



Concours national d'informatique
Épreuve écrite d'algorithmique
Bordeaux, Lyon I

4 Février 2017

LES DESSOUS DE LA FINALE

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.
- 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.
- 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.

2 Sujet

Introduction

Bravo, vous avez été sélectionné en finale¹ ! Vous êtes arrivé tard après un long trajet, et à part l'organisateur solitaire qui vous a fait rentrer, les couloirs sont déserts... Armé de votre sac de voyage, vous entreprenez la recherche périlleuse d'un endroit où dormir. Désespéré après avoir erré de longues minutes dans les méandres de l'EPITA sans voir aucun matelas, vous trouvez une salle avec un ordinateur allumé. Intrigué, vous vous installez, dans l'espoir de trouver un plan. Dans le terminal, vous contemplez le prompt :

```
root@epita:/#
```

Avoir tous les droits sur les machines de prologin, vous n'allez pas laisser passer ça² ! Mais là, vous cherchez juste un matelas. Ni vu ni connu, vous créez une backdoor dans le serveur³, trouvez un plan, et filez vous coucher.

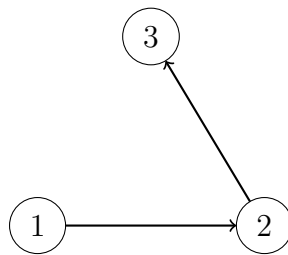
48h plus tard, la finale est finie, il va être l'heure de lancer les affrontements d'IA. Cependant, vous n'êtes pas satisfaits de votre champion... Vous décidez donc d'utiliser votre expérience nocturne pour essayer de modifier vos résultats. Ni une, ni deux, vous allez chercher une crêpe⁴ et trouvez un poste tranquille pour mener à bien votre affaire.

Partie I : Ancien système

En fouillant le serveur, vous avez trouvé une description du système de classement. Les joueurs sont numérotés de 1 à n . Le jeu⁵ oppose deux joueurs, et fournit un gagnant et un perdant, aucune égalité n'est possible. L'ensemble des matchs joués est appelé *tournoi*. Ils sont représentés sous la forme suivante :

- chaque joueur est représenté par son numéro i ;
- on dessine une flèche de i vers j si le joueur i a gagné dans le match entre i et j .

Par exemple, le résultat suivant :



signifie que le joueur 1 a gagné contre 2, qui a gagné contre 3.

Question 1

(1 point)

Dessinez le tournoi obtenu si Alex a gagné contre Bob, que Bob a perdu contre Clément, et que Clément a gagné contre Alex.

Le système de classement commence par lancer tous les matchs possibles une et une seule fois.

1. Enfin, presque, tout dépend de la suite...
2. Osez dire le contraire !
3. N'essayez pas de faire ça pour de vrai, sinon ça va barder.
4. Oui, c'est obligatoire !
5. Vous avez perdu.

Question 2

(2 points)

Combien de matchs vont être lancés au total (en fonction du nombre n de joueurs) ?

Le système de classement actuel consiste à lancer tous les matchs possibles entre deux joueurs, et à trier les joueurs selon leur nombre de victoires.

Question 3

(3 points)

Donnez un exemple de tournoi dans lequel certains joueurs ont le même nombre de victoires. Proposez une méthode qui permet de les départager, si c'est possible.

Question 4

(3 points)

Proposez un moyen de représenter un tournoi dans votre langage favori, et implémentez une fonction qui prend en entrée un tournoi dont les joueurs sont numérotés de 1 à n et qui les ordonne selon les résultats obtenus. Par exemple, sur l'exemple de la question 1, si on numérote les joueurs par ordre alphabétique, vous renverrez : [3; 1; 2], ce qui correspond au classement Clément⁶, Alex, Bob. En cas d'égalité entre deux joueurs, ils pourront être classés dans n'importe quel ordre l'un par rapport à l'autre.

