



Concours national d'informatique

Épreuve écrite d'algorithmique

ATELIER CUISINE

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 (3 heures 30 minutes).

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.

2 À propos du sujet

Dans ce sujet écrit, Joseph Marchand vous propose de le suivre dans différents ateliers cuisine.

- Dans la Section 3, Joseph Marchand fera goûter à ses proches un gâteau multi-couches. Elle contient un total de 15 points.
- Dans la Section 4, Joseph Marchand a l'ambition de réaliser une pièce montée composée d'une multitude de choux à la crème. Elle contient un total de 17 points.
- Dans la Section 5, Joseph Marchand voudrait prototyper ses desserts afin d'avoir le meilleur goût possible. Elle contient un total de 12 points.
- Dans la Section 6, il est possible de remporter des points bonus sur divers questions. Elle contient un total de -4235 points.

Les exercices suivent la même histoire mais sont indépendants. Il est conseillé de ne pas bloquer sur un exercice et de passer au suivant si nécessaire.

3 Gâteau multi-couches

Joseph Marchand a l'ambition de réaliser le meilleur gâteau composé de plusieurs couches. Étant un très bon cuisinier, il a déjà préparé et assemblé les différentes couches de son gâteau. Pour réaliser le meilleur gâteau, il voudrait faire goûter à ses proches sa création pour l'améliorer dans le futur.

3.1 Gâteau unique

Joseph Marchand s'apprête à faire goûter son gâteau. Le gâteau est composé de N couches plus ou moins sucrées. La couche i a `sucreri` grammes de sucre¹, représentée par un nombre entier.

Question 1 (1 point)

Joseph Marchand fait goûter son gâteau à sa sœur mais elle n'a pas très faim (c'est déjà le 6^e gâteau cette semaine), elle voudrait goûter seulement certaines couches. Proposer un algorithme permettant de calculer la quantité totale de sucre entre les deux couches L et R ($L \leq R$) correspondant au calcul `sucrerL + sucrerL+1 + ... + sucrerR`. Les couches L et R sont comprises dans la somme.

Question 2 (1 point)

Maintenant, nous définissons une liste de $N+1$ entiers nommée `cumul` tel que `cumul1 = 0`, `cumuli = sucrer1 + sucrer2 + ... + sucreri-1`. Montrer comment calculer `cumuli+1` en fonction de `cumuli` et de `sucreri`.

Question 3 (1 point)

Joseph Marchand trouve qu'il est possible de calculer simplement la quantité de sucre entre deux couches en utilisant `cumul`. Montrer sur le schéma correspondant à la liste `sucrer`, quelles couches sont incluses dans `cumul5`. Vous pouvez ajouter une phrase d'explication si nécessaire.

3
2
9
8
3
1

Voici ci-dessus une représentation du gâteau multi-couches. La couche ayant 1g de sucre est la **première**, celle ayant 3g de sucre la **dernière**.

Question 4 (2 points)

Proposer une formule utilisant `cumul` pour aider Joseph Marchand à trouver la somme des valeurs `sucrerL + sucrerL+1 + ... + sucrerR`. Elle correspond à la quantité de sucre entre la couche L et R (deux entiers, $L \leq R$).

Question 5 (2 points)

Joseph Marchand s'intéresse ici au goût ainsi qu'à la texture et l'acidité du gâteau. Voici comment calculer chaque valeur :

- Goût : `goutL × goutL+1 × ... × goutR`.
- Texture : `textureL ⊕ textureL+1 ⊕ ... ⊕ textureR` (\oplus étant l'opération **ou exclusif** (XOR) bit-par-bit).
- Acidité : `pgcd(acideL, acideL+1, ..., acideR)` (pgcd étant le plus grand diviseur commun).

1. L'abus de sucre est dangereux pour la santé, à consommer avec modération (non vous n'avez pas d'ami s'appelant "modération")

Pour chaque valeur, indiquer s'il est possible d'utiliser la même formule que précédemment en changeant seulement les opérateurs par d'autres opérateurs ou fonctions. De même, la liste `cumul` pourra être initialisée d'une autre manière mais il n'est pas nécessaire d'expliquer comment.

Chaque liste de valeur contient des entiers compris entre 0 et 10^9 . Si ce n'est pas possible, préciser pourquoi.

