У меня разработка компилятора подмножества процедурного для JVM

Компилятор состоит из следующих основных компонентов: file.w, parser, bytecodeGen и .class. Ниже представлено краткое описание каждого блока, сейчас мы их рассмотрим поподробнее.

Блок file.w подразумевает файл с исходным кодом формата .w, на слайде два примера программ на Wlang.

Блок parser состоит из трех подблоков, wlangLexer-лексический анализ на основе файла грамматики, wlangParser-синтаксический анализ на основе файла грамматики и classDeclaration-создание описания класса на основе синтаксического анализа

Подблок wlangLexer занимается разбиением входного файла на токены, за это отвечает соответствующий класс wlangLexer, который сгенерирован библиотекой antlr4 на основе файла грамматики, подробное описание файла грамматики представлено в докуменет “Описания программы”, также тут показан пример токенов для программы HelloWorld.

Подблок wlangParser на основе входящего потока токенов строит дерево токенов (parse tree), этим занимается класс wlangParser, который тоже сгенерирован на основе файла грамматики, вот пример дерева токенов для программы HelloWorld.

Подблок classDeclaration создает описание класса на основе дерева токенов, этим занимается класс classDeclaration, на слайде представлен пример описания класса для HelloWorld

Блок BytecodeGen на основе описания класса генерирует байт-код для JVM, этим занимается класс BytecodeGenerator и метод generate класса classGenerator, по итогу мы получаем массив байтов, который записывается в файл формата .сlass, вот пример для HelloWorld.

Вот пример компиляции нашей программы HelloWorld и последующего запуска на JVM.