Regular Expressions Applications

- Finite state machines are a useful model for many important kinds of hardware and software. e.g.
  - Software for designing digital circuits
  - Lexical analyzer of a compiler
  - Searching for keywords in a file or on the Web
  - Software for verifying finite state systems, such as communication protocols
  - Operating System (UNIX grep)
  - Text Editors
  - Markup Languages (HTML, XML)
  - Natural Language Processing

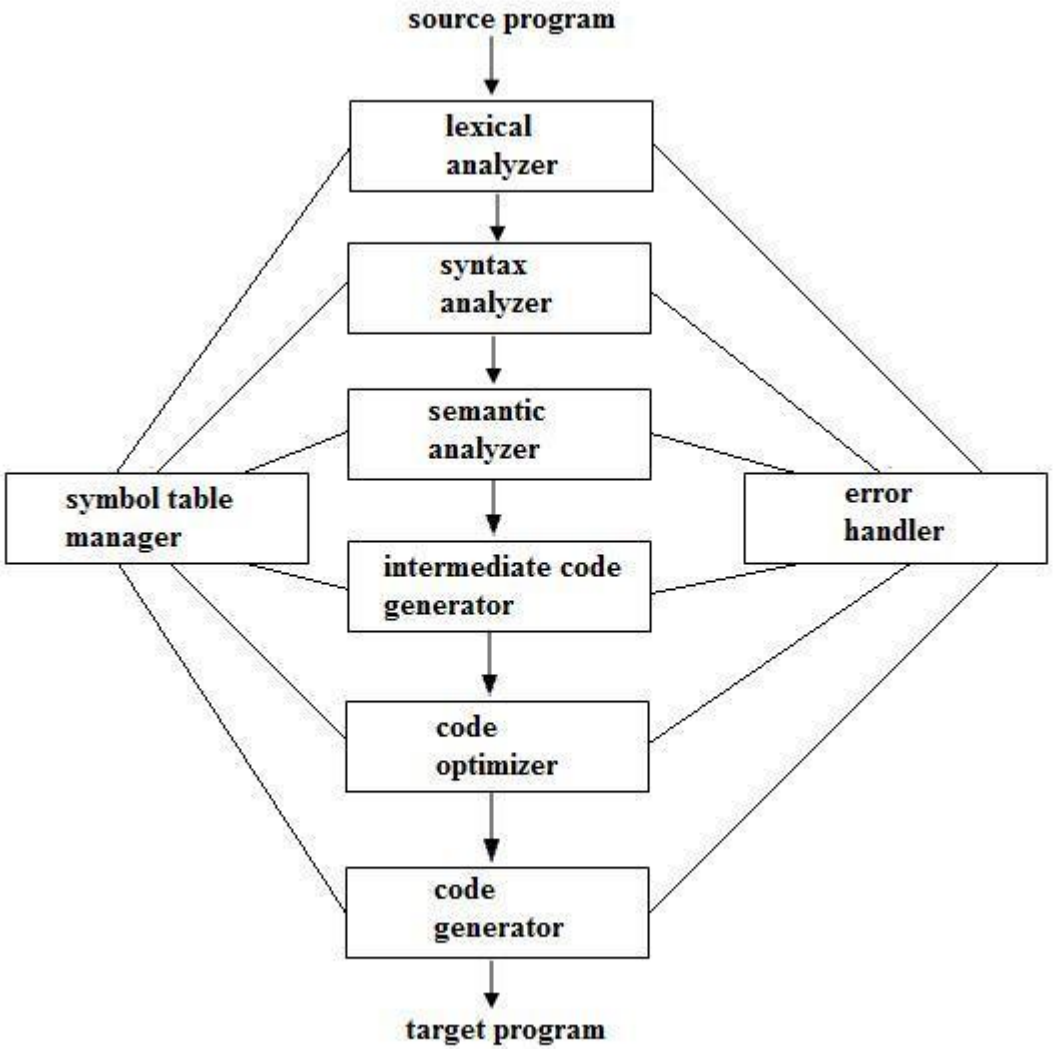
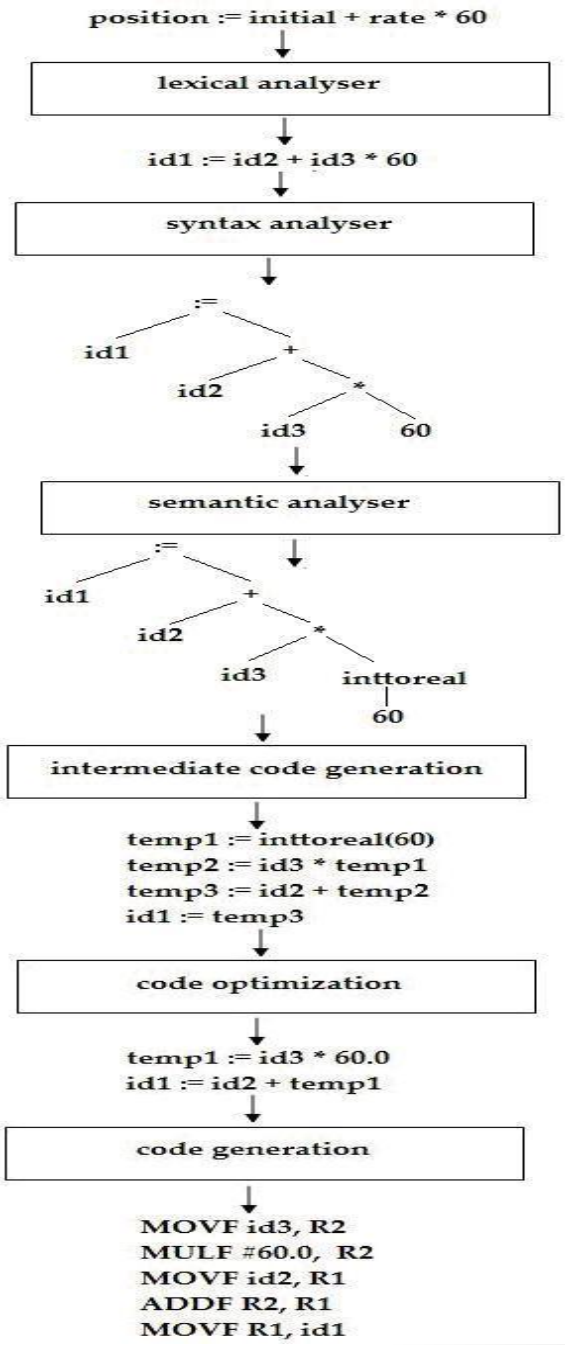


Fig 1.5 Phases of a compiler



Lexical Analyzer

- **Lexical Analyzer** reads the source program character by character and returns the *tokens* of the source program.
- A *token* describes a pattern of characters having same meaning in the source program. (such as identifiers, operators, keywords, numbers, delimeters and so on)

Ex: newval := oldval + 12 => tokens:

newval	identifier
:=	assignment operator
oldval	identifier
+	add operator
12	a number

- Puts information about identifiers into the symbol table.
- Regular expressions are used to describe tokens (lexical constructs).
- A (Deterministic) Finite State Automaton can be used in the implementation of a lexical analyzer.