

TP 7 : De la récursivité et des types

Antoine Frénoy

frenoy@gmx.com

<http://perso.crans.org/frenoy/caml2012.html>

Mardi 27 mars 2012

Exercice 1 : Exponentiation rapide

Soit $x \in \mathbf{R}$ et $n \in \mathbf{N}$. On a la relation suivante :

- Si n est pair alors $x^n = (x^{\frac{n}{2}})^2$
- Sinon $x^n = x * (x^{\frac{n-1}{2}})^2$

1. Écrivez une fonction récursive `ExpoRapide` prenant deux arguments `x` et `n` et renvoyant `x` exposant `n` en utilisant la méthode décrite ci-dessus.
2. Quel est son avantage par rapport à la méthode « naïve » vue la semaine précédente ?

Exercice 2 : Tours de Hanoi

On dispose de trois tiges sur lesquelles on empile des disques de diamètres deux à deux distincts. On peut déplacer un disque d'une pile à une autre, avec les règles de déplacement suivantes :

- On ne peut déplacer qu'un disque à la fois.
- On ne peut prendre un disque que s'il est en haut de sa pile.
- On ne peut poser un disque que sur un disque (strictement) plus grand que lui ou sur une pile vide.

On part d'une position dans laquelle les n disques sont empilés sur la tige n°1 du plus grand au plus petit (le plus grand en dessous, le plus petit au dessus). On cherche à arriver en un nombre minimal de déplacements à une position dans laquelle les n disques sont empilés sur la tige n°3 du plus grand au plus petit.

Pour mieux comprendre, voici un exemple dans le cas où $n = 3$ disques :

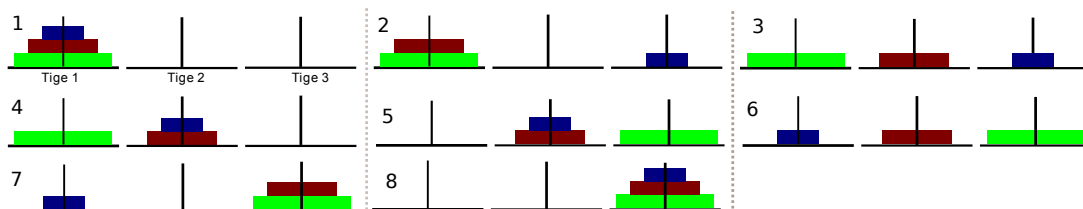


FIGURE 1 – Tours de Hanoi : exemple avec trois disques

1. Définir un type `tige` permettant de représenter les trois tiges.
2. Puisqu'on ne peut déplacer que le disque en haut d'une pile, pour représenter un mouvement il suffit de préciser la tige de départ et la tige de destination. Définir un type `mouvement`.

3. Écrivez une fonction `hanoi` prenant un argument `n` et renvoyant une liste de mouvements qui résout le problème dans le cas où il y a `n` disques.

Indice : *Écrivez une fonction récursive `hanoi_g` plus générale, qui prend en arguments `x` (de type `int`), `i`, `j` et `k` (de type `tige`), et qui renvoie la liste des mouvements nécessaires pour déplacer les `x` disques les plus en haut de la pile `i` vers le haut de la pile `j` (en conservant leur ordre), où `k` est la pile restante (la pile qui n'est ni `i` ni `j`).*