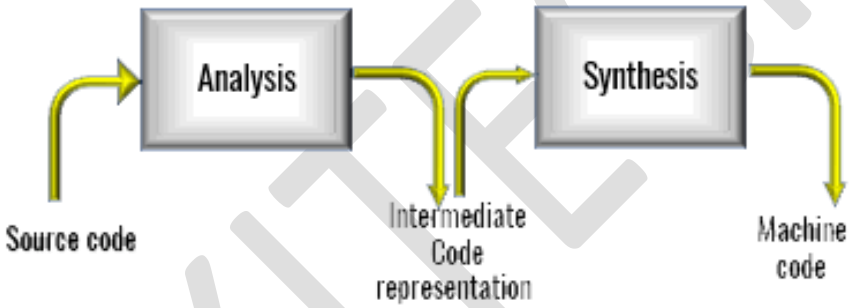
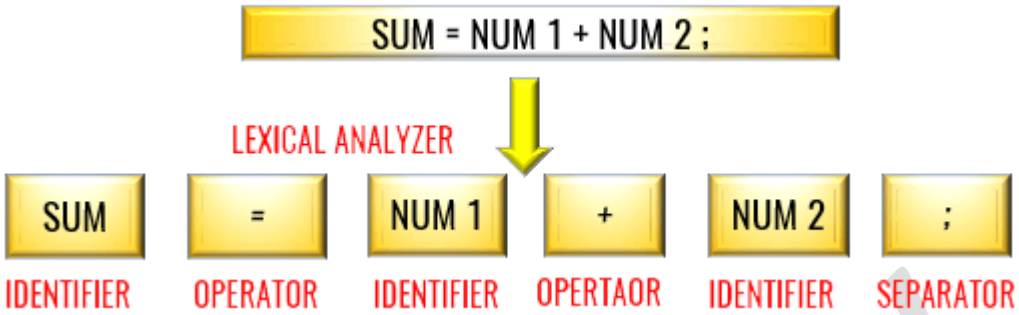
Phases of Compiler

- A compiler operates in phases. A phase is a logically interrelated operation that takes source program in one representation and produces output in another representation.
- There are two phases of compilation.
    - Analysis (Machine Independent/Language Dependent)
    - Synthesis (Machine Dependent/Language independent)



Lexical Analysis

- Lexical analysis is the first phase of a compiler. It takes the modified source code from language preprocessors that are written in the form of sentences. The lexical analyzer breaks these syntaxes into a series of tokens, by removing any whitespace or comments in the source code.



Syntax Analysis

- The second stage of translation is called syntax analysis or parsing. In this phase expressions, statements, declarations are identified by using the results of lexical analysis.
- A Syntax analyzer creates the syntactic structure of the given program.

For More Details Click Here:

https://www.wikitechy.com/tutorials/compiler-design/phases-of-compiler

