



Concours Individuel National d'Informatique

Sujet de demi-finale Nantes / Paris

Samedi 3 février 2001

@!!# DE ELUCNE!!

1 Préambule

Ce sujet constitue la première des trois parties de votre demi-finale Prologin. Cette épreuve a une durée de 3 heures. Par la suite, vous passerez un entretien et une épreuve de programmation sur machine.

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.
- Passez une bonne journée.

Remarques

- Si vous trouvez le sujet trop simple, relisez-le, réfléchissez bien, puis dites-le nous, nous pouvons ajouter des questions plus difficiles.
- Le barème récompense les algorithmes les plus efficaces : écrivez des fonctions qui trouvent la solution le plus rapidement possible.
- N'essayez pas d'acheter ou de torturer les organisateurs, ils n'ont de toutes façons aucune idée des solutions. Celui qui a fait le sujet est à l'autre demi-finale.
- Nous vous conseillons de bien dormir avant de venir.

2 Sujet

Introduction

Vous voilà dans la peau de Séno Janaïd'Ni, un célèbre pilleur de tombes. Vous sévissez aux alentours du Nil à la recherche de votre prochaine "grosse prise", le tombeau du roi Sesram III, où serait soi-disant caché le secret du peuple Sit'Nalta...

Après trois ans de recherche acharnée, vous avez enfin réussi à localiser l'endroit exact de l'emplacement de la pyramide (enfouie) abritant le précieux tombeau. Après quinze jours de fouilles, vous découvrez l'entrée de cette foutue pyramide et vous vous précipitez à travers le dédale de couloirs. Vous arrivez peu de temps après devant une porte qui semble être (si vous en croyez les indications de votre plan) la porte gardienne du trésor Sit'Naltéen.

C'est là que les choses se compliquent : Vous trouvez devant vous une porte scellée sur laquelle est dessinée une grille de caractères. La pièce est tapissée de fresques murales semblant raconter l'histoire de ce lieu. Bien décidé à franchir cet ultime obstacle, vous commencez à déchiffrer les fresques mystérieuses ; vous avez toujours été une brute en traduction.

Après trois heures, voilà que, ouh là là là, vous croyez que vous commencez à comprendre : Le gardien de la porte n'est autre que le Dieu égyptien de la politesse, ce bon vieux Icrem. Il y eut autrefois un combat sans merci entre lui et le Dieu de la grossièreté, Elucne. Ce dernier fut vaincu et banni dans un autre monde astral.

Vous vous rendez ainsi compte que le seul moyen d'ouvrir la porte sera d'invoquer Elucne. Seul lui pourra briser le sortilège qui protège la porte. Pour cela vous disposez d'un grand nombre de barres de longueur variable qu'il vous faudra disposer sur la grille de caractères de manière à proférer les insultes les plus terribles.

N'étant pas le quart d'un mi-céphale, vous retroussiez vos manches et vous vous mettez à préparer quelques insultes copieuses.

Question 1

Chaque insulte est un mot composé de plusieurs caractères. Vous avez retrouvé dans vos anciens cours une liste d'insultes égyptiennes ; vous saviez bien que vous aurez à vous en servir un jour... A chaque insulte est associée un score de grossièreté.

Proposez une structure de données permettant de stocker une liste d'insultes avec le score associé. (On pourra utiliser une constante `NB_INSULTES`). Justifiez votre choix.

Attention, cette question est importante, car vous utiliserez votre solution pour écrire les fonctions des questions suivantes.

Question 2

Ecrivez une fonction qui prend une insulte en paramètre et qui renvoie le score de grossièreté qui lui est associé.

Question 3

Ecrivez une fonction qui prend un mot et une liste d'insultes (cf. question 1) en paramètre et qui renvoie 1 si ce mot est une insulte et 0 dans le cas contraire.

Question 4

Ecrivez une fonction qui prend plusieurs insultes en paramètre et qui renvoie le score total de grossièreté obtenu.

Question 5

On considèrera dans un premier temps une seule ligne de la grille. Notez bien que les barres en pierre ne pourront pas se chevaucher.

Ecrivez une fonction qui prend une chaîne de caractères et une liste d'insultes en paramètre et qui renvoie le score maximal de grossièreté que l'on peut espérer obtenir.

Dans un premier temps, le faire sous la forme d'une fonction qui explore récursivement toutes les combinaisons.

Ensuite, estimez grossièrement le temps d'exécution de votre algorithme sachant qu'on l'exécute sur une Dreamcast avec un processeur Hitachi cadencé à 360 MIPS (millions d'instructions par secondes).

Question 6

Proposez et implémentez une optimisation de la question précédente. Estimez le temps d'exécution de votre nouvel algorithme. Concluez.