

1 Задание

Необходимо написать некоторый упрощенный аналог генератора трансляторов. Рекомендуется брать за основу синтаксис ANTLR или Bison. Рекомендуется для чтения входного файла с грамматикой сгенерировать разборщик с помощью ANTLR или Bison.

Необходимо набрать в сумме хотя бы 35 баллов.

Обязательное требование: должен быть лексический анализатор, не должно быть ограничений, что токен это один символ.

Необходимо из каждого пункта выполнить хотя бы 1 вариант.

Выбор класса грамматик

1. (10 баллов) LL(1)-грамматики, нисходящий разбор
2. (15 баллов) SLR-грамматики, восходящий разбор
3. (20 баллов) LALR-грамматики, восходящий разбор

Работа с атрибутами

1. (10 баллов) поддержка синтезируемых атрибутов
2. (10 баллов) поддержка наследуемых атрибутов

Тестирование получившегося генератора

1. (обязательно) сгенерировать с помощью вашего генератора калькулятор
2. (5 баллов) выполнить с помощью вашего генератора ваше задание второй лабораторной
3. (10 баллов) выполнить с помощью вашего генератора ваше задание третьей лабораторной

2 Реализация

В работе реализован генератор парсеров для LL(1) грамматики с поддержкой синтезируемых и наследуемых атрибутов. Входная грамматика описывается с помощью синтаксиса похожего на синтаксис ANLR. (см ParserGrammar). Для генерации используется класс Generator, который принимает путь для входной грамматики и опционально путь до выходной директории (дефолтная директория: gen).

Особенности реализации

1. Переменная `_localctx` зарезервирована.
2. Для обозначения конца файла используется EOF.