



**UNIVERSITÉ
DE NAMUR**

**FACULTÉ
D'INFORMATIQUE**

INFOB314 – THEORIE DES LANGAGES

RAPPORT DE PROJET – GROUPE 09

ANNEE 2017 – 2018

**ADEGNON Kodjo
NAZILI Christian
RAJAONA Andry**

TABLE OF CONTENTS

Introduction	2
1. Deroulement du projet	3
1.1 Organisation – Rôle.....	3
1.2 Avancement du projet	4
2. Grammaire	5
Représentation de la grammaire BNF.....	5
Structure de la table des symboles.....	10
Diagramme des classes.....	10
Description.....	10
Architecture du compilateur.....	11
Conclusion.....	12

INTRODUCTION

Dans le cadre du cours de Théorie des langages, nous avons dû créer un compilateur permettant de traduire un fichier «*B314*» en code machine.

Ce compilateur, implémenté en JAVA, accepte 2 types de fichier :

1. Un fichier décrivant le monde du jeu.
2. Un fichier concernant la stratégie du joueur

1. DEROULEMENT DU PROJET

1.1 Organisation – Rôle

Nous avons mis en place, un Dashboard sur GitHub, au lieu de Trello, pour mieux gérer notre projet. Pour la réalisation du projet, nous avons ainsi créé des tâches converties en Issue, permettant, de la sorte, à chacun de choisir sa tâche.

Nous avons mis en place un Git Flow en créant une branche «*develop*», établis certaines règles implicites :

1. Chaque Issue créée doit être bien renseignée sur le contexte (syntaxe, sémantique, test, ...) et contenir une description de la tâche à effectuer.
2. Une nouvelle branche pour chaque Issue avec le nom contenant le numéro de l'Issue et une courte description en mots-clés.
3. Chaque ajout ou changement doit être accepté par les autres membres via un Pull Request sur la branche «*develop*». Aucun merge automatique ne doit être effectuée.
4. La branche «*master*» ne reçoit des commit seulement pour faire un build sur Jenkins.
5. Avant de réaliser un build sur Jenkins, nous avons créé des release avec un système de versioning.

Nous n'avons pas désigné de « chef ». Nous avons, pour chaque remise, choisi une personne comme responsable du repository et de la remise. Celui-ci avait pour tâche de :

- ✓ Créer les tâches-issues à réaliser pour cette remise.
- ✓ S'assurer de leurs bonnes réalisations.
- ✓ Suivre les pull-request, résoudre les éventuels conflits.
- ✓ Accepter et merge les pull-request.
- ✓ Effectuer les release hebdomadaires et les build sur Jenkins.
- ✓ Compléter le rapport.

1.2 Avancement du projet

- **Lundi 19/02 au Samedi 24/02**
 - Mise en place du git flow - Création de la branche develop.
 - Mise en place d'un Project board dans le repo. GitHub au lieu de Trello.
 - Mise en place d'un espace de travail Slack pour communiquer.
 - Création des différentes tâches/issues pour la semaine.
 - Ajout de toutes les tokens et règles pour le lexeur et parseur.
 - Ajout de tests unitaires basiques.
 - Complétion de la méthode `Main.compile()`.
 - Premier release pour build de Jenkins.
 - Début de rédaction du rapport.
- **Lundi 26/02 au Samedi 03/03 :**
 - Création des différentes tâches/issues pour la semaine.
 - Amélioration des tests unitaires sur la grammaire B314.
 - Release pour build de Jenkins.
 - Rédaction du rapport.
- **Lundi 12/03 au Samedi 24/03**
 - Amélioration de la grammaire avec les vérifications de types, ...
 - Release pour build de Jenkins
 - Rédaction du rapport pour la partie - Remise 1.
-

