

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт Космических и информационных технологий

институт

Кафедра «Информатика»

кафедра

ОТЧЕТ О ПРАКТИЧЕСКОЙ РАБОТЕ №1

Конечные автоматы

тема

Преподаватель

подпись, дата

А. С. Кузнецов

инициалы, фамилия

Студент КИ18-17/16 031830504

номер группы, зачетной книжки

подпись, дата

Е.В. Железкин

инициалы, фамилия

Красноярск 2021

1 Цель работы

Необходимо построить ДКА и НКА в системе JFLAP и произвести программную реализацию. В коде программы обязательно наличие сущностей и процедур, относящихся к табличному представлению автомата.

Использование функций обработки строковых данных запрещено. Результат работы, выдаваемый программой на экран, внешне должен быть схож, а фактически эквивалентен результату, выдаваемому JFLAP на тех же тестовых цепочках.

2 Задача работы

В каждом варианте задания в части,

- а) задается цепочка или набор цепочек для распознавания ДКА. В части
- б) задается цепочка или набор цепочек для распознавания НКА.

Вариант 7

Вариант 7. а) Построить ДКА, допускающий в алфавите $\{0, 1\}$ множество всех цепочек, в которых число нулей нацело делится на 5, а число единиц — на 3.

б) Построить НКА с количеством состояний, не превышающим 4, для языка $\{a^n : n \geq 0\} \cup \{b^n a : n \geq 1\}$.

Рисунок 1 - Задание

3 Ход работы

- а) Была составлена таблица состояний (множество всех состояний Q) с использованием условия «остаток от деления количества единиц на 3 и нулей на 5»:

1	0
0	0
0	1
0	2
0	3
0	4
1	0
1	1
1	2
1	3
1	4
2	0
2	1
2	2
2	3
2	4

Рисунок 2 – Таблица состояний

Где первый столбец отвечает за остаток от деления количества единиц в данном состоянии, а второй столбец остаток от деления кол-ва нулей в данном состоянии.

По условию, входное состояние q_0 у нас $(0, 0)$;

По условию, в качестве выходного состояния (F) нас устраивает только $(0,0)$;

