

# INFO—F403

## Introduction to language theory and compiling Project Report - Part 3

Berthion Antoine — 566199      Bontridder Milan - 493709

December 23, 2025

### Abstract

This project implement the translation to the language llvm ir pashe of the YaLCC compiler. The parser tree is covered, and each terminal are written to llvm ir syntax. This report present the design and implementation of the compiler.

## 1 Project Overview

The project is organized as follows:

- `src/` : Contains all source files necessary for the **parser implementation**, including `NonTerminal.java`, `ParseException.java`, `Parser.java`, `ParserTree.java`, `Main.java` and the llvm translator : `LLVMIRGenerator.java`.
- `doc/` : Contains the **Javadoc documentation** and this report PDF.
- `test/` : Contains example **YaLCC source files** and unit tests implemented using **JUnit**.
- `dist/` : Contains the executable JAR file `part2.jar`.
- `more/` : Contains other supporting files, including the project specification document.

The main **workflow** of the project can be summarized as:

YaLCC source file (.ycc) → LexicalAnalyzer → sequence of lexical tokens → Parser →  
Parse Tree → llvm code

## 2 Parcouring the Parse Tree

The for each non terminal, there is a function dispatching and writing llvm code in a string. The tree is read from the top to each leaf in a recursive way.

### 2.1 Error in the parse tree

During the creation of **LLVMIRGneerator**, we discovered that the part 2 of the projet missed some non-terminal. This led to a temporary solution of testing more lexical unit and non-terminal than normaly necessary. The llvm code is compiling and running. The full correction requires refactoring LLVMIRGenerator and rethinking path through the tree.

### 3 Use of LLMs

In this brief section, we discuss the use of large language models (LLMs) in the context of this project. No LLM was used for the implementation or for understanding the project itself. However, this report was reviewed for syntax and grammar by DeepL as well as a GPT model. It should be noted that none of the information contained in this report was generated by anyone other than the authors.

### 4 Conclusion

LLVMIR is compiling through there are mistakes in the parse tree.