



Concours national d'informatique  
Épreuve écrite d'algorithmique  
Louvain-la-Neuve, Rennes, Toulouse

25 Février 2017



# BIENVENUE CHEZ M-CORP.



## 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

**M-Corp** : Chez M-Corp, on n'est peut-être pas gentil, mais on est optimisé, et c'est ça le plus important. Et comme on est optimisé, on engage des stagiaires, comme toi. C'est moins cher puisque c'est gratuit. On aime aussi l'ironie, alors tu vas nous aider à embaucher des gens qui eux seront payés et auront un avenir.

### Partie 1 : Tu fais de l'algo. C'est mieux qu'apporter le café.<sup>1</sup>

#### Question 1 (2 points)

**M-Corp** : Commençons par l'optimisation de ton quotidien. Le bureau de la tour d'M-Corp dans lequel tu travailles est à l'étage  $n$ . Monter ou descendre un étage prend un temps  $t$  par l'escalier et un temps  $t/3$  par l'ascenseur. Mais l'ascenseur ne s'arrête que tous les 10 étages<sup>3</sup>.

Écris un programme donnant la durée du trajet optimal en fonction de  $n$  et  $t$ .

#### Question 2 (1 point)

**M-Corp** : Bien sûr, l'ascenseur ne sera pas toujours libre au moment où tu arrives<sup>4</sup>. Un temps d'attente  $T_a$  est affiché sur l'ascenseur. Tu ne peux monter dans l'ascenseur qu'au premier étage.

Écris un programme donnant ton temps de trajet en fonction de  $T_a$ .

**M-Corp** : Tu es à présent assuré d'être à l'heure tous les matins, bravo. Maintenant, au travail.

#### Question 3 (3 points)

**M-Corp** : M-Corp a reçu une liste de candidats. Elle n'est pas triée et c'est inadmissible. Je suis sûr que as l'œil pour déterminer le talent de chacun, tu sais donc, face à deux candidats, dire lequel est meilleur.

Décris un algorithme qui trie cette liste pour que je sache quels candidats refuser directement.

#### Question 4 (3 points)

Je vais pas coder ça moi-même non plus hein. Implémente cet algorithme, imprime la liste et dépose-la dans mon bureau.

**A-Society** : Salut, je fais partie de la A-Society et j'ai entendu dire que tu travaillais pour M-Corp. Si tu veux vraiment vendre ton âme à ces démons capitalistes, libre à toi. Mais sinon, suis mes conseils. Tu pourras quand même réaliser ton travail, mais d'une façon bien plus amusante !

---

1. Enfin, tu peux<sup>2</sup> aussi apporter le café.

2. Comprendre : dois.

3. Tu imagines bien que les réparations ne sont pas gratuites. A moins que... Tu te débrouilles en bricolage ?

4. Tu n'es pas toujours le centre du monde tu sais ?

### Question 5

(2 points)

**A-Society** : Tu ne seras pas viré pour si peu<sup>5</sup>, alors commençons par nous amuser un peu.

Donne moi un algorithme de mélange aléatoire uniforme d'une liste de candidats. (C'est à dire que chaque mélange a la même probabilité d'être choisi.) Comme ça tous le monde a ses chances!<sup>6</sup>

**A-Society** : Chez A-society on n'est pas comme M-corp, on ne veut pas le meilleur, c'est pas juste. Alors on va prendre la médiane, parce que tout le monde a le droit à sa chance! J'ai volé la liste entière  $L$  des candidats et de leurs compétences.

### Question 6

(2 points)

Étant donné un candidat  $c$  et une liste  $L$ , écris un programme qui renvoie la liste  $L_{<c}$  des candidats moins bons que  $c$  et la liste  $L_{>c}$  des candidats meilleurs que  $c$ .

### Question 7

(3 points)

En utilisant la fonction précédente, implémente un algorithme  $\text{rang}(L, i)$  qui trouve le  $i$ -ème meilleur candidat. Selon les choix de  $c$ , dans le pire des cas, combien de temps va-t-il prendre?

### Question 8

(4 points)

Si, à chaque étape,  $c$  est choisi aléatoirement, combien de temps en moyenne l'algorithme précédent prendra-t-il?

On cherche maintenant à choisir  $c$  à chaque étape pour que le temps d'exécution de l'algorithme soit proportionnel à la taille  $n$  de la liste  $L$  dans tous les cas. Pour ce faire, on divise la liste en blocs de taille au plus 5, on calcule les médianes  $m_1, \dots, m_k$  de chacun des  $k = \lceil \frac{n}{5} \rceil$  blocs, puis on renvoie l'élément de rang  $\lfloor \frac{n+5}{10} \rfloor$  dans la liste  $[m_1, \dots, m_k]$ .

