Group members: Quy An Le, Bao Long Duong, An Nguyen

COT 4210

**Interactive Visual Simulation of Regular Expression to Minimal DFA: Project Proposal**

Regular expressions and finite automata form the backbone of numerous computational applications, from compiler design to text processing. While the theoretical pathways for converting between these representations are well-established, there remains a significant gap in educational tools that visually demonstrate these transformations. Our project aims to develop an interactive simulator that visualizes the complete conversion process from regular expressions to minimal deterministic finite automata (DFA), enhancing understanding of these fundamental automata theory concepts.

The conversion pipeline encompasses three key algorithms: Thompson's construction for converting regular expressions to non-deterministic finite automata (NFA), subset construction for transforming NFAs to DFAs, and Hopcroft's algorithm for DFA minimization. Each step represents a significant theoretical concept, but their interrelationships can be challenging to grasp without visualization. Our tool will provide an intuitive visual representation of each transformation phase, allowing users to observe how formal language representation evolves while preserving language equivalence.

Our implementation will follow a modular approach with four interconnected components. First, we will develop a regular expression parser and Thompson's construction algorithm to generate NFAs. Second, we will implement the subset construction algorithm for NFA-to-DFA conversion. Third, we will incorporate Hopcroft's algorithm for DFA minimization. Finally, we will create an interactive visualization layer that renders these automata as graphs, highlights state transitions during algorithm execution, and allows users to test the resulting automata with input strings. The interface will enable users to progress through transformations step-by-step, with explanatory annotations highlighting the theoretical principles being applied at each stage.

We will evaluate our implementation through three primary methods. For correctness, we will verify our results against established theoretical outcomes and existing academic implementations using diverse test cases ranging from simple to complex regular expressions. For usability and educational value, we will conduct small-scale user testing with computer science students, collecting qualitative feedback on the clarity and effectiveness of our visualizations. For performance, we will measure the efficiency of our algorithms with increasingly complex inputs, ensuring our implementation remains responsive even with larger automata. Through this project, we aim to create not just a technical implementation of automata transformations, but an effective educational tool that bridges theoretical understanding with visual intuition, making these fundamental concepts more accessible to students of computational theory.