4I505 CPA – Examen Mai 2015

Les documents distribués dans le cadre du cours et les notes personnelles sont autorisés.

1 Question de cours

Pour chaque question suivante donnez une preuve formelle justifiant votre réponse.

- Quelle est la complexité de l'algorithme Ritter calculant une approximation du cercle minimum couvrant un ensemble de points?
- Quelle est la complexité du filtrage Alk-Toussaint vu en cours?
- Quelle est la complexité de l'algorithme QuickHull calculant l'enveloppe convexe d'un ensemble de points?
- Peut on concevoir un algorithme calculant le contour positif de l'enveloppe convexe d'un ensemble de n points ayant une complexité temporelle en $O(n)$?

2 L'élément en double

On utilise la notation: $[m] = \{1, \ldots, m\}$. Étant donné un flux $x_1, \ldots, x_n \in [m]$, avec $m = n - 1$ et la promesse que tout élément $i \in [m]$ apparaît exactement une fois dans le flux, sauf un élément particulier $k \in [m]$ qui apparaît 2 fois. Concevez un algorithme à une seule passe qui détermine cet élément double avec une complexité espace $O(\log n)$ bits.

3 L'élément majoritaire

Étant donné un flux $x_1, \ldots, x_n \in [m]$, avec la promesse qu'il existe un élément $k \in [m]$ qui apparaît strictement plus $n/2$ fois dans le flux. Concevez un algorithme à une seule passe qui détermine cet élément double avec une complexité espace $O(\log n + \log m)$ bits.

4 Minimiser distance Manhattan

La distance Manhattan entre deux points du plan $(x, y), (x', y')$ est définie comme $|x - x'| + |y - y'|$. Étant donnée un flux de points $p_1, \ldots, p_n\{-m, +m\}$ aux coordonnées entières, on cherche un point q qui minimise

$$\sum_{i=1}^{n} d_1(p_i, q),$$

où d_1 est la distance de Manhattan. Ce point q ne fait pas forcément partie du flux d'entrée et les coordonnées de q ne sont pas nécessairement entières.

Concevez un algorithme à une seule passe pour ce problème. Montrez qu'il est valide et analysez sa complexité en espace.

5 Échantillonner un couple de valeur

Soit un flux $x_1, \ldots, x_n \in [m]$, avec la promesse simplificatrice $x_i \leq x_j$ pour tout $i, j \in [m]$ avec $i \neq j$. On veut générer deux valeurs x_i, x_j choisis uniformément sur tous les couples $i, j \in [m]$ avec $i \neq j$. Chaque couple de valeur distincts du flux a donc la même probabilité $1/(n(n-1))$ d'être choisi.

Concevez un algorithme à une seule passe pour ce problème. Montrez qu'il est valide et analysez sa complexité en espace. L'algorithme ne connaît pas n en avance.

Nous avons vu en cours un algorithme pour générer uniformément au hasard un élément du flux. On serait tenté de s'en servir avec deux exécutions indépendantes en parallèle, mais dans ce cas on ne pourrait pas assurer que les deux éléments générés sont distincts.

A Corrections

A.1 Élément en double

Soit $S = (n+1)n/2$. Pour déterminer l'élément en double il suffit de faire la somme des éléments du flux et soustraire S. La valeur de la somme est de l'ordre $O(n^2)$, qui prend $O(\log n)$ bits pour le stockage.

A.2 L'élément majoritaire

Initialement $c := 0$ et h quelconque. Sur la vue de x, si $h = x$ alors c est incrémenté, sinon si c est positif c est décrémenté, sinon $c := 1$ et $h := x$. La réponse finale est h.

Pour la validité, considérons une incrémentation suivi d'une décrémentation provoqués par deux éléments x, y successifs dans le flux. La suppression de x, y du flux, ne change pas le comportement de l'algorithme, dans le sens que lors du traitement d'un élément z, les valeurs c, h restent inchangées par cette suppression. La suppression préserve aussi que k est l'élément majoritaire, car au plus une occurrence de k est supprimé du flux sur deux suppressions. En répétant cette suppression on trouve un flux qui ne contient que k est clairement l'algorithme produit k sur ce flux.

Pour la complexité en espace, la valeur de s est au plus n et nécessite donc $O(\log n)$ bits alors que h peut stocker toute valeur de $[m]$ et nécessite donc $O(\log m)$ bits.

A.3 Minimiser distance Manhattan

Avec cette distance le problème peut être résolu indépendamment pour chaque coordonnée. Il suffit de choisir q comme le point dont la coordonnée x est la moyenne des coordonnées x des points du flux et idem pour la coordonnée y. Choisir le médian ne minimise pas la distance. Pour calculer la moyenne il suffit de faire la somme sur les coordonnées et compter le nombre d'élément dans le flux. Ceci génère une complexité en espace de $O(\log(nm)) = O(\log n + \log m)$ bits.

A.4 Échantillonner un couple de valeur

Soit S l'ensemble des valeurs de la partie du flux déjà observé. On veut un algorithme randomisé qui maintient deux valeurs aléatoires X, Y dans S, avec pour tout $i, j \in S$

$$P[X = i, Y = j] = \begin{cases} 0 & \text{si } i = j \\ \frac{1}{n(n-1)} & \text{sinon.} \end{cases} \tag{1}$$

L'algorithme initiale après avoir vu x_1, x_2 du flux avec probabilité $1/2$ $X = x_1, Y = x_2$ et avec probabilité $1/2$, $X = x_2, Y = x_1$.

Puis pour chaque élément x_k observé, l'algorithme avec probabilité $1/k$ pose $X = x_k$, avec probabilité $1/k$ pose $Y = x_k$ et avec probabilité $(k-2)/k$ les variables restent inchangées.

Pour la validité de l'algorithme notez que (1) implique que $P[X = i] = 1/n$ et $P[Y = j] = 1/n$. Pour montrer que l'invariant (1) est respecté après avoir traité x_k, soient i, j deux valeurs distincts parmi $\{x_1, \dots, x_k\}$.

Si ni i ni j sont x_k, alors la probabilité $P[X = i, Y = j]$ est la probabilité de cet événement avant le traitement de x_k, fois la probabilité que les variables sont restés inchangés. Donc

$$\frac{2}{(k-1)(k-2)} \cdot \frac{k-2}{k} = \frac{2}{(k-1)k}$$

comme demandé.

Si $i = x_k$ on utilise le fait qu'avant le traitement de x_k on ait $P[Y = j] = 1/(k-1)$. Donc avec probabilité $1/k$ X prend la valeur x_k et donc

$$P[X = i, Y = j] = \frac{1}{k} \cdot \frac{1}{k-1} = \frac{1}{k(k-1)},$$

comme demandé.

Le cas restant $j = x_k$ est symétrique.