

Differentiate between top-down and bottom-up parser. Under which conditions predictive parsing can be constructed for a grammar ?

Aspect	Top-Down Parser	Bottom-Up Parser
Approach	Starts from the highest level and works downwards.	Starts from the lowest level and builds upwards.
Analogy	Reading a story: starts from the title and reads down.	Solving a puzzle: starts with pieces, builds upwards.
Parsing Strategy	Begins with the start symbol and expands it.	Begins with terminal symbols, combines to form the start symbol.
Prediction Capability	Less predictive without lookahead.	More predictive with proper lookahead.
Example	Recipe: starts with dish name, breaks down to steps.	Sentence: starts with words, builds to complete sentence.
Efficiency	Can be less efficient due to backtracking.	Generally more efficient with proper algorithms.

Predictive parsing can be constructed if the following condition holds :

1. Every grammar must be recursive in nature.
2. Each grammar must be left factored.

Related Posts:

1. What are the types of passes in compiler ?
2. Discuss the role of compiler writing tools. Describe various compiler writing tools.
3. What do you mean by regular expression ? Write the formal recursive definition of a regular expression.
4. How does finite automata useful for lexical analysis ?
5. Explain the implementation of lexical analyzer.
6. Write short notes on lexical analyzer generator.
7. Explain the automatic generation of lexical analyzer.

Differentiate between top-down and bottom-up parser. Under which conditions predictive parsing can be constructed for a grammar ?

8. Explain the term token, lexeme and pattern.
9. What are the various LEX actions that are used in LEX programming ?
10. Describe grammar.
11. Explain formal grammar and its application to syntax analyzer.
12. Define parse tree. What are the conditions for constructing a parse tree from a CFG ?
13. Describe the capabilities of CFG.
14. What is parser ? Write the role of parser. What are the most popular parsing techniques ? OR Explain about basic parsing techniques. What is top-down parsing ? Explain in detail.
15. What are the common conflicts that can be encountered in shift-reduce parser ?
16. Differentiate between recursive descent parsing and predictive parsing.
17. What is the difference between S-attributed and L-attributed definitions ?
18. What is intermediate code generation and discuss benefits of intermediate code ?
19. Define parse tree. Why parse tree construction is only possible for CFG ?
20. Discuss symbol table with its capabilities ?
21. What are the symbol table requirements ? What are the demerits in the uniform structure of symbol table ?