

Proposition de correction

Analyse d'algorithme

Remarque : Dans l'objectif d'appliquer les objectifs SEED de cette année, merci de ne pas imprimer cette première page (si vous souhaitiez imprimer le document complet).

1 Complexité moyenne sur des listes ou des tableaux

Moyenne sur x

Directement, en utilisant la définition de la complexité moyenne il vient :

$$C_{moy}(n) = \sum_{X||x|=n} \delta(x)C(x) = \sum_{k=1}^n \frac{k}{n} = \frac{1}{n} \frac{n(n+1)}{2} = \mathcal{O}(n)$$

Moyenne sur L

La complexité en moyenne sur une liste L est la complexité en moyenne sur une liste dans laquelle l'élément considéré est dedans plus celle sur une liste dans laquelle l'élément considéré est manquant.

La formule de la complexité en moyenne est :

$$C_{moy}(n) = \sum_{x||x|=n} \delta(x)C(x)$$

Pour être plus précis, on distingue alors deux cas :

→ Si x est dans la liste :

$$C_{moy}(n) = \sum_{k=1}^n \frac{1}{M} k = \frac{n(n+1)}{2M}$$

→ Si x n'est pas dans la liste :

$$C_{moy}(n) = \frac{n(n+1)}{2}$$

La complexité en moyenne sur une liste L de taille n et dont les valeurs sont aléatoirement dans $[1;M]$ est :

$$\begin{aligned} C_{moy}(n) &= \sum_{x \in L} \delta(x)C(x) + \sum_{x \notin L} \delta(x)C(x) \\ &= \sum_{k=1}^n \left(\frac{1}{M} \left(1 - \frac{1}{M}\right)^{k-1} \right) k \\ &= \frac{1}{M} \sum_{k=1}^n \left(1 - \frac{1}{M}\right)^{k-1} \\ &= \frac{1}{M} \left(\sum_{k=1}^n \left(1 - \frac{1}{M}\right)^k \right)' \\ &= \left(\frac{1}{M^n} \right)' \\ &= (M^{-n})' \\ &= M \left(1 - \left(\frac{M-1}{M} \right)^n \right) \end{aligned}$$

Commentons la complexité : lorsque n est grand devant M , c'est-à-dire quand la taille de la liste est grande devant le nombre de valeur, la complexité est linéaire en M ; lorsque que c'est l'inverse, est que la liste est petite devant le nombre de valeurs, la complexité est constante en moyenne.

Moyenne sur x et L

La complexité en moyenne d'une liste L est :

$$C_{moy}(n) = M \left(1 - \left(\frac{M-1}{M} \right)^n \right)$$

La complexité ne dépendant pas de x, la complexité en moyenne sur x et L est la complexité sur L.

2 Dénombrement fruité

On dispose de pommes, de bananes, d'oranges et d'ananas. On souhaite dénombrer les manières de choisir n fruits tout en respectant les contraintes suivantes :

- les pommes sont en nombre pair
- le nombre de bananes est un multiple de 5
- il y a au plus 4 oranges
- il y a strictement moins de deux ananas.

Donner une série génératrice pour chacun de ces contraintes.

L'intuition à avoir pour introduire les coefficients des séries génératrices est la suivante : on souhaite qu'ils nous permettent de compter le nombre de cas favorables lorsqu'on effectuera le produit des dites séries génératrices – à savoir un produit de Cauchy.

Aussi, pour les pommes, on choisit les coefficients $(p_k) := \begin{cases} 0 & \text{si } k \in 2\mathbb{Z} + 1 \\ 1 & \text{si } k \in 2\mathbb{Z} \end{cases}$.

Alors, il vient la série

$$P(z) := \sum_{k=0}^{\infty} p_k z^k = \sum_{k=0}^{\infty} z^{2k} = \sum_{k=0}^{\infty} (z^2)^k = \frac{1}{1 - z^2}$$

De même, on introduit pour les bananes la série

$$B(z) := \sum_{k=0}^{\infty} (z^5)^k = \frac{1}{1 - z^5}$$

Comme les conditions sur les oranges et les ananas peuvent se résumer à l'énumération exhaustive des cas – il y a soit pas d'ananas soit un ananas – on introduit :

$$O(z) := 1 + z + z^2 + z^3 + z^4 \quad \text{et} \quad A(z) := 1 + z$$

En déduire le nombre de possibilités de choisir n fruits en respectant ces contraintes.

