fait à ce monde de cubes.

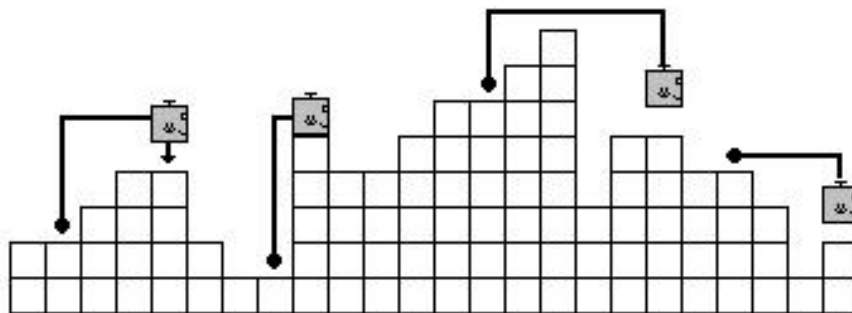
Les Sphères avancent par petits sauts successifs. Un saut se compose de trois parties : la première consiste à monter verticalement d'un nombre entier de cubes. La seconde consiste à avancer horizontalement, toujours d'un nombre entier de cubes. Finalement, la sphère tombe verticalement, jusqu'à ce qu'elle rencontre le sol.

Pour chaque saut, les Sphères disposent d'une certaine quantité d'énergie qu'elles peuvent répartir entre leur déplacement vertical (montée), et leur déplacement horizontal. Cette énergie est de 64 points par saut. Lorsqu'elles ne transportent rien (ce qui est le cas ici), les sphères utilisent 14 points pour monter d'une case, et 7 points pour se déplacer horizontalement d'une case.

Les différents déplacements possibles sont donc les suivants :

- Saut de 4 cases vers le haut, puis zéro ou une case horizontalement ( $4*14 + 1*7 = 63$ )
- Saut de 3 cases vers le haut, puis jusqu'à 3 cases horizontalement ( $3*14 + 3*7 = 63$ )
- Saut de 2 cases vers le haut, puis jusqu'à 5 cases horizontalement ( $2*14 + 5*7 = 63$ )
- Saut d'une case vers le haut, puis jusqu'à 7 cases horizontalement ( $1*14 + 7*7 = 63$ )
- Pas de saut, mais jusqu'à 9 cases horizontalement ( $0*14 + 9*7 = 63$ )

Voici quelques exemples :



Vous comprenez sans doute maintenant pourquoi une Sphère doit bien faire attention à ses choix lors de ses déplacements, pour qu'il soient le plus rentables possibles. On considère qu'un déplacement complet (monter, avancer, puis retomber) prend toujours le même temps. Ce qu'il faut optimiser est donc le nombre de sauts nécessaires pour arriver à destination.

Dans les différentes questions, on considère que le point de départ de la sphère est sur la toute première colonne (à gauche), et que le point d'arrivée est sur la dernière colonne (à droite) du terrain.

Vous pouvez utiliser les constantes LONGUEUR\_TERRAIN et HAUTEUR\_TERRAIN, qui définissent en nombre de cubes, le nombre maximal de cases du terrain, en longueur et hauteur.

### Question 1 (2 points)

Décrivez les structures de données nécessaires pour stocker la composition du terrain le long du chemin qu'empruntera la Sphère, ainsi que la description des sauts successifs que celle-ci souhaite faire.

### Question 2 (3 points)

On vous donne la composition du terrain, une position de départ, et le prochain déplacement qu'a prévu de faire la Sphère (en utilisant vos structures définies à la question 1).

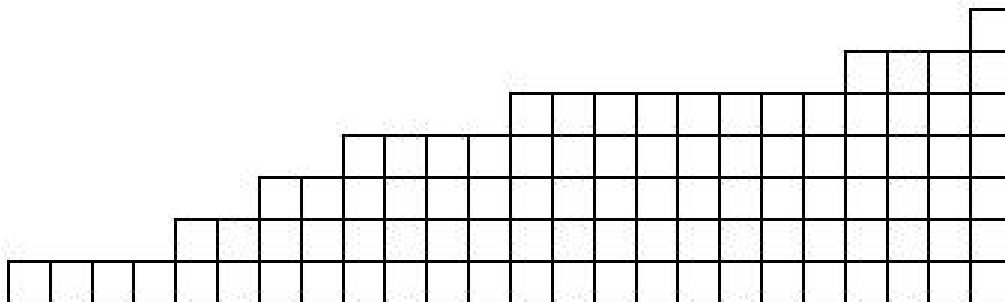
Ecrivez une fonction qui détermine si le déplacement est valide : le saut n'utilise pas plus de 64 points d'énergie et respecte les règles de déplacement d'une sphère. Il ne doit bien sûr pas essayer de faire traverser un cube à la Sphère.

### Question 3 (3 points)

On vous donne maintenant une suite de déplacements complète. Ecrivez une fonction qui détermine si cette suite de déplacements est valide, et mène au point de rendez-vous.

### Question 4 (4 points)

La sphère n'a pas beaucoup de chance, pour son premier rendez-vous : le terrain ne fait que monter, du début à la fin. Il y a bien sûr des zones plates, mais à aucun moment le terrain ne redescend quand on va vers la droite.



Ecrivez une fonction qui à partir de la description du terrain, construit une liste de sauts permettant d'atteindre le point de rendez-vous

### Question 5

Lors d'un deuxième rendez-vous, les choses ne sont plus aussi simples : le terrain est maintenant beaucoup plus irrégulier, et contient aussi bien des descentes que des montées. Le nombre de sauts à effectuer dépend maintenant beaucoup des choix effectués par la Sphère. Un mauvais calcul, et la sphère peut maintenant se retrouver coincée dans une crevasse, sans pouvoir remonter.

- **Partie 1** (5 points)

Ecrivez une fonction qui prend en argument la structure du terrain, et qui retourne le nombre minimum de sauts nécessaires pour traverser le terrain de gauche à droite.

- **Partie 2** (1 point)

Donnez un ordre de grandeur du nombre d'opérations que votre fonction va effectuer.

- **Partie 3** (1 point)

Votre ordinateur est un Pentium 3 à 1 GHz. Donnez un ordre de grandeur du temps qu'il faudra à votre fonction pour fournir le résultat, ainsi qu'une approximation de la quantité de mémoire nécessaire.

## Question 6 (Bonus)

Cette question ne rapportera des points que si vous avez répondu correctement à toutes les questions précédentes.

Pour aider votre sphère à atteindre son point de rendez-vous le plus vite possible, d'autres sphères habitant le long du chemin ont décidé de lui faciliter les choses en modifiant le parcours. En effet, les Sphères sont capables de transporter des cubes de terrain d'un endroit à un autre.

Ecrivez une fonction qui détermine sur quelles colonnes des cubes de terrain doivent être retirés ou ajoutés pour accélérer le trajet de notre Sphère. Votre fonction peut retirer ou ajouter un total de 10 cubes maximum, le long du terrain. Elle peut bien sûr en retirer ou ajouter plusieurs sur une même colonne. Votre fonction devra afficher la liste des colonnes sur lesquelles il faut ajouter, et retirer des cubes, pour obtenir le trajet le plus court possible.

Si vous avez répondu à toutes les questions, relisez bien. Si vous êtes sûr que tout est bon, et qu'il vous reste encore du temps, demandez à un organisateur de vous donner une question supplémentaire.

(+1 point présentation)