

# Rapport

Alessandro Pepegna & Loïc Filippi

Pour la table des symboles, nous utilisons une table de hachage. On y stocke des informations sur les identifiants mais elle n'est pas exploitée lors de la compilation, notamment pour récupérer ces informations. Elle n'est pas structurée en portée, cela amènerait donc les problèmes qu'on connaît lors de la définitions des identifiants.

Concernant les tests, tous les fichiers sont correctement analysés mais le seul code généré à être correct dans le langage destination est "variables.c".

La difficulté majeure a été de produire les instructions en code à trois adresses.

Des changements ont été apportés à la grammaire frontend:

- L'incrément : '+' car présent dans les fichiers tests.
- Les décalages binaires : '<<' et '>>' car présents dans les fichiers tests.
- Factorisation gauche pour le branchement conditionnel (mais le conflit décalage/réduction n'a pas été réglé pour autant)
- Moins unaire : '-' pour prendre en compte les constantes négatives.

Pour l'aspect technique, nous avons acquis une certaine maîtrise des outils lex et yacc. Nous avons appris à gérer correctement les *tokens*, à modifier la grammaire lorsque cela était nécessaire et à introduire des routines sémantiques. La génération de code s'est avérée compliquée, mais cela nous a quand même permis de mieux appréhender les fonctionnements internes d'un compilateur.