Prologin 2024

Concours national d'informatique

Épreuve écrite d'algorithmique
Paris Toulouse Lille

11 Février 2024

Peintures

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. Pendant cette épreuve, vous passerez un entretien (10 minutes) avec un organisateur. Ensuite, vous aurez une épreuve de programmation sur machine (3 heures et 30 minutes) l'après-midi.

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.

— Lorsqu'un algorithme est demandé, vous pouvez le décrire avec suffisamment de précision, le pseudo-coder ou l'implémenter dans le langage de votre choix. Dans le dernier cas, veuillez néanmoins préciser le langage 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 Informations

En l'an 2142, l'humanité est composée de robots et d'humains, dans une atmosphère d'égalité. Dans cet univers, Joseph Marchand est un artiste peintre spécialisé dans la création d'œuvres d'art, quand un jour, il reçoit des commandes bien particulières de robots : ils veulent tous des œuvres générées par algorithmes qui respectent certaines conditions.

À travers ce sujet nous allons explorer différentes peintures demandées. Le sujet est composé de 4 parties indépendantes :

1. Analyse de peintures 1D sous contraintes (28 points)
2. Compositions de peintures 1D (12 points)
3. Réparation de peintures et tableaux (18 points)
4. Peindre des sculptures 3D (42 points)

Dans toutes les sections, une *couleur* est un nombre entier positif.

3 Peinture de paysages simples

Dans cette section, tous les robots demandent des œuvres représentant des paysages longitudinaux. Une toile est un tableau à une dimension, découpé en cases.
On notera $T[i]$ la *couleur* de la case i.
Les indices commencent en 0, ainsi $T[0]$ est la première case.

3.1 Alice et Bob

Question 1 (1 point)

Si nous voulons utiliser N couleurs distinctes au moins deux fois chacune, quelle doit être la taille minimale de la toile ?

Question 2 (3 points)

Un tableau T de taille N est dit symétrique si la lecture de ses éléments de gauche à droite est identique à celle de droite à gauche.

Le premier client (Alice) demande une peinture de longueur $2N$ symétrique et composée de N couleurs différentes. On considère le code suivant :

Entrée : Un entier $N \in \mathbb{N}$
Sortie : Un tableau de taille $2N$ symétrique et composé de N couleurs différentes.

Soit T un tableau rempli de 0 de longueur $2N$
pour i de 0 inclus à N exclus **faire**
  $T[i] \leftarrow i$
  $T[2N - i] \leftarrow i$
fin pour

Expliquez pourquoi cet algorithme est incorrect, puis proposez un correctif.

Une couleur paire est une couleur dont la valeur est paire, une couleur impaire est une couleur qui n'est pas paire.

Question 3 (3 points)

Un nouveau client, Bob, demande une peinture respectant ces deux conditions :

1. Toutes les couleurs paires strictement positives sont entourées uniquement de couleurs paires
2. Toutes les couleurs impaires sont soit à coté d'autres couleurs impaires ou soit de 0

Donnez une peinture pour Bob de longueur 5, qui respecte ses conditions, composée des couleurs $\{0, 1, 3, 4, 6\}$.

S'il y a des nombres paires et impaires dans l'ensemble des couleurs, que remarque-t-on sur la position du 0 dans un tableau satisfaisant la condition de Bob ?

Question 4 (5 points)

On considère cet algorithme qui prend une liste de couleurs (pas nécessairement contiguës) et qui renvoie un tableau respectant les conditions de Bob contenant toutes les couleurs, et un tableau vide si c'est impossible.

Entrée : Une liste L de couleurs, de longueur N
Sortie : Un tableau T de longueur N satisfaisant la condition de Bob si c'est possible.

Soit T un tableau de 0 de taille N
Soit $j = 0$
Soit `contient_zero` = FAUX
pour i de 0 inclus à N exclus **faire**
    si $L[i]$ impair **alors**
        $T[j] \leftarrow L[i]$
        $j \leftarrow j + 1$
    fin si
    si $L[i] = 0$ **alors**
        `contient_zero` ← VRAI
    fin si
fin pour
si (non `contient_zero`) et $j \notin \{N, 0\}$ **alors**
    renvoyer Tableau vide
fin si
pour i de 0 inclus à N exclus **faire**
    si $L[i]$ pair **alors**
        $T[j] \leftarrow L[i]$
        $j \leftarrow j + 1$
    fin si
fin pour
renvoyer T

Expliquer pourquoi cet algorithme est incorrect en s'appuyant d'un contre-exemple, puis proposez un correctif.

