
Analyse d'Algorithmes et Complexité

Devoir 1

Description du problème

Soit $T[1, \dots, n]$ un tableau de n entiers relatifs.

Notons $S(k, l) = \sum_{j=k}^l T[j]$ la valeur de la sous-séquence $T[k, \dots, l]$ pour $1 \leq k \leq l \leq n$.

Déterminer quelle est la plus grande valeur des sous-séquences possibles.

On propose quatre stratégies algorithmiques pour résoudre ce problème :

- *Algorithme naïf* : examiner toutes les sous-séquences possibles.
- *Algorithme moins naïf* : optimiser l'algorithme précédent en observant que $S(k, l) = S(k, l - 1) + T[l]$.
- *Algorithme diviser pour régner* : diviser la séquence en deux. Calculer une sous-séquence de somme maximale de chaque moitié. Calculer une sous-séquence de somme maximale qui contient l'élément du milieu. Finalement, prendre le maximum des trois.
- *Algorithme incrémental* : supposer le problème résolu pour $T[1, \dots, i]$. Observer que la solution pour $T[1, \dots, i + 1]$ est soit la solution précédente, soit la sous-séquence de somme maximale qui se termine par $T[i + 1]$.

Travail demandé

Effectuer une étude des différents algorithmes présentés. Cette étude est à rendre, avant le 20 Mars 2024, sous la forme d'un rapport et comprendra trois parties :

1. *Analyse théorique* : écrire le pseudocode de chaque algorithme et calculer leur
2. *Implémentation et mesure* : écrire dans le langage C ou C++ chaque algorithme puis comparer leur performance sur des instances aléatoires. Attention : vos implémentations devront suivre les indications suivantes :
 - L'entrée standard attendue prendra l'entier n (correspondant au nombre d'entiers dans le tableau), suivi du tableau. Les algorithmes devront retourner sur la sortie standard la valeur maximale de la plus grande sous-séquence sous la forme d'un entier.
3. *Confronter votre analyse théorique et vos résultats expérimentaux.*

Bonus : Ce problème peut se généraliser aux matrices. Le problème consiste alors à trouver une sous-matrice de poids maximal dans une matrice initialement donnée. Généraliser les différents algorithmes afin de résoudre ce problème.