Creating a Finite Automation Simulator and Visualiser

Description, Objectives, Ethics & Resources

Matriculation Number: 210010341

Description

This project aims to create a standalone application in which users can create a diagram for a finite automaton and either export the automation diagram or simulate a chosen input word on the automation. Currently, there are no highly usable applications for creating such automata diagrams, and most that exist do not provide the functionality to simulate an input word. Because of this, university lecturers find it difficult to create and configure automata easily for teaching purposes. The application will provide lecturers, and possibly others with a simple tool to design, test and mockup finite automata.

The application will be developed using React.js and allow for deterministic and non-deterministic finite automata. Users will be provided with a graphical user interface for creating a finite automation, with the application recognising if the automation is valid or not. To ensure that diagrams are neat and organised, the software will be able to reorganise the diagram's layout to improve readability. Given a valid automation, the user may export the diagram or simulate the automation using an input word with its output displayed. Additional features may be added for further convenience, however, the description above covers the core of this project.

Objectives

Primary

The application should:

- Allow users to create finite automata with their diagrams graphically displayed
- Allow users to choose and manipulate the exact positions of nodes
- Verify that the automaton is valid
- Allow for both deterministic and non-deterministic finite state automata
- Provide users with a way to automatically reconfigure the layout of the diagram to be in an organised and presentable manner, suitable for lectures
- Allow users to export the finite automata diagram in at least one known image format
- Allow the user to simulate an input word on the finite automata by displaying its path, output and whether the word was accepted or rejected

Secondary

The application may:

- Allow users to control the exact angle of edges between nodes
- Implement additional exporting formats such as: SVG, PNG and LaTeX
- Utilise animations when simulating an input word to clearly show the path taken
- Allow users to save a video file of the simulation
- Allow users to create their finite automata configuration through text entry which uses a specific language communicating the automation's formal definition

Ethics

Given that this project only requires questionnaires solely about the artifact produced, it may be covered by the Artifact Evaluation Form for Computer Science projects. The student agrees to completely abide by the terms stated within the ethics application for "Evaluation of artefacts produced for CS projects" (with the ethics approval code CS15727). **See signed copy below.**

Resources

No additional resources will be required, as the project will be developed on a personal computer and tested on the lab machines.

Risks

There are no identifiable risks to the project as it contains no dependencies and is unlikely to be rejected for ethical approval.

UNIVERSITY OF ST ANDREWS
TEACHING AND RESEARCH ETHICS COMMITTEE (UTREC)
SCHOOL OF COMPUTER SCIENCE
ARTIFACT EVALUATION FORM

Title of project

Creating a Finite Automation Simulator and Visualiser

Name of researcher(s)

James Edward Hart

Name of supervisor

Michael Young

Self audit has been conducted **YES** ☑ **NO** ☐

This project is covered by the ethical application CS15727.

Signature Student or Researcher

James Hart

Print Name

James Hart

Date

25/09/2024

Signature Lead Researcher or Supervisor

M. Young

Print Name

MICHAEL YOUNG

Date

25/09/2024