— Algorithmique & Programmation – D.E.S.S. Imoi —

# **TD 3 :** Fonctions et récursivité

## Exercice 1 : Définition de fonctions

Reprenez les exercices du TD 2 en utilisant des fonctions, afin que le programme principal ne consiste qu'en appels de fonctions. Dans un premier temps, on pourra se limiter au calcul du PGCD (ex. 2.1), de la suite de Fibonacci (ex. 2.2), de la puissance (ex. 2.5) et de la méthode de Hörner (ex. 3.3). Ainsi, il faudra généralement définir une fonction de lecture de données, une fonction de calcul et une d'affichage du résultat.

## Exercice 2 : Récursion

1. Reprenez les programmes de l'exercice 1 de manière à ce que les fonctions de calcul soient récursives, et non itératives. Comparez les temps de calcul.
   En particulier, on pourra noter que le calcul récursif du $35^{\text{ème}}$ terme de la suite de Fibonacci est déjà très long, alors que le calcul itératif reste très rapide longtemps après ce terme (jusqu'au dépassement de capacité pour le $75^{\text{ème}}$ terme).

2. Programmez différentes méthodes pour calculer $C_n^p$ (on rappelle que $C_n^0 = C_n^n = 1$, $\forall n \geq 0$, et que $C_n^p = n!/(p!(n-p)!) = C_{n-1}^p + C_{n-1}^{p-1}$, $\forall n > p > 0$). Comparez leur efficacité en terme de vitesse et de place mémoire.

3. Le jeu des tours de Hanoï consiste à déplacer un ensemble de pièces d'un pilier à un autre en en utilisant untroisième, et ce de manière à ne jamais placer une pièce sur une autre plus petite.
   Programmez ce jeu, en utilisant un double appel récursif. On pourra proposer deux versions d'affichage textuel, une numérique et une graphique.