

# Programmation multi-cœurs — TD 5

## Algorithmes non-bloquants

**Exercice 1.** *Skip list*. Présentation selon Wikipédia : « Une skip list se présente comme une amélioration d’une liste chaînée triée. Elle contient des pointeurs supplémentaires vers l’avant, ajoutés de façon aléatoire, de sorte que la recherche dans la liste puisse “sauter” (to skip en anglais) de nombreux éléments.

La skip list est organisée en couches. La couche la plus basse est simplement une liste chaînée standard. Chaque couche supérieure  $i + 1$  est une “voie rapide” pour parcourir les couches inférieures  $0, \dots, i$ . Un élément présent sur la couche  $i$  a une probabilité fixée  $p$  de faire partie de la couche  $i + 1$ . En moyenne, chaque élément apparaît dans  $\frac{1}{1-p}$  listes, et l’élément le plus haut (souvent un élément factice plus petit que tous les autres) apparaît dans  $\mathcal{O}\left(\log_{\frac{1}{p}}(n)\right)$  couches. »

La figure ci-dessous représente l’une des structures possibles encodant l’ensemble  $\{0, 10, 25, 30, 50\}$ . Proposez une implémentation linéarisable et lock-free de la skip list. On prendra  $p = \frac{1}{2}$ , et on pourra fixer la taille de la couche la plus haute à 32.

