Министерство образования и науки РФ

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра информационных технологий и автоматизированных систем

Дискретная математика

Лабораторная работа № 4

Тема: «Синтезирование и реализация конечного автомата»

Выполнил: студент группы ИВТ-22-1б

Игошев Матвей Иванович

Проверил: Ст. Преподаватель кафедры ИТАС

Рустамханова Г.И.

г. Пермь – 2024

Оглавление

Цель работы 3

Задачи работы 4

Этапы выполнения 5

Заключение 6

Список используемой литературы 7

# Цель работы

Синтезировать конечный автомат и реализовать его в виде консольного приложения.

# Задачи работы

1. Изучить теорию связанную с конечными автоматами.
2. Реализовать функционал приложения.
3. Получить регулярное выражение, для указанного в задании автомата.
4. Протестировать приложение

# Этапы выполнения

**Теоретическая часть и синтез автомата**

Конечные автоматы представляют собой математическую абстракцию, модель дискретного устройства, имеющего один вход, один выход в каждый момент времени находящегося в одном состоянии из множества возможных, помимо этого для автомата задается определенный алфавит — конечный набор различных символов. Однако количество таких состояния для конечного автомата ограниченно.

Конечный автомат может задаваться как в виде диаграммы мура, так и через таблицы выходов и таблицы переходов. Помимо этих способов есть еще способ задания конечного автомата используя формальные языки. Регулярные выражения являются одними из видов формальных языков.

Таблица выходов и переходов представляют собой таблицы, в которых в соответствие поставлены символы алфавита и состояний автомата. Таблица переходов хранит номер следующего состояния. Таблица выходов хранит выход на каждом из состояний. Диаграмма Мура представляет собой граф, на котором изображены все существующие переходы из одного состояния в другое и помимо этого, если выход не привязан к определенным состояниям, то также указывается вывод для каждого из переходов.

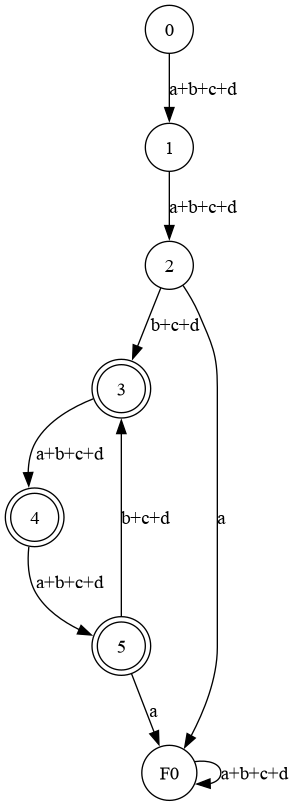
В рамках работы было необходимо синтезировать автомат для распознавания слов некоторого языка, который изначально задан был в текстуальной форме.

Нужно было выполнить задание под вариантом 10(каждая 3 буква должна быть согласной, остальные могут быть любыми). Для данного варианта таблица переходов будет выполнена следующим образом:



В данном автомате состояния: 3, 4, 5 являются распознающими.

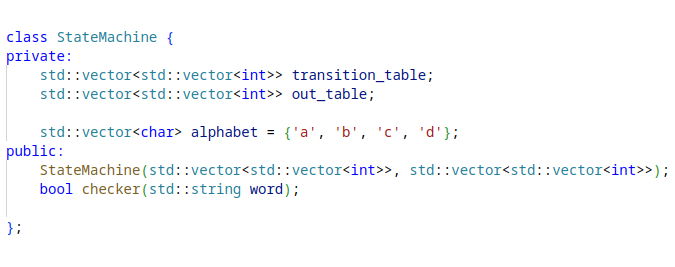
Диаграмма Мура будет выглядеть следующим образом:

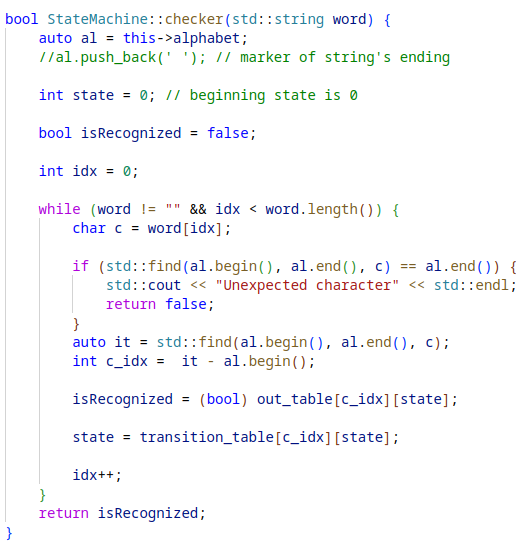
Регулярное выражение для данного автомата будет выглядеть следующим образом:

\*

**Объяснение работы кода программы**

Для выполнения поставленной цели был написан класс StateMachine, который содержит в себе таблицы переводов и выходов. На основе этих таблиц класс выполняет свою задачу. Выглядит класс следующим образом:

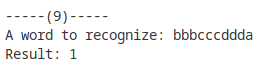
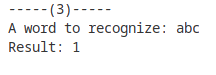
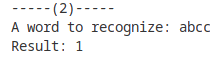
Класс содержит в себе также и алфавит, конструктор и функцию checker, которая выполняет функцию автомата. Выглядит следующим образом:

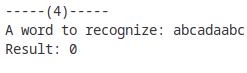
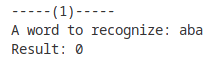
Функция checker объявляет 0 состояние начальным, далее перемещается в рамках цикла по слову. В цикле значение isRecognized меняется в зависимости от состояния согласно таблице выходов. Вместе со значением isRecognized меняется и состояние автомата. Как только заканчиваются символы в слове, цикл завершается, возвращается значение isRecognized.

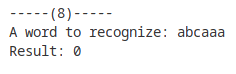
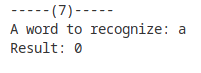
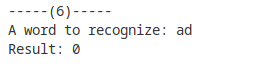
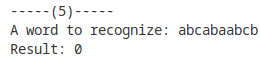
**Тестирование программы**

Для тестирования программы были подобраны принадлежащие и не принадлежащие языку слова.

Тестирование со словами принадлежащим языку:

Тестирование слов не принадлежащих языку:





# Заключение

В ходе работы были закреплены знания о конечных автоматах. Был получен опыт в синтезировании автоматов, в построении регулярных выражений. Реализовано консольное приложение в рамках, которого автомат выполняет распознавание.

# Список используемой литературы

1. https://habr.com/ru/articles/661577/
2. https://neerc.ifmo.ru/wiki/index.php?title=%D0%A0%D0%B0%D1%81%D0%BA%D1%80%D0%B0%D1%81%D0%BA%D0%B0\_%D0%B3%D1%80%D0%B0%D1%84%D0%B0