2. GRAMMAIRE

Représentation de la grammaire BNF

Declaration de variables

varDecl : H — ID — AS — type — H

type : H — scalar — H
 H — array — H

scalar : H — BOOL_TYPE — H
 H — INT_TYPE — H
 H — SQR_TYPE — H

array : H — scalar — LBRACK — intVal — COMMA — intVal — RBRACK — H

Plateau de Jeu

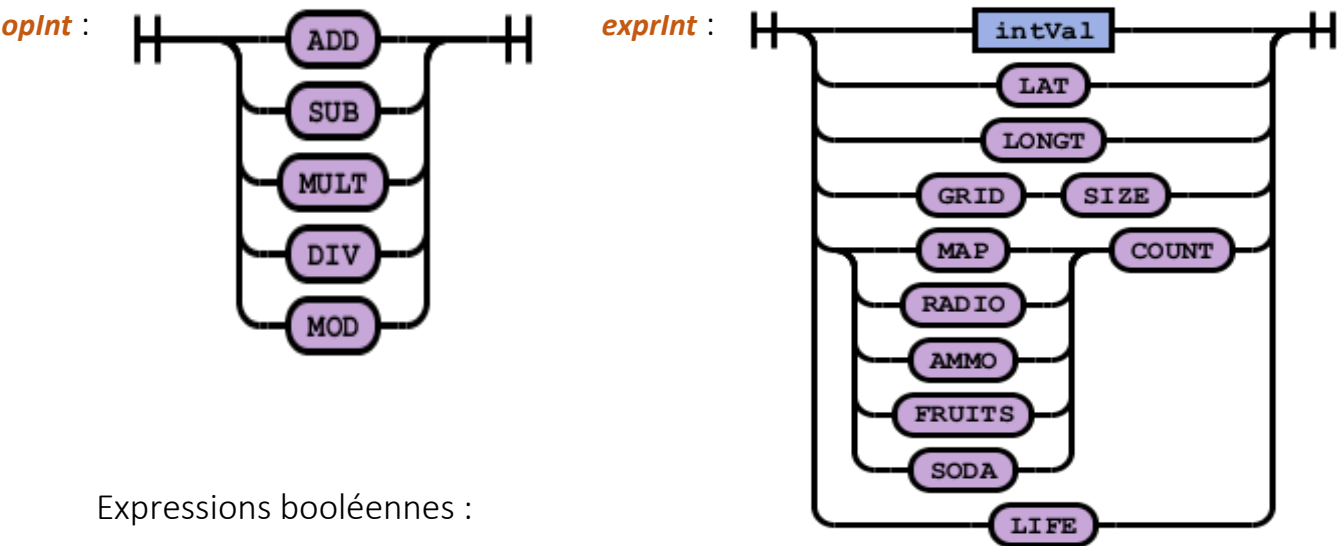
arena : H — ARENA — AS — SQR_TYPE — LBRACK — intVal — COMMA — intVal — RBRACK — H

Expressions droites

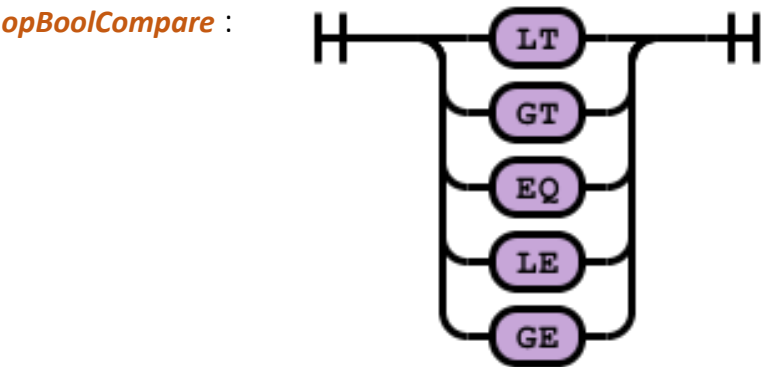
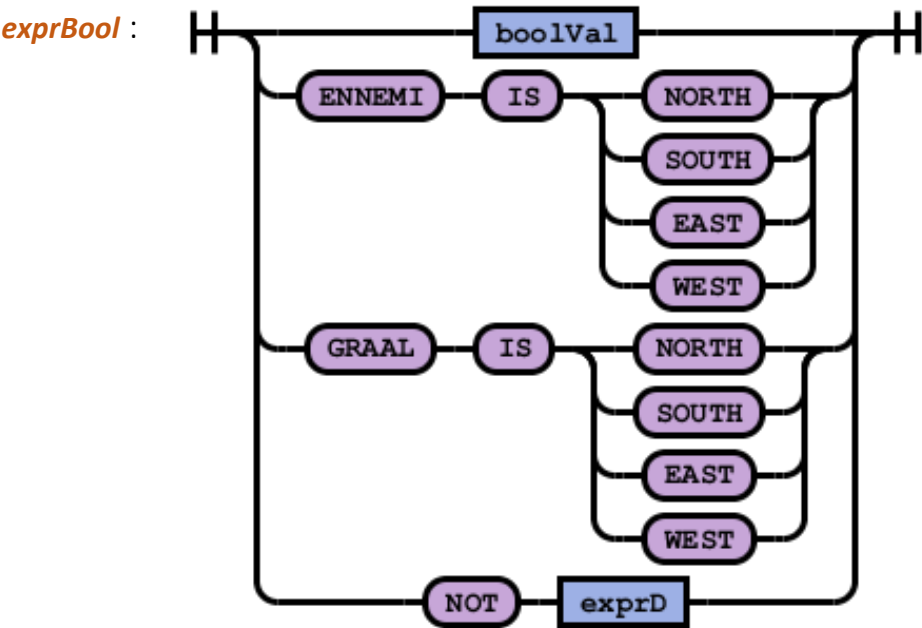
exprD : H — LPAR — exprD — RPAR — H
 H — exprInt — H
 H — exprD — opInt — exprD — H
 H — exprBool — H
 H — exprD — opBool — exprD — H
 H — exprD — opBoolCompare — exprD — H
 H — exprCase — H
 H — exprG — H
 H — exprDFct — H

