



Concours national d'informatique  
Épreuve écrite d'algorithmique  
Paris II

Dimanche 14 février 2016



# JOURNALISTES EN HERBE



T'as vu, il paraît qu'Alice a fait un bisou à Bob! Ça c'est du vrai journalisme d'investigation!

## 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

Lycéen, vous faites partie d'un petit groupe d'étudiants qui aimerait lancer un journal du lycée<sup>1</sup>. Après quelques semaines à convaincre vos amis de tous bords<sup>2</sup> que votre projet est cool, votre équipe est assez grande pour se lancer dans cette aventure. Après avoir rédigé dur, votre premier numéro est lancé, et c'est un franc succès!<sup>3</sup> Seulement, certains de vos amis vous disent que certains de leurs amis ont entendu dire que quelques lecteurs cherchaient désespérément à contacter l'équipe de rédaction pour soumettre des articles, vous féliciter, ou vous lancer des tomates trop mûres<sup>4</sup>.

Bref, après un tour de table où certains proposent de laisser une adresse email, ou le numéro de téléphone de votre local, vous décidez à la majorité que c'est quand même plus chouette de rencontrer les personnes. Mais ça n'est pas si simple...

### Partie I

Vous devez d'abord décider du jour durant lequel vous allez effectuer une permanence pour vos lecteurs. Vous demandez ainsi à chaque membre du comité de rédaction d'indiquer quels jours il sera disponible.

#### Question 1

(2 points)

On vous donne un entier  $n$ , le nombre de personnes, suivi pour chacune de sept booléens (0 ou 1), chacun représentant un jour de la semaine (commençant par lundi). 0 signifie que la personne ne pourra pas venir ce jour-là, tandis qu'un 1 indique sa disponibilité. Écrivez une fonction qui détermine quel jour faire votre permanence, c'est-à-dire le jour où le plus grand nombre de personnes sont disponibles.

Suite à votre sondage, vous vous rendez compte que le jour optimal est de loin le mercredi. Mais ce n'est pas fini, maintenant se pose la questions des horaires. Chacun vous donne ses disponibilités : une liste d'intervalles  $[a_1; b_1[, \dots, [a_p, b_p[$  signifiant que ce membre est disponible entre les temps  $a_i$  (inclus) et le temps  $b_i$  (exclu). On vous garantit que les intervalles d'un même membre sont disjoints.

On vous donne ainsi un tableau de listes (ou un tableau de tableaux/vecteurs, si vous préférez) `disponibilites` contenant ces intervalles.

#### Question 2

(2 points)

On vous donne un temps  $t$ . Écrivez une fonction qui renvoie vrai si une personne est disponible au temps  $t$ .

Vous souhaitez à présent faire un planning d'ouverture du local du journal.

- 
1. Si vous n'avez pas vu *La Colline aux Coquelicots*, vous savez ce qu'il vous reste à faire ;-)
  2. Oui, même les scientifiques!
  3. D'ailleurs bravo, vous avez toute notre admiration!
  4. Bah oui, gâcher c'est mal.<sup>5</sup>
  5. Et les accents circonflexes, c'est mignon.

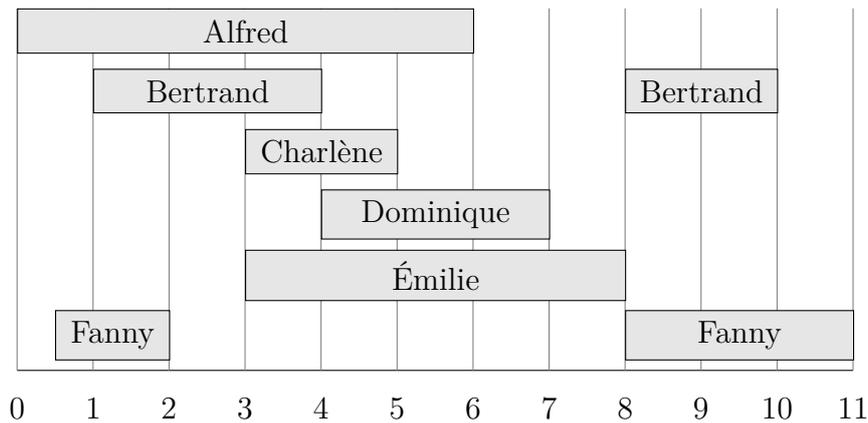


FIGURE 1 – Oui, votre permanence commence à minuit.

### Question 3

(5 points)

- Supposez que les temps sont exprimés en secondes (entre 0 et 86 400 donc). Écrivez une fonction qui détermine la liste des périodes où personne n'est disponible.
- Comment faire si les temps ne sont pas bornés<sup>6</sup> ? On cherchera<sup>7</sup> une réponse qui ne dépend pas du<sup>8</sup> temps maximum donné<sup>9</sup>.

