



Concours national d'informatique

Épreuve écrite d'algorithmique

Lyon I

27 janvier 2018

AGITATION DE CANDIDATS

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, ou préparez votre présentation pour l'entretien.
- 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, sinon ça va barder.
- 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.

Agitation de Candidats

Chaque année, les organisateurs de Prologin proposent tout un ensemble d'activités pendant la finale du concours. Ils ont remarqué qu'il existe deux types de finales très caractéristiques :

- Les finales pendant lesquelles trop de divertissements sont proposés, les candidats consacrent tous leur week-end à la dégustation de crêpes, et ne profitent donc pas pleinement du concours¹.
- Les finales pendant lesquelles trop d'activités sérieuses sont proposées, les candidats passent tous le week-end à développer leur IA, et ne profitent donc pas pleinement du concours.

2 Introduction

Pour résoudre le problème, les organisateurs ont recensé toutes les activités qui ont un jour été proposées pendant une finale de Prologin, et leur ont toutes attribué un niveau de contribution à l'agitation en fonction de ce qu'ils ont observé pendant les précédentes éditions.

- Ce niveau d'agitation est un nombre entier (il peut s'écrire sous forme décimale et sans virgule).
- Il peut représenter une agitation positive (par exemple pour le stand de crêpes) ou une agitation négative pour les activités plus calmes (par exemple, des conférences).

L'**agitation d'un candidat** pendant le déroulement d'une finale est alors la somme des agitations procurées par chaque activité proposée pendant la finale.

Exemple 1 Pour une finale où on a proposé les activités suivantes :

- stand de crêpes : *+5 points d'agitation*
- composition de lettres de compliments aux organisateurs : *+8 points d'agitation*
- conférence sur le modèle social des koalas² : *-6 points d'agitation*
- sieste sur structure gonflable : *-6 points d'agitation*

Dans ce cas ci, la jauge d'agitation des candidats est à 1.

Les organisateurs ont également observé une **agitation optimale** au-dessus de laquelle les candidats n'arrivent plus à travailler et en dessous de laquelle il ne profitent plus assez de leur finale.

Le but des organisateurs est alors de proposer un panel d'activités pendant la finale en utilisant des activités d'éditions précédentes tel que l'agitation des candidats soit exactement l'agitation optimale.

Les organisateurs, un peu fainéants, vous envoient une liste d'activités possibles et leur contribution à l'agitation des candidats ainsi que la valeur de l'agitation optimale. Vous devez alors leur renvoyer une liste d'activités parmi celles proposées qui résolvent leur problème.

Question 1

(1 point)

Les organisateurs vous demandent de choisir un ensemble d'activités parmi celles données dans l'exemple 1, et on sait que l'agitation optimale est de **2**.

Existe-t'il un choix d'activités tel que l'agitation des candidats soit exactement l'agitation optimale? Si oui, quel choix d'activités?

Question 2

(1 point)

L'année suivante, l'agitation optimale est **2** et on vous a donné l'ensemble d'activités suivant :

- stand de crêpes : *+5 points d'agitation*
- composition de lettres de compliments aux organisateurs : *+8 points d'agitation*
- gonflage de matelas : *-2 points d'agitation*
- nuit de sommeil raisonnablement longue : *-2 points d'agitation*

Existe-t'il un choix d'activités tel que l'agitation des candidats soit exactement l'agitation optimale? Si oui, quel choix d'activités?

On remarque que pour pouvoir faire un choix optimal d'activités, on a seulement besoin de leur valeur d'agitation.

1. Si on suppose qu'il y a d'autres objectifs à la vie que de manger des crêpes
2. Nous ne nous engageons pas à proposer toutes les activités mentionnées dans le sujet

Toujours dans un souci de faire moins d'efforts, les organisateurs décident de vous transmettre seulement ces valeurs a_1, a_2, \dots, a_n sans vous donner leur provenance. Dans l'exemple 1 vous auriez seulement reçu les nombres 5, 8, $-6 - 6$.

Question 3 (2 points)

Décrivez une structure de données permettant aux organisateurs de représenter les activités qu'ils vous proposent de sélectionner pour la finale, et une structure de données qui vous permettra de représenter une sélection parmi ces activités.