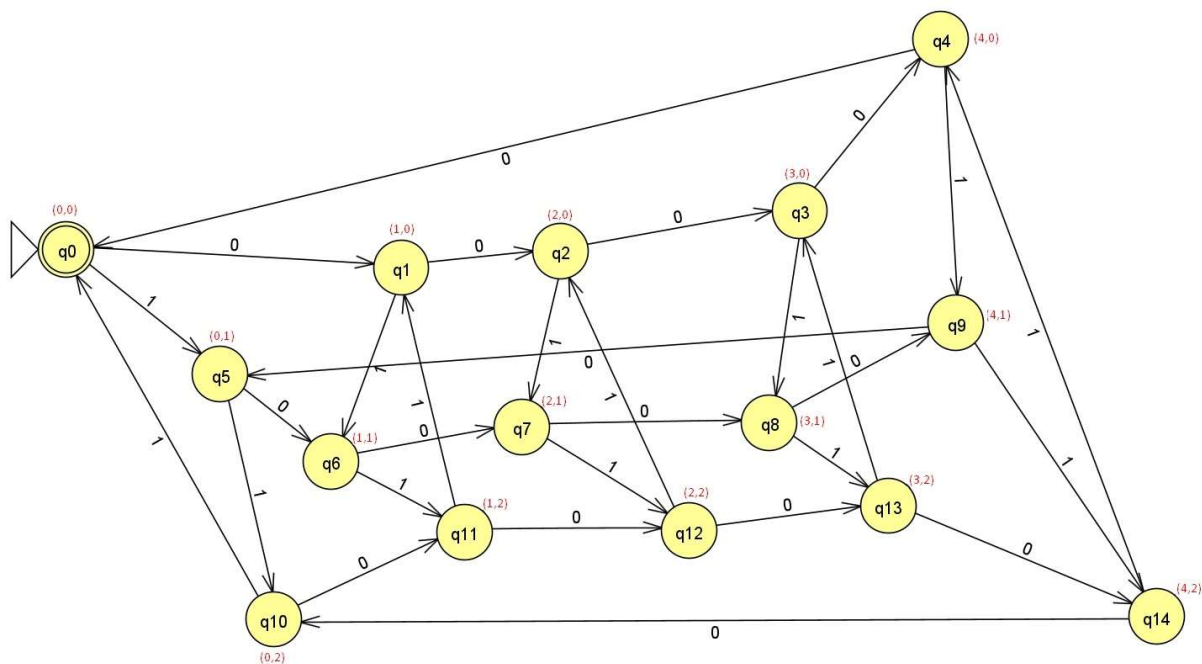


Рисунок 3 – Диаграмма переходов ДКА (у вершин подписаны их состояния из таблицы)

```

Enter a string with '0' s and '1's:
Press Enter Key to stop

01010100
0: DFA_STATES.q0 -> DFA_STATES.q1
1: DFA_STATES.q1 -> DFA_STATES.q6
0: DFA_STATES.q6 -> DFA_STATES.q7
1: DFA_STATES.q7 -> DFA_STATES.q12
0: DFA_STATES.q12 -> DFA_STATES.q13
1: DFA_STATES.q13 -> DFA_STATES.q3
0: DFA_STATES.q3 -> DFA_STATES.q4
0: DFA_STATES.q4 -> DFA_STATES.q0
Accepted

Process finished with exit code 0

```

Рисунок 4 – Пример выполнения ДКА 1

```

Enter a string with '0' s and '1's:
Press Enter Key to stop

Accepted

```

Рисунок 5 – Пример выполнения ДКА 2

```

Enter a string with '0' s and '1's:
Press Enter Key to stop

101
1: DFA_STATES.q0 -> DFA_STATES.q5
0: DFA_STATES.q5 -> DFA_STATES.q6
1: DFA_STATES.q6 -> DFA_STATES.q11
Rejected

Process finished with exit code 0

```

Рисунок 6 – Пример выполнения ДКА 3

```
Enter a string with '0' s and '1's:
Press Enter Key to stop

11000000100000111
1: 0 -> 5
1: 5 -> 10
0: 10 -> 11
0: 11 -> 12
0: 12 -> 13
0: 13 -> 14
0: 14 -> 10
1: 10 -> 0
0: 0 -> 1
0: 1 -> 2
0: 2 -> 3
0: 3 -> 4
0: 4 -> 0
1: 0 -> 5
1: 5 -> 10
1: 10 -> 0
Accepted

Process finished with exit code 0
```

Рисунок 7 - Пример выполнения ДКА 4

б) Для выполнения данного задания была составлена диаграмма переходов в соответствии с языком, описанным в задании:

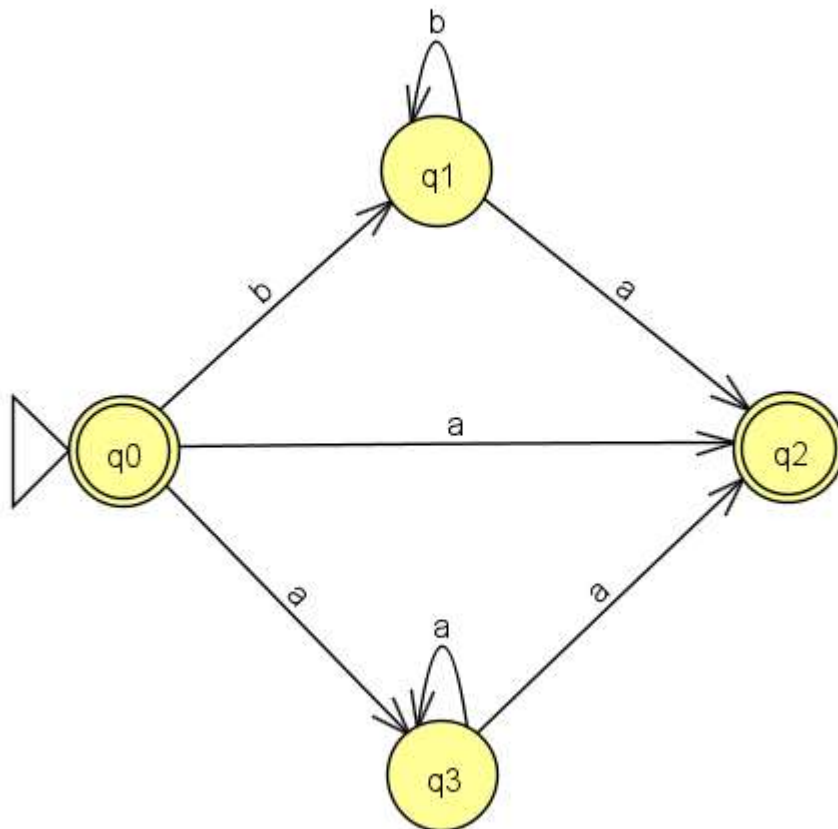


Рисунок 8 – Диаграмма переходов НКА

Далее была составлена таблица функции переходов:

	a	b
$\rightarrow q_0$	$\{q_2, q_3\}$	$\{q_1\}$
q_1	$\{q_2\}$	$\{q_1\}$
q_2	\emptyset	\emptyset
q_3	$\{q_2, q_3\}$	\emptyset

Рисунок 9 – Таблица функции переходов

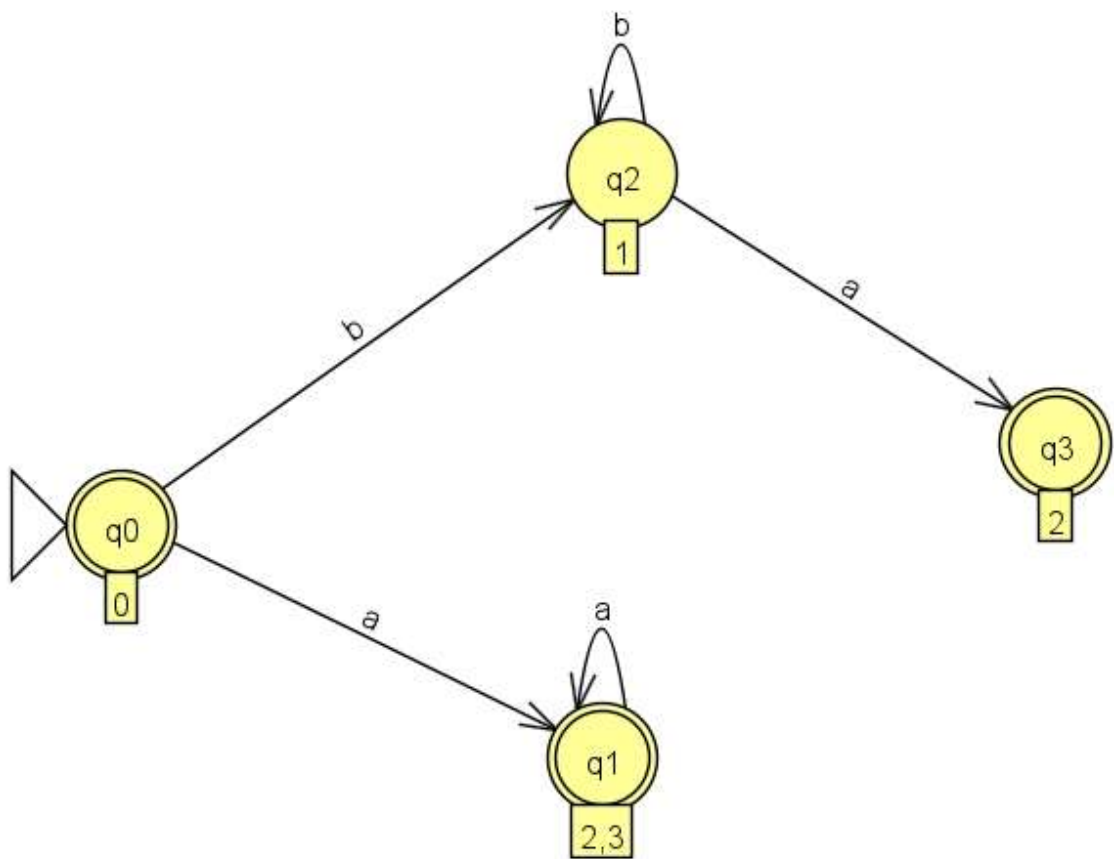


Рисунок 10 – НКА, конвертированный в ДКА с помощью средств JFLAP

ከክርስቲያንነት ጋር

```
Process finished with exit code 0
```

8


```
Enter a string with 'a' s and 'b's:
Press Enter Key to stop

aaa
a: DFA_STATES.q0 -> DFA_STATES.q1
a: DFA_STATES.q1 -> DFA_STATES.q1
a: DFA_STATES.q1 -> DFA_STATES.q1
Accepted

Process finished with exit code 0
```

