

Concours National d'Informatique

Sujet de demi-finale Paris I

Samedi 25 février 2006

LES SMS DE CIMONE

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. 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.
- 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.
- Si vous trouvez le sujet trop simple, relisez-le, réfléchissez bien, puis dites-le nous, nous pouvons ajouter des questions plus difficiles.
- Tous les langages sont autorisés. Néanmoins, veuillez préciser celui que vous utilisez.
- Le barème récompense les algorithmes les plus efficaces : écrivez des fonctions qui trouvent la solution le plus rapidement possible.
- 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.

2 Sujet

Introduction

Cimone (il y a un "C", donc ce n'est pas un prénom de grand-mère mais bel et bien un prénom jeune et dans le vent qui lui permettra peut-être même d'être admise à la Star Academy) est une jeune adolescente qui mène une vie palpitante, avec tous ses amis. En fait, elle a trop d'amis : son forfait de 100 SMS par semaine ne suffit pas, elle ne peut plus leur répondre à tous, ceux ci le prennent mal et risquent de la rejeter... La situation est grave.

Cimone décide de s'organiser, et de choisir judicieusement les SMS qu'elle va envoyer et les amis qu'elle va conserver. Elle attribue donc à chacun de ses amis une note, qui correspond à l'intérêt qu'elle lui porte, et un nombre, qui correspond au nombre minimum de SMS qu'elle doit lui envoyer chaque semaine pour conserver son amitié. (Au cas où vous en douteriez, ces valeurs ne sont absolument pas arbitraires et approximatives, elles ont été déterminées par de savants calculs faisant intervenir des régressions bi-linéaires sur les statistiques de ses précédentes communications, la position des constellations, et même l'âge du capitaine (de l'équipe de foot). Elles sont rigoureusement exactes, inutile donc de les remettre en question ou d'espérer conserver l'amitié d'une personne sans lui envoyer le nombre prescrit de SMS). Cimone a choisi de trier sa liste de copines par ordre décroissant du nombre prescrit de SMS à envoyer (elle connaît parfaitement bien les différents algorithmes de tris). Malheureusement, Cimone se rend vite compte qu'elle va avoir du mal à trouver la configuration qui lui permettra de conserver le plus possible d'amis intéressants. Son désespoir est grand, d'autant plus qu'elle ne peut même pas se confier à sa meilleure amie, pour ne pas gaspiller son précieux forfait.

Heureusement, alors qu'elle défoulait sa colère dans une partie épique de Tétris sur son téléphone portable, elle s'aperçoit qu'en dépassant le score maximum, elle peut exploiter un buffer overflow qui déplace le pointeur d'exécution sur la mémoire vidéo de l'appareil photo. Dès lors, il ne lui reste plus qu'à prendre une photo judicieusement choisie pour que la bitmap correspondante déclenche l'exécution des instructions qu'elle souhaite. Comment Cimone va-t-elle faire pour programmer un algorithme qui lui permettra de résoudre son problème ? Vous pouvez programmer dans le langage de votre choix, on vous épargne la phase de transcription en dessin puis photo puis instructions, Cimone sait le faire parfaitement (encore une fois, inutile de douter : elle dessine merveilleusement bien, connaît l'algorithme de conversion jpg par cœur, et peut faire à volonté des scores inhumains à Tétris).

Question 1 (2 points)

Proposez une structure de données permettant de mémoriser l'ensemble des copines de Cimone. Justifiez votre choix.

Attention, cette question est importante car vous utiliserez votre solution pour écrire les fonctions des questions suivantes. Relisez donc entièrement le sujet (et les questions qui suivent) avant de répondre.

Question 2 (2 points)

Quelle est le nom de la meilleure amie de Cimone ?
Si vous ne le savez pas, écrivez une fonction qui le retrouve. La fonction pourra prendre en argument la liste des copines.

Question 3 (3 points)

Écrivez une fonction qui renvoie la note que Cimone a le plus souvent donnée. Si plusieurs notes ont la même fréquence, renvoyer la note la plus élevée.

Écrivez une fonction qui renvoie le nombre de copines à qui Cimone doit envoyer au moins 5 SMS.

Souvenez-vous comment est triée la liste de copines.

Question 4 (2 points)

Combien d'amies peut-elle conserver au maximum ?

Question 5

Partie 1 (4 points)

Comment optimiser la liste des amis que Cimone gardera ? Elle souhaite que la somme des notes des amis à conserver soit maximale. Aidez-là.

Ecrire une fonction qui prend en argument la liste des copines de Cimone et qui renvoie la somme des notes des amies gardées. Cete somme doit être maximale.

Partie 2 (2 points)

Cimone souhaite envoyer ses SMS le plus rapidement possible. Estimez grossièrement le temps d'exécution de votre algorithme, sachant que Cimone a 100 SMS à envoyer et 42 copines. Le programme sera testé sur son téléphone portable, qui exécute 50 MIPS (millions d'instructions par seconde) et qui possède 4 Mo de mémoire.

Question 6

Pour une raison qui dépasse la compréhension de bon nombre d'hommes, les copines de Cimone sont régulièrement de mauvaise humeur et leur envoyer des SMS durant ces périodes provoque le contraire de l'effet escompté. Fort heureusement, des peta-heures de Tetris ont permis à Cimone d'acquérir un sens de l'observation hors du commun et elle est donc parfaitement capable pour chacune de ses copines de savoir quels jours il ne faut surtout pas leur envoyer de SMS.

Faites intervenir cette information supplémentaire dans votre structure de données, et modifiez votre algorithme pour le rendre optimal en tenant compte de cela.

Questions bonus

Ces questions peuvent vous rapporter des points seulement si vous avez répondu juste à toutes les questions précédentes.

Question 6++

Écrire une fonction qui renvoie le nombre de manières différentes d'obtenir le résultat renvoyé à la question 6.

Question 42

Pourquoi l'âge du capitaine de l'équipe de foot intervient dans les calculs ?
Si vous ne le savez pas, imaginez.

(+1 point présentation)