Question 5

(7 points)

On suppose que notre serveur a p processeurs qui peuvent chacun simuler un match à la fois. Cependant, un joueur ne peut pas intervenir simultanément dans deux simulations de matchs.⁷

1. Est-il utile d'avoir plus de $\lfloor \frac{n}{2} \rfloor$ processeurs⁸? (Justifier.)
2. Heureux hasard, le serveur de la finale a justement $p = \lfloor \frac{n}{2} \rfloor$ processeurs⁹! Supposons qu'un match dure d secondes. Déterminez en fonction de n et d le temps minimum que prend la simulation du tournoi. Vous préciserez la méthode utilisée pour obtenir cette borne. Indication : on pourra commencer par gérer le cas où n est pair, puis étendre la méthode au cas général.

Partie II : Retour à la réalité

Rien ne va plus ! À force de fouiner dans le serveur, vous avez perdu la notion du temps ! Le serveur a détecté que le tournoi aurait dû commencer depuis longtemps... Il ne reste plus assez de temps pour que le système de classement précédent ait le temps de classer les candidats. Un protocole de secours prend alors le relais. Il fonctionne comme suit :

- au premier tour, on lance $\frac{n}{2}$ matchs où n est le nombre de joueurs. On suppose que n est une puissance de 2 ($n = 2^k$, avec k entier) ;
- les perdants sont définitivement éliminés ;
- au tour 2, on réitère avec les $\frac{n}{2}$ gagnants, et on continue jusqu'à ce qu'il reste un seul joueur ;
- le joueur restant est donc classé premier. Si un joueur perd au i -ème tour, il est classé $(2^{n-i} + 1)$ -ème *ex æquo*¹⁰.

6. Normal, c'est le meilleur dev de France !

7. Pour des raisons techniques que vous ne cherchez pas à investiguer.

8. On rappelle que $\lfloor x \rfloor$ est le plus grand entier inférieur ou égal à x .

9. Le nombre de candidats est vraiment bien choisi...

10. Seuls les meilleurs comptent, départager les perdants, c'est dépassé.

Par exemple : 4 joueurs, Alex, Bob, Clément et Damien s'affrontent. Au premier tour, Alex joue contre Bob, et Clément contre Damien. Bob et Damien perdent, et sont donc classés tous les deux troisièmes. Alex et Clément s'affrontent alors, le gagnant, Alex, finit premier, et le perdant deuxième.

Question 6

(2 points)

Proposez un moyen de représenter ce nouveau format de tournoi, et dessinez le tournoi décrit précédemment.

Question 7

(6 points)

Intrigué par ce nouveau système, vous vous demandez s'il va vraiment finir à temps. Vous vous posez donc quelques questions statistiques :

1. Combien de matchs vont se dérouler au total ?
2. Combien de matchs chaque joueur va-t-il jouer dans le meilleur des cas ? dans le pire cas ? en moyenne ?

Après réflexion, vous vous rendez compte que le classement obtenu dépend fortement de l'ordre des matchs à chaque tour : si Alex et Bob sont tous deux beaucoup plus forts que Clément et Damien, Bob finit quand même moins bien classé que Clément. Vous essayez alors de tirer parti de cette faille.

Question 8

(4 points)

1. Que peut-on dire d'un joueur J qui est meilleur que tous les autres (c'est-à-dire qu'il gagnera tous les matchs, quel que soit son adversaire). De même, que dire d'un joueur qui est plus mauvais que tous les autres ?
2. Montrez que si un joueur est meilleur que tous les autres sauf J (autrement dit, il perd s'il affronte J , mais gagne s'il affronte n'importe quel autre joueur), vous pouvez choisir l'ordre des matchs pour le faire finir à n'importe quelle place sauf la première.

Cependant, les développeurs ont pensé à tout. Pour éviter ce genre de magouilles, le serveur utilise les archives des précédentes finales pour établir un pré-classement. Par exemple, si on sait que Alex a toujours eu une meilleure place que Bob, on pré-classera Alex au-dessus que Bob. En revanche, si c'est la première finale pour Clément et Damien, ils seront pré-classés aléatoirement parmi les derniers¹¹.