exprDFct : H — ID — LPAR — exprD — COMMA — exprD — RPAR — H

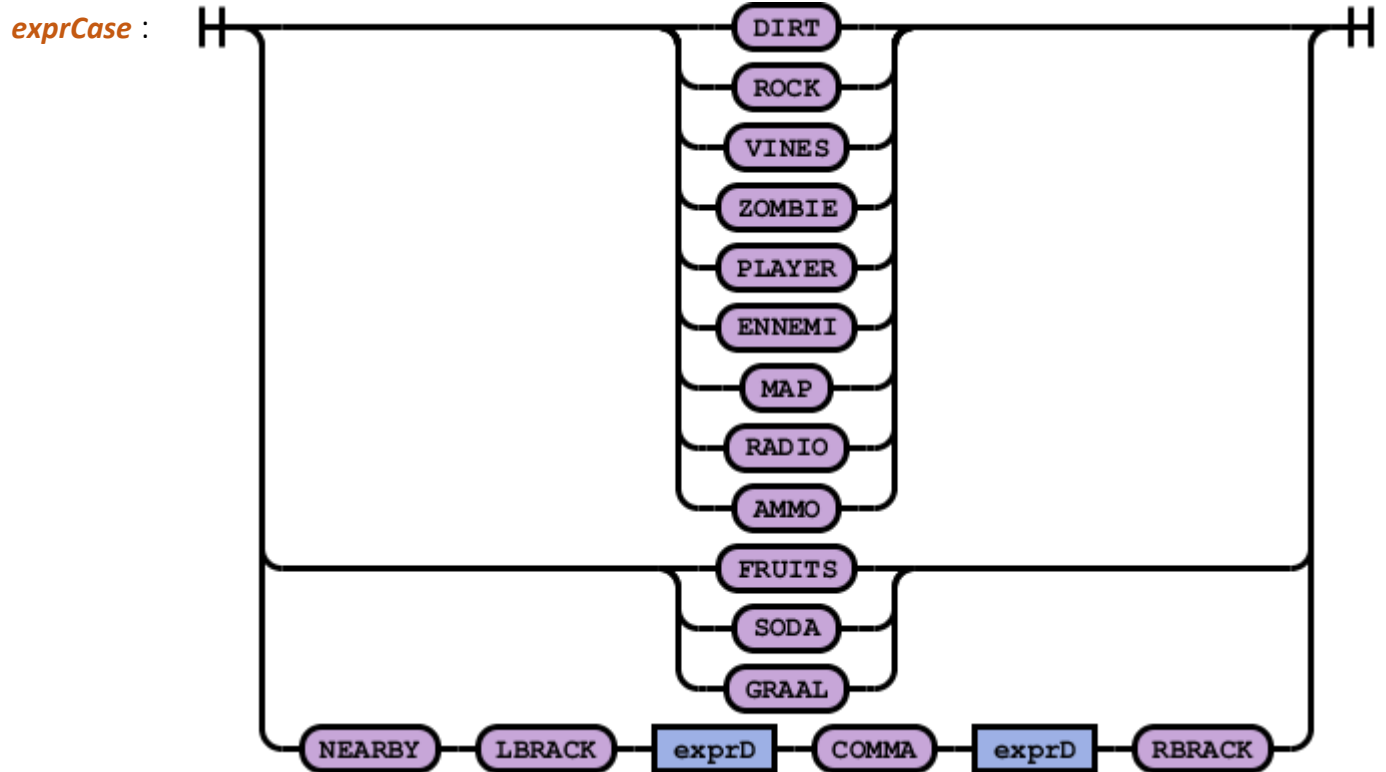
Expressions entières



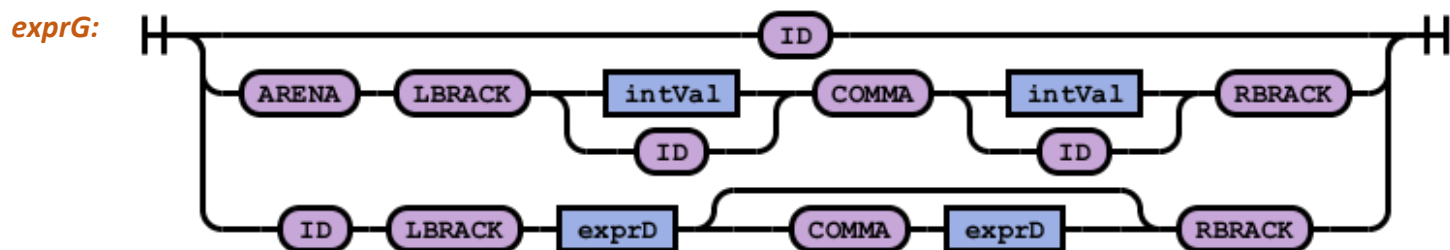
Expressions booléennes :