Question 6

(1 point)

Joseph Marchand a une liste contenant les intervalles de couches que veulent chaque proche. Compléter l'algorithme pour afficher la quantité de sucre qu'obtiendra chaque proche, soit la somme $\text{sucre}_L + \text{sucre}_{L+1} + \dots + \text{sucre}_R$. La complexité temporelle attendue est $O(Q)$.

Algorithme 1 : Affichage des quantités de sucre pour chaque proche

Entrées : Une liste d'entiers `cumul`, une liste `proche` de Q paires d'entiers (L, R)

pour chaque (L, R) *dans* `proche` **faire**

$\text{quantite} \leftarrow 0$;

 Afficher *quantite*;

fin

3.2 Série de gâteaux

Toute la famille Marchand s'est mise à cuisiner aujourd'hui ! Un gâteau rectangulaire gigantesque a fait son apparition dans la maison des Marchand. Maintenant, chaque personne va pouvoir prendre une part rectangulaire du gâteau.

- Le gâteau forme un rectangle de largeur M et de longueur N , divisé en $M \times N$ carrés de taille 1×1 .
- Un proche peut goûter une part de gâteau, formant un rectangle de coin inférieur gauche (x_1, y_1) et de coin supérieur droit (x_2, y_2) .
- À présent, **sucre** est un tableau bidimensionnel tel que **sucre** $_{x,y}$ ($1 \leq x \leq M$ et $1 \leq y \leq N$ entiers) correspond à la quantité de sucre dans le carré (x, y) .
- De la même manière, **cumul** est maintenant bidimensionnel et correspond à une matrice de $N + 1$ lignes et $M + 1$ colonnes.
- **cumul** $_{x,y}$ correspond à la somme du sucre contenu dans la part de coin inférieur gauche $(1, 1)$ et de coin supérieur droit $(x - 1, y - 1)$. Nous avons donc **cumul** $_{1,j} = \text{cumul}_{i,1} = 0$ peu importe les valeurs de i et j .

8	5	4	0	8	4	8	1	2	8
7	3	5	5	6	2	4	3	2	2
6	2	5	6	9	7	2	8	4	2
5	1	3	7	0	0	0	8	2	6
4	8	9	0	6	5	6	4	6	0
3	6	0	9	6	5	6	4	3	5
2	4	3	4	7	7	3	0	2	9
1	8	5	4	7	1	2	6	4	2
	1	2	3	4	5	6	7	8	9

Le gâteau fait $N = 8$ de longueur et $M = 9$ de largeur. Le tableau représente les valeurs de **sucre**. La valeur de **cumul** $_{9,6}$ est alors la somme des valeurs de **sucre** étant dans la zone rouge (en gras) dans la grille ci-dessus.

Question 7

(2 points)

8	5	4	0	8	4	8	1	2	8
7	3	5	5	6	2	4	3	2	2
6	2	5	6	9	7	2	8	4	2
5	1	3	7	0	0	0	8	2	6
4	8	9	0	6	5	6	4	6	0
3	6	0	9	6	5	6	4	3	5
2	4	3	4	7	7	3	0	2	9
1	8	5	4	7	1	2	6	4	2
	1	2	3	4	5	6	7	8	9

Le but est de calculer la quantité de sucre dans le rectangle de valeurs en gras, grâce au tableau **cumul**.

En particulier, quatre valeurs du tableau **cumul** nous intéressent.

- **cumul** $_{5,4}$, en orange, en bas à gauche.
- **cumul** $_{9,4}$, en vert, en bas à droite.
- **cumul** $_{9,6}$, en bleu, en haut à droite.
- **cumul** $_{5,6}$, en rouge, en haut à gauche.

Les rectangles indiqués dans le schéma illustrent les valeurs incluses dans ces quatre valeur.

Proposer une formule pour calculer la quantité de sucre dans le rectangle de valeur en gras, en utilisant les valeurs **cumul** $_{5,4}$, **cumul** $_{9,4}$, **cumul** $_{9,6}$ et **cumul** $_{5,6}$.

Question 8

(2 points)

Montrer comment calculer $\text{cumul}_{x+1,y+1}$ en fonction de $\text{cumul}_{x,y}$, $\text{cumul}_{x+1,y}$, $\text{cumul}_{x,y+1}$ et $\text{sucre}_{x,y}$.

Question 9

(3 points)

Pour que la fête soit plus fun, la famille Marchand a glissé plusieurs fèves dans leur gâteau. Il y a F fèves représentées par une liste de paires (x, y) .

