T.P. 03

Programmation Récursive : 50 nuances de Fibonnaci

■ ■ ■

Ce TP va s'attarder deux heures sur la suite F définie par $F_0 = F_1 = 1$ et $F_{n+2} = F_{n+1} + F_n \dots$

Partie I : Montée en charge

Exercice 1.

* Ecrire la fonction récursive naïve qui calcule F_n.
* Quelle est le nombre d'additions nécessaires à calculer F_n par cette méthode ? A quelle classe de compléxité cela appartient t'il ?

Exercice 2.

* Ecrivez une fonction qui prend en argument une paire (F_n, F_{n+1}) et renvoie la paire (F_{n+1}, F_{n+2}).
* Déduisez en une implementation linéaire en nombres d'addition de F.

Partie II : Mémoisation

Ici, se suivenir des deux dernières valeurs suffit. C'est néanmoins l'occation de découvrir une méthode de programmation plus générale. Pour ne pas réeffectuer des calculs déja fait, on peut les stocker dans une table juste après les avoir calculer.

Exercice 3. En utilisant le module `map` de camllight (référez vous à sa documentation) et en particulier les fonctions `map__add: 'a -> 'b -> ('a, 'b) map__t -> ('a, 'b) map__t` et `map__find:'a -> ('a, 'b) map__t -> 'b`, écrivez une fonction *memo_fib* telle que `memo_fib (map__empty eq__compare) n` renvoie une paire formée de F_n et de la table des valeurs F_k que la fonction a eu besoin de calculer (mais n'a calculé qu'une fois !).

Partie III : « Rien à voir »

Exercice 4. (Exponentiation rapide) On sait que $x^0 = 1$, $x^{2k} = (x^2)^k$ et $x^{2k+1} = x(x^2)^k$. En déduire une fonction récursive pour calculer x^n (x et n donnés).
Indication : étudier la parité de n.

Exercice 5. Nous allons écrire une bibliothèque super élémentaire de manipulation des matrices 2,2 représentée par des paires de paires d'entiers : `type mat = ((int * int) * (int * int))`

* Ecrire `addition : mat -> mat -> mat`
* Ecrire `multiplication : mat -> mat -> mat`
* En déduire une fonction optimisée `puissance : mat -> int -> mat`

Partie IV : tout ce tient

Exercice 6.

* Calculez $\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix} \times \begin{pmatrix} F_{n-1} & F_n \\ F_n & F_{n+1} \end{pmatrix}$. Que vaut $\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}^n$?
* Ecrire une fonction qui calcule F_n en temps logarithmique ?
* Que dire de ses résultats à partir de 30 ? Que se passe-t-il ? Trouvez dans la bibliothèque de CamlLight un moyen d'y remedier ?

P. Boutillier & A. Camanes