En fait, vous vouliez l'inverse pour votre planning : la liste des périodes où quelqu'un sera présent. M'enfin vous avez réussi à vous débrouiller, et vous avez imprimé un joli emploi du temps, où les cases noires sont celles où personne ne sera là, et celles blanches sont les périodes d'accueil du public. Après quelques semaines, le résultat est correct, mais vos lecteurs ont la fâcheuse tendance à venir en même temps aux périodes où il y a peu de membres au local. La dernière fois le pauvre Joseph Marchand a dû faire face à 17 personnes en étant tout seul ! Heureusement... vous avez un stabilo ! Vous aimeriez ainsi colorier en jaune fluo la case où le plus grand nombre de personnes sont disponibles.

### Question 4

(5 points)

- Supposez que les temps sont exprimés en secondes (entre 0 et 86 400 donc). Écrivez une fonction qui détermine la période où le plus grand nombre de personnes sont disponibles, ainsi que leur nombre.
- Comment faire si les temps ne sont pas bornés<sup>10</sup> ? On cherchera une réponse qui ne dépend pas du temps maximum donné<sup>11</sup>.

6. Enfin, on supposera qu'ils tiennent dans un entier 64 bits fourni dans votre langage.

7. Ce qu'il y a de drôle dans le comique de répétition, c'est la répétition...

8. Sisi, la répétition vous dis-je !

9. Imaginez qu'on vous fournisse des temps en femtosecondes !

10. enfin, on supposera qu'ils tiennent dans un entier 64 bits

11. Imaginez qu'on vous fournisse des temps en yoctosecondes !

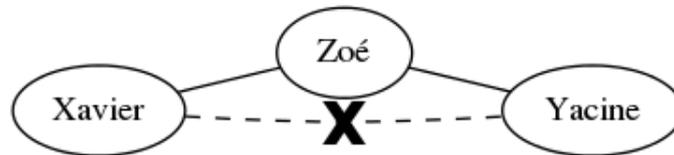
## Partie II

Après les épreuves du bac, vous passez tous d'excellentes vacances d'été. Mais à la rentrée, certains membres (ceux qui étaient en terminale) partent du lycée, alors vous recrutez à nouveau. Vous faites une première réunion, et il s'avère que cette année il y a énormément de volontaires alors chacun ne fera **qu'une seule** permanence le mercredi<sup>12</sup>, c'est-à-dire restera au local pendant **un unique intervalle de temps**.

Grâce à vos compétences en algorithmique, vous réalisez l'affectation des membres aux permanences. Le lendemain matin en revanche c'est la panique! Le secrétaire improvisé de la réunion n'a pas été fichu de prendre le compte-rendu correctement, et le planning des permanences est inutilisable. Heureusement les membres ont une bonne mémoire, et sont capables de vous dire sans l'ombre d'un doute avec qui ils étaient censés coexister pendant cette journée. Vous essayez donc de reconstituer les permanences partagées. Par exemple :

- Zoé et Yacine se souviennent qu'ils partageaient leurs permanences.
- Zoé et Xavier se souviennent qu'ils partageaient leurs permanences.
- Ils sont aussi sûrs qu'ils ne partageaient leur permanence avec personne d'autre.

Ceci ne signifie pas qu'ils étaient au local pendant exactement la même période, juste qu'à un instant ils se sont trouvés au local ensemble.



### Question 5

(2 points)

Donnez un exemple de planning (vous pouvez représenter chaque intervalle par un segment) correspondant aux souvenirs de Zoé, Yacine et Xavier. Donnez un autre exemple de planning cohérent, de sorte à montrer que ces souvenirs ne sont pas suffisants pour déterminer qui était le premier à venir effectuer sa permanence.

### Question 6

(2 points)

Vous souhaitez stocker l'information des permanences « partagées ». Quelle structure de données utiliser pour modéliser cette information ? Attention, vous ne disposez pas des horaires, seulement des réponses oui/non aux questions de type « X et Y ont-ils été tous deux présents à un même moment ? ».

Bon... en fait vous retrouvez sous votre canapé une version 2 du compte-rendu dont vous aviez oublié l'existence, avec un planning de permanences complet. Plein de mauvaise foi, vous vous demandez s'il correspond aux souvenirs des membres que vous avez glanés<sup>13</sup>.

12. Le jour n'a pas changé, malgré la présence de quelques maîtres Jedi à la réunion.

13. Vous savez, à l'époque où vous aviez perdu le deuxième compte-rendu.

### Question 7

(3 points)

Écrivez une fonction qui prend en paramètre deux tableaux d'entiers **debut**s et **fin**s correspondant aux temps de début et de fin de permanence de chaque membre, et votre structure de données de la question précédente déjà remplie. Votre fonction doit renvoyer un booléen : vrai si les données concordent, ou faux si comme vous le pensiez les données de la version 2 du compte-rendu ne correspondent pas aux souvenirs récoltés.