### Question 9

(3 points)

Montre que le candidat  $c$  obtenu est tel que  $L_{<c}$  et  $L_{>c}$  ont pour taille au plus  $\frac{7n}{10}$ .

*Indication : on pourra remarquer que dans chaque bloc de taille 5,  $m_i$  est inférieur à trois éléments, et supérieur à trois éléments.*

### Question 10

(4 points)

Déduis-en un algorithme pour trouver le  $i$ -ème meilleur candidat de  $L$  en un temps proportionnel à  $n$  dans le pire cas.

## Partie 2 : L'argent, ça prend du temps

**M-Corp** : Cette liste de candidats que tu m'as apportée me paraît bien étrange. Mais passons. J'ai besoin de toi sur un autre travail. Tu ne devais, au départ, n'aider qu'au recrutement. Je sais. mais je viens juste de renvoyer l'un de tes collègues stagiaires pour incompétence, et j'ai besoin que son travail soit complété. Je suis sûr que seras ravi de t'en charger.

5. J'en suis *presque* certain.

6. N'hésite pas à répondre à cette question même si tu es un grand fan de la stratégie M-Corp.

### Question 11

(3 points)

**M-Corp** : Il est enfin temps pour M-Corp de lancer sa propre monnaie, qui éclipsera toutes les autres. Il nous faut cependant être capable de calculer automatiquement la décomposition de cette monnaie en centimes. Nous n'avons pas encore décidé quelles pièces seront disponibles. Supposons pour l'instant que nous suivions le système européen (1, 2, 5, 10, 20, 50, 100, 200 centimes).

Donne moi un algorithme simple et rapide `decomp_rapide` qui décompose n'importe quelle somme d'argent en un nombre le plus petit possible de pièces. Implémente-le. Combien d'opérations va-t-il utiliser ?

### Question 12

(2 points)

**A-Society** : C'est bien trop facile de pouvoir décomposer les sommes d'argent automatiquement.

Donne-leur un système de décomposition de la monnaie qui n'est pas correctement décomposé par ton algorithme précédent ! Juste au cas où<sup>7</sup>, donne-moi une somme d'argent pour laquelle la décomposition obtenue n'est pas optimale.

### Question 13

(6 points)

**A-Society** : Je ne voudrais pas te faire virer, alors prépare au cas où un algorithme `decomp_lent`, sûrement lent cette fois, qui fonctionne pour tous les systèmes de décomposition de pièces ! Combien de temps prendra-t-il ?

### Question 14

(1 point)

**M-Corp** : Nous ne sommes bien évidemment pas rattachés à l'UE, et ne suivrons donc pas leur décomposition de l'argent.

Donne moi un algorithme permettant de décider si un système de monnaie fait en sorte que l'algorithme `decomp_rapide` permette de décomposer optimalement toutes les valeurs. On dit alors que le système est *canonique*. (J'ai tout mon temps, ne nous inquiétons pas de la durée d'exécution de ta réponse. Tu peux aussi supposer qu'on ne possèdera jamais plus d'un milliard de milliards d'unités de notre monnaie).

On note maintenant  $c_1, \dots, c_n$  le système de décomposition de M-Corp, trié par ordre croissant de valeurs.

### Question 15

(1 point)

Donne une condition sur  $c_1$  pour que toutes les sommes soient décomposables. On supposera à partir de maintenant que cette condition est vérifiée.

---

7. Même si j'ai entièrement confiance en toi!<sup>8</sup>

8. Puisqu'on se connaît depuis presque une heure !

**Question 16**

(5 points)

Soit  $R(x)$  le nombre de pièces utilisées pour décomposer la somme  $x$  avec l'algorithme `decomp_rapide`, et  $L(x)$  avec l'algorithme `decomp_lent`.

1. Donne une condition sur  $L$  et  $R$  pour que le système soit canonique.
2. Pour chaque valeur  $x$ , donne une relation entre  $R(x)$  et  $R(x - c_i)$  où  $c_i$  est la plus grande pièce inférieure à  $x$ .
3. Montre que pour toute valeur  $x$  et pièce  $c_i$ ,  $L(x) \leq L(x - c_i) + 1$ , et qu'il y a égalité si et seulement si il existe une décomposition optimale de  $x$  contenant  $c_i$ .

**Question 17**

(1 point)

Montre que pour toute valeur  $x$ ,  $R(x) \geq L(x)$ .

**Question 18**

(3 points)

Montre que si un système de monnaie n'a que une ou deux pièces (c'est-à-dire  $n \leq 2$ ), alors il est canonique. Dédus-en que, si  $n \geq 3$  et  $x < c_3$ , alors  $R(x) = L(x)$ .