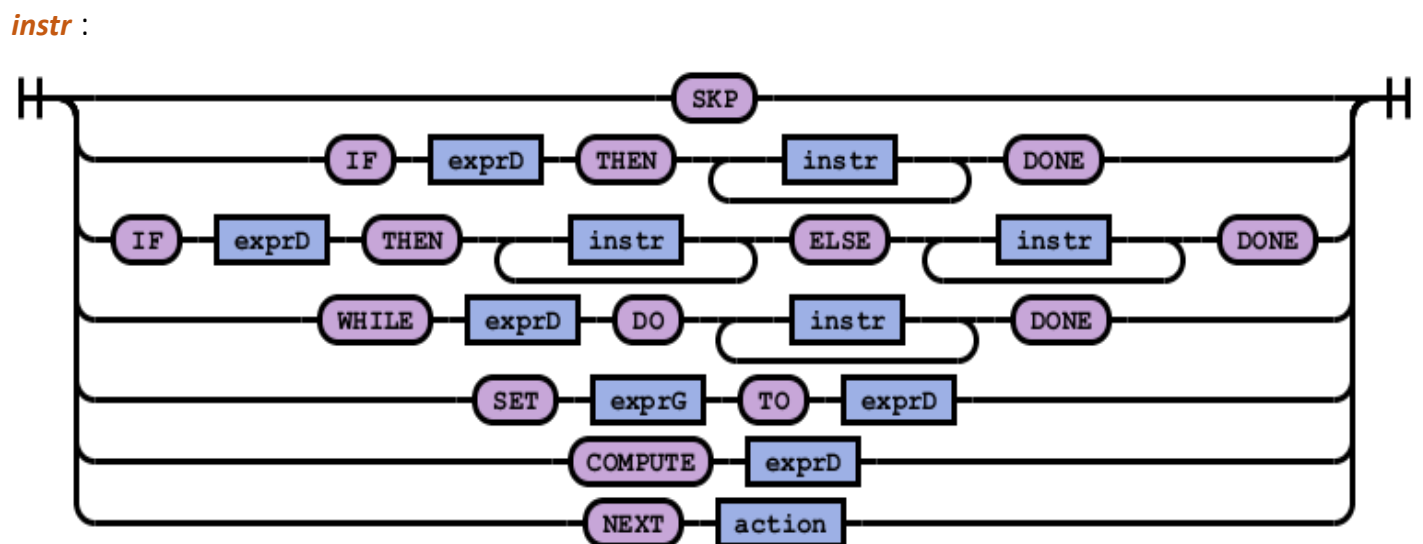
Expressions sur les types de case

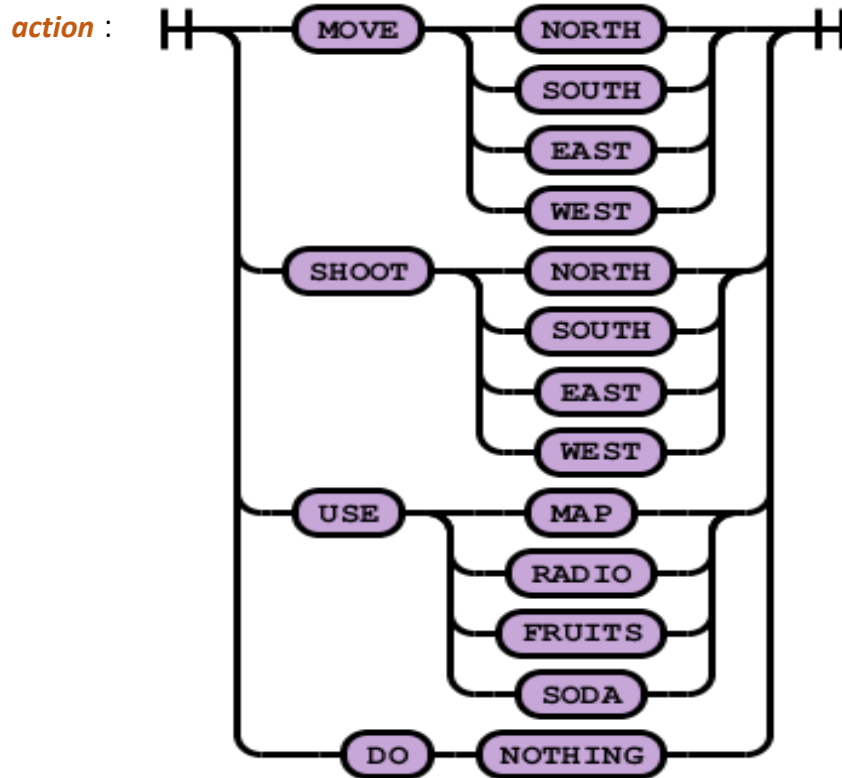


Expressions gauches



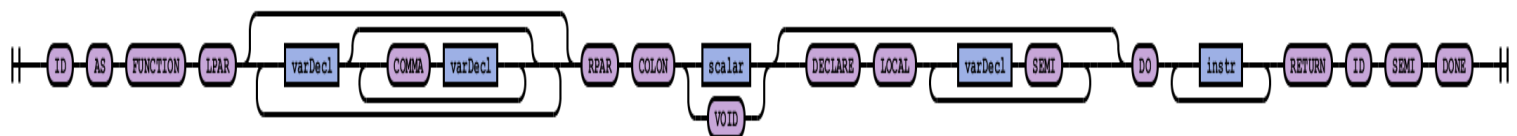
Instructions



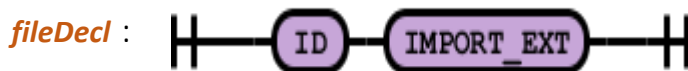


Déclarations de fonctions

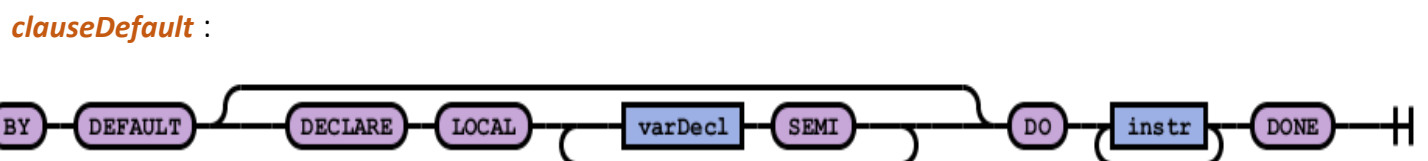
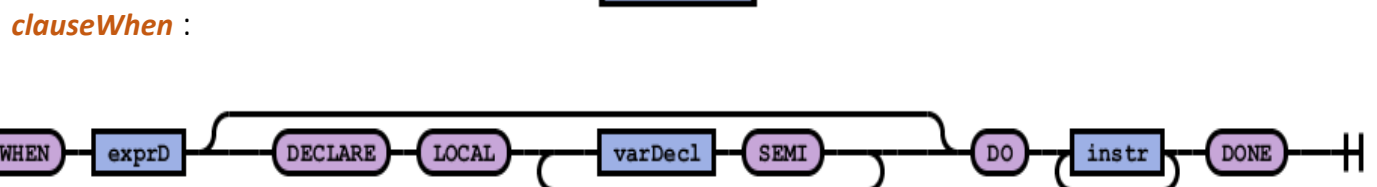
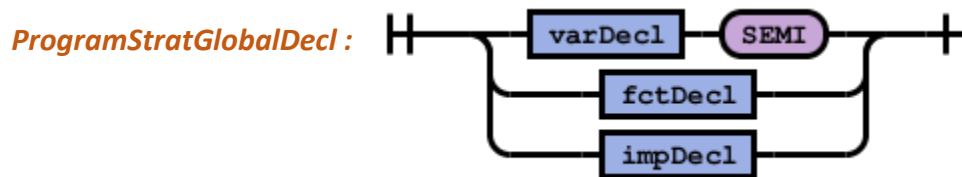