Question 5 (5 points)

Donnez un algorithme qui prend une liste de couleurs (pas nécessairement contiguës) et qui renvoie un tableau respectant les conditions de Bob ET d'Alice en même temps, contenant toutes les couleurs de la liste, et un tableau vide si c'est impossible. On peut ré-utiliser une couleur plusieurs fois. On ne met aucune contrainte sur la taille du tableau de sortie.

3.2 Coloration spécifique

De bouche à oreille, et grâce aux services rendus à Alice et Bob, Joseph Marchand se fait connaître par Camille, une investisseuse d'œuvre d'art ! Étant minimaliste, elle va faire une série de commissions chez Joseph Marchand, et pour chacune il faudra utiliser un minimum de couleurs possibles.

Question 6 (2 points)

La première demande de Camille est un tableau de taille $N \in \mathbb{N}$ tel que

$$\forall i \quad 0 \leq i < \frac{N}{2} \quad T[i] \neq T[2i+1] \neq T[2i+2] \neq T[i]$$

Dessinez un tel tableau de longueur 10 utilisant un minimum de couleurs.

Question 7 (5 points)

Proposez un algorithme qui prend en argument un N et qui génère un tel tableau.

Question 8 (4 points)

Montrez que le nombre de tableaux de taille N n'utilisant que 3 couleurs respectant cette condition en fonction de N est $3 \times 2^{2^{k-1}-1}$, quand N est de la forme $2^k - 1$.

4 Redessiner des peintures

Joseph Marchand commence réellement à se faire connaître ! Il reçoit maintenant des commission de clients un peu célèbres, mais qui lui demandent des choses bien particulières : corriger voire combiner des peintures déjà faites ! Évidemment, afin de bien réussir sa carrière dans le monde artistique, il ne peut pas refuser une occasion pareille... mais il a besoin d'aide !

4.1 Redessiner des peintures

Ici, le client vous donne une liste de peintures 1D, et aimerait que vous combiniez les peintures en une seule grande œuvre. Dans cette combinaison, chaque élément présent dans les peintures données doivent se retrouver une fois dans le tableau résultant. La taille doit donc être la somme des tailles des tableaux donnés en entrée.

De plus, chaque élément d'un tableau donné en entrée doit se retrouver dans le même ordre (mais potentiellement séparés par d'autres éléments) dans le tableau résultant. Par exemple, si vous avez les 3 peintures $[1, 2, 3, 4]$, $[5, 6]$ et $[1, 1, 2]$, votre œuvre finale pourrait être $[1, 1, 1, 5, 2, 6, 3, 2, 4]$.

Question 9 (3 points)

Le client vous donne une peinture combinée P et deux tableaux T et T' et aimerait s'assurer qu'elle puisse bien provenir des deux tableaux T et T' originaux. On considère l'algorithme suivant :

Entrée : Trois tableaux T, T', P de longueurs respectives N, N', N_p
Sortie : VRAI ou FAUX

si $N + N' \neq N_p$ **alors**
  renvoyer FAUX
fin si
Soit T_s un tableau à deux dimensions de FAUX de taille $N \times N'$
$T[0][0] \leftarrow$ VRAI
pour i_1 de 0 inclus à N exclus **faire**
  pour i_2 de 0 inclus à N' exclus **faire**
    $T_s[i_1][i_2] \leftarrow P[i_1] == T[i_1]$ et $T_s[i_1 - 1][i_2]$
    $T_s[i_1][i_2] \leftarrow P[i_2] == T'[i_2]$ et $T_s[i_1][i_2 - 1]$
  fin pour
fin pour
renvoyer $T_s[N - 1][N' - 1]$

Expliquez pourquoi cet algorithme est incorrect, puis proposez un correctif.

Question 10 (4 points)

Ecrivez un algorithme qui vérifie si P est possiblement la combinaison des tableaux T et T', cette fois-ci en autorisant le partage des éléments entre plusieurs tableaux d'origine.

Par exemple, [1,2,2,2,3] est désormais une combinaison valide de [1,2,2,2] et de [2,2,3]. On pourra s'inspirer du code de la question précédente et le faire en programmation dynamique.

Question 11 (5 points)

Ecrivez un algorithme qui renvoie une des plus petites (en longueur) peintures P possible que l'on peut faire avec T et T' (les deux tableaux originaux), en pouvant partager des couleurs.
On peut prendre le même exemple que la question précédente, [1,2,2,2,3] est une combinaison valide de [1,2,2,2] et de [2,2,3]. On pourra s'inspirer du code de la question précédente et le faire en programmation dynamique.

