

IFT 2035

Travail pratique 1 - Interpréteur Slip

Rapport

Marie-Anne Prud'Homme-Maurice (Ajout Matricule)
Olivier Guénette (20154866)

21 octobre 2021

Une sorte de Lisp

Dans le cadre du cours IFT 2035. Il nous a été demandé de concevoir une sorte d'interpréteur Lisp en utilisant le langage fonctionnel Haskell. Le travail a pour but d'implanter une fonction qui finalise l'analyse syntaxique de l'expression fournie ainsi que de la fonction eval qui permet d'évaluer celle-ci.

Ce rapport décrit notre processus d'analyse, les problèmes rencontrés, les décisions prises et notre expérience durant la création de cet interpréteur.

Analyse et compréhension de l'énoncé

Comme dans tout travail la première étape consistait à comprendre la tâche à réaliser. Sans le cacher, Haskell et Lisp sont des nouveaux langages pour nous. Juste ce fait rend la tâche du projet plus complexe.

Suite à plusieurs lectures, nous avons commencé à reconnaître des similitudes entre la structure de Slip et Haskell.

Problèmes rencontrés

Problèmes de compréhension

À faire (Marie)

Élimination dynamique du sucre syntaxique

Le premier problèmes que nous avons rencontrés était le manque de dynamisme dans l'analyse des Scons.

À faire (Marie)

Évaluation des Lfn

L'évaluation des Lfn est quelque chose que nous avons résolue vers la fin du travail pratique.

À Faire (Olivier)

Solutions rejetées et choisies

Implentation initial de eval slet et dlet

Lorsque nous étions rendu à l'évaluation des slet et des dlet dans la fonction eval, la première solution trouvée était de décortiquer l'information du let en plusieurs parties. Nous avons donc implémenté des fonctions auxiliaires permettant de trouver toutes les variables définies dans le let ainsi que l'expression final à évaluer.

Cette méthode fonctionnait particulièrement bien pour les expressions n'incluant pas de récursion, car cette solution remplaçait les variables du let par leur valeur. Ainsi, si une expression faisait référence à une variable du let, le système ne s'en souvenait pas.

Clairement cette méthode ne permettait pas l'évaluation de toutes les expressions de slet et de dlet. En corrigeant ce problème, nous nous sommes rendu compte que la source du problème venait en effet du manque d'évaluation en Vfn des Lfn. Suite à une bonne évaluation des Lfn en Vfn, cela nous a permis

d'ajouter les variables du let dans leur environnement respectif pour ensuite évaluer l'expression finale.

Implentation initial de eval pour Lfn

À faire (Olivier)

Conclusion

À faire