Project Proposal: Automata and Computability

Cain Hertoghs, Emir Murat, Jonathan van der Sluijs Mathis De Herdt
Bachelor in Computer Science, University of Antwerp, Belgium

October 29, 2024

Introduction

Our project, developed as part of the Automata and Computability course, aims to create an advanced multi-tape Turing machine (MTM) capable of performing various arithmetic operations. Comprising Cain Hertoghs, Emir Murat, Jonathan van der Sluijs and Mathis De Herdt, our group has designed a system that bridges theoretical knowledge of automata with practical application, providing both functional and educational value. Our project centers on implementing a GUI, expression validation, and mathematical operations such as addition, subtraction, multiplication, exponentiation, and handling expressions with brackets using a multi-tape TM. The system will not only function as an effective computational tool but also as an accessible resource for understanding each calculation step.

This project's educational value is twofold: it serves as a resource for elementary school students seeking support with arithmetic, offering a step-by-step breakdown of calculations, and as a study aid for Computer Science students to gain a deeper understanding of the concept and practical implementation of a Turing Machine. By visually simulating the Turing Machine's operations, the project facilitates both basic math skills development and advanced computational concepts, providing a unique educational tool applicable across different learning levels.

Our project falls within the "gold" evaluation category, given the challenging yet accessible nature of implementing GUI, Earley Parser and Turing Machine concepts. We chose this project specifically for its potential to connect theory and practice, demonstrating how foundational concepts in automata theory can be applied to educational tools.

Group Composition and Project Plan

Team Members and Roles

Our team, consisting of Cain Hertoghs, Emir Murat, Jonathan van der Sluijs and Mathis De Herdt, has assigned tasks as follows:

- Cain Hertoghs: Expression validation, Compatibility between member code, Bracket precedence and implementation of MTM
- Emir Murat: Multiplication with MTM, Powers with MTM, Expression validation with Earley parser, Logs in HTML and implementation of MTM
- Jonathan van der Sluijs: Modulo-operation with MTM, Expression validation with Earley parser and Implementation of MTM
- Mathis De Herdt: Input GUI in QT, Addition and subtraction with MTM, Logs in HTML and Implementation of MTM

Features and Time Allocation

The table below outlines the core features of our project, including estimated time for each component. These features form the foundation of our project and will serve as benchmarks for evaluation.

Feature	**Workload**	**Estimated time spent**
GUI in QT	6%	15
CFG, CNF, CYK	4%	10
Expression Validation	8%	20
Addition and subtraction with MTM	8%	20
Multiplication with MTM	8%	20
Modulo-Operations with MTM	12%	30
Expression Validation with Earley parser	10%	25
Brackets and precedence of expressions on MTM	10%	25
Powers with MTM	8%	20
Logs with HTML of calculation steps	6%	15
Logs in HTML of calculation steps	8%	20
Implementation of the MTM	16%	40
Total		**260**

Societal and Educational Benefits

This project offers substantial educational value and societal benefits by providing an accessible computational tool that can be used by a broad range of users. For younger students, particularly those in elementary school, this project provides an interactive way to understand arithmetic, breaking down calculations into clear steps and allowing students to follow the reasoning behind each result. This fosters foundational math skills and builds confidence in mathematical reasoning from an early age.

For Computer Science students, the project offers a practical learning experience with Turing Machines, providing an interactive simulation of computations with visual feedback. This aids students in understanding the complexities and applications of automata theory, supporting deeper learning of theoretical concepts. Overall, this project combines practical computation and theoretical exploration, making it a valuable educational tool for diverse audiences.

Project Phases

- **Phase 1:** Implement the core Turing Machine structure and define a Context-Free Grammar (CFG) for the arithmetic expressions. This will set the foundation for subsequent calculations and syntax validation.
- **Phase 2:** Simulate arithmetic tasks on the Turing Machine and validate input expressions using the Earley Parsing algorithm. This phase focuses on implementing core functionality to perform and verify calculations according to CFG rules.
- **Phase 3:** Develop the visual representation of calculation steps and the Turing Machine's processing using Qt for the GUI and HTML for output. This phase enhances user interaction, allowing both arithmetic learners and CS students to engage with the step-by-step execution of the Turing Machine.

Project Planning and Motivation

This project aims to develop a functional and educational system valuable to both students and educators in understanding arithmetic and Turing Machine concepts. Each team member's time commitment, set to between 60 and 75 hours, aligns with the project requirements, ensuring a balanced workload. The project aligns well with conference objectives, as it demonstrates how core theoretical knowledge can be applied to create a tangible educational tool accessible to different age and education levels.