



Concours National d'Informatique  
Sujet de demi-finale Toulouse

16 février 2008

# Mais d'où vient le fric de Joseph Marchand ?

## 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

Depuis le temps que Prologin existe, Joseph Marchand a eu une vie très occupée : inviter des centaines d'amis à boire du Ricard chez lui, partir en vacances à Barcelone, faire du ski en essayant de passer partout avant que les remontées mécaniques ne ferment, refaire sa salle de bains... Nous sommes sûrs que vous vous demandez tous quelque chose : mais d'où Joseph Marchand tire-t-il tout son fric ? A votre connaissance, il n'a pas de boulot. Il ne programme même pas, puisqu'il vous exploite pour résoudre ses problèmes.

En fait, Joseph Marchand est un fin financier. Après avoir bossé chez Ricard dans un passé lointain, il fait fructifier son capital en spéculant sur les changes. Le principe est simple : si un euro vaut 1,5\$, qu'un dollar vaut 40 roupies (parce que Joseph Marchand connaît bien l'informatique et qu'il connaît les Indiens parce que maintenant ce sont eux qui codent tout), et qu'une roupie vaut 0,02 €, en investissant un euro, il gagnera  $1,5 \times 40 \times 0,02 = 1,20$  €, soit un rendement de 20%.

Joseph Marchand récupère donc les taux de change entre un grand nombre (une constante NB\_DEVISES) de devises sur Internet (il pourrait vous demander de coder le XML/RPC pour faire ça, mais il pense que vous avez mieux à faire, il va plutôt aller demander à des Indiens), et maintenant, il veut gagner de l'argent... à vous de l'aider !

### Question 1 (2 points)

Proposez une ou des structures de données permettant de stocker les taux de changes récupérés par le programme que les Indiens ont réalisé pour Joseph, ainsi qu'une suite d'opérations de change.

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 (1 point)

Écrivez une fonction qui renvoie le rendement pour Joseph Marchand (sous forme d'un pourcentage (positif ou négatif)) d'un aller-retour entre deux devises, euro-dollar-euro par exemple.

### Question 3 (2 points)

Écrivez une fonction qui renvoie la somme que Joseph Marchand gagnera en convertissant une certaine somme d'une devise donnée selon une suite d'échanges que l'on vous donne dans le format que vous avez défini à la question 1.

#### Question 4 (3 points)

Écrivez une fonction qui renvoie les trois meilleurs aller-retours, comme dans la deuxième question, que Joseph Marchand puisse faire.

#### Question 5

##### Question 5a (1 point)

Joseph Marchand veut maintenant que vous fassiez le boulot à sa place (normal, c'est un fainéant) et que vous lui donniez la suite d'opérations de change qui lui rapportera le plus d'argent. Il a besoin d'argent assez vite, et sa banque met un peu de temps à chaque fois qu'il fait une transaction. Il veut donc se limiter à `NB_DEVISES` opérations. Par ailleurs, comme il ne veut pas qu'on se rende compte de ce qu'il fait, il ne veut jamais "passer" deux fois par la même devise dans sa séquence de changes.

Supposez que les sous-traitants indiens de Joseph Marchand aient codé l'algorithme nécessaire pour répondre à cette question. Joseph Marchand désire ajouter une nouvelle monnaie au programme, mais il se rend compte qu'il est très mal codé et très difficilement modifiable. En supposant que le programme indien renvoie néanmoins pour chaque couple de devises  $i$  et  $j$  la meilleure séquence possible de changes les reliant et son rendement, écrivez une fonction qui renvoie les nouveaux rendements optimaux en incluant la monnaie rajoutée par Joseph Marchand.

##### Question 5b (3 points)

Joseph Marchand, content de votre travail, désire maintenant que vous remplaciez le programme indien afin d'avoir un code plus flexible et plus facilement adaptable. Essayez de vous servir de la fonction précédente et de généraliser la méthode appliquée (vous pouvez commencer par écrire un bout de code qui se contraint, pour calculer la meilleure séquence reliant la devise  $i$  à la devise  $j$ , à passer uniquement par les  $k$  premières devises).

A titre indicatif, le nombre de devises sur lesquelles Joseph Marchand spéculé est de l'ordre de 500.

Bien entendu, Joseph Marchand veut partir d'euros et arriver à des euros (si, dans votre structure de données, les devises sont indexées, vous pouvez considérer que les euros ont l'id 0).

#### Question 6

##### Question 6a (4 points)

Joseph Marchand a vraiment besoin d'argent très très vite (il veut un MacBook Air, et il est tellement fan qu'il ne veut pas attendre). Maintenant, il ne veut plus que vous lui donniez la séquence qui va lui rapporter le plus, mais la plus courte, du moment qu'elle lui rapporte de l'argent. Il se limite toujours à `NB_DEVISES` opérations, mais cette fois-ci, comme le problème est un peu plus compliqué, il va se limiter à 100 devises.

##### Question 6b (2 points)

Évaluez le temps que mettra votre programme pour calculer la plus courte séquence de changes telle que définie en question 5b. Joseph Marchand exécute votre programme sur son Qonos (il a chopé un prototype!) à 400 MHz, émulant une HP-49G, qu'il utilisait comme calculatrice financière, et tournait sur un processeur Saturn à 4 MHz.

## Questions bonus

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

### Question 6++

Pour faire plus simple, Joseph Marchand va s'en tenir aux allers-retours, comme dans la question 2. Seulement, il veut pouvoir faire beaucoup d'opérations à la fois, donc il ne veut pas les 3 meilleurs échanges, comme à la question 4, mais les  $k$  meilleurs échanges. Écrivez le programme le plus efficace possible.

### Question 7++

A force de filouter, Joseph Marchand a été interdit de changes par certaines banques centrales. Il y a donc certains types d'échanges qu'il ne peut plus faire. Par exemple, il ne peut plus changer de dollars en euros (mais il peut changer des euros en dollars), ni de roupies en birrs (la monnaie éthiopienne). Il veut quand même continuer ses opérations, mais cette fois-ci, il ne veut surtout pas être détecté. Quelle est la plus longue séquence de changes qui lui rapporte de l'argent et qu'il puisse effectuer sans jamais passer deux fois par la même devise ?

Comme il pense que vous aurez beaucoup de mal à résoudre le problème, ou du moins à lui proposer une solution qui s'exécute en temps raisonnable, il accepte aussi que vous lui donniez une solution approchée.

(+2 points présentation)