МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«Тверской государственный технический университет»

(ТвГТУ)

Кафедра “Программного обеспечения”

**Курсовая работа**

по дисциплине “Теория алгоритов и формальных языков”

Тема: “Преобразование НКА в эквивалентный ему автомат без ε - команд”

Выполнил: студент группы

ПИН 17.06

Завгороднев Е. Ю

Проверил:

к.ф.- м.н. Карлов Б. Н

Тверь 2020

Оглавление

[Введение 3](#_Toc41039759)

[ДКА. НКА. ε-НКА 3](#_Toc41039760)

[Аналитическая часть 5](#_Toc41039761)

[Алгоритм 5](#_Toc41039762)

[Проектная часть 6](#_Toc41039763)

[Описание 6](#_Toc41039764)

[Функционал 6](#_Toc41039765)

[Пример работы 6](#_Toc41039766)

[Архитектура приложения 8](#_Toc41039767)

[Заключение 9](#_Toc41039768)

[Литература 9](#_Toc41039769)

# 

# Введение

**Цель**: разобраться в работе недетерминированных конечных автоматов. Ознакомится с ε – переходами. Примерить алгоритм преобразования НКА в эквивалентный ему автомат без ε – команд.

## ДКА. НКА. ε-НКА

Недетерминированные конечные автоматы – одна из моделей, используемых в теории вычислений.

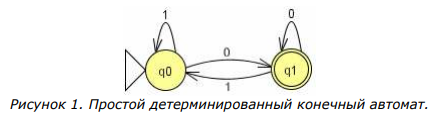
Детерминированным конечным автоматом (ДКА) называется устройство, описываемое

следующими параметрами:

* Q – конечное множество состояний.
* Σ – конечное множество входных символов (алфавит).
* δ – функция перехода. Аргументы – состояние и входной символ, результат – состояние.
* q0 – начальное состояние, принадлежит Q.
* F – множество допускающих состояний, является подмножеством Q.

И функционирующее следующим образом:

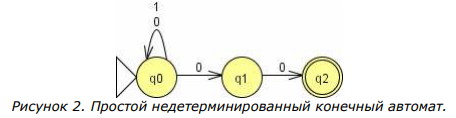
* Автомат начинает работу в состоянии q0.
* Если автомат находится в состоянии qi , а на вход поступает символ b, то автомат переходит в состояние δ(qi,b).



Определение недетерминированного конечного автомата (НКА) практически полностью повторяет приведённое выше определение ДКА. Отличий всего два:

* δ – функция перехода. Аргументы – состояние и входной символ, результат – множество состояний (возможно – пустое).
* Если автомат находится в состоянии qi, а на вход поступает символ b, то автомат переходит во множество состояний δ(qi , b). Если автомат находится во множестве состояний {qi}, то он переходит во множество состояний, получаемое объединением множеств δ(qi , b).





Тот же автомат в виде таблицы:

|  |  |  |
| --- | --- | --- |
|  | 0 | 1 |
| q0 | {q0, q1} | {q0} |
| q1 | {q2} | ∅ |
| q2 | ∅ | ∅ |

Небольшое полезное расширение стандартного НКА – ε-НКА, или НКА с эпсилон-переходами.

«Эпсилон-переходом» называется переход между состояниями, который может быть выполнен автоматом «просто так», без входного символа. На графах и в таблицах такие переходы обычно помечаются символом ε.

Введём понятие ε-замыкание.

* ε-замыканием состояния qi называется множество состояний ε-НКА, в которые из qi можно попасть по цепочке ε-переходов. Как минимум, в это множество входит само qi .
* Функцию, аргументом которой является состояние, а значением – соответствующее ε-замыкание, назовём eclose.

Функцию eclose можно определить так:



А теперь мы можем строго определить функционирование ε-НКА.

* Автомат начинает работать во множестве состояний eclose(q0).
* Если автомат находится во множестве состояний {qi}, то он переходит во множество состояний, получаемое ε-замыканием всех состояний из объединения множеств δ(qi , a).



# Аналитическая часть

**Задача**:

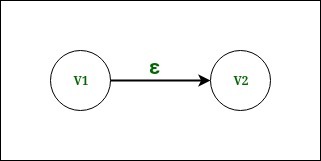
Написать программу для задачи:

Преобразовать недетерминированный конечный автомат в эквиваленый ему автомат без ε-команд.

Разработать формат входных и выходных данных, удобный для человека.

Входные данные должны считываться из файла, выходные данные должны выводится в файл. Прокоментировать каждый логически законченный фрагмент программы.

