

TD 6 : Opérations de listes et complexité

NB : quelques exercices de ce TD proviennent du livre “Algorithm Design with Haskell” de Richard Bird et Jeremy Gibbons (qui est fortement recommandé)

1 La Complexité de la Concaténation

L’objectif de cet exercice est d’étudier de plus près la complexité de l’opération `++` et l’impact de l’utilisation de `foldr` ou `foldl`. Nous commençons avec une réimplémentation de `++` et puis on l’utilise pour réimplémenter la fonction `concat`.

1. Donner une implémentation de la fonction `++`.
2. Donner une implémentation de la fonction `++` en utilisant `foldr`.
3. Si on applique l’opération `++` sur deux listes `xs`, `ys` de taille n , m respectivement, quelle est sa complexité en tant que fonction de n , m ? Vous pouvez supposer que l’opération `(:)` est de complexité $O(1)$.
4. Donner une implémentation de la fonction `concat` en utilisant `++` et `foldr`.
5. Donner une implémentation de la fonction `concat` en utilisant `++` et `foldl`.
6. Quelle est la complexité de chacune de vos implémentations? Laquelle est préférable?

2 Steep Lists

On dira qu’une liste d’entiers est “steep” (raide) si on a la propriété suivante : pour chaque élément x de la liste, x est strictement supérieur à la somme de tous les éléments suivants. Par exemple `[4, 2, 1]` et `[23, 11, 8, 2]` sont steep, alors que `[5, 4, 2]` et `[125, 15, 11, 5]` ne le sont pas.

1. Programmer une fonction qui teste si une liste donnée est steep.
2. En utilisant `foldr`, programmer une version améliorée de votre fonction de complexité $O(n)$, où n est le nombre d’éléments de la liste. (Vous pouvez supposer que les opérations arithmétiques ont complexité $O(1)$).

3 Foldr vs Foldl

Dans cet exercice on va comparer un autre aspect de `foldr` et `foldl` : parfois les deux fonctions retournent le même résultat pour des listes finies mais ne sont pas équivalentes quand on passe comme argument une liste infinie. L’exemple qu’on va utiliser est la fonction `takeWhile`.

1. Donner une implémentation de la fonction `takeWhile` en utilisant `foldr`. Quelle est la complexité de votre implémentation?
2. Donner une implémentation de `takeWhile` en utilisant `foldl`. Quelle est la complexité de votre implémentation?
3. Que se passe-t-il si on passe comme argument une liste infinie?