



Course	Compiler Design	Year / Semester	III / I
Topic	Methodologies for Parsing Techniques	Innovative Method Chosen	Brainstorming

Brainstorming

Brainstorming is a group creativity technique that is often used to find a solution to a specific problem. This is accomplished by gathering and recording new ideas from team members in a free-flowing manner. Brainstorming sessions are usually made up of a handful of core team members, and typically are led by a director or facilitator.

In compiler design, parsing is the process of analyzing the syntax of a program written in a programming language to determine its structure according to the rules of a formal grammar. There are several methodologies and techniques for parsing, each with its own advantages and disadvantages. Here are some of the common methodologies for parsing:

1. Recursive Descent Parsing:

- Recursive descent parsing is a top-down parsing technique where the parser starts from the root of the parse tree and works its way down to the leaves.
- Each non-terminal in the grammar corresponds to a parsing function.
- It is easy to implement and understand, especially when the grammar is simple and LL(1) (Left-to-right, Leftmost derivation, 1 lookahead).
- However, it can be inefficient for complex grammars and may lead to issues with left recursion.

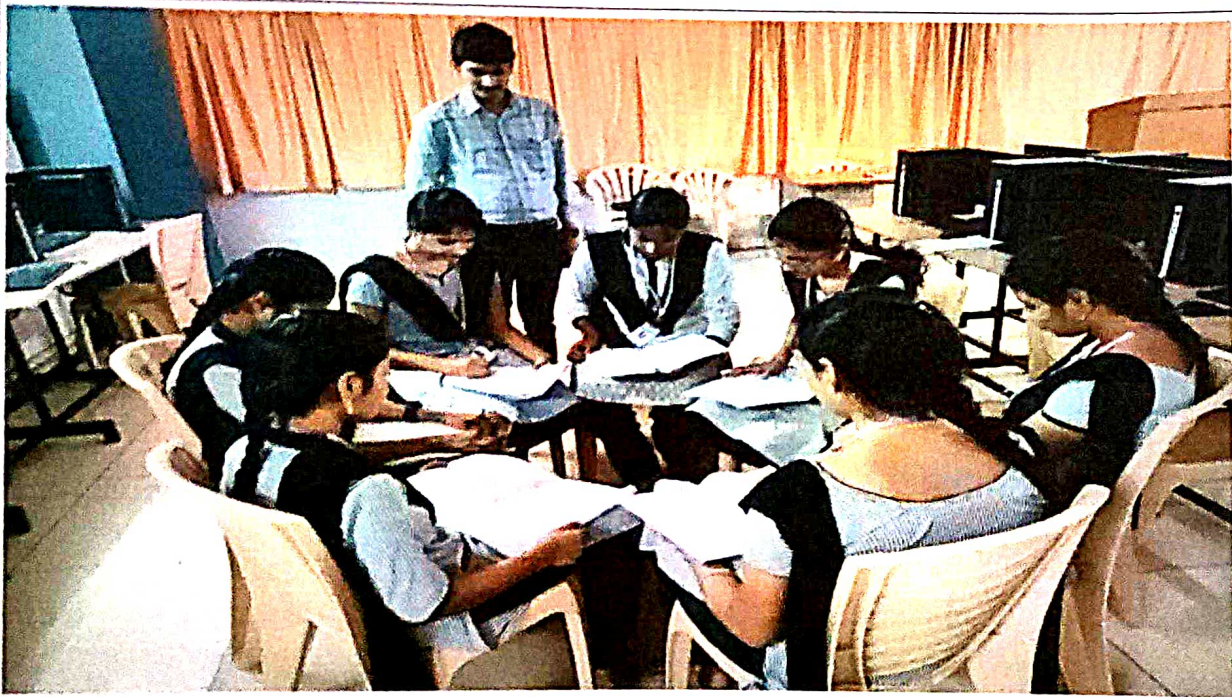
2. LL Parser (Table-driven LL parsing):

- LL parsers use a table-driven approach to parsing where parsing decisions are made based on the current input symbol and the top of the parsing stack.
- LL parsers are commonly used for implementing recursive descent parsers and are efficient for parsing LL(k) grammars.
- LL parsers are used in many parser generators like ANTLR, JavaCC, etc.

3. LR Parsing:

- LR parsing is a bottom-up parsing technique where the parser starts from the input symbols and works up to the start symbol of the grammar.
- LR parsers can handle a larger class of grammars than LL parsers, including left-recursive grammars.
- LR parsers are typically more powerful but also more complex to implement compared to LL parsers.

LR parsers can be further categorized into SLR, LALR, and CLR parsers, with each category having different levels of lookahead and parsing power.



Sw
Coordinator

[Signature]
Head of the Department
Head of the Department
Department of Computer Science & Engineering
D.N.R. College of Engineering & Technology
BHIMAVARAM-534 202.