Les K amis de la famille se demandent si leur part contient ou non une fève. Les parts sont représentées par une liste de K couples (x_1, y_1, x_2, y_2) . (x_1, y_1) est la position du coin inférieur gauche et (x_2, y_2) la position du coin supérieur droit du rectangle représentant la part de gâteau.

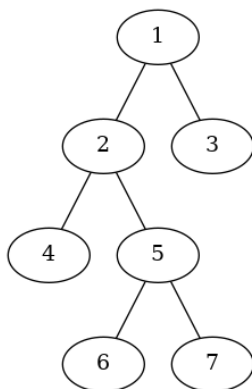
Proposer une idée² d'algorithme en réutilisant les méthodes de cet exercice permettant de déterminer combien d'amis aura au moins une fève. La complexité temporelle attendue est $O(F + K + N \times M)$.

2. Indiquer les étapes principales sans détailler.

4 Pièce montée

Joseph Marchand en a marre des gâteaux à plusieurs couches. Il veut réaliser à présent une pièce montée composée de choux à la crème. Nous pouvons décrire la pièce montée comme un chou se reposant sur d'autres choux formant des sous-pièces montées. Il y a N choux composant le dessert.

Voici un exemple de pièce montée composée de $N = 7$ choux, le sommet est le chou 1 :



4.1 Choux numérotés

Joseph Marchand s'amuse à numéroter ses choux d'un numéro unique allant de 1 à N . En attendant que ses invités arrivent, il code un peu pour se détendre (comme tout le monde le fait).

$piece_{chou}$ donne la liste des choux sur lequel repose le chou numéro $chou$, par ordre croissant. Par exemple, $piece_2$ donne la liste $[4, 5]$.

Voici son algorithme :

Algorithme 2 : Fonction Affichage

Entrées : Une liste formée de listes d'entiers $piece$, et le numéro d'un chou $chou$.

Afficher "Chou : " + $chou$;

pour tous suivant $piece_{chou}$ **faire**

 | Affichage($piece$, $suivant$) ;

fin

Question 10

(1 point)

Que va être affiché si nous appelons la fonction **Affichage** sur le chou numéroté 1 ?

Question 11

(1 point)

L'ordre ne convenant pas à Joseph Marchand, il a légèrement modifié la fonction **Affichage**. Après avoir appelé la fonction sur le chou 1, il obtient le résultat suivant :

```
Chou : 4
Chou : 6
Chou : 7
Chou : 5
Chou : 2
Chou : 3
Chou : 1
```

Qu'a-t-il modifié à l'algorithme pour obtenir ce résultat ?

Question 12

(1 point)

Joseph Marchand n'a plus beaucoup de temps avant que ses invités arrivent. Indiquer à Joseph Marchand comment modifier l'algorithme pour ajouter à une liste nommée **ordre** les choux dans l'ordre dans lequel ils sont affichés avec l'algorithme donné avant la Question 10. **ordre** est initialement une liste vide.

Question 13

(1 point)

Aider Joseph Marchand en créant un algorithme non récursif affichant les choux tel qu'un chou A posé sur un chou B soit toujours affiché avant le chou B . La liste `ordre` doit être utilisée.

4.2 Remplissage des choux

Afin que la pièce montée soit équilibrée, Joseph Marchand doit remplir de manière différente les choux. Les choux au sommet de son dessert sont souvent moins remplis que ceux à la base de celui-ci.

Pour savoir combien de grammes de crème pâtissière sont composés les choux, Joseph Marchand a fait ce programme :

Algorithme 3 : Fonction Remplissage

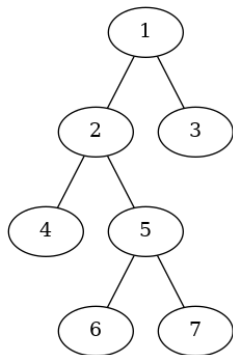
Entrées : Une liste formée de listes d'entiers *piece*, le numéro d'un chou *chou*, son remplissage *r*
Afficher *chou* + " → " + *r*;
pour tous suivant *piece*_{chou} **faire**
| Remplissage(*piece*, *suivant*, *r* + 1);
fin

Initialement, la fonction **Remplissage** est appelée sur le chou au sommet du dessert avec un remplissage $r = 1$.

Question 14

(1 point)

Qu'affiche ce code sur cette pièce montée³ ? Si un chou est posé sur plusieurs choux, les choux seront visités de gauche à droite (2 sera visité avant 3 sur le schéma).