## Алгоритм



Шаг 1: Рассмотрим две вершины, имеющие эпсилон-ход. Здесь, на рисунке, у нас есть вершина v1, а вершина v2, имеющая эпсилон, перемещается из v1 в v2.

Шаг 2: Теперь найдите все ходы к любой другой вершине, которая начинается с вершины v2 (кроме рассматриваемого движения эпсилона). Найдя ходы, продублируйте все ходы, начинающиеся с вершины v2, с тем же входом, чтобы начать с вершины v1, и удалите эпсилон-переход из вершины v1 в вершину v2.

Шаг 3: Посмотрите, если вершина v1 является начальным состоянием или нет. Если вершина v1 является начальным состоянием, то мы также сделаем вершину v2 начальным состоянием. Если вершина v1 не является начальным состоянием, то никаких изменений не будет.

Шаг 4: Посмотрите, является ли вершина v2 конечным состоянием или нет. Если вершина v2 является конечным состоянием, то мы также сделаем вершину v1 конечным состоянием. Если вершина v2 не является конечным состоянием, то никаких изменений не будет. Повторяйте шаги (от шага 1 до шага 4), пока все движения эпсилона не будут удалены из NFA.

# Проектная часть

## Описание

Программа разрабатывалась на языке Java (8-ой версии).

Используемые библиотеки:

* [OpenCSV](http://opencsv.sourceforge.net/)

## Функционал

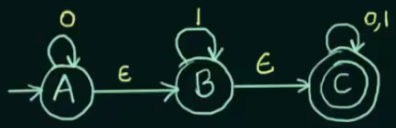
Программа принимает на вход НКА с ε - переходами (в виде csv файла):

* Количество символов в алфавите
* Алфавит
* Количество состояний
* Функции перехода

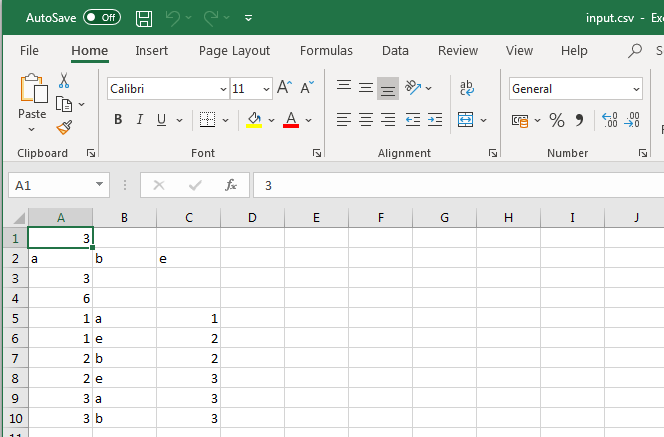
На выходе программа выдает эквивалентный НКА без ε-команд (в виде txt файла):

* Алфавит
* Состояния
* НКА в виде таблицы

## Пример работы

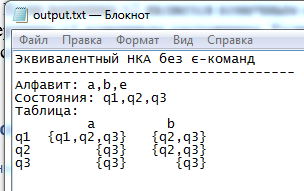


input.csv





output.txt



## Архитектура приложения

**Converter** - класс для преобразования НКА в эквивалентную.

Методы:

* printEquivalentNFA() - печать в выходной поток результатов выполнения программы.
* findClosure() – поиск ε-замыкания.
* insertIntoTransitionTable() – вставка элементов в таблицу НКА.
* findCharacterInAlphabet() – поиск нужного символа в заданном алфавите.
* fillEClosure() – поиск всех ε-замыканий.
* unionClosure() – заполнение массива для построения таблицы с эквивалентным НКА.

**FileIO** - класс для чтения данных из файла и запись результата в файл.

Методы:

* readFromFile() – чтение данных из csv файла.
* writeToFile() – запись результата в txt файл.

**Node –** структура (ребра графа) для таблицы НКА, содержащая:

* начальное состояние.
* конечное состояние, которое является начальным для другого Node.

# Заключение

В ходе выполнения курсовой работы я усвоил принципы работы конечных автоматов (детерминированных и недетерминированных), их представлением в виде графов и таблиц, ознакомился с ε-командами и ε-замыканием. Понял и применил алгоритм преобразования НКА с ε-командами в НКА без ε-команд.

# Литература

* Сергей Холодилов - Недетерминированные конечные автоматы. Статья в журнале RSDN Magazine №2, 2007 – 14 c.
* Conversion of Epsilon-NFA to NFA. Статья с сайта geeksforgeeks.org (<https://www.geeksforgeeks.org/conversion-of-epsilon-nfa-to-nfa/?ref=rp>)