

TP 4 – Temps de calcul, tours, terminaison

1. EXPONENTIATION RAPIDE

On cherche à écrire un programme calculant la puissance n -ième d'un réel a , avec n entier.

- (1) Ecrire un programme récursif "naïf" réalisant cette fonction.
- (2) Montrer que sa complexité en temps est linéaire en n .
- (3) Il est possible d'écrire un programme plus rapide, dont la complexité est logarithmique en n (donc un coût en temps en $\log_2(n)$). Ecrire ce programme.

2. TOURS DE HANOI

Le jeu des tours de Hanoi est un jeu de permutation : il consiste en un ensemble de trois piquets sur lesquels peuvent être placés des disques de diamètre tous différents. Tous les disques sont initialement situés sur le piquet de gauche du plus grand au plus petit. Le but est de tous les transférer sur le piquet de droite, en respectant la règle suivante : il est impossible de placer un disque au-dessus d'un disque de diamètre inférieur.

On représentera les piquets comme trois piles t_1 , t_2 et t_3 , contenant chacune la liste des diamètres des disques du bas vers le haut. Au début du jeu, la pile t_1 est donc $n, n-1, \dots, 2, 1$ et les deux autres sont vides.

- (4) Ecrire un programme permettant de résoudre le problème des tours de Hanoi pour un nombre n quelconque de disque.
- (5) Quelle est sa complexité ?

3. ALGORITHME DE MORRIS

On donne la fonction de Morris définie par récurrence de la manière suivante :

$$morris(m, n) = \begin{cases} 1 & \text{si } m = 0; \\ morris(m - 1, morris(m, n)) & \text{sinon} \end{cases}$$

avec m et n deux entiers.

- (6) Déterminer à la main ce que vaut $morris(1, 0)$.
- (7) Ecrire un algorithme récursif pour calculer cette fonction.
- (8) Utiliser cet algorithme pour calculer $morris(1, 0)$. Commenter