**Question 19**

(5 points)

Soit  $y \geq c_n + c_{n-1}$ . Montre que si pour toutes les valeurs  $x \leq y$ ,  $R(x) = L(x)$  alors  $R(y + 1) = L(y + 1)$ .

**Question 20**

(3 points)

Dédus des questions précédentes une amélioration (significative!) de l'algorithme de test de canonicité de la question 14. Détermine en fonction de  $c_1, \dots, c_n$  combien d'opérations vont être effectuées.

**Question 21**

(10 points)

**M-Corp** : Ce n'est pas assez rapide, trouve un meilleur algorithme.

1. Soit  $x$  une valeur et  $c$  une pièce telles que  $R(x) > R(x - c) + 1$ . Montre que  $R(x) \neq L(x)$ .
2. Montre que si  $x$  est la plus petite valeur telle que  $R(x) \neq L(x)$  alors il existe une pièce  $c$  telle que  $R(x) > R(x - c) + 1$ .
3. Dédus-en, à l'aide des questions précédentes, un algorithme qui teste si un système est canonique plus rapidement que le précédent.

**Partie 3 : Tu commences presque à être utile.****Question 22**

(1 point)

**M-Corp** : Le système de décomposition que tu nous as fourni me semble étrange mais ton algorithme décompose bien toutes les valeurs, alors ça ira. Arrange-toi pour que tout le monde dans cet étage entende parler de notre nouveau système.

Nous sommes très exactement  $n$  employés, chacun possédant un téléphone. Tu es pour l'instant le seul à avoir l'information. Je veux qu'en au plus  $n$  étapes de la durée d'un appel téléphonique, tout le monde soit au courant à coup sûr. Comment vas-tu faire ? (Tes collègues sont très occupés, ils n'ont pas le temps d'appeler d'autres personnes!)

### Question 23

(5 points)

**M-Corp** : Ça irait quand même plus vite si tes fainéants de collègues pouvaient transmettre l'information à leur tour.

Comment procèderais-tu pour propager l'information ? En combien d'étapes de la durée d'un appel, l'information est propagée ?

### Question 24

(3 points)

**A-Society** : J'ai une meilleure idée pour annoncer ton grand œuvre à tous tes copains ! Moi je pense que tu peux faire quelque chose qui en moyenne marchera aussi vite, mais qui aura un risque non nul de marcher beaucoup moins vite ! Par contre, tu n'auras pas besoin de passer autant d'informations à tes collègues à chaque appel !

Propose un algorithme aléatoire pour diffuser le message. En combien d'étapes au mieux/en moyenne/au pire le message a atteint tous les employés ?

### Question 25

(1 point)

**A-Society** : On pourrait croire que c'est risqué de faire comme cela parce que parfois c'est long. Mais en réalité c'est l'autre technique qui est risquée !

Tu vois pourquoi ?

*Indication : C'est fou qu'un groupe d'humains puisse avoir les mêmes défauts qu'un groupe d'ordinateurs !*

Tu ne peux attaquer la partie suivante que si tu as fini d'imprimer nos milliards de billets, employé trois stagiaires et vérifié le bon fonctionnement de tous les téléphones du bâtiment.

## Partie 4 : Bonus

### Question bonus 26

(3 points)

À quel condition le système de monnaie  $c_1, c_2, c_3$  permet de décomposer toutes les sommes et est canonique ?

### Question bonus 27

(7 points)

**A-Society** : Je vais penser à un numéro de candidat entre 1 et  $n$ , et tu vas le deviner. Et tu feras embaucher le numéro que tu penseras avoir deviné. C'est, cette fois, quelqu'un qui le mérite alors ne te trompe pas ! Tu peux me poser des questions et je ne répondrai que par oui ou non !

1. Combien de questions te faudra-t-il et comment vas-tu t'y prendre ?
2. Et si j'ai le droit de te mentir au plus  $p$  fois, quelle stratégie vas-tu utiliser ? De combien de questions auras-tu besoin ?

**Question bonus 28**

(5 ou 1d10 point(s))

Plutôt M-Corp ou A-society ? **M-Corp** : Justifie en très exactement 273 mots. **A-Society** : Ou en 10d20 mots, c'est ton choix !

**Question bonus 29**

(1d20 point(s) ?)

Si on note la copie en lisant simplement le lancer de 1d20, peut-on considérer cette question comme une question bonus ?

Le sujet comporte 6 pages (sans compter la page de garde) et 29 questions, parmi lesquelles 4 questions bonus. Les questions normales sont notées sur 77 points, et les questions bonus rapportent au total quelques points, plus 1 point de présentation.