



UNIVERSITÉ DE FRIBOURG  
UNIVERSITÄT FREIBURG

# Programmation fonctionnelle et Logique Haskell (Listes et récursivité)

Yael Iseli & A. Schaller  
14-8215-24 & 16-896-375  
Prof.: Stéphane Le Peutrec

30 septembre 2020

## Série 2

À travers les 4 exercices de la deuxième série, nous devons réaliser plusieurs implémentations de fonction en utilisant ou non les motifs.

Comme plusieurs exemples sont donnés pour vérifier le comportement des fonctions, nous étions intéressés de découvrir les solutions de tests disponibles dans l'écosystème *Haskell*.

Rapidement, nous avons découvert *hspec*<sup>1</sup>. Ses avantages sont :

- il utilise une structure similaire à de nombreux autres frameworks de tests, tels que *Jasmine*<sup>2</sup>, *mocha*<sup>3</sup> et pleins d'autres.
- une prise en main qui paraît rapide et simple avec *cabal*<sup>4</sup>.

Pour le dernier point, l'environnement du packages/librairies manager *cabal* semble varier entre les systèmes. Dans tous les cas, c'est une question d'un argument en plus un moins et de l'installation d'un, deux packages supplémentaires.

Dans tous les cas, nous sommes arrivé à pouvoir lancer les tests sur nos implémentations sans devoir configurer un projet *cabal* complet.

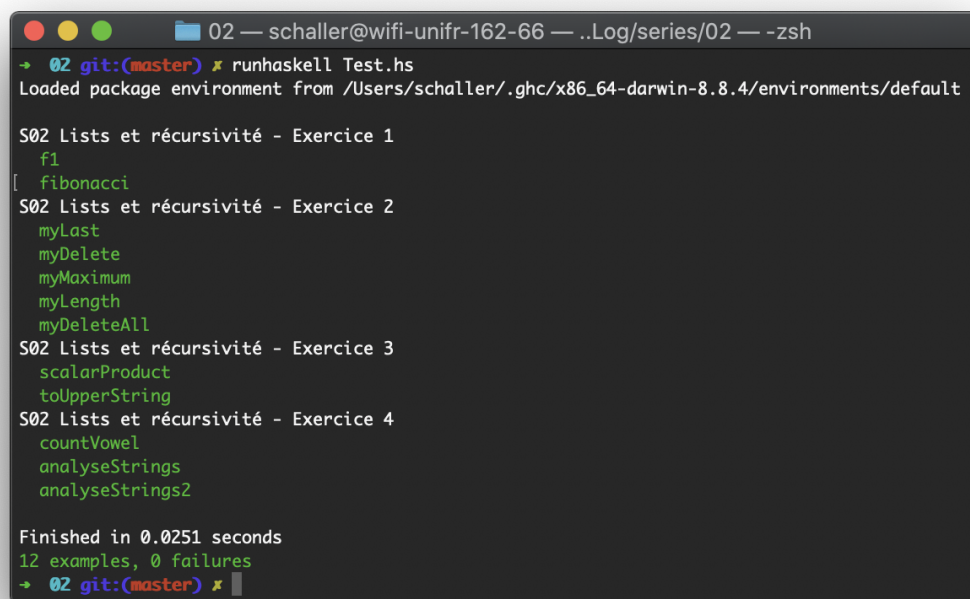
---

1. <https://hspec.github.io/> (visited on sept. 2020)  
2. <https://jasmine.github.io/> (visited on sept. 2020)  
3. <https://mochajs.org/> (visited on sept. 2020)  
4. <https://www.haskell.org/cabal/> (visited on sept. 2020)

### Code 1 Installation du package *hspec*

```
cabal update && cabal install hspec
# dans certains cas, l'option --lib est nécessaire
# il est également possible de devoir installer 'QuickCheck'
# et 'base'
```

Finalement, nous pouvons exécuter nos tests que nous déclarons dans le `main` avec la commande `runhaskell`. Le résultat obtenu dans le terminal est présenté dans la Figure 1.



```
02 — schaller@wifi-unifr-162-66 — ..Log/series/02 — -zsh
→ 02 git:(master) ✕ runhaskell Test.hs
Loaded package environment from /Users/schaller/.ghc/x86_64-darwin-8.8.4/environments/default

S02 Lists et récursivité - Exercice 1
  f1
[ fibonacci ]
S02 Lists et récursivité - Exercice 2
  myLast
  myDelete
  myMaximum
  myLength
  myDeleteAll
S02 Lists et récursivité - Exercice 3
  scalarProduct
  toUpperString
S02 Lists et récursivité - Exercice 4
  countVowel
  analyseStrings
  analyseStrings2

Finished in 0.0251 seconds
12 examples, 0 failures
→ 02 git:(master) ✕
```

FIGURE 1 – *hspec* tests results

## Accès au code source

Comme convenu, nous partageons le code source des séries sur notre *Github* à l'adresse :

— <https://github.com/schalleralala/MSc-ProgFunLog>