

Calculabilité

Rapport de travaux pratiques sur un éliminateur d'expressions complexes

Glen Le Baill

Antoine Bouffard

Introduction

Un éliminateur d'expressions complexes permet de transformer un AST `a1` comprenant des expressions ayant plusieurs constructeurs en un AST `a2` ne comprenant que des expressions ayant au maximum un seul constructeur, tout en préservant la sémantique.

Un éliminateur d'expressions complexes peut donc être une étape de pré-compilation permettant de simplifier l'interprétation du programme `a1` afin de pouvoir y apporter des optimisations lors de la future étape de compilation principale.

Programme

Structure du programme

Dans le fichier principal `while1cons.scala`, le squelette du projet qui nous a été fourni comporte une structure en plusieurs parties. Les plus importantes sont les parties dédiées au traitement des Expressions, des Commandes et des Programmes du langage While.

La partie Expressions, Commandes et Programmes ont respectivement pour rôles de décomplexifier les Expressions, les Commandes et les Programmes du langage While.

Méthodes

Chacune des sections mentionnées ci-dessus sont composées d'une ou plusieurs méthodes.

Nous avons créé une méthode auxiliaire dans la section Expressions pour faciliter la conversion d'une `Variable` en `VarExp`.

La méthode principale de l'application est `while1ConsProgr()`, qui est la racine de l'arbre des méthodes du programme principal. Ainsi la fragmentation des traitements à effectuer en sections et méthodes permet une meilleure organisation du code et une maintenance plus simple grâce à cette modularité.

Le reste des méthodes n'a pas besoin selon nous d'explications supplémentaires.

Difficultés rencontrées

Les méthodes `while1ConsExprV()` et `while1ConsCommand()` nous ont posé quelques difficultés. En effet le fait de jongler entre la syntaxe de Scala et de While n'a pas été évident, et parfois les traitements à implémenter nous ont paru assez abstraits aux premiers abords.

Cependant avec de la persévérance et de la réflexion (et les tests) nous sommes parvenus à compléter ce programme correctement.

Conclusion

La réalisation de ce projet nous a permis de mettre en pratique le cours de Calculabilité et de poursuivre dans le thème des traitements à apporter à un AST du langage While.

Par ailleurs, il nous a semblé que le fait d'éliminer les expressions complexes en créant beaucoup de variables pouvait cependant avoir pour conséquence de faire exploser les besoins en mémoire de l'application.