

Projet : Compilateur SLIP vers NBC  
IHDC B332 - Théorie des langages : Syntaxe et sémantique

Nicolay Matthias  
Evrard Cédric

UNamur

# 1 Introduction

Dans ce rapport, nous allons présenter les différents aspects techniques du projet de syntaxe et sémantique ainsi que les choix techniques que nous avons effectués lors de la création du compilateur SLIP.

## 2 Démarche générale

Notre groupe était composé de deux personnes. La démarche que nous avons adoptée a été de laisser Cédric commencer à travailler en chaque début de partie afin de créer l'architecture de l'application ainsi que la structure des différents packages. Une fois que cette structure était créée, une répartition était faite du travail.

Il a parfois été difficile de découper la réalisation du travail de manière équitable. En effet, une dépendance assez importante dans la grammaire de la partie *expression droite* a fait que celle-ci devait souvent être réalisée avant tout le reste et conditionnait énormément la structure du projet.

Avant la fin de la phase 2, un gros travail de refactoring a été effectué afin de rendre le code plus lisible et d'en faciliter la maintenance et l'écriture du code. Cela nous a aussi permis de travailler plus rapidement sur la phase 3 en reprenant le même principe de structure.

De manière générale, la réalisation du projet c'est bien passée même si nous aurions pu mieux répartir le travail entre nous. Malheureusement, la charge de travail en dehors du projet ainsi que le commencement des phases de façon un peu tardif a fait que nous avons dû aller assez vite pour réaliser chacune d'elle et donc se répartir le travail sur celui qui pouvait avancer le plus rapidement.

## 3 Structure de la table des symboles

## 4 Architecture du compilateur

## 5 Conclusion

Nous sommes assez content de notre compilateur, particulièrement l'architecture de celui-ci qui, suivant le contexte ANTLR, redirige vers la partie qui s'occupe de la gestion de ce contexte. Cela nous a permis de facilement réutiliser nos codes et de gérer la plus part des parties du compilateur de manière indépendante.

Une faiblesse de celui-ci pourrait être l'utilisation importante de *instanceof*. Nous aurions pu regarder à une alternative suivant les informations qui étaient contenues dans le contexte ANTLR sur lequel on travaillait pour voir si celui-ci ne contenait pas une information qui nous aurait permis d'éviter l'*instanceof*.

Une amélioration que nous aimerions apporter est un *refactoring* de la grammaire. En effet, nous avons pu remarquer lors de l'écriture du compilateur que nous avions à passer par certains contextes qui ne faisaient rien. Nous pourrions donc retirer certains éléments de la grammaire pour améliorer celle-ci.

Le projet nous a appris l'écriture d'une grammaire assez complète ainsi que la rigueur dans la compilation d'un code vers un autre. De fait, il y a vite beaucoup d'éléments à

gérer dans le code NBC de sortie.

Une difficulté du projet a été de ne pas pouvoir tester le code compilé sur un vrai robot LEGO® car le simulateur *Bricx* n'est pas complet et donc nous ne pouvions savoir si ce que nous générions était correcte et correspondait au comportement attendu.