Elements d'algorithmique

TP 7 : Fonctions récursives

1 Principe

Qu'est-ce qu'un oignon ? Soit rien, soit un oignon entouré d'une peau d'oignon ! Cette définition est dite *récursive*, au sens où il s'agit d'une définition qui fait référence à elle-même. En mathématique, la notion de récursivité apparaît à travers la notion de récurrence. Soit par exemple la définition suivante de factorielle :

$$U_n = \begin{cases} 1 & \text{si } n = 0 \\ n.U_{n-1} & \text{sinon} \end{cases}$$

La traduction Pascal de cette fonction est la suivante :

```
FUNCTION factorielle(n : INTEGER) : INTEGER;
BEGIN
   IF n = 0 THEN factorielle := 1
   ELSE factorielle := n * factorielle(n - 1);
END;
```

2 Exercices

2.1 Exercice 1

On rappelle les propriétés du PGCD de deux nombres a et b :
– si $a > b$, pgcd(a,b) = pgcb(b,a-b)
– si $a = b$, pgcd(a,b) = a = b
Écrire une fonction récursive qui calcule le PGCD de deux entiers.

2.2 Exercice 2

Écrire une fonction récursive qui calcule C_n^p où $0 \leq p \leq n$, en utilisant la propriété suivante :

$$C_n^p = \begin{cases} 1 & \text{si } p = 0 \text{ ou } p = n \\ C_{n-1}^p + C_{n-1}^{p-1} & \text{sinon} \end{cases}$$

2.3 Exercice 3

Écrire une fonction récursive qui calcule x^n pour tout $n \geq 0$. Mettez à profit la remarque suivante :

$$x^n = \begin{cases} x.x^{n-1} & \text{si n est impair} \\ (x^2)^{n/2} & \text{si n est pair} \end{cases}$$