Recursive Descent Parser

- ## Introduction

In the realm of compiler design, *syntax analysis* plays a crucial role in processing programming languages. A *Recursive Descent Parser* is a type of *top-down parser* that uses recursive procedures to navigate through the input program based on a set of mutually recursive predictive parsing functions. This approach to parsing allows for the direct translation of grammar rules into parsing functions, offering simplicity and ease of implementation.

- ## Purpose

The primary objective of a *Recursive Descent Parser* is to recognize the structure of the input program and verify its adherence to the defined grammar rules. By constructing parsing functions corresponding to the production rules of the grammar, the parser can recursively explore the input program, making decisions based on the current token being processed.

* Algorithm

1. **Create parsing functions for each non-terminal symbol in the grammar, mimicking the production rules of the grammar.**

2. **Begin parsing by invoking the starting parsing function corresponding to the start symbol of the grammar.**

3. **In each parsing function, match the current token against the expected token(s) based on the corresponding production rule.**

4. **If a match is found, proceed to parse the next token in the input stream using recursive calls to other parsing functions.**

5. **If a mismatch occurs, handle error and recovery strategies to maintain parsing synchronization.**

- **Example**

Consider a simple grammar:

S → E$

E → T + E | T

T → int | (E)

Define parsing functions for each non-terminal symbol (S, E, T) following the grammar rules and recursive parsing logic. Implement error handling and recovery mechanisms to gracefully handle parsing errors during the parsing process.

- **Conclusion**

The *Recursive Descent Parser* is a fundamental parsing technique in compiler design, offering a straightforward and intuitive approach to syntax analysis. By directly mapping grammar rules to parsing functions and utilizing recursive calls for exploring the input program, this parser provides a clear and effective method for language processing tasks.

Producer: Elham Jafari

Computer Engineering