On note $\Pi(z)$ la série produit. Il vient alors :

$$\begin{aligned}
\Pi(z) &= P(z)B(z)O(z)A(z) \\
&= \frac{\left(\frac{1-z^5}{1-z}\right)(1+z)}{(1+z)(1-z)(1-z^5)} \\
&= \frac{1}{(z-1)^2} \\
&= \frac{d}{dz} \left(\frac{1}{1-z} \right) \\
&= \frac{d}{dz} \sum_{k=0}^{\infty} z^k \\
\Pi(z) &= \sum_{k=0}^{\infty} (k+1)z^k
\end{aligned}$$

Étant donnée la construction de nos séries, et par unicité des séries entières, on en déduit par identification qu'il y a $n + 1$ possibilités de choisir n fruits sous de telles contraintes.

3 Analyse amortie : Table dynamique

On considère la structure de données table dynamique, qui combine des avantages des tableaux et des listes. On utilise pour cela des tableaux de taille variées.

Des insertions uniquement

Dans un premier temps, comme en cours, on considère uniquement des insertions comme suit :

S'il y a de la place, on ajoute le nouvel élément au tableau. Sinon, on réalloue un tableau de capacité double, on recopie les premiers éléments, et on ajoute le nouvel élément dans la entière case vide.

Par convention, à partir d'un tableau vide, on obtient un tableau de taille 1.

On va évaluer la complexité amortie par la méthode du potentiel : pour cela, on pose

$$\phi(t) := 2\#\text{cases_remplies}(T) - \#\text{cases}(T)$$

Illustrer la valeur de la fonction ϕ sur une table de taille 8 comportant 6 éléments. Vérifier que le potentiel est toujours positif.

Dans un tel tableau, il vient :

$$\phi(T) = 2 \times 6 - 8 = 4$$



Pour démontrer que le potentiel est toujours positif, on introduit, pour $n \in \mathbb{N}$ l'invariant $\mathcal{I}(n)$: "Après la n^e insertion, le potentiel est positif" ; puis l'on procède par récurrence.

Naturellement, lors de la première insertion, le tableau est crée de taille 1 et contient un élément : l'invariant est vérifié.

Supposons, pour $n \in \mathbb{N}$, que $\mathcal{I}(n)$ est vrai. En procédant à la $(n + 1)^e$ insertion, on distingue alors deux cas :

→ Si il n'y a pas besoin de doubler la taille du tableau alors, en notant T' le tableau après insertion de l'élément, il vient :

$$\phi(T') = \phi(T) + 2 \geq \phi(T) \geq 0$$

Et cela vérifie l'invariant de rang $n + 1$.

→ Sinon, on a exactement, après avoir doublé la taille et inséré l'élément :

$$\phi(T') = 2 \times (|T| + 1) - (2 * |T|) = 2 \geq 0$$

Et cela vérifie l'invariant de rang $n + 1$.

Le principe de récurrence conclut.

Calculer la complexité amortie d'une opération d'insertion, en distinguant s'il y a une opération de réallocation ou non.

Pour un tableau T de taille n données, devenant T' , on trouve :

→ Sans réallocation du tableau, en notant x le nombre de données :

$$C_{amo}(op) := C_{reel}(op) + \phi(T') - \phi(T) = 1 + 2 * (x + 1) - n - (2 * x - n) = 3$$

→ Avec réallocation :

$$C_{amo}(op) := C_{reel}(op) + \phi(T') - \phi(T) = (n + 1) + 2 * (x + 1) - 2 * n - (2 * x - n) = 3$$

Ainsi, la complexité amortie de cette opération est $C_{amo}(op) = 3$ opérations.

Remarque : On rappelle que la complexité donnée par la méthode du potentiel est une majoration de la complexité amortie – et qu'elle est d'autant plus fine que la fonction de coût est judicieusement choisie !

