Programmation fonctionnelle et logique
TD1 : récursivité et polymorphisme

Suite de Fibonacci

La suite de Fibonacci est définie comme suit :

$$\mathrm{Fib}(n) = \begin{cases} 1 \text{ si } n = 0 \\ 1 \text{ si } n = 1 \\ \mathrm{Fib}(n-1) + \mathrm{Fib}(n-2) \text{ sinon.} \end{cases}$$

1. Écrire une fonction récursive qui calcule $\mathrm{Fib}(n)$. Écrire deux versions de cette fonction : une qui utilise des tests (expressions booléennes) et l'autre le filtrage.
2. Écrire une fonction récursive qui calcule $(\mathrm{Fib}(n), \mathrm{Fib}(n-1))$. L'utiliser pour écrire une fonction calculant $\mathrm{Fib}(n)$.
3. Quelle est la complexité des différentes fonctions ?

Maximum sur un tableau

Écrire une fonction qui prend en entrée un tableau et renvoie son élément maximal.

Tri sur les entiers

1. Écrire une fonction qui prend en entrée un tableau t dont les $n-1$ premiers éléments sont triés et renvoie ce même tableau mais avec les n premiers éléments triés.
2. Au moyen de la fonction précédente, écrire une fonction de tri par insertion d'un tableau d'entiers.

Tri polymorphe

Écrire une fonction de tri polymorphe sur tableau.