

# Overall Structure of a generic Compiler

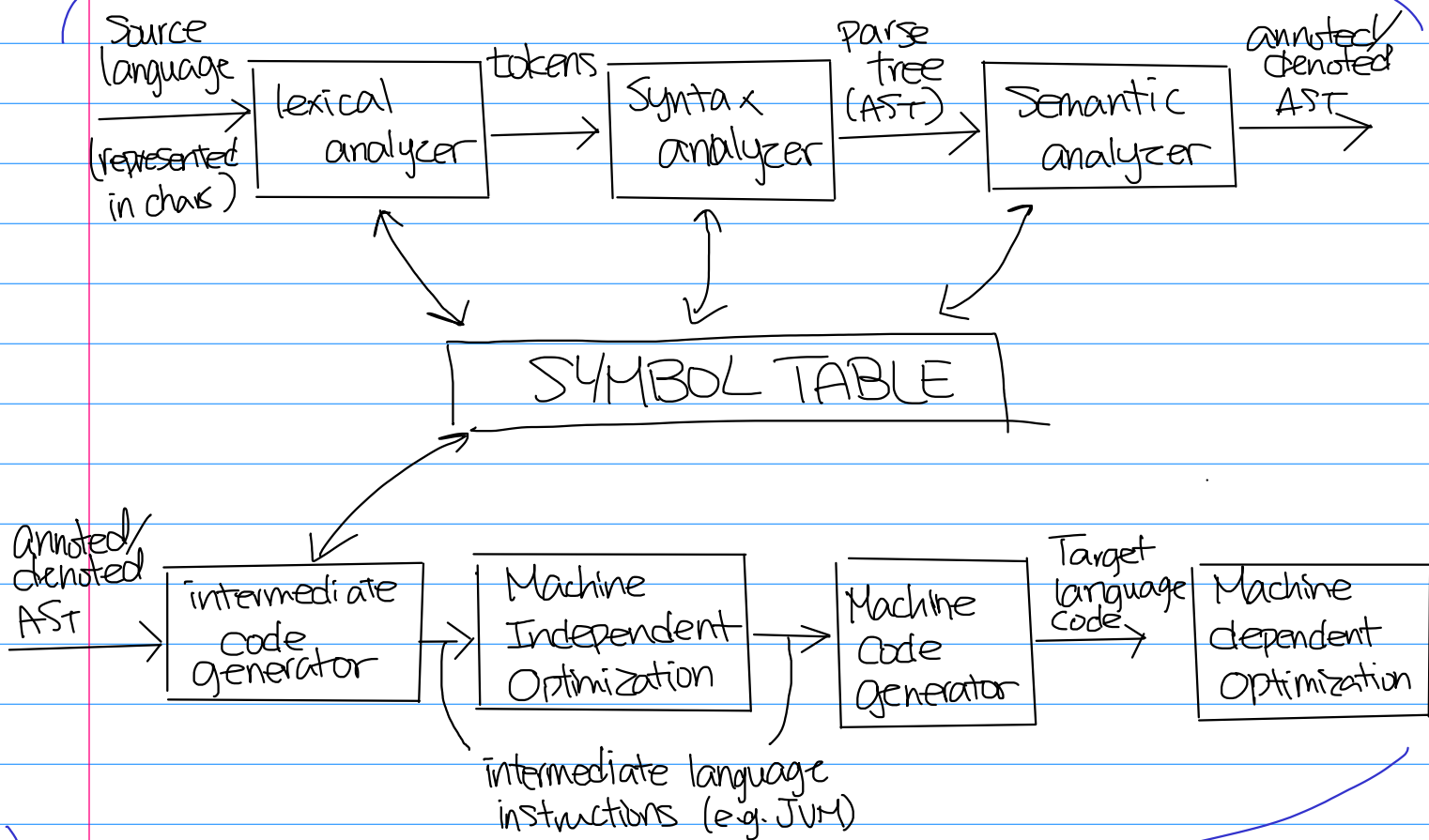
high-level  
computing lang.

assembly or  
machine language.