Pour ne pas déséquilibrer le tournoi, les candidats sont ordonnés afin que les matchs entre les meilleurs pré-classés arrivent le plus tard possible dans le tournoi. Pour ce faire, vous observez qu'à chaque tour, le niveau des matchs (la moyenne des pré-classements des joueurs) est toujours le même, en supposant que l'issue des matchs respecte le pré-classement.

Par exemple, si le pré-classement est : Alex (1), Bob (2), Clément (3) et Damien (4), on commence par Alex vs Damien (de niveau moyen 2.5) et Bob vs Clément.

11. Eh oui, la vie est dure !

Question 9

(6 points)

Écrivez une fonction qui, étant donné un pré-classement, renvoie la liste des matchs qui se dérouleront (en supposant que le gagnant d'un match est toujours le joueur ayant le meilleur pré-classement).

Question 10

(3 points)

Vous devez gérer un tournoi à n joueurs. Cependant, le programme interne du serveur ne gère que des nombres de joueurs qui sont des puissances entières de 2. Comment l'utiliser en conservant la même structure et en exploitant l'idée de pré-classement ? Donnez un exemple de tournoi à six joueurs¹².

Question 11

(5 points)

Supposons qu'on ait n joueurs et p processeurs. Si un match dure d secondes, combien de temps avez-vous besoin pour simuler du tournoi entier avec ce modèle ? On précisera dans quel ordre jouer les matchs. Comme précédemment, chaque joueur ne peut jouer qu'un match à la fois. *Cette question est ouverte, il n'y a pas de réponse spécifique attendue. Vous êtes libres de la traiter comme bon vous semble.*

Partie III : Pré-classement

Youpi ! Vous avez réussi à couper l'accès du serveur à ses archives. Il ne peut alors plus les utiliser pour élaborer son pré-classement. En revanche, vous n'avez pas encore gagné ! Les développeurs ayant pensé à tout, le serveur est programmé pour générer un pré-classement en faisant s'affronter les joueurs.

Question 12

(3 points)

Vous remarquez un bout de code en commentaire¹³ qui gérait un système de poules. Plus précisément, il génère aléatoirement des sous-ensembles de joueurs de tailles similaires, il effectue ensuite un tournoi restreint avec ces joueurs, puis il sélectionne les meilleurs. Donnez un exemple où les joueurs issus des poules ne sont pas les meilleurs parmi les joueurs initiaux.

Après un 42^e examen du code, vous avez trouvé la procédure utilisée. Le serveur attribue aléatoirement à chaque joueur un adversaire et lance $\frac{n}{2}$ matches¹⁴. Il sépare alors d'un côté les $\frac{n}{2}$ gagnants et de l'autre les $\frac{n}{2}$ perdants. Il réitère alors le processus sur les deux groupes de joueurs, jusqu'à ce que chaque joueur soit séparé des autres. Le pré-classement est défini comme suit :

- on commence par trier selon le nombre de victoires ;
- si deux joueurs ont le même nombre de victoires, on cherche le premier match pour lequel leurs résultats sont différents, le gagnant est alors pré-classé au-dessus du perdant.

Question 13

(2 points)

12. La recherche des noms des joueurs sera valorisée !

13. Que vous décryptez quand même, au point où vous en êtes...

14. Un seul match par joueur.

Écrivez une fonction qui prend en entrée les résultats de deux joueurs et qui renvoie celui qui a le meilleur pré-classement.

Question 14

(3 points)

Combien d'itérations de ce processus sont nécessaires pour faire en sorte que tous les joueurs aient un pré-classement différent ? Combien de matchs cela ajoute-t-il au tournoi ?