Finalement, vous décidez que le planning des permanences que vous avez retrouvé est moins fiable que les souvenirs des gens ; le secrétaire a dû le falsifier pour faire oublier son incompetence. Vous renoncez donc à déduire quoi que ce soit d'une source d'information autre que les permanences partagées.

### Question 8

(8 points)

Écrivez des fonctions qui, à partir uniquement de la donnée des permanences partagées, renvoient :

- (a) un booléen indiquant si le local est occupé en continu pendant la journée à partir de son ouverture jusqu'à sa fermeture définitive, sans période où personne n'est présent ;
- (b) (bonus) le nombre de périodes où personne n'est disponible, qui vaudra 0 lorsque la fonction précédente renverra vrai, et correspondra au nombre d'intervalles dans la liste que vous fournirait la fonction de la question 3 si vous lui fournissiez en entrée le vrai planning des permanences.

On pourra supposer ici qu'il ne se produit jamais d'événement où tous les rédacteurs présents s'en vont tandis que d'autres arrivent exactement au même moment.

Afin d'augmenter la convivialité du local durant les permanences, vous avez durement négocié l'achat d'une théière<sup>14</sup> à l'assemblée générale budgétaire du lycée. Cependant, l'instauration d'une minute thé et biscuits se heurte à la difficulté suivante : il est rare que tous les membres de la rédaction soient présents en même temps ! On appellera *sous-équipe* un ensemble de rédacteurs qui sont présents simultanément à un moment, et qui peuvent donc partager le thé sans qu'il refroidisse.

### Question 9

(7 points)

- (a) Considérons le dernier rédacteur à arriver au local. Montrez qu'il vérifie la propriété suivante : lui et tous ceux avec qui il partage une permanence forment une sous-équipe. Montrez qu'il en est de même pour le premier rédacteur à s'en aller.
- (b) Déduisez-en que si un groupe de personnes partage des permanences deux à deux, alors il constitue une sous-équipe.
- (c) Comment savoir si la rédaction se retrouvera au complet à un moment de la journée et pourra partager du thé et des biscuits ? On vous demande toujours de répondre à partir uniquement des partages de permanence.

---

14. Le café étant bien entendu un psychotrope qu'il est impensable de laisser à disposition de mineurs, on se rabat sur une approximation.

### Question 10

(5 points)

Écrivez une fonction qui trouve, à partir des permanences partagées, un membre vérifiant la propriété de la question précédente (pour rappel : lui et tous ceux avec qui il partage une permanence forment une sous-équipe).

### Question 11

(10 points)

En vous appuyant sur la question précédente, écrivez une fonction qui calcule le pic de fréquentation à partir des permanences partagées. Celle-ci devra renvoyer le même nombre de personnes que si le planning horaire des permanences avait été fourni à la fonction de la question 4.

## Partie bonus

### Question 12

(6 points)

Écrivez une fonction qui construit, si c'est possible, *deux* sous-équipes de sorte que chaque rédacteur se retrouve dans l'une des sous-équipes, et signale si c'est impossible.

### Question bonus 13

(20 points)

Plutôt que de considérer des intersections entre intervalles dans une droite des temps, généralisons aux intersections entre un ensemble de parties connexes d'un *arbre*. Traitez les questions 9 à 11 dans ce cas généralisé, puis montrez qu'il s'agit en fait d'une caractérisation des graphes où « tout ceci marche bien », dans un sens rigoureux que vous préciserez.

### Question bonus 14

(1 point)

Créez du contenu que vous aimeriez voir passer dans le journal! Dessins, poèmes, chagrins d'amour et jeux débiles acceptés, les articles restant prioritaires.<sup>15</sup>

# FIN

Le sujet comporte 6 pages (sans compter la page de garde) et 14 questions, parmi lesquelles 2 questions bonus. Les questions normales sont notées sur 57 points, et les questions bonus rapportent au total 21 points, plus 1 point de présentation.

---

15. Les sujets écrits de Prologin sont la dépense de papier du Concours National d'Informatique. Ils paraissent tous les week-ends pendant 7 semaines consécutives au nombre galère à imprimer de quelques dizaines d'exemplaires. Les notes de pied de page à l'humour douteux n'engagent que ceux qui les écrivent et resteront accessibles à la vue de tous jusqu'à la fin de l'ère informatique, puisque que ces sujets sont sur le Web : <https://prologin.org/archives>. En fait, Prologin est un magazine hebdomadaire (directeur de publication<sup>16</sup> : Joseph Marchand), il faudrait lui attribuer un numéro ISBN et le déposer à la BnF.

16. cf. Loi du 29 juillet 1881 sur la liberté de la presse.