## Pipes & Filters

- pattern
- similar to unix shell

## ANALYSIS



## SYNTHESIS

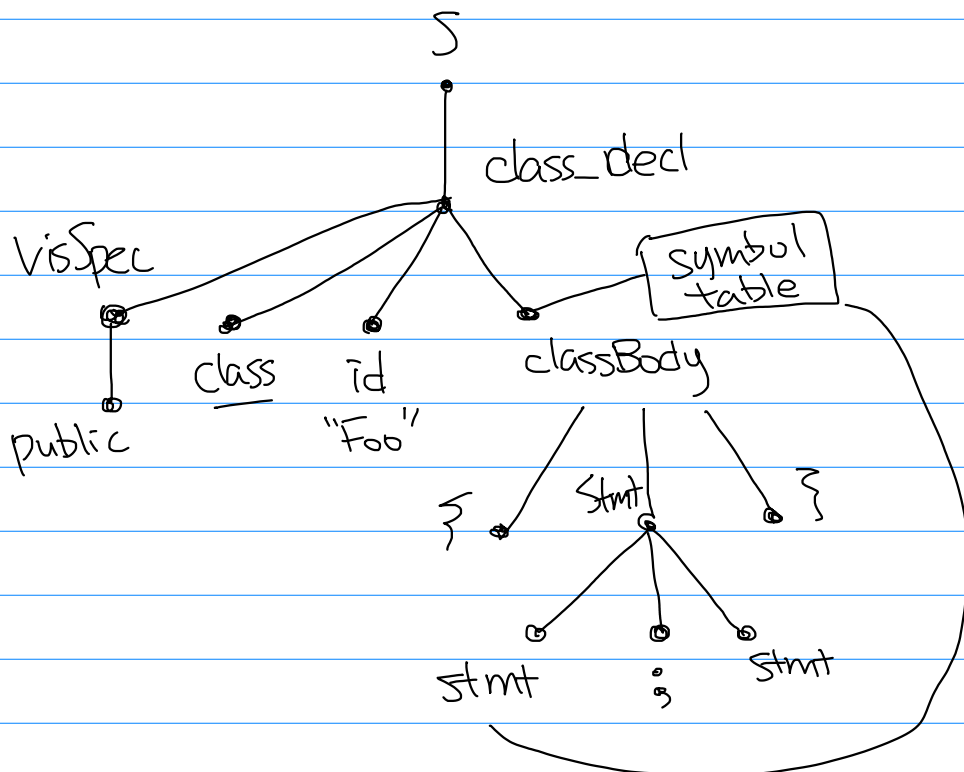
# Syntax Analyzer]

Grammar Example :

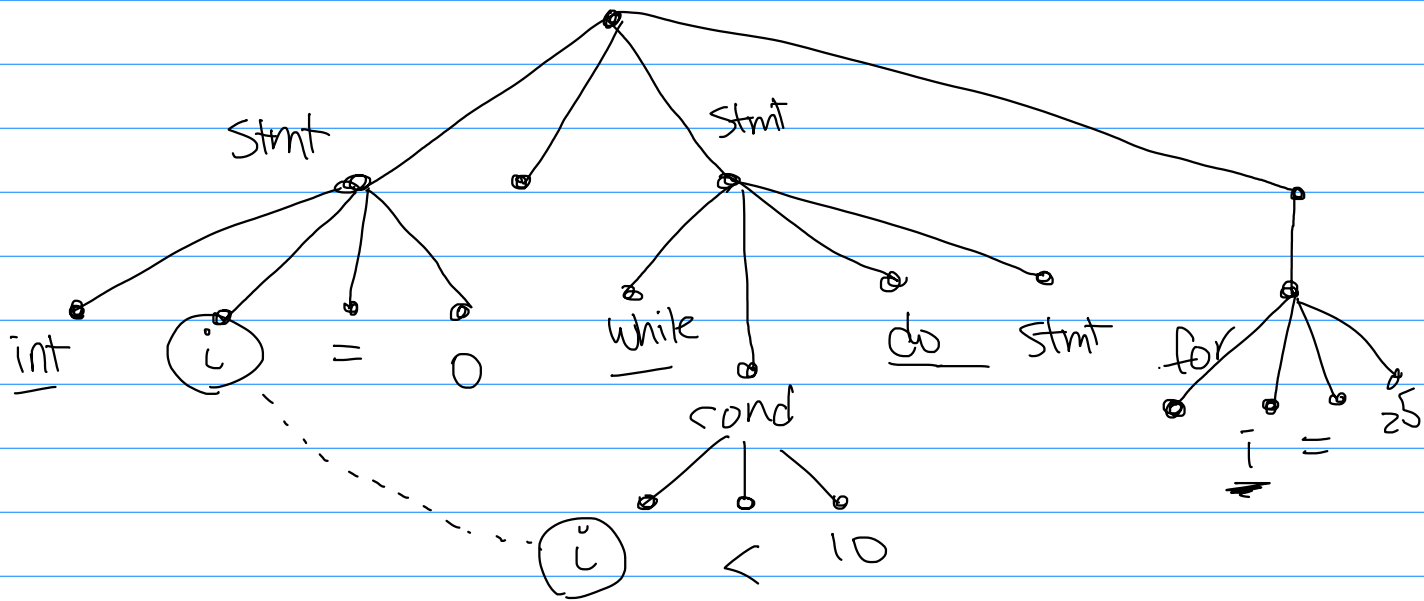
- stmt  $\rightarrow$  if condition then stmt else stmt
- $\rightarrow$  while condition do stmt
- $\rightarrow$  variable = expr

— are explicit symbols or keywords

- $S \rightarrow$  class\_decl  $\leftarrow$  visibility specifier
- class\_decl  $\rightarrow$  visSpec class id classBody
- visSpec  $\rightarrow$  public { private } protocol
- classBody  $\rightarrow$  { stmt }



# Semantic Analyzer



link with  $\frac{i}{T}$

## Semantic Analyzer

- scoping
- detects type error  $\Rightarrow$  type check
- variable analysis
- misc. rule enforce.