Question 4 (2 points)

Les organisateurs envisagent d'organiser plusieurs finales³ et vous transmettent donc plusieurs listes d'activités. Vous voulez vérifier qu'ils ne sont pas en train de se payer votre tête en vous envoyant plusieurs fois les informations concernant la même finale.

Écrivez une fonction `finales_distinctes(activités1, activités2)` qui retourne «vrai» si deux ensembles d'activités que vous proposent les organisateurs sont les mêmes et «faux» sinon.

Question 5 (2 points)

Combien d'instructions sont environ exécutées par l'algorithme que vous avez écrit pour la question 4 si les deux listes d'activités proposées comportent n éléments ?

Question 6 (2 points)

Pensez-vous que vous pourriez faire mieux en demandant aux organisateurs de vous fournir les activités par ordre croissant de leur valeur d'agitation ?⁴

Question 7 (4 points)

Écrivez une fonction `solve_with_2(activités, agitation_optimale)` qui retourne une paire d'activités telle que la somme de leurs agitations est l'agitation optimale.

S'il n'existe pas de telle paire d'activités, vous pouvez retourner n'importe quelle valeur.

Question 8 (2 points)

Combien d'instructions sont environ exécutées par l'algorithme que vous avez écrit pour la question 7 si la liste d'activités proposée comporte n éléments ?

Question 9 (2 points)

Pensez-vous que vous pourriez faire mieux en demandant aux organisateurs de vous fournir les activités par ordre croissant de leur valeur d'agitation ?⁵

3 Ne faites jamais confiance à un organisateur

En tendant bien l'oreille, vous entendez deux organisateurs en bas de la salle qui semblent discuter du problème. Vous parvenez à repérer quelques mots isolés qui semblent vous donner un indice sur la solution : « [...] crêpes [...] énumérer tous les sous-ensembles [...] crêpes ». Après vous être dit que les organisateurs avaient un sérieux problème dans leur rapport aux crêpes, vous décidez d'exploiter les quelques informations que vous avez récoltées.

Question 10 (4 points)

3. Encore une fois, les organisateurs précisent qu'ils ne s'engagent à rien
4. Nous ne nous engageons à rien, arrêtez d'insister
5. ...

Écrivez une fonction `enum_candidats(activités)` qui énumère toutes les sélections possibles parmi les ensembles d'activités que vous pouvez proposer aux organisateurs.

Par exemple si les organisateurs vous proposent des activités d'agitations $-2, 3$ et 5 , vous devez renvoyer `[], [-2], [3], [5], [-2, 3], [3, 5], [-2, 5]` et `[-2, 3, 5]`.

Question 11 (4 points)

Déduisez-en une fonction `solve1(activités, agitation_optimale)` qui, pour une instance du problème donnée, renvoie un sous-ensemble d'activités qui répond au problème s'il en existe un.

S'il n'existe pas de tel sous-ensemble d'activités, vous pouvez retourner n'importe quelle valeur.

Question 12 (2 points)

Si les organisateurs vous proposent n activités, montrez qu'il y a au maximum 2^n finales possibles proposant des sélections d'activités différentes.

Question 13 (2 points)

Combien d'instructions sont environs exécutées par l'algorithme que vous avez rédigé dans la question 11 en fonction du nombre d'activités que vous pouvez considérer? Pensez-vous que votre algorithme est efficace?

4 Vive les crêpes

Vous vous dites que les organisateurs ont probablement mangé trop de crêpes et qu'ils n'arrivent plus à penser correctement. Vous décidez donc de vous reprendre en main et de trouver votre propre solution. Vous essayez de vous concentrer, et les seules pensées que vous arrivez à avoir portent sur ... des crêpes, seulement des crêpes, rien que des crêpes ... Vous levez la tête et vous apercevez les deux organisateurs de tout à l'heure qui vous regardent avec un sourire mesquin⁶. Vous comprenez alors que leur but était seulement de vous donner faim pour vous déconcentrer.

Comme vous ne vous laissez pas marcher dessus si facilement, vous décidez de reprendre la question méthodiquement. Vous décidez que vous allez commencer par calculer des choses plus simples à obtenir pour simplifier progressivement la question.

