Turing Recognizable vs Co-Turing Recognizable Checker

A yellow sign with black text

Description automatically generated

The Turing Recognizable vs Co-Turing Recognizable Checker project aims to determine if a language is Turing recognizable or co-Turing recognizable, an important distinction in theoretical computer science and automata theory for understanding the computational characteristics of languages. This project simulates a Turing machine using its description and an input string. By processing the input, the machine checks if it halts and accepts the string, signifying the language is Turing recognizable. Conversely, if it halts and accepts the complement of the string, the language is co-Turing recognizable.

It takes a detailed description of the Turing machine, including its states, alphabet, transition functions, start state, accept state, and reject state, along with an input string. The output clearly indicates whether the language is Turing recognizable, co-Turing recognizable, or neither, based on the machine's response to the input string.

The project operates through several steps: first, the Turing machine's description and the input string are parsed. Next, the Turing machine simulation processes the input string to see if it halts and accepts it. If it does, the language is deemed Turing recognizable. If it halts and accepts the complement of the string, the language is deemed co-Turing recognizable. If neither condition is met, the language is classified as neither.

This project is implemented in Python, utilizing key libraries such as a custom Turing machine simulation library, numpy for efficient array and matrix operations, and matplotlib for visualizing the Turing machine's steps.