4.2 En 2D !

Le client décide maintenant de vous faire confiance pour réparer certaines grandes peintures 2D. Une peinture 2D est représentée par un tableau à 2 dimensions, si bien que $T[i][j]$ représente l'élément à la $(i+1)$-ème ligne et $(j+1)$-ème colonne.

Le client possède un tableau T de 0 et 1, à deux couleurs donc. Mais... quelque chose est louche. Il était sûr que sur l'original le nombre de 1 dans chaque ligne et dans chaque colonne était pair ! Il vous demande donc votre aide.

Question 12 (3 points)

Écrivez un algorithme qui prend en argument un tableau T en entrée et qui vérifie si c'est un original (si le nombre de 1 dans chaque ligne et chaque colonne est bien pair)

Question 13 (3 points)

Il se demande quel est le nombre minimal de cases qu'il aurait à changer pour revenir sur un tableau valide (qui a un nombre pair de 1 dans chaque ligne et chaque colonne).

Dessinez un des pires cas possibles (dont le minimum de cases à changer est le plus élevé possible) pour un tableau de taille $N = 3$ et $M = 4$. Combien de cases faudrait-il changer ?

Question 14 (4 points)

Montrez par récurrence sur le nombre de cases noires que si I est le nombre de lignes avec un nombre impair de 1 et J le nombre de colonnes avec un nombre impair de 1, alors $I + J$ est toujours pair.

Question 15 (8 points)

Donnez une expression du nombre de cases minimal à changer en fonction de I et J. Proposer un algorithme corrigeant un tableau donné en entrée avec un nombre minimal de modifications.

5 Des peintures de sculptures 3D

Waou ! Joseph Marchand fait fureur, et a été invité à peindre des structures 3D très étranges d'un artiste très très réputé, Madeline Mollet. Précisément, les structures sont composées de boules à colorier uniformément qui sont reliées entre elles par des tiges de metal. On représentera donc une sculpture à colorier par un graphe non orienté : les boules à colorier sont les nœuds et les tiges sont les arêtes.

Pour une telle sculpture $G = (N, E)$ avec N l'ensemble des nœuds et E l'ensemble des arêtes (on a $E \subseteq N \times N$), on pose $\deg(s)$ comme étant le nombre d'arêtes attachées à $s \in N$ (c'est son degré dans le graphe G). De plus on pose $\Delta(G) = \max_{s \in N}\{\deg(s)\}$ le degré maximum sur tout le graphe.
Toutes les sculptures que l'on considère ont au moins 2 sommets. Aucun sommet n'est relié à lui-même.

5.1 Colorier des sommets

Question 16 (2 points)

On suppose que la sculpture est en un seul morceau (une seule composante connexe).
Montrez que $1 \leq \Delta(G) < |N|$, avec $|N|$ le nombre de nœuds du graphe.

Question 17 (4 points)

Proposez un algorithme qui détermine si la sculpture est en un seul morceau. On indiquera quelle structure de données a été choisie pour représenter le graphe.

On cherche maintenant à colorier le graphe de manière à ce que deux couleurs ne se touchent jamais. On note $\chi(G)$ le nombre minimum de couleurs nécessaires pour colorier G en respectant cette condition.

Question 18 (2 points)

Dessinez un graphe à 6 sommets dont le nombre minimum de couleurs nécessaires pour le colorier sous cette contrainte est 4.

Question 19 (5 points)

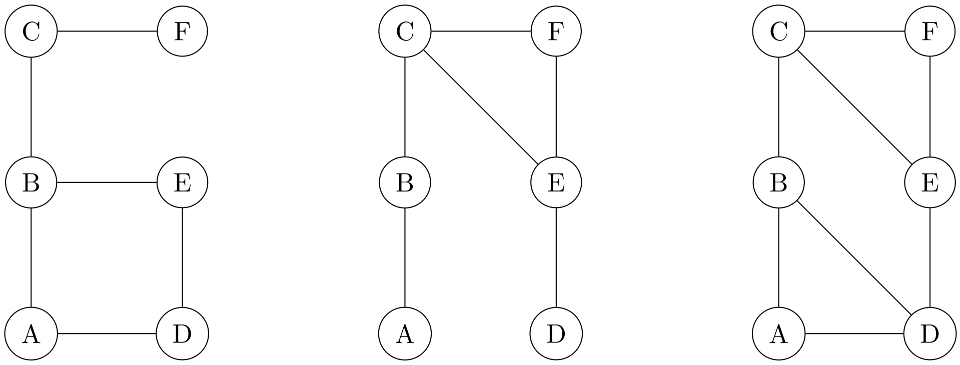
Montrez que si $\Delta(G) = 2$, alors il y a un algorithme qui colorie le graphe en le minimum de couleurs en $O(|N|)$ en respectant la condition. Dans quel cas on a besoin de 3 couleurs ?