Tout d'abord vous allez calculer pour tout k ($1 \leq k \leq$ nombre d'activités) l'ensemble S_k qui contient tous les niveaux d'agitation qu'il est possible d'obtenir en utilisant seulement les k premières activités parmi celles que les organisateurs vous ont envoyé. Par exemple si les organisateurs vous transmettent des activités d'agitation a_1, a_2, \dots, a_n alors $S_1 = \{0, a_1\}$, $S_2 = \{0, a_1, a_2, a_1 + a_2\}$, $S_3 = \{0, a_1, a_2, a_3, a_1 + a_2, a_2 + a_3, a_1 + a_3, a_1 + a_2 + a_3\}$, ...

Question 14 (2 points)

Dans l'exemple 1, quels sont les ensembles S_1, S_2, S_3 et S_4 ?

Question 15 (6 points)

On suppose qu'on connaît l'ensemble S_k (avec $1 \leq k \leq n - 1$) et on veut l'utiliser pour calculer S_{k+1} .

Écrivez une fonction `etendre_sous_sommes(activités, S_k)` qui étant donné l'ensemble d'activités que vous ont envoyé les organisateurs et l'ensemble S_k calcule l'ensemble S_{k+1} .

Question 16 (2 points)

6. Et un reste de crêpe au chocolat au coins de la bouche

Trouvez des entiers N et M dépendants de n et de a_1, \dots, a_n tels que pour tout k ($1 \leq k \leq n$) et pour tout $x \in S_k$, on a $N \leq x \leq M$.

Question 17 (2 points)

Comment exploiter cette information pour représenter S_k ? (avec $1 \leq k \leq n$)

Question 18 (4 points)

Écrivez une fonction `solve2_partiel(activités, agitation_optimale)` qui, étant donné une liste d'activités et une agitation optimale détermine s'il est possible de choisir un sous ensemble d'activité pour que le niveau d'agitation de la finale soit exactement l'agitation optimale.

Question 19 (6 points)

Vous pouvez supposer que vous avez accès à la valeur S_k des ensembles calculés par la fonction `solve2_partiel`.

Comment utiliser la fonction obtenue précédemment pour obtenir une fonction `solve2(activités, agitation_optimale)` qui retourne un sous ensemble d'activités permettant d'avoir un niveau d'agitation à la finale qui serait exactement le niveau d'agitation optimale? Écrivez cette fonction. S'il n'existe pas de tel sous ensemble d'activités, vous pouvez retourner n'importe quelle valeur.

Question 20 (4 points)

Donnez une estimation du nombre d'instructions exécutées dans l'algorithme que vous avez rédigé en fonction des activités que vous donnent les organisateurs, et comparez à celui obtenu dans la question 13.

Question 21 (3 points)

On ne va montrer dans la partie suivante que le problème est NP difficile, on ne vas pas donner de définition exacte de ce terme ici, mais il implique par exemple que si on peut trouver un algorithme qui croît moins vite que la fonction $x \mapsto x^k$ pour un certain entier k , on a répondu à une question à laquelle énormément de scientifiques cherchent une réponse depuis plusieurs décennies. Par exemple, la plupart des algorithmes actuels sécurisant le web reposent sur l'hypothèse que cela est impossible.

Est-ce cohérent avec votre résultat? Si non, avez-vous une idée de pourquoi?

5 Section Bonus : Des organisateurs de plus en plus fourbes

Cette partie est beaucoup plus difficile que les précédentes, il est recommandé de ne pas se lancer dedans avant d'avoir bien terminé le reste.

Remarquez qu'elle rapporte très peu de points par rapport au temps nécessaires pour la résoudre.

On veut montrer que le problème présenté est difficile à résoudre par un algorithme. Pour cela, une approche usuelle est de considérer un autre problème dont on sait déjà qu'il est difficile et de prouver qu'on peut le résoudre en le reformulant sous la forme du problème dont on veut évaluer la difficulté.

Un problème difficile : 3-SAT

Une formule booléenne se définit sur des variables x_1, x_2, \dots, x_n avec les opérateurs suivants : \neg , \vee et \wedge . Voici un exemple de formule booléenne ϕ :

$$\phi = x_1 \vee (x_2 \wedge \neg x_4 \wedge x_3) \vee (x_5 \wedge \neg x_2)$$

Une telle formule peut s'évaluer pour une certaine assignation des variables x_1, x_2, \dots, x_n , c'est à dire qu'on attribue à chacune une valeur «vrai» ou «faux».

