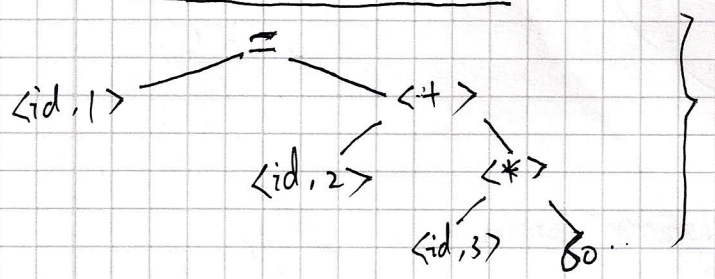
↓

output of the lexical analyzer:

<id,1> <=> <id,2> <+> <id,3> <*> 60.

= input of the syntax analyzer ⟶ context-free grammer.



parse trees.

↓

semantic analyzer

- type checking e.g. (float) 60 → 60.0
  report errors.

↓

Code generation.

- intermediate code.

  t1 = (float) 60.
  t2 = id3 * t1
  t3 = id2 + t2
  id1 = t3

  optimize ↓

  t = id3 * 60.0
  id1 = id2 * t.

- target machine code