Avec des suppressions

Justifier que ϕ' satisfait les contraintes des fonctions de potentiel

On vérifiera que la fonction est bien nulle pour un tableau vide et qu'elle reste toujours positive – on pourra pour cela étudier l'invariant $\mathcal{I}(n)$: "À l'étape n , plus d'un quart des cases sont remplies" ; comme fait précédemment.

Exprimer les coûts réels et l'évolution de la fonction de potentiel ϕ pour les opérations d'insertion et de suppression.

Dans la suite, on pose fonction $\mu(T) := \begin{cases} 1 & \text{si } 2 * \#cases_remplies(T) \geq \#cases(T) \\ 0 & \text{sinon} \end{cases}$

Traitons l'insertion dans un premier temps :

→ $\mu(T) = \mu(T') = 1$.

❖ $C_{amo} = 3$ (cf. question précédente)

→ $\mu(T) = \mu(T') = 0$.

❖ $C_{reel}(ins) = 1$

❖ $\Delta(ins) := \phi(T') - \phi(T) = -1$

$$\clubsuit C_{amo} = 0$$

$$\rightarrow \mu(T) < \mu(T').$$

$$\clubsuit C_{reel}(ins) = 1$$

$$\clubsuit \Delta(ins) := \phi(T') - \phi(T) = 2 * (n + 1) - 2 * n - \lfloor \frac{n}{2} \rfloor + n = 2 + \lfloor \frac{n}{2} \rfloor$$

$$\clubsuit C_{amo}(ins) = 3 + \lfloor \frac{n}{2} \rfloor$$

On fait ensuite de même avec l'opération de suppression :

$$\rightarrow \mu(T) = \mu(T') = 1.$$

$$\clubsuit C_{reel}(supp) = 1$$

$$\clubsuit \Delta(supp) := \phi(T') - \phi(T) = -2$$

$$\clubsuit C_{amo}(supp) = -1$$

$$\rightarrow \mu(T) = \mu(T') = 0.$$

\clubsuit Pas de réallocation.

$$- C_{reel}(supp) = 1$$

$$- \Delta(supp) := \phi(T') - \phi(T) = 1$$

$$- C_{amo}(supp) = 2$$

\clubsuit Avec réallocation.

$$- C_{reel}(supp) = \frac{n}{2} + \frac{n}{4}$$

$$- \Delta(supp) := \phi(T') - \phi(T) = \lfloor \frac{n}{4} \rfloor - \frac{n}{4} - \lfloor \frac{n}{2} \rfloor + \frac{n}{4} + 1$$

$$- C_{amo}(supp) = 1 + \frac{n}{2}$$

$$\rightarrow \mu(T) < \mu(T').$$

$$\clubsuit C_{reel}(supp) = 1$$

$$\clubsuit \Delta(supp) := \phi(T') - \phi(T) = \frac{n}{2} - \frac{n}{2} - 2 * (\frac{n}{2} - 1) + n = 1$$

$$\clubsuit C_{amo}(supp) = 2$$

Remarque : Compte tenu de la non constance des coûts, nous ne voyons pas comment conclure sur les coûts amortis globaux. On peut tout de même remarquer que les opérations d'insertions sont en temps constants presque tout le temps (hors cas spécifiques arrivant assez peu souvent) : on gagne donc beaucoup à faire cette étude amortie (on aurait conclut sur un temps linéaire sinon).

4 Nombres de Catalan

Valeurs de C_0 , C_1 , C_2 , C_3 et une relation de récurrence sur C_n

$C_0 = 1$. Il n'y a qu'un arbre qui ne contienne aucun noeud.

$C_1 = 1$. Le seul arbre ne contenant qu'un noeud est l'arbre ne contenant que la racine.

$C_2 = 2$. Il y a deux arbres qui contiennent deux noeuds : celui qui contient la racine et un noeud dans son fils droit, et celui qui contient la racine et un noeud dans son fils gauche.

$C_3 = 5$. Il y a cinq arbres, tous sont composés de la racine. Pour le premier, il y a un noeud par fils. Pour les autres, il y a deux noeuds sur le fils droit, ou deux noeuds sur le fils gauche.

