Grammar Class

https://github.com/917-SzaboBalazs/FLCD/tree/main/lab5

Description

The `Grammar` class represents a context-free grammar (CFG) and provides methods to access information about the grammar. It reads grammar specifications from a file, initializes the object, and checks if it is a context-free grammar.

Class Members

- `__input_file`: Private attribute representing the input file path.
- `__nonterminals`: Private list of nonterminal symbols.
- `__terminals`: Private list of terminal symbols.
- `__input_symbol`: Private attribute representing the input symbol.
- `__productions`: Private instance of the `ProductionSet` class containing production rules.
- `__is_cfg`: Private attribute indicating whether the grammar is a CFG.

Methods

`__init__(self, input_file)`

- Constructor method that initializes the Grammar object.
- Reads grammar specifications from the specified input file.
- Sets nonterminals, terminals, input symbol, productions, and checks if it is a CFG.

`get_input_file(self)`

- Returns the path of the input file.

`get_nonterminals(self)`

- Returns the list of nonterminal symbols.

`get_terminals(self)`

- Returns the list of terminal symbols.

`get_input_symbol(self)`

- Returns the input symbol.

`get_productions(self)`

- Returns the `ProductionSet` instance containing production rules.

`get_is_cfg(self)`

- Returns a boolean indicating whether the grammar is a CFG.

`__read_from_file(self)`

- Private method that reads grammar specifications from the input file.
- Populates nonterminals, terminals, input symbol, and production rules.

`__check_if_cfg(self)`

- Private method that checks if the grammar is a CFG.
- Returns a boolean indicating the result.

ProductionSet Class

Description

The `ProductionSet` class represents a set of production rules in a CFG. It provides methods to access and manipulate these rules.

Class Members

- `__productions`: Private dictionary storing production rules.

Methods

`__init__(self)`

- Constructor method that initializes the ProductionSet object.

`get_all(self)`

- Returns the dictionary containing all production rules.

`__getitem__(self, key)`

- Returns the production rules associated with the given key.

`__setitem__(self, key, value)`

- Sets the production rules for the given key.

`keys(self)`

- Returns the keys (nonterminal symbols) in the production set.

`__str__(self)`

- Returns a string representation of the production set in a readable format.