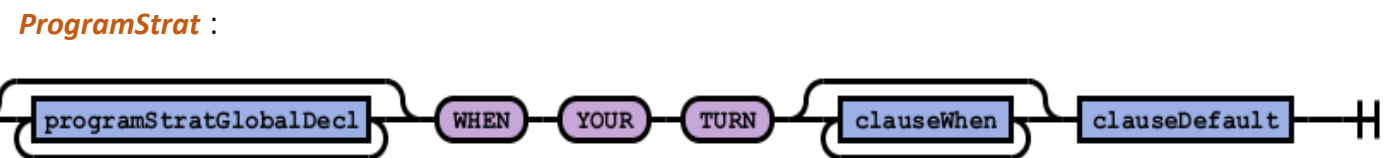
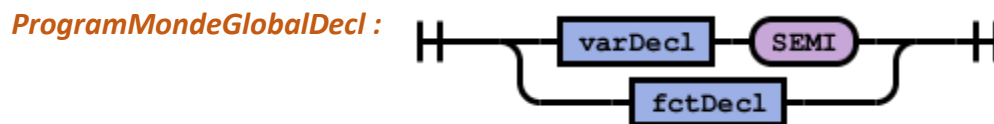
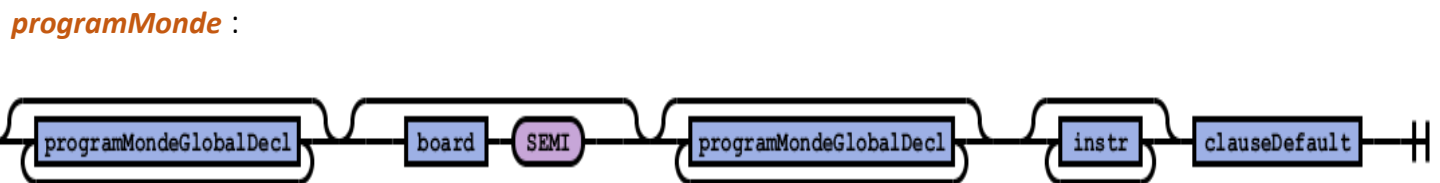
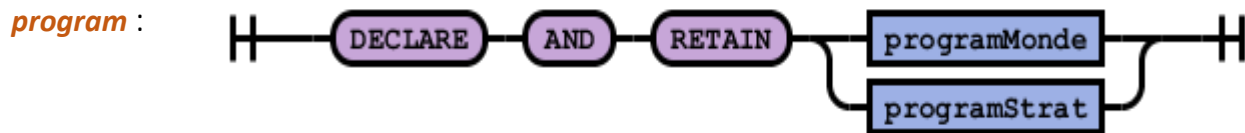
fctDecl :



Déclarations d'import



Programme



STRUCTURE DE LA TABLE DES SYMBOLES

Dans le cadre du cours de Théorie des langages,

Diagramme des classes

(Créer et insérer un diagramme des classes)

Description

ARCHITECTURE DU COMPILATEUR

(Explication des fichiers de grammaire,
de la generation faite par ANTRL,
le parcours de l' AST
et comment le PCode est generé)

(Faire un diagramme de tout ce processus)

CONCLUSION

Ce projet nous a permis de mieux à appréhender le magnifique processus exécuté par un compilateur

***CETTE PAGE CONCLUT
CE DOCUMENT***