Lab Terminal

Fa20-bcs-009

Maryam Amjad

Q1: Brief of the project

The Compiler project seamlessly transforms regular expressions into Nondeterministic Finite Automata (NFA) and optimizes performance through deterministic conversions to Finite Automata (DFA), with a focus on DFA minimization. The Lexical Analyzer Generator facilitates the creation of customized analyzers, ensuring smooth integration into projects. Additionally, the project supports a variety of parsers, including SLR (Simple LR), LALR (Look-Ahead LR) from LR(1), LALR from LR(0), and LR(1). The integration of Graphviz enhances visualization, providing a practical aid for developers to understand Regular Expression transitions and structures visually.

Key Features:

- **Regular Expression (RE) to Finite Automata Conversions:**
  - Conversion of regular expressions into Nondeterministic Finite Automata (NFA).
  - Subsequent transformation from NFA to Deterministic Finite Automata (DFA).
  - Minimization of DFA to optimize performance.
- **Lexical Analyzer Generator:**
  - Empower your compiler with a customized Lexical Analyzer using our user-friendly generator.
- **Parser Generators:**
  - Support for SLR (Simple LR), LALR (Look-Ahead LR) from LR(1), LALR from LR(0), and LR(1) parsers.
- **Graph Visualization with Graphviz:**
  - Visualize Regular Expression transitions and structures with Graphviz.
  - Ensure Graphviz is installed for optimal graph visualization: