System Software (5KS03)

Unit 1 : Introduction to Compiling

Lecture : 1 Introduction to Compiling: Phases of a compiler,

A S Kapse,
Assistant Professor,
Department Of Computer Sci. & Engineering
Anuradha Engineering College, Chikhli

Contents…

- Introduction to Compilers

Objectives…

