

# Домашняя работа по формальным языкам 4

Полупанова Анна

03.10

## Задание №1

### 1. Даниил Плотников.

Мне кажется, что язык хорошо описан, то есть мне после прочтения и прочтения примера все стало понятно.

Кажется не совсем удобным для использования из-за отсутствия ключевых слов + наверное было бы лучше, если бы в языке задавался алфавит.

Поддержка конкретно этого языка не делалась.

Лексер на некорректный ввод выдает ошибку «некорректный символ», не принимает символы не входящие в язык, а также некорректные имена вершин или их состояний.

Честно говоря, у меня не получилось найти ошибок в лексере, потому что регулярные выражения написаны так, что все токены однозначно определяются (благодаря тому, что после элементов стоят скобки разных видов).

Примеры некорректных входов:

некорректный вход, на котором не ломается лексер:

```
3 }
123 ] -> 0
aaa ] -> 2
abb ] -> 3
123,aaa : #a a a a# #b_hello\\s#
abb,123 : #abc# #kdfjd123#
```

некорректный вход, на котором уже ломается лексер:

```
3 }
123 -> 0
aaa -> 2
abb -> 3
123,aaa : #a a a a# #b_hello\\s#
abb,123 : #abc# #kdfjd123#
```

из-за отсутствия квадратной скобки не может распарсить имена вершин

## 2. Артем Полушин.

На момент описания языка рассматривалась версия из второго дз, все было понятно даже без примеров, единственное: была опечатка, в конце перепутано терминальное и не терминальное состояние.

Несмотря на то, что ключевых слов в языке тоже нет, он кажется более удобным, нежели предыдущий рассматриваемый. То есть, на мой взгляд, он более легкочитаем.

Была сделана подсветка другого языка, а не языка описания конечных автоматов.

Лексер выдает сообщения об ошибках в консоль, а не в файл, но в целом тоже стандартно все распознает и выдает информацию о некорректных символах.

На корректных примерах лексер работает верно.

Примеры некорректных входов:

```
3
"aaa"
ge
"Q1"
"T2"
Q0 : 0 -> Q1; 1 -> T2;
Q1 : 0 -> T2; 1 -> Q0;
T2 : 0 -> T2; 1 -> Q0;
```

Не может распарсить слово без кавычек.

```
3
"aaa"
"Q1"
"T2"
Q0 : 0 -> Q1; 1 T2;
Q1 : 0 -> T2; 1 Q0;
T2 : 0 -> T2; 1 Q0;
```

отсутствие стрелок воспринимает спокойно.

## Задание №2

Написан парсер на плюсах + два теста к нему.