https://github.com/cs-ubbcluj-ro/lab-work-computer-science-2024-915-Micu-AlexiaClaudia/tree/main/3-Parser

The `Grammar` class provides functionality to represent and manipulate context-free grammars (CFGs). It supports reading grammar definitions from a file, checking CFG validity, and querying grammar elements such as productions and symbols.

`__init__(self)`

- Initializes the grammar with the following attributes:
  - `non_terminals`: A list of non-terminal symbols.
  - `terminals`: A list of terminal symbols.
  - `start_symbol`: The starting symbol of the grammar.
  - `productions`: A dictionary mapping non-terminals to their production rules.

`load_from_file(self, file_name: str)`

- Reads a grammar definition from a file and populates the grammar's attributes.

`check_if_CFG(self) -> bool`

- Validates if the grammar adheres to the rules of a context-free grammar (CFG):
  - All left-hand sides (LHS) are single non-terminals.
  - All symbols in the right-hand side (RHS) are in the grammar's alphabet.
  - The start symbol is defined in the grammar.
- Returns:
  - `True` if valid CFG.
  - `False` otherwise.

`get_productions_for_non_terminal(self)`

- Prompts the user for a non-terminal and prints its productions.
- Checks if the input is a valid non-terminal and provides an error message if invalid.

`__str__(self)`

  - Provides a string representation of the grammar, listing non-terminals, terminals, the start symbol, and production rules.