School of Computing Science

Course Code: CSEG3015

Course Name: Compiler Design

Lecture No. 3
Unit-1st
(Introduction)
Compiler Design
(CSEG3015)

Objectives

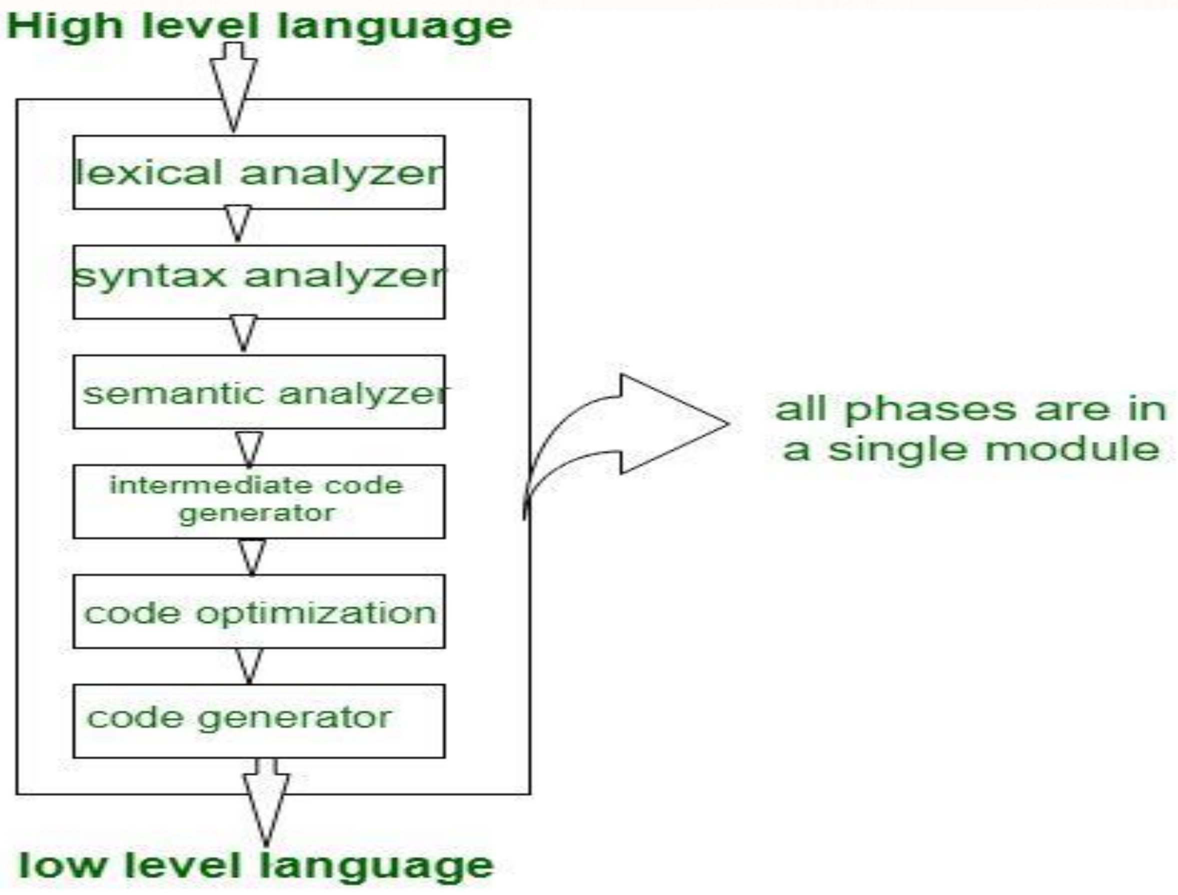
- Passes
- Bootstrapping

Passes

- **Passes**: We already know about all the Phases of Compiler design, now the Compiler Passes.
- A Compiler pass refers to the traversal of a compiler through the entire program.
- Compiler pass are two types: **Single Pass** Compiler and **Two Pass** Compiler or **Multi Pass** Compiler.

Single Passes Compiler

- If we combine or group all the phases of compiler design in a single module known as single pass compiler.
- In the given diagram, six phases are grouped in to a single module
- One pass compiler read the program only once & then translate it.



Advantages & Disadvantages of Single Pass Compiler

- A one pass/single pass compiler is that type of compiler that passes through the part of each compilation unit exactly once.

- Single pass compiler is faster and smaller than the multi pass compiler.

- As a disadvantage of single pass compiler is that it is less efficient in comparison with multipass compiler. (Less efficient code optimization and code generation.)

- Single pass compiler is one that processes the input exactly once, so going directly from lexical analysis to code generator, and then going back for the next read.