Partie IV : Quelques améliorations

C'est fini, il est venu le temps¹⁵ de publier les résultats, vous n'avez pas réussi¹⁶ à changer le fonctionnement du serveur. De plus, votre incursion a été repérée, et, malgré toutes vos précautions, vous avez été démasqué. Voulant esquiver une condamnation à d'atroces supplices¹⁷ promise par les organisateurs, vous essayez de marchander en proposant d'améliorer leur système. Intéressés par votre proposition, ceux-ci rangent leurs fouets-RJ45, et attendent impatiemment vos explications.

Après quelques instants de réflexion, vous leur faites remarquer que vous trouvez anormal de venir jusqu'à l'EPITA et être éliminé après un seul match¹⁸. Vous cherchez alors (rapidement) un système qui respecte le *principe de double élimination* : un joueur est éliminé si et seulement s'il a perdu deux matchs.

Vous mettez donc en place un tournoi de repêchage, auquel on accède en perdant un match dans le tournoi principal. À chaque étape, deux joueurs Alex et Bob du tournoi de repêchage s'affrontent. Le perdant Alex est éliminé. Pendant ce temps, dans le tournoi principal, Clément et Damien s'affrontent. Le gagnant Clément reste en lice, pendant que Damien arrive dans le tournoi de repêchage, et affronte Bob. Encore une fois, le perdant est définitivement éliminé, pendant que le gagnant poursuit sa course dans le tournoi de repêchage. Pour initialiser le processus, on lance $\frac{n}{2}$ matchs, les gagnants restent dans le tournoi principal, les perdants commencent le tournoi de repêchage.

Question 15

(6 points)

Les organisateurs écoutent avec attention. Surgissent alors des questions que vous n'aviez pas prévues :

1. Le gagnant du tournoi de repêchage n'a perdu qu'un seul match. Adaptez la méthode pour préserver le principe de double élimination.
2. Comment classer les joueurs une fois qu'on a obtenu les résultats du tournoi ?

Question 16

(10 points)

Voulant défendre votre format de tournoi, vous devez aussi parler des spécifications. Avec ce nouveau format,

1. Combien de matchs vont être joués ?

15. Des cathédraaaa-leuh !

16. Bien fait, sale tricheur !

17. Par exemple, coder en PHP enfermé dans une cave secrète pour le restant de vos jours.

18. Les crêpes, c'est bien, gagner, c'est mieux !

2. Combien de matchs chaque joueur jouera-t-il au pire? au mieux? en moyenne?
3. Combien de joueurs vont jouer au moins 3 matchs?

Partie bonus

Question bonus 17

(5 points)

Montrez que dans les tournois de la partie I, on peut toujours numéroter les joueurs de 1 à n de manière à ce que, pour $1 \leq i < n$, le joueur i ait gagné son match contre le joueur $i + 1$.

Question bonus 18

(4 points)

On considère un tournoi entre n joueurs qui respecte les règles suivantes :

- le perdant d'un match est définitivement éliminé ;
- un match ne peut avoir lieu que si le nombre de victoires des joueurs concernés diffère d'au plus 1 ;
- le tournoi se termine quand il ne reste plus qu'un joueur en lice.

Vous avez le choix de quels matchs lancer, et dans quel ordre. En supposant qu'un des joueurs gagne tous les matches qu'il joue, combien de matches peut-il jouer au maximum ?

Question bonus 19

(4 points)

Donnez des inconvénients du système décrit en partie III.

Question bonus 20

(3 points)

Proposez une méthode pour placer les perdants dans le tournoi de repêchage (cf partie IV) de manière à minimiser le nombre de matchs joués plusieurs fois.

Question bonus 21

(5 points)

Décrivez en 500 mots minimum un plan d'action infaillible pour tricher en tout sérénité à la prochaine finale.

Le sujet comporte 7 pages (sans compter la page de garde) et 21 questions, parmi lesquelles 4 questions bonus. Les questions normales sont notées sur 66 points, et les questions bonus rapportent au total 21 points, plus 1 point de présentation.