Commentaires DM n°6 : Somme minimale d'une tranche de tableau

Généralités

- un exercice avec différentes approches algorithmiques pour résoudre le problème initial
- attention, ici $j = n$ est possible. $j - 1$ indique le dernier indice de la tranche A_{ij}
- beaucoup d'algorithmes à écrire recherchent un minimum. Attention à la valeur initiale (`INT_MAX`, ou bien une des valeurs à considérer)
- éviter de réaliser deux appels identiques. Utiliser une variable pour stocker le résultat de l'appel.
- justifier toujours un minimum les calculs de complexité

Q1. On demande la valeur de la somme, pour un tableau donné, dont tous les éléments sont positifs. La justification souvent trop légère. Recopier l'énoncé puis "donc" puis donner la réponse ne peut pas convenir.

Approche 1 - exhaustive

Q2. `somme`. OK. On peut ajouter un `assert (i < j)`.

Q3. `somme_mini_1`. Attention aux bornes sur les indices. On doit tester *toutes* les tranches, *une et une seule fois* pour avoir tous les points à ce genre de question.

Q4. Nombre de termes sommés. $3 + 5 + 2$ est une somme de 3 termes. C'est un calcul mathématique, pas facile, mais tout à fait faisable.

Q5. Complexité en $\mathcal{O}(n^3)$, en utilisant la question précédente ou non.

Approche 2 - (plus ou moins gloutonne)

Q6. `somme_mini_depuis`. Pour obtenir la complexité attendue, on doit calculer les sommes de proche en proche.

Q7. `somme_mini_2`. OK.

Approche 3 - diviser pour régner

Q8. OK. On doit envisager les tranches "à cheval" sur les demi-tableaux

Q9. Pour avoir la relation sur la complexité, on doit trouver la somme minimale "à cheval" en temps linéaire. On s'inspire de la question 6.

Q10. on reconnaît la relation du tri fusion ! et donc $\mathcal{O}(n \log n)$ (ou bien on déroule la récurrence, mais c'est beaucoup plus long ...)

Q11. `somme_mini_3`. plus difficile, il faut avoir bien compris cette approche. La fonction `min` est à définir en C.

Approche 4 - programmation dynamique

Q12. Q13. les deux questions les plus difficiles de l'exercice. Il faut bien "visualiser" les ensembles de tranches. On reconnaît le principe de la programmation dynamique.

Q14. `somme_mini_4`. On calcule de proche en proche (approche tabulaire "bottom-up"), la réponse attendue est m_n. Complexité en $\mathcal{O}(n)$