Question 15

(2 points)

Joseph Marchand attend toujours ses invités et est à court d'idées pour coder d'autres algorithmes. Il veut alors faire le même algorithme mais de manière itérative. La variable **ordre** est une liste d'entiers, itérer cette liste donne un ordre tel qu'un chou *A* posé sur un chou *B* soit affiché toujours en premier. Finir son algorithme.

Algorithme 4 : Fonction Remplissage

Entrées : Une liste formée de listes d'entiers *piece*, et la liste *ordre*.
remplissage ← Liste contenant *N* fois l'entier 1;
pour tous chou *dans* *ordre* **faire**
| $r \leftarrow$ remplissage_{chou};
fin
pour tous chou *dans* *ordre* **faire**
| Afficher chou + " → " + remplissage_{chou};
fin

3. Merci de ne pas critiquer la beauté de ce dessert, Joseph Marchand a travaillé dur pour réaliser celui-ci.

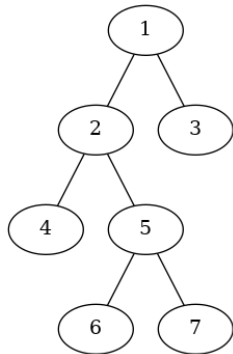
4.3 Décoration

Joseph Marchand veut maintenant décorer son dessert. Pour cela, il dépose un coulis de chocolat d'une manière bien précise : il choisit un chou, commence à déposer son coulis dessus puis remonte d'un certain nombre de choux jusqu'à ne plus avoir de coulis dans sa poche à douille.

Question 16

(1 point)

Le coulis est déposé initialement sur le chou numéro **chou**. S'il existe au moins K choux qui reposent sur le chou numéro **chou**, combien de fois Joseph Marchand doit faire remonter sa douille d'un chou afin que le coulis parcoure K choux au total? Donner la réponse en fonction de K .



Par exemple, dans l'image si dessus, si $\text{chou} = 7$ et $K = 3$ alors il doit remonter de 2 choux.

Question 17

(2 points)

Joseph Marchand voudrait gagner du temps en pouvant remonter sa douille de deux choux à la fois. Sachant qu'une action consiste à remonter 1 ou 2 choux, combien doit-il réaliser d'actions au minimum pour que le coulis passe sur K choux?

Par exemple, si $K = 4$, il peut remonter de deux choux puis d'un seul chou pour un total de 2 actions.

Il est possible d'utiliser la fonction `arrondi` permettant d'arrondir la valeur à l'entier le plus proche ou bien `entiere` donnant la partie entière d'une valeur. `arrondi(1.5) = 2`, `entiere(1.5) = 1`.

Question 18

(3 points)

Joseph Marchand réalise une très grande pièce montée. Il voudrait réaliser le moins d'actions possibles sachant qu'il peut en une action remonter de n'importe quel nombre de choux étant une puissance de 2. (1, 2, 4, 8, ...)

Donner une méthode pour réaliser le moins d'actions possibles en décorant K choux avec le chou de départ numéro **chou**.

Question 19

(1 point)

Si nous réalisons un tel algorithme, quel serait l'ordre de grandeur du nombre d'étapes uniquement en fonction de K ? Donner une réponse exacte ou en utilisant une notation de Landau, l'opérateur "Grand O" par exemple.

Question 20

(3 points)

Pour modéliser ces actions, le tableau bi-dimensionnel `audessus` est défini tel que `audessusi,chou` donne le (2^{i-1}) -ème chou au dessus du chou numéroté **chou**. La valeur est 1 (correspondant au chou au sommet du dessert) s'il y a moins de 2^{i-1} choux posés sur **chou**. Nous remarquons également que `audessus1,chou` donne le chou au dessus du chou **chou**. Ce tableau possède D lignes et N colonnes.

Proposer un algorithme permettant d'initialiser `audessus` sachant que `audessus1,chou` est déjà initialisé pour chaque **chou** possible.

5 Prototypage

Joseph Marchand essaye différentes recettes de son gâteau préféré : le 42 quarts. Pour avoir un goût encore meilleur, il n'utilise pas des quarts mais modifie sa recette en variant la quantité de ses ingrédients.