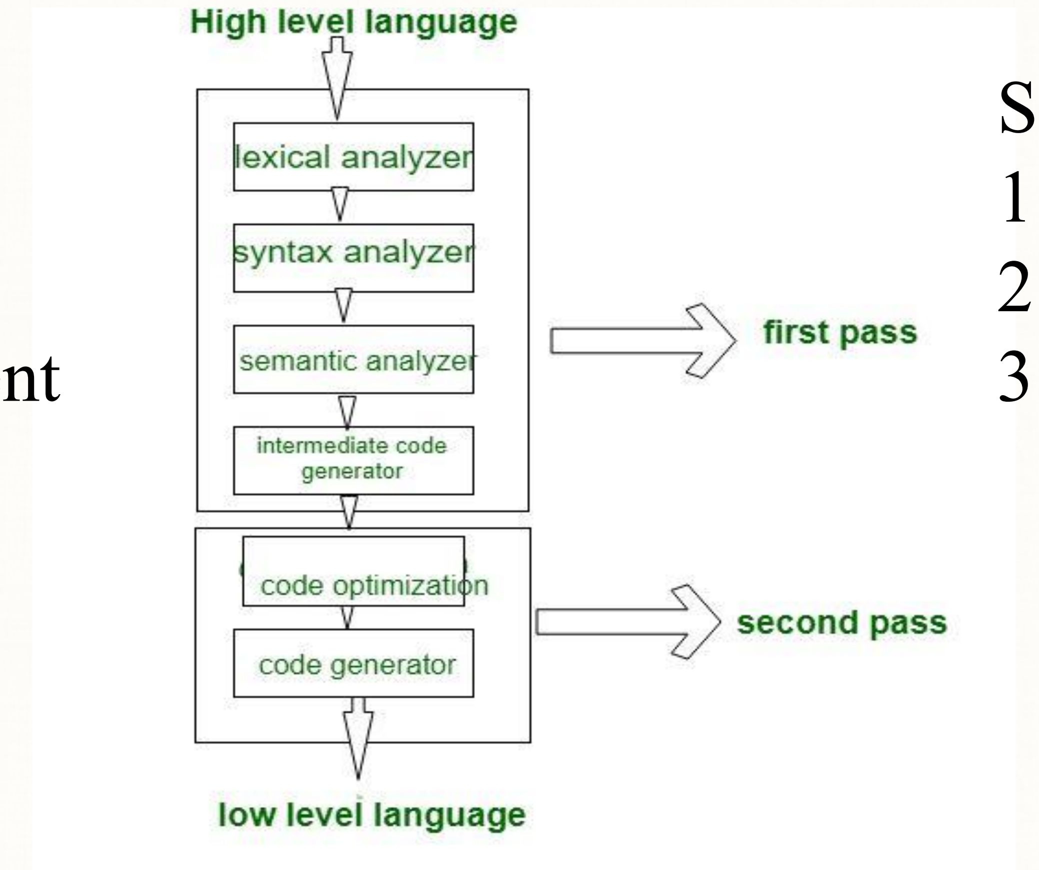
Advantages & Disadvantages of Single Pass Compiler

- It reads the program only once, and transform it at the same time.
- Less efficient code optimization and code generation.
- Also called “Narrow compiler”. As it has limited scope.
- Large memory is required by the compiler..

Multi Pass Compiler

- A Two pass/multi-pass Compiler is a type of compiler that processes the source code multiple times. In multipass Compiler, phases can be divided in two pass as:

First Pass:
1. Front end
2. Analytic part
3. Platform independent



Second Pass:
1. Back end
2. Synthesis Part
3. Platform Dependent

Advantages & Disadvantages of Multi Pass Compiler

- It reads the program several times, each time transforming it into different form.
- Slower, As more number of passes means more execution time.
- Better code optimization and code generation.
- Also called “wide compiler”. As they can scan each and every portion of program.
- Memory occupied by one pass can be reused by subsequent pass, therefore small memory is required by compiler.

Bootstrapping

- It is a process to build new compiler from existing one

- A compiler is characterized by the three languages

1. Source Language

2.Object / Target Language

3.Language in which it is written

Bootstrapping

- Direct Compiler: C_S^{LA} ----> C_A^{SA} -----> C_A^{LA}

- Cross Compiler

  - C_L^{LB} ----> C_A^{LA} -----> C_A^{LB}

  - C_L^{LB} ----> C_A^{LB} -----> C_B^{LB}

- A compiler may run on one machine and produce the target code for another machine. Such a compiler is often called as cross compiler.

References

References:

Text Books

1. ALFRED V AHO, JEFFREY D ULLMAN “Principles of Compiler Design”.
2. V Raghavan, “ Principles of Compiler Design”, TMH
3. Kenneth Louden,” Compiler Construction”, Cengage Learning

Reference Books

1. Aho, Sethi & Ullman, "Compilers: Principles, Techniques and Tools”,Pearson Education 2
2. Charles Fischer and Ricard LeBlanc,” Crafting a Compiler with C”, Pearson Education

Thank You