Compilers

LL(k)-Parser

Definition

A grammar is LL(k) if it can be parsed from Left to right, using Leftmost derivations, with k lookahead tokens.

Features

- Predictive Parsing: Uses a parsing table to predict which production to apply based on the lookahead tokens.
- No Backtracking: Decisions are made based on the lookahead, eliminating the need for backtracking.

Conditions

- No Left Recursion: The grammar should not have left recursion.
- Factoring: Left-factoring may be necessary to resolve ambiguities.
- Unambiguous: The grammar must be unambiguous for effective parsing.

SLR(k)-Parser

Definition

A grammar is SLR(1) if it can be parsed using a Simple LR parser with 1 lookahead token.

Features

- Shift-Reduce Parsing: Utilizes a shift-reduce mechanism based on a parsing table.
- LR(0) Items: Relies on LR(0) items and follow sets for handling reduce actions.

Conditions

- No Ambiguity: The grammar should be unambiguous and clearly defined for reductions.
- Conflict Resolution: Resolves conflicts using follow sets.

LR(k)-Parser

Definition

A grammar is LR(k) if it can be parsed using a Left-to-right scan with Rightmost derivations and k lookahead tokens.

Features

- More Powerful: Handles a broader class of grammars compared to LL(k) and SLR(k) parsers.
- Constructing Parsing Tables: Involves complex state-based parsing and lookahead management.

Conditions

- General LR Parsing: LR(1) is commonly used due to a balance between complexity and capability.
- Handles Ambiguity: Capable of resolving more complex ambiguities.

LALR(k)-Parser

Definition

A grammar is LALR(1) if it can be parsed using a Look-Ahead LR parser with 1 lookahead token and a simplified set of LR(1) states.

Features

- Efficient LR Parsing: Combines the power of LR(1) parsing with more efficient state management.
- Reduction of States: Reduces the number of states compared to full LR(1) parsing.

Conditions

- LR(1) Compatibility: Every LALR(1) grammar is LR(1), but not every LR(1) grammar is LALR(1).
- State Merging: Simplifies state management but may introduce conflicts.

yello