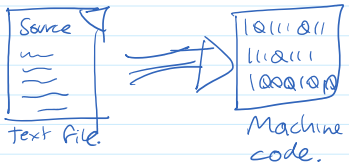
2 Programming Language Implementation

Friday, August 25, 2023 10:59 AM

è How are programming languages implemented?

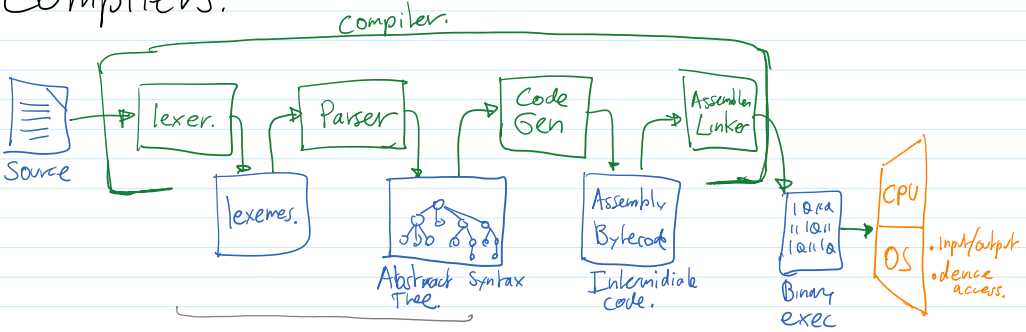
- Compilers
- Interpreters
- Hybrids: Combination of both

- the objective:



- Machine language is relatively simple.
  - fetch value from memory to cpu
  - store value from CPU to memory
  - perform operation(s) on two values in CPU (Add, Multiply, Subtract)
  - if two values are equal, skip, or jump
  So many instructions

- Compilers.



lexer:- split input into atomic components
e.g. apple = banana * (orange + 5)

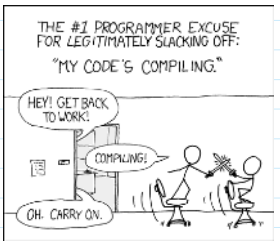
Parser:- recognize weather lexemes form a valid program under the rules of the language.

apple = 3dfx (* orange + [] 7 }

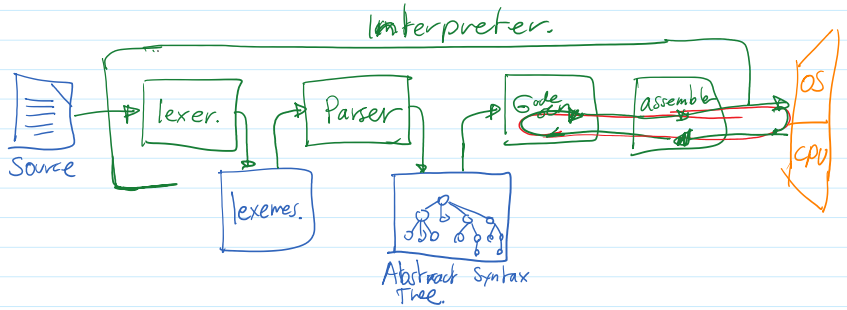
E.G. FORTRAN, C, C++, Pascal, D, Go, Rust.

Pro: - speed.
- intellectual property protection.

Cons: - Portability.
- compilation can be slow.



- INTERPETERS



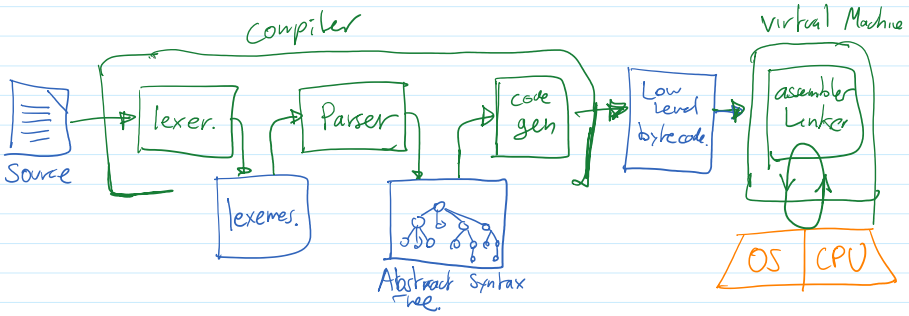
REPL :- Read-Evaluate-Print-Loop.

E.G. Python, JavaScript, ML, Prolog, Lisp, BASIC, Ruby, Perl

Pros: + interactivity REPL
+ Speedy program development.
+ Portability.

Cons: - Speed of execution.
- Code is not hidden.

- HYBRID



E.G. Java, C#, Scala, Kotlin

Pros: + Speed > Interpreters.
+ portability
+ I.P. Protection.
+ inter-Language operability

Cons - No R.E.P.L.
- Less speed than Binary

- EOF.