Complier design phase

Lexical analysis

Lexical analysis is the first phase of a compiler. Lexical analysis also called a scanning, scans a source program from left to right character by character and group them into token having a collective meaning. It performs two important tasks.
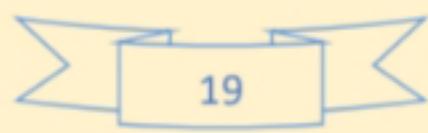
First, it scans a source program character by character from left to right and group them into tokens (or synthetic elements). It's token or basic synthetic element represents a logically. Cohesive sequence of character such as identifier (also called variables) Multi character operator <=, etc. The output of this phase goes to the next phase. i.e. syntax analysis, or parsing.
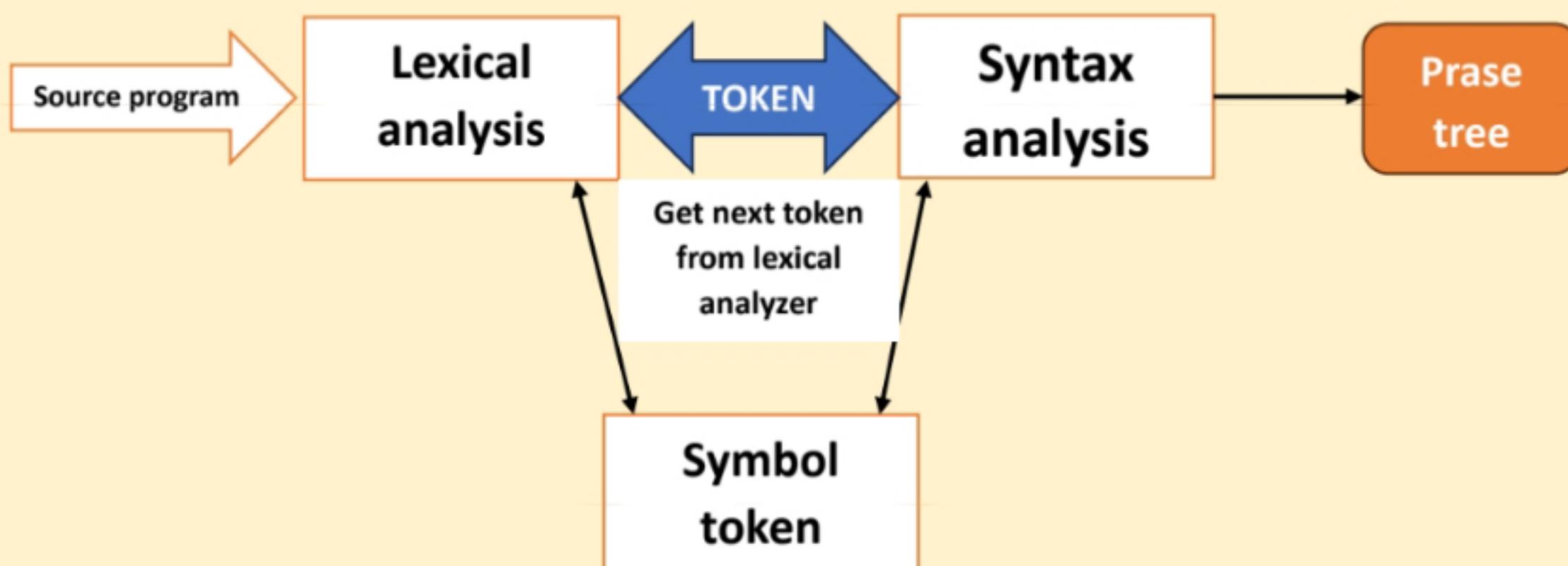
The second task performed during lexical analysis is to make a entry of token into a symbol table If it is not there.

Some other tasks performed during lexical analysis are:

➢ To remove all comments tab blank space and machine characters.
➢ To produce error messages (Also called diagnostics) occurred in a source program.

Syntax analysis.

Every long ways whether it is a programming language of any natural language follows certain grammatical tools that define Send taxi call a structure of a long ways.