Рисунок 12 - Пример выполнения НКА 2

```
Enter a string with 'a' s and 'b's:
Press Enter Key to stop

ababa
a: DFA_STATES.q0 -> DFA_STATES.q1
b: DFA_STATES.q1 -> 0
Rejected

Process finished with exit code 0
```

Рисунок 13 - Пример выполнения НКА 3

```
Enter a string with 'a' s and 'b's:
Press Enter Key to stop

Accepted

Process finished with exit code 0
```

Рисунок 14 - Пример выполнения НКА 4

4 Вывод

В ходе данной лабораторной работы были изучены и реализованы в систему JFLAP, а также программно детерминированные и недетерминированные конечные автоматы.

ПРИЛОЖЕНИЕ А

Листинг 1 – файл PyDFA.py

```
import sys
from enum import Enum

TOTAL_STATES = 15
FINAL_STATES = 1
ALPHABET_CHARCTERS = 2

UNKNOWN_SYMBOL_ERR = 0
NOT_REACHED_FINAL_STATE = 1
REACHED_FINAL_STATE = 2

class DFA_STATES(Enum):
    q0 = 0
    q1 = 1
    q2 = 2
    q3 = 3
    q4 = 4
    q5 = 5
    q6 = 6
    q7 = 7
    q8 = 8
    q9 = 9
    q10 = 10
    q11 = 11
    q12 = 12
    q13 = 13
    q14 = 14

class Input(Enum):
    zero = 0
    one = 1

gAcceptedStates = [DFA_STATES.q0]
gAlphabet = ('0', '1')
gTransitionTable = [[0] * ALPHABET_CHARCTERS for i in range(TOTAL_STATES)]
gCurrentState = DFA_STATES.q0

def setDFATransitions() -> None:
    gTransitionTable[DFA_STATES.q0.value][Input.zero.value] =
DFA_STATES.q1
    gTransitionTable[DFA_STATES.q0.value][Input.one.value] =
```

```

DFA_STATES.q5
    gTransitionTable[DFA_STATES.q1.value][Input.zero.value] =
DFA_STATES.q2
    gTransitionTable[DFA_STATES.q1.value][Input.one.value] =
DFA_STATES.q6
    gTransitionTable[DFA_STATES.q2.value][Input.zero.value] =
DFA_STATES.q3
    gTransitionTable[DFA_STATES.q2.value][Input.one.value] =
DFA_STATES.q7
    gTransitionTable[DFA_STATES.q3.value][Input.zero.value] =
DFA_STATES.q4
    gTransitionTable[DFA_STATES.q3.value][Input.one.value] =
DFA_STATES.q8
    gTransitionTable[DFA_STATES.q4.value][Input.zero.value] =
DFA_STATES.q0
    gTransitionTable[DFA_STATES.q4.value][Input.one.value] =
DFA_STATES.q9
    gTransitionTable[DFA_STATES.q5.value][Input.zero.value] =
DFA_STATES.q6
    gTransitionTable[DFA_STATES.q5.value][Input.one.value] =
DFA_STATES.q10
    gTransitionTable[DFA_STATES.q6.value][Input.zero.value] =
DFA_STATES.q7
    gTransitionTable[DFA_STATES.q6.value][Input.one.value] =
DFA_STATES.q11
    gTransitionTable[DFA_STATES.q7.value][Input.zero.value] =
DFA_STATES.q8
    gTransitionTable[DFA_STATES.q7.value][Input.one.value] =
DFA_STATES.q12
    gTransitionTable[DFA_STATES.q8.value][Input.zero.value] =
DFA_STATES.q9
    gTransitionTable[DFA_STATES.q8.value][Input.one.value] =
DFA_STATES.q13
    gTransitionTable[DFA_STATES.q9.value][Input.zero.value] =
DFA_STATES.q5
    gTransitionTable[DFA_STATES.q9.value][Input.one.value] =
DFA_STATES.q14
    gTransitionTable[DFA_STATES.q10.value][Input.zero.value] =
DFA_STATES.q11
    gTransitionTable[DFA_STATES.q10.value][Input.one.value] =
DFA_STATES.q0
    gTransitionTable[DFA_STATES.q11.value][Input.zero.value] =
DFA_STATES.q12
    gTransitionTable[DFA_STATES.q11.value][Input.one.value] =
DFA_STATES.q1
    gTransitionTable[DFA_STATES.q12.value][Input.zero.value] =
DFA_STATES.q13
    gTransitionTable[DFA_STATES.q12.value][Input.one.value] =
DFA_STATES.q2
    gTransitionTable[DFA_STATES.q13.value][Input.zero.value] =
DFA_STATES.q14