5.1 Quantité de chocolat

Afin de mettre la quantité parfaite de chocolat, il veut minimiser le nombre de gâteaux à faire et à goûter. Pour cela, il utilise cet algorithme prenant en paramètre la liste de toutes les quantités de chocolat (N entiers) triées par ordre croissant et renvoyant la quantité parfaite :

Algorithme 5 : Fonction Chocolat

```
Entrées : Liste d'entiers choco
g ← 1;
d ← N;
tant que g ≤ d faire
  | mil ← entiere( $\frac{g+d}{2}$ );
  | si ManqueChoco(chocomid) alors
  | | g ← mil + 1;
  | fin
  | sinon
  | | d ← mil;
  | fin
fin
retourner chocog
```

entiere renvoie la partie entière du nombre donné en paramètre. Par exemple, `entiere(2.5) = 2`.

Quand `ManqueChoco` est appelée, Joseph Marchand fait goûter à sa sœur Elle dit alors s'il manque de chocolat mais ne dit pas s'il y en a trop car elle est très gourmande. Joseph Marchand sait que la quantité de chocolat est parfaite **s'il ne manque pas de chocolat et que la quantité est la plus petite possible**.

Question 21

(1 point)

En fonction de N , quelle est la complexité temporelle de cet algorithme ?

Question 22

(1 point)

Modifier l'algorithme pour qu'il fonctionne avec une liste de quantités de chocolat triée en ordre décroissant.

5.2 Quantité de lait

Joseph Marchand aime utiliser plusieurs types de laits végétaux pour avoir un goût plus varié dans son gâteau. La recette de Joseph Marchand est assez complexe, il l'a noté sous forme de liste. Il y a N étapes, à chaque étape il verse une quantité de lait qu'il a notée puis il remue.

La recette étant assez compliquée, il ne veut pas changer de type de lait à moins que cela soit nécessaire. Le goût qu'apporte un lait végétal est **démultiplié** quand on l'ajoute plusieurs fois - c'est là le secret de sa recette. Par exemple, s'il ajoute 2cl puis 5cl et enfin 3cl du même type de lait, alors le goût a une intensité de $2 \times 5 \times 3 = 30$.

Question 23

(2 points)

Joseph Marchand possède 3 types de lait dans son placard - amande, avoine et noisette. Pour que l'on sente encore le chocolat, Joseph Marchand ne doit pas avoir une intensité dépassant K .

Réaliser une fonction prenant en argument la recette et l'intensité permettant de savoir s'il est possible de découper en 3 ou moins sa liste (sans changer son ordre) tel que cette condition soit respectée.

Par exemple, si $K = 6$ la liste 1 3 2 4 1 5 peut être découpée comme cela : 1 3 2 - 4 - 1 5 mais la liste 1 3 2 4 2 1 5 ne peut pas être découpée sans dépasser l'intensité maximale.

Question 24

(1 point)

Pour que le chocolat se ressente au maximum, Joseph Marchand voudrait minimiser l'intensité de goût des laits végétaux.

Proposer un algorithme simple utilisant la fonction précédente permettant de trouver la valeur minimale K tel qu'il soit possible d'utiliser 3 laits différents au maximum (la liste représentant la recette doit pouvoir être coupée en 3 ou moins). Il n'est pas ici question d'optimisation.

Question 25

(1 point)

Quelle est la complexité de votre algorithme ?

Question 26

(3 points)

Joseph Marchand n'est pas satisfait, attendre que l'algorithme se termine lui fait passer sa faim.

Grâce au premier algorithme donné en début d'exercice (fonction Chocolat), optimiser la fonction de la question précédente.

Question 27

(1 point)

Quelle est la complexité temporelle de cet algorithme ?

Question 28

(2 points)

Proposer un algorithme affichant trois listes correspondants aux trois découpes d'une recette, obtenant une intensité minimale.

6 Bonus

Question bonus 29 (1 point)

▮ Pile ou face ?

Question bonus 30 (1 point)

▮ Quelle est l'année de la première apparition de Joseph Marchand dans les exercices Prologin ?

- 2005
- 2007
- 2018
- 2022
- 2042

Question bonus 31 (1 point)

▮ Citer 8 lettres non présentes dans les noms ou pseudos des organisateurs.

Question bonus 32 (1 point)

▮ Nommer un ustensile de cuisine que Joseph n'a pour sûr jamais utilisé lors de ses aventures culinaires.

Question bonus 33 (3 points)

▮ Dévoiler votre meilleure recette de gâteau.

Question bonus 34 (-4242 point)

▮ Aller acheter les ingrédients de la recette énoncée précédemment et préparer le gâteau.