5.2 Colorier des arêtes

On s'intéresse ici à colorier non pas des sommets, mais des arêtes, de manière à ce que deux arêtes de la même couleur n'ont jamais de sommets en commun.

Question 20 (3 points)

Pour chacun des graphes suivants, donnez une colorisation des arêtes utilisant un minimum de couleurs :



Le client se demande combien de couleurs au minimum il a besoin pour colorier les arêtes de cette manière (on note cela $\chi'(G)$). On va montrer que $\chi'(G) = \Delta(G)$ ou $\chi'(G) = \Delta(G) + 1$. Cela va répartir les graphes en deux classes :

1. Ceux de Classe 1 tel que $\chi'(G) = \Delta(G)$
2. Ceux de Classe 2 tel que $\chi'(G) = \Delta(G) + 1$

Question 21 (2 points)

Donnez deux graphes G_1, G_2 ayant strictement plus de 3 sommets tel que $\chi'(G_1) = \Delta(G_1)$ et tel que $\chi'(G_2) = \Delta(G_2) + 1$. Pour chacun de ces graphes, indiquez la coloration et leur degré maximal.

Question 22 (2 points)

Montrez que $\chi'(G) \geq \Delta(G)$

On va montrer le théorème par récurrence sur le nombre d'arêtes. On pose

$$P(n) : \text{“Pour tout graphe } G = (N, E) \text{ avec } |E| = n \text{ on a } \chi'(G) \leq \Delta(G) + 1\text{”}$$

Question 23 (4 points)

Montrez $P(i)$ pour i de 0 inclus à 3 inclus.

Question 24 (2 points)

On suppose $P(n)$ vraie et on va montrer $P(n+1)$ par l'absurde. On considère G un graphe à n arêtes que l'on suppose non $\Delta(G) + 1$ coloriable et on y retire une arête (a, b). On note $G - ab$ le graphe ou l'on a retiré l'arête (a, b), qui est par hypothèse de récurrence $\Delta(G) + 1$ coloriable.

Pour deux couleurs α et β et un sommet x, on note $x : \alpha/\beta$ le plus long chemin commençant à x et alternant en couleur entre α et β. On dit qu'un sommet x *utilise* une couleur c si c est la colorisation d'une des arêtes connectées à x.

Montrez que pour tout sommet x de $G - ab$ on peut choisir une couleur non utilisée par x.

Question 25 (3 points)

Montrez que pour x un nœud et α, β deux couleurs, $x : \alpha/\beta$ est unique.

Question 26 (5 points)

Si α est une couleur non utilisée par le sommet a et β une couleur non utilisée par le sommet b, montrez que si b n'est pas dans le chemin $a : \alpha/\beta$ (dans $G - ab$) alors G est $\Delta(G) + 1$ coloriable.

Question 27 (8 points)

Montrez que si b est dans le chemin $a : \alpha/\beta$ (dans $G - ab$), alors G est $\Delta(G) + 1$ coloriable. Conclure sur la démonstration du théorème.

6 Questions bonus

N'abordez ces questions que si vous vous êtes relu 42 fois.

Question bonus 28 (1 point)

Écrire l'algorithme d'Al-Kashi.
Note : cet algorithme n'existe pas.

Question bonus 29 (3 points)

Ecrivez un algorithme qui renvoie le plus grand nombre possible avant de terminer. Si vous renvoyez le plus grand nombre de votre centre d'examination, vous gagnez 3 points.

Question bonus 30 (2 points)

Inventez une règle qui est valide pour tous les tableaux suivants. Est jugée la créativité de la règle

1. $[0, 10, 20]$
2. $[-5, -10, -15, -20]$
3. $[1, 2, 2, 3, 4, 5, 999]$
4. $[0, 1, 0, 0, 0, 1]$

Question bonus 31 (2 points)

Redessinez la molécule suivante sur votre copie et proposez une bicoloration des atomes.