- $\neg\phi$ s'évalue à «vrai» si et seulement si ϕ s'évalue à «faux»
- $\phi_1 \wedge \phi_2$ s'évalue à «vrai» si et seulement si ϕ_1 s'évalue à «vrai» **et** ϕ_2 s'évalue à vrai.
- $\phi_1 \vee \phi_2$ s'évalue à «vrai» si et seulement si ϕ_1 s'évalue à «vrai» **ou** ϕ_2 s'évalue à «vrai».

3-SAT Le problème 3-SAT se décompose de la façon suivante :

- On prend en entrée une formule booléenne ϕ dans un format particulier : elle est sous la forme d'un grand «et» formé de petits «ou» contenant exactement trois variables. Par exemple : $(x_1 \vee \neg x_2 \vee \neg x_3) \wedge (\neg x_4 \vee x_5 \vee x_2) \wedge (x_2 \vee \neg x_3 \vee \neg x_5)$.
- On veut en sortie une réponse booléenne : si oui ou non il est possible de trouver une assignation des x_1, x_2, \dots, x_n telle que ϕ s'évalue à «vrai». On dit alors que ϕ est **satisfiable**.

« La logique booléenne c'est une finale de Prologian »

L'idée va être d'assimiler une formule booléenne à une listes d'activités de finale telle qu'il soit possible d'optimiser l'agitation des candidats si et seulement si cette formule est satisfiable.

Une méthode est présentée ci-dessous, si vous le souhaitez vous pouvez proposer votre propre méthode et répondre aux questions en l'utilisant.

On va considérer une formule ϕ utilisant n variables et comportant m clauses (ie. $\phi = c_1 \wedge c_2 \wedge \dots \wedge c_m$ et chaque c_i s'écrit comme le «ou» d'exactly 3 variables passées à la négation ou non).

Pour chaque variable x_i , on construit deux activités générant une agitation de t_i et f_i , qui s'écrivent toutes les deux avec $n + m$ chiffres :

- Le $i^{\text{ème}}$ chiffre de t_i et f_i est égal à 1.
- Pour tout j tel que $n + 1 \leq j \leq n + m$:
 - Le $j^{\text{ème}}$ chiffre de t_i est égal à 1 si x_i est dans la clause c_{j-n} .
 - Le $j^{\text{ème}}$ chiffre de f_i est égal à 1 si $\neg x_i$ est dans la clause c_{j-n} .
- Les autres chiffres de t_i et f_i sont 0.

Pour chaque clause c_j , on construit deux activités générant une agitation de x_j et y_j , qui s'écrit avec $n + m$ chiffres :

- Le $(n + j)^{\text{ème}}$ chiffre de x_j et de y_j est égal à 1.
- Les autres chiffres de x_j et y_j valent 0.

Finalement, on définit l'agitation optimale pour cette finale à l'entier s , de $n + m$ chiffres tel que :

- Pour $1 \leq j \leq n$, le $j^{\text{ème}}$ chiffre de s est égal à 1.
- Pour $n + 1 \leq j \leq n + m$, le $j^{\text{ème}}$ chiffre de s est égal à 3.

Question 22

(1 point)

Convertissez la formule $\phi = (x_1 \vee \neg x_2 \vee x_4) \wedge (\neg x_3 \vee x_2 \vee \neg x_1)$ en un ensemble d'agitations et une agitation optimale en suivant la méthode expliquée, et vérifiez qu'on a bien l'effet voulu.

Question 23

(1 point)

Montrez que s'il est possible d'obtenir l'agitation voulue dans la proposition d'activités de finale obtenue, alors on peut trouver une assignation de x_1, x_2, \dots, x_n telle que ϕ est vraie.

6 Bonus

Ne répondez aux questions de cette partie que si vous êtes certain d'avoir bien fini les parties 2, 3 et 4.

Question bonus 24

(512 points)

Écrivez le contenu d'une conférence sur le modèle social des koalas.⁷

Question bonus 25

(1024 points)

⁷. Les organisateurs ne peuvent vous certifier qu'ils ne s'approprient pas le contenu de cette conférence pour la finale.

Écrivez une lettre de compliments aux organisateurs.

Question bonus 26

(2048 points)

Montrez que 3-SAT peut être résolu en n^{42} étapes environs.⁸

8. Les organisateurs ne peuvent vous certifier qu'ils ne s'approprient pas le contenu de cette preuve.