```

```

        gTransitionTable[DFA_STATES.q13.value][Input.one.value] =
DFA_STATES.q3
        gTransitionTable[DFA_STATES.q14.value][Input.zero.value] =
DFA_STATES.q10
        gTransitionTable[DFA_STATES.q14.value][Input.one.value] =
DFA_STATES.q4

def dfa(current_symbol: chr) -> int:
    global gAcceptedStates
    global gCurrentState
    global gTransitionTable

    if current_symbol not in gAlphabet:
        return UNKNOWN_SYMBOL_ERR

    old_state = gCurrentState
    gCurrentState =
gTransitionTable[gCurrentState.value][gAlphabet.index(current_symb
ol)]
    print(str(old_state) + " -> " + str(gCurrentState))    # route
info
    if gCurrentState in gAcceptedStates:
        return REACHED_FINAL_STATE

    return NOT_REACHED_FINAL_STATE

if __name__ == "__main__":
    result = RuntimeError

    setDFATransitions()

    print("Enter a string with '0' s and '1's:\nPress Enter Key to
stop\n")

    symbol = sys.stdin.read(1)

    while symbol != '\n':
        result = dfa(symbol)

        symbol = sys.stdin.read(1)

    if REACHED_FINAL_STATE == result:
        print('Accepted')

    if NOT_REACHED_FINAL_STATE == result:
        print('Rejected')

```

Листинг 2 – файл PyNFA.py

```
import sys
from enum import Enum

TOTAL_STATES = 3
FINAL_STATES = 1
ALPHABET_CHARCTERS = 2

UNKNOWN_SYMBOL_ERR = 0
NOT_REACHED_FINAL_STATE = 1
REACHED_FINAL_STATE = 2

class DFA_STATES(Enum):
    q0 = 0
    q1 = 1
    q2 = 2
    q3 = 3

class Input(Enum):
    a = 0
    b = 1

gAcceptedStates = [DFA_STATES.q0, DFA_STATES.q1, DFA_STATES.q3]
gAlphabet = ('a', 'b')
gTransitionTable = [[0] * ALPHABET_CHARCTERS for i in
range(TOTAL_STATES)]
gCurrentState = DFA_STATES.q0

def setDFATransitions() -> None:
    gTransitionTable[DFA_STATES.q0.value][Input.a.value] =
DFA_STATES.q1
    gTransitionTable[DFA_STATES.q0.value][Input.b.value] =
DFA_STATES.q2
    gTransitionTable[DFA_STATES.q1.value][Input.a.value] =
DFA_STATES.q1
    gTransitionTable[DFA_STATES.q2.value][Input.a.value] =
DFA_STATES.q3
    gTransitionTable[DFA_STATES.q2.value][Input.b.value] =
DFA_STATES.q2

def dfa(current_symbol: chr) -> int:
    global gAcceptedStates
    global gCurrentState
    global gTransitionTable
```

```

    if current_symbol not in gAlphabet:
        return UNKNOWN_SYMBOL_ERR

    old_state = gCurrentState
    try:
        gCurrentState =
gTransitionTable[gCurrentState.value][gAlphabet.index(current_symb
ol)]
    except Exception as e:
        return NOT_REACHED_FINAL_STATE

    print(current_symbol + ": " + str(old_state) + " -> " +
str(gCurrentState))    # route info

    if gCurrentState in gAcceptedStates:
        return REACHED_FINAL_STATE

    return NOT_REACHED_FINAL_STATE

if __name__ == "__main__":
    result = REACHED_FINAL_STATE

    setDFATransitions()

    print("Enter a string with 'a' s and 'b's:\nPress Enter Key to
stop\n")

    symbol = sys.stdin.read(1)

    while symbol != '\n':
        result = dfa(symbol)

        if UNKNOWN_SYMBOL_ERR == result:
            print('Unknown symbol error')
            break

        symbol = sys.stdin.read(1)

    if REACHED_FINAL_STATE == result:
        print('Accepted')

    if NOT_REACHED_FINAL_STATE == result:
        print('Rejected')

```