Travaux Pratiques d'algorithmique n°3

Cours d'Informatique de Deuxième Année

—DEUG Sciences M3-MI—

Récursivité

L'objet de cette séance est d'introduire la notion de récursivité. Dans un premier temps, on s'intéressera à plusieurs exemples basiques qui permettent de se familiariser avec cette technique. Ensuite, on étudiera un problème plus technique : les tours de Hanoï.

▸ **Exercice 1.** *Ecrire une fonction récursive qui affiche les N premiers entiers dans l'ordre croissant. Idem dans l'ordre décroissant.*
Que constatez-vous ?

▸ **Exercice 2.** **(Somme et produit récursifs)** *Ecrire deux fonctions qui calculent, de façon récursive, la somme et le produit de deux nombres entiers* `a` *et* `b` *positifs ou nuls.*
Indication : pour calculer la somme, nous disposons des relations suivantes :

$$\begin{cases} SOMME(a,0) = a \ , \\ SOMME(a,b) = SOMME(a,b-1)+1 \ . \end{cases}$$

Evaluer les complexités des algorithmes programmés.

▸ **Exercice 3.** **(Coefficients du binôme)** *On sait que les coefficients du binôme C_n^p définis par :*

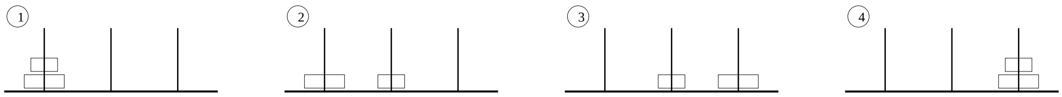
$$\begin{cases} C_n^p = \frac{n!}{p!\,(n-p)!} & si\ n \geq p \ , \\ C_n^p = 0 & si\ n < p \ , \\ C_n^p = 1 & si\ p = 0 \ , \end{cases}$$

vérifient la récurrence suivante :

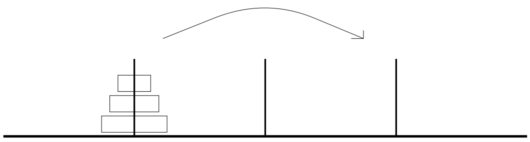
$$C_n^p = C_{n-1}^{p-1} + C_{n-1}^p \qquad pour\ 0 < p < n \ .$$

Ecrire une fonction récursive qui calcule C_n^p .
Cette méthode de calcul vous paraît-elle efficace ?

Exercice 4. *Nous cherchons à résoudre le problème des "tours de Hanoï". Ce problème consiste à déplacer* `n` *disques d'un piquet de départ à un piquet d'arrivée en utilisant un troisième piquet. La seule contrainte tient au fait qu'il est interdit de poser un disque sur un disque plus petit que lui.*



Résolution du problème pour $\mathtt{n} = 2$.



Exemple du problème pour $\mathtt{n} = 3$.

1. *résoudre "à la main" l'algorithme pour* $\mathtt{n} = 3$ *;*

2. *en déduire un algorithme récursif de résolution du problème ;*

3. *écrire la fonction correspondant à cet algorithme : la fonction devra afficher la suite des déplacements des disques ; elle retournera le nombre de déplacements nécessaires.* `int Hanoi(int n, char dep, char tmp, char arr)` *où* `n` *est la hauteur de la tour, et* `dep,tmp,arr` *sont les noms (un seul caractère) des trois piquets de départ, de transition, et d'arrivée ;*

4. *donner enfin, la complexité en temps de votre algorithme.*