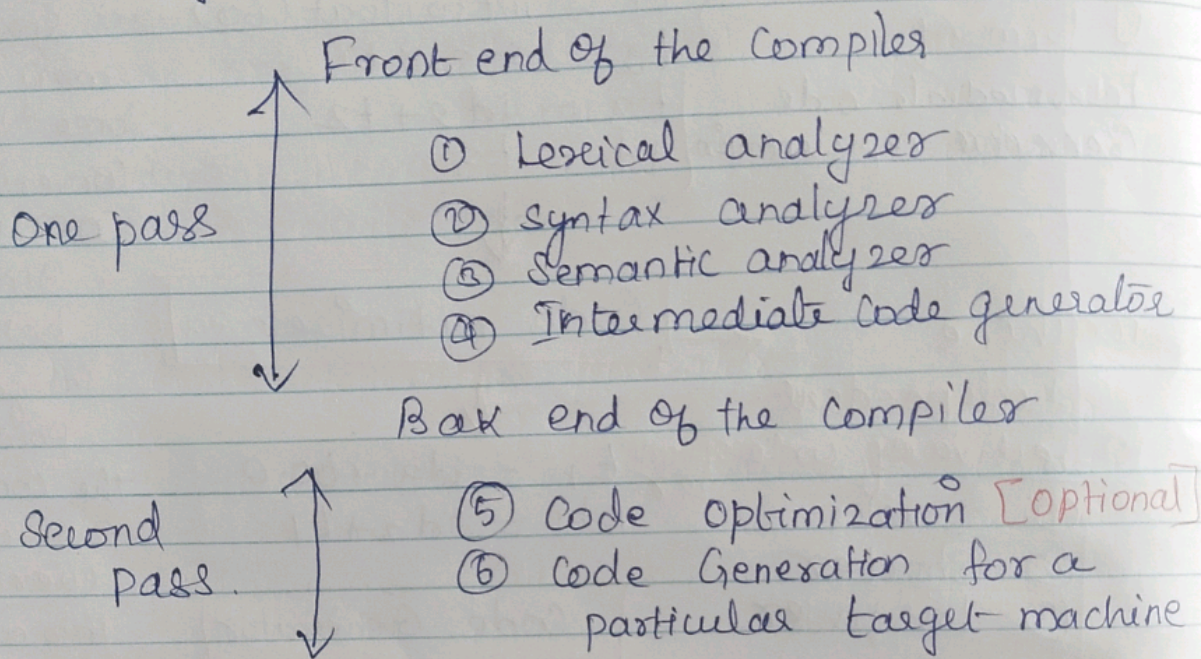


The Grouping of Phases into Passes:-

- ✓ Activities of several phases are grouped into passes.



Compilers are constructed carefully to interface Intermediate code representation of a particular language with the backend of target machine.

Advantage:-

✓ Design ~~one~~ compilers for different source language for one target machine

⇐ Similarly different target machine can be combined with one front end.

Compiler Construction Tools:-

Compiler writers use modern software development environments containing tools like,

- * Language editor
- * debugger
- * Version Manager
- * Profilers
- * Test harnesses

Other than these tools, other compiler construction tools uses specialized languages for specifying and implementing specific components and may use sophisticated algorithms.

Required properties of compiler construction tools:-

- (i) It should hide the details of generation algorithms
- (ii) It should produce components that can be easily integrated into the remainder of the compiler.

Some commonly used tools:-

1. Parser Generator:-

Produce syntax analyzer from a grammatical description of a programming language.

2. Scanner Generators:-

produces lexical analyzers from a regular expression description of the tokens of a language.

3. Syntax-directed translation engine:-

produces collections of routines for walking a parse tree and generating intermediate code.

4. Code Generator generator:-

produces code generator from a collection of rules for translating each operators of the intermediate language into machine language for a target machine.

5. Data-flow analysis engines:-

Facilitates the gathering of information about how values are transmitted from one part of a program to each other part.

Data flow Analysis \Rightarrow Key part of Code optimization.

6. Compiler Construction Toolkits:-

Provides integrated set of routines for constructing various phases of Compiler.