En raisonnant sur la construction des arbres binaires précédents, on en déduit la relation de récurrence :

$$C_{n+1} = \sum_{k=0}^n C_k C_{n-k}$$

L'idée est de voir que l'on ajoute un arbre binaire de taille k à un arbre binaire de taille $n-k$.

Une autre façon de voir les choses, plus complexe, est de voir le parenthésage d'une expression comme le produit de deux grands membres de tailles k et $n-k$, puis de chercher à parenthéser ces membres. Ce qui donne la même formule de récurrence. Noter qu'il faut aussi remarquer une bijection entre les arbres binaires et les parenthésage de Dick, ce qui se fait avec un *parcours extérieur* de l'arbre.

Série génératrice

On cherche à établir une relation de récurrence pour déterminer $G(z)$:

$$\begin{aligned} G(z) &= C_0 + z \sum_{k=0}^{\infty} C_{k'+1} z^{k'} \\ &= C_0 + \left(\sum_{k=0}^{\infty} C_{k'} z^{k'} \right)^2 \\ &= 1 + zG^2(z) \\ &= \frac{1 \pm \sqrt{1-4z}}{2z} \end{aligned}$$

De plus, $G(0) = 0$ ce qui, avec une étude asymptotique, permet d'affirmer que :

$$G(z) = \frac{1 - \sqrt{1-4z}}{2z}$$

Formule de Taylor

Nous voulons maintenant déterminer les coefficients de la série génératrice : $H(z) = zG(z)$.

Or, nous savons que $G(z) = \frac{1-\sqrt{1-4z}}{2z}$.

Nous pouvons donc utiliser la formule de Taylor sur $\sqrt{1-4x}$ pour obtenir, sur $[-\frac{1}{4}; \frac{1}{4}]$:

$$\sqrt{1-4x} = \sum_{n=0}^{+\infty} \binom{-\frac{1}{2}}{n} (-4x)^n = 1 + \sum_{n=1}^{+\infty} \binom{-\frac{1}{2}}{n} (-4x)^n$$

De plus, nous avons :

$$\binom{-\frac{1}{2}}{n} = (-1)^n \frac{(2n)!}{2^{2n}(n!)^2}$$

Nous avons donc :

$$\begin{aligned} H(z) &= \frac{\sum_{n=1}^{+\infty} \binom{-\frac{1}{2}}{n} (-4)^n z^{n+1}}{2z} \\ &= \frac{\sum_{n=1}^{+\infty} \binom{-\frac{1}{2}}{n} (-4)^n z^n}{2} \\ &= \frac{\sum_{n=1}^{+\infty} (-1)^n \frac{(2n)!}{2^{2n}(n!)^2} (-1)^n 2^{2n} z^n}{2} \\ &= \frac{\sum_{n=1}^{+\infty} \frac{(2n)!}{(n!)^2} z^n}{2} \\ &= \sum_{n=1}^{+\infty} \frac{(2n)!}{2(n!)^2} z^n \end{aligned}$$

Série génératrice de $G(z)$

Nous avons donc prouvé :

$$H(z) = zG(z) = \sum_{n=1}^{+\infty} \frac{(2n)!}{2(n!)^2} z^n$$

Donc :

$$\begin{aligned} G(z) &= \sum_{n=1}^{+\infty} \frac{(2n)!}{2(n!)^2} z^{n-1} \\ &= \sum_{n=0}^{+\infty} \frac{(2(n+1))!}{2((n+1)!)^2} z^n \\ &= \sum_{n=0}^{+\infty} \frac{(2n)!2(n+1)}{2(n!)^2(n+1)^2} z^n \\ &= \sum_{n=0}^{+\infty} \frac{(2n)!}{(n!)^2(n+1)} z^n \\ &= \sum_{n=0}^{+\infty} \frac{1}{n+1} \binom{2n}{n} z^n \end{aligned}$$

Or nous avons :

$$G(z) = \sum_{k=0}^{+\infty} C_k z^k$$

Donc

$$\forall n \in \mathbb{N}, C_n = \frac{1}{n+1} \binom{2n}{n}$$

