



Projet du module Compilation  
Master 1 Informatique  
Groupe 1B3M

Maxime DESGRANDCHAMPS  
Baptiste FOREST  
Miguel GERVILLA  
Maxime HOHL

12 janvier 2020

# Table des matières

<b>1</b>	<b>Compte rendu</b>	<b>2</b>
1.1	Travail Réalisé . . . . .	2
1.1.1	Langage Supporté . . . . .	2
1.1.2	Reconnaissance des BLAS . . . . .	2
1.1.3	Paramètres de la ligne de commande . . . . .	2
1.2	Architecture . . . . .	3
1.2.1	Makefile . . . . .	3
1.2.2	Les fichiers . . . . .	3

# 1. Compte rendu

## 1.1 Travail Réalisé

Notre programme permet de passer d'un fichier C à un autre fichier C en passant par un *Abstract Syntax Tree*.

### 1.1.1 Langage Supporté

Le langage supporte toutes les fonctionnalités suivantes :

- Les types int et float
- Les variables classiques et les tableaux
- Les opérateurs +, - (unaires et binaires), \*, /, ++ et -- (prefix et postfix)
- Les structures de contrôle if, else, while et for
- Les appels de fonctions (mais pas la déclaration)
- La fonction printf
- Le symbole return

### 1.1.2 Reconnaissance des BLAS

Faute de temps, on ne reconnait aucun *Basic Linear Algebra Subprograms* (BLAS).

### 1.1.3 Paramètres de la ligne de commande

Le programme reconnait tout les paramètres de la ligne de commande demandés, c'est à dire :

- -version : Affiche le nom du grappe ainsi que ses membres
  - -tos : Affiche la table des symboles
  - -ast : Affiche l'AST avant et après reconnaissance des BLAS
  - -o <name> : Définis le fichier où écrire le code C modifié
- Ainsi que les paramètres supplémentaires suivants :
- -h qui affiche une aide sur comment utiliser le programme

## 1.2 Architecture

### 1.2.1 Makefile

La Makefile supporte les commandes suivantes :

- make : Compile le programme avec les symboles de debug
- make MODE=release : Compile le programme sans les symboles de debug et avec l'option -O2
- make rebuild : Force la re-compilation de tout les fichiers
- make clean : Supprime les fichiers générés
- make mrproper : Supprime les dossier build et doc
- make doc : Génère la documentation
- make test : Lance le programme sur tout les fichiers de test
- make clean\_test : Supprime tout les fichiers générés par les tests
- make mrproper\_test : Supprime le dossier test/out

Tout les fichiers générés se trouvent dans le dossier build sauf la documentation qui se trouve dans le dossier doc et les tests qui se génèrent dans le dossier test/out.

### 1.2.2 Les fichiers

Le programme est composé des fichiers suivants :

- main.c : Appels les différentes partie du programme
- ast.c/.h : Structure de donnée permettant de gérer les AST
- blas.c/.h : Structure de donnée représentant les BLAS
- c\_generator.c/.h : Fonctions permettant de générer un fichier C à partir d'un AST
- c\_grammar.l : L'analyse lexicale du fichier C d'entrée
- c\_grammar.y : L'analyse syntaxique du fichier C d'entrée
- cl\_parameters.c/.h : Fonctions permettant de gérer les paramètres de la ligne de commande
- logging.c/.h : Fonctions permettant d'afficher des messages, des avertissements et des erreurs. On affiche sur stdout ou stderr avec des couleurs
- symbol\_table.c/.h : Structure de donnée permettant de gérer la table des symboles