Mid Lab

Subject: Compiler Construction

Submitted To: Ms Aqsa Kazmi

Submitted By: Faisal Hafeez

Registration No: CIIT/FA21-BCS-037/SWL

Section: A

COMSATS University Islamabad,

Sahiwal Campus.

1. LL(1) Parser Code Visual Tool Documentation

1 Overview

This tool provides a graphical interface for working with LL(1) parsers, allowing users to:

- Input grammar productions
- Generate parsing tables
- Test input strings against the grammar
- Visualize the parsing process

![Application Screenshot]([INSERT SCREENSHOT HERE])

2 Installation

1. Requires Visual Studio with .NET Framework support
2. Simply copy the provided code into a new Windows Forms project
3. No additional dependencies required

3 User Interface Components

1. Grammar Input Section

![Grammar Input Section]([INSERT SCREENSHOT HERE])

- **Grammar Textbox**: Multiline textbox for entering grammar productions
- **Generate Parser Button**: Processes the grammar and generates the parsing table

2. Parsing Table Display

![Parsing Table Display]([INSERT SCREENSHOT HERE])

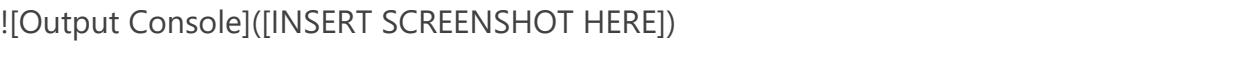
- Color-coded DataGridView showing the complete parsing table
- Non-terminals in light blue header column
- Terminals across the top row

- Productions highlighted in light yellow

3. Test Input Section

- **Input Textbox**: For entering strings to test against the grammar
- **Parse Button**: Initiates the parsing process

4. Output Console



- Rich text box with color-coded messages
- Green for success messages
- Red for error messages
- Dark gray for informational text

4 Using the Application

Step 1: Enter Grammar

1. Type or paste your grammar productions in the Grammar Input section
2. Format: One production per line, using "->" for derivation
3. Example:

```
4. E  -> T E'
5. E' -> + T E' | ε
6. T  -> F T'
7. T' -> * F T' | ε
8. F  -> ( E ) | id
```

Step 2: Generate Parser

1. Click the "Generate Parser" button
2. The system will:

- Validate the grammar
- Compute FIRST sets
- Generate the parsing table
- Display results in the output console
- Show the parsing table in the grid view

Step 3: Test Input Strings

1. Enter a string to test in the Input Textbox
2. Click "Parse Input"
3. View parsing results in the output console

4.1 Code Structure

The entire implementation is contained within the Form1_Load method, organized as:

1. **UI Initialization**
   - Creates all controls programmatically
   - Sets up colors, fonts, and layouts
   - Configures event handlers
2. **Grammar Processing**
   - Parses grammar productions
   - Identifies terminals and non-terminals
   - Computes FIRST sets
3. **Table Generation**
   - Builds the parsing table
   - Populates the DataGridView
4. **Parsing Logic**
   - Simulates the parsing process
   - Provides visual feedback

4.2 Customization Options

1. **Color Scheme**: Modify the RGB values in the control initializations
2. **Fonts**: Change font families and sizes in the control properties
3. **Layout**: Adjust the Top, Left, Width, and Height properties of controls

4.3 Limitations

1. Simplified FIRST set calculation
2. FOLLOW sets not implemented
3. Actual parsing algorithm is simulated

4.4 Future Enhancements

1. Add FOLLOW set calculation
2. Implement full parsing stack visualization
3. Add parse tree generation
4. Include grammar validation checks