

T.D. 1 : Analyse Amortie

Institut Galilée - USPN

Exercice 1

Considérons que l'on maintienne une structure de données sur laquelle on effectue une série d'opérations. Soit $f(n)$ le coût réel de la n -ième opération. Pour chacune des fonctions f suivantes, déterminez le coût amorti élémentaire (c'est-à-dire d'une seule opération).

1. $f(n)$ est la plus grande puissance de 2 qui divise n .
2. $f(n)$ est l'exposant de la plus grande puissance de 2 qui divise n .
3. $f(n) = n$ si n est une puissance de 2, et $f(n) = 1$ sinon.
4. $f(n) = n^2$ si n est une puissance de 2, et $f(n) = 1$ sinon.

Exercice 2

Une MultiPile se compose d'une série infinie de piles P_0, P_1, P_2, \dots où la i -ème pile P_i peut contenir jusqu'à 3^i éléments. Chaque fois qu'un utilisateur essaie de mettre un élément dans une pile complète P_i , on vide d'abord tous les éléments de P_i et on les met dans la pile P_{i+1} pour faire de la place. Ainsi, si P_{i+1} est déjà pleine, on doit d'abord déplacer de manière récursive tous ses éléments dans P_{i+2} . Déplacer un seul élément d'une pile à la suivante coûte 1.

1. Dans le pire des cas, combien de temps faut-il pour pousser un élément supplémentaire dans un MultiStack contenant n éléments ?
2. Montrez que le coût amorti d'une opération de poussée est $O(\log(n))$, où n est le nombre maximum d'éléments dans le MultiStack.