

Description de notre experience pour les points suivants :

## **1 Parfaire sa connaissance de Haskell**

Problèmes, surprises, choix, options rejetées, etc.

## **2 Lire et comprendre la donnée**

Problèmes, surprises, choix, options rejetées, etc.

## **3 Lire, trouver et comprendre les parties importantes du code fourni**

Problèmes, surprises, choix, options rejetées, etc.

## **4 Compléter le code fourni**

Problèmes, surprises, choix, options rejetées, etc.

Nous pensions qu'une approche en largeur plutôt qu'en profondeur était une bonne stratégie telle que suggérée dans la donnée. Après avoir noté par écrit le problème dans son ensemble, nous avons essayé de nous attaquer à *s2l* pas à pas de manière systématique et logique, en regardant tous les cas possibles. Nous en sommes convenu que les éléments les plus simples pour débiter étaient les booléens et les structures conditionnelles (if, then, else). Pour chaque cas de *s2l*, nous faisons immédiatement le correspondant dans la fonction *eval* pour pouvoir faire des tests et debugger en cours de route.