Lab 5-6

Jeler Andrei-Iulian & Hoha Mihai

934

Github link: https://github.com/SummerRolls99/FLCD/tree/main/lab%20-%20parser

Statement: Implement a parser algorithm

Lab 5

One of the following parsing methods will be chosen (assigned by teaching staff):

1.a. recursive descendent

The representation of the parsing tree (output) will be (decided by the team):

2.a. productions string (max grade = 8.5)

2.b. derivations string (max grade = 9)

2.c. table (using father and sibling relation) (max grade = 10)

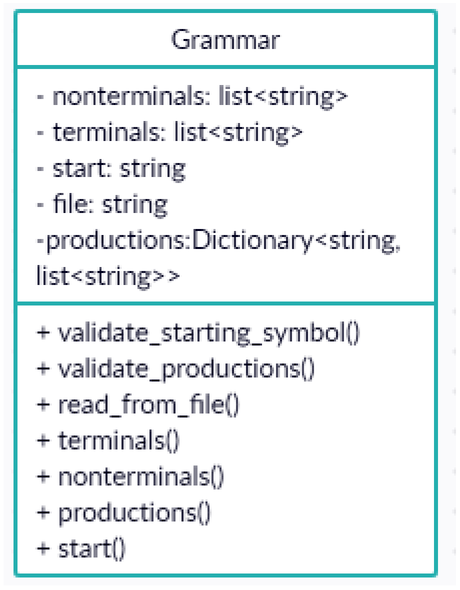
Lab 6

PART 2: Deliverables

1. Algorithm corresponding to parsing tables (if needed) and parsing strategy

2. Class ParserOutput - DS and operations corresponding to choice 2.c (required operations: transform parsing tree into representation; print DS to screen and to file)

Class diagram:



Class structure:

The grammar class stores the necessary information as follows:

- The nonterminals are stored as a list of strings
- The terminals are stored as a list of strings
- The starting symbol is stored as a string
- The productions are stored as a dictionary that has as key the left hand side, and as value a list which has elements lists of string corresponding to each value in the right hand side

File structure:

The grammar is stored in the file as follows:

- First line: list of nonterminals
- Second line: list of terminals
- Third line: starting symbol
- Rest of file: the productions as follows: each line has a production, the lhs and rhs are separated by '-' and each possible value of the production is separated by '|'