

LEXICAL ANALYSIS: (also called scanner)

- Reach the successive line.
- Breaks into Hums like identifier, operator, de-limiter.
- constructs symbol table
- symbol table allocates memory.
- Lexical Analysis is the first phase of the compiler also known as a scanner. It converts the high level input program into a sequence of Tokens.

\* Lexical Analysis can be implemented with the Deterministic finite Automata.

\* The output is sequence of tokens that is sent to the parser for syntax analysis.

Syntactic Analysis: (also called parsing)

- Expression, statement, declaration identified.
- Aided by formal grammar of programming language

Syntactic analysis or parsing or syntax analysis is the third phase of NLP. The purpose of this phase is to draw exact meaning from line text. Syntax analysis checks the text for meaningfulness comparing to the rules of formal grammar. For example, the sentence like, "not icecream" would be rejected by semantic analyzer.

Semantic analysis: (also called Phase bridge).

→ Analysis Phase of Syntax.

→ Last phase of translation is code generation.

Tools:

LEX - Lexical Analyser

YACC - Yet Another Compiler

Semantic Analysis makes sure that declarations and statements of program are semantically correct. It is a collection of procedures which is called by parser as and when required by grammar.

Both Syntax tree of previous phase and symbol table are used to check the consistency of the given code. Type checking is an important part of semantic analysis where compiler makes sure that each operator has matching operands.