

МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Кафедра теории вероятностей и компьютерного моделирования

Лабораторная работа № 1
По спецкурсу «Теория сложности алгоритмов»

Регулярные языки

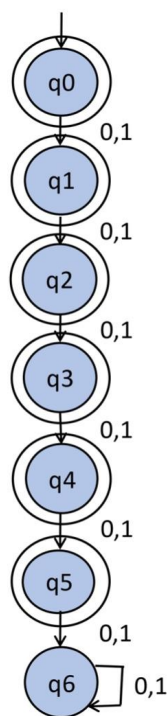
Выполнил: Покхарел П.К.
Группа: М8О-101М-22, Вариант 7
Преподаватель: Рассказова В.А.

Москва, 2023

Задание. Для заданного регулярного языка A над алфавитом $\Sigma = \{0, 1\}$

1. построить диаграмму состояний ДКА, распознающего A ;
2. реализовать данный ДКА в виде программы, которая для произвольной входной строки w должна выводить историю вычислений ДКА на ней в виде последовательности состояний.

$$A = \{w: w \text{ имеет длину не более пяти}\};$$



	0	1
q0	q1	q1
q1	q2	q2
q2	q3	q3
q3	q4	q4
q4	q5	q5
q5	q6	q6
q6	q6	q6

Рис 1 Диаграмма состояний МП-автомата

Программная реализация

```
table = {'q0': ['q1', 'q1'],
```

```
        'q1': ['q2', 'q2'],
```

```
        'q2': ['q3', 'q3'],
```

```
        'q3': ['q4', 'q4'],
```

```
        'q4': ['q5', 'q5'],
```

```
        'q5': ['q6', 'q6'],
```

```
'q6': ['q6', 'q6']  
}
```

```
def test(s, finish):
```

```
    key_cur = 'q0'
```

```
    path = 'q0'
```

```
    for bit in list(s):
```

```
        key_cur = table[key_cur][int(bit)]
```

```
        path += '->' + key_cur
```

```
    return path, key_cur in finish
```

```
need_state = ['q0', 'q1', 'q2', 'q3', 'q4', 'q5']
```

```
commands = input('Введите последовательность команд из 0 и 1: ')
```

```
seq, result = test(commands, need_state)
```

```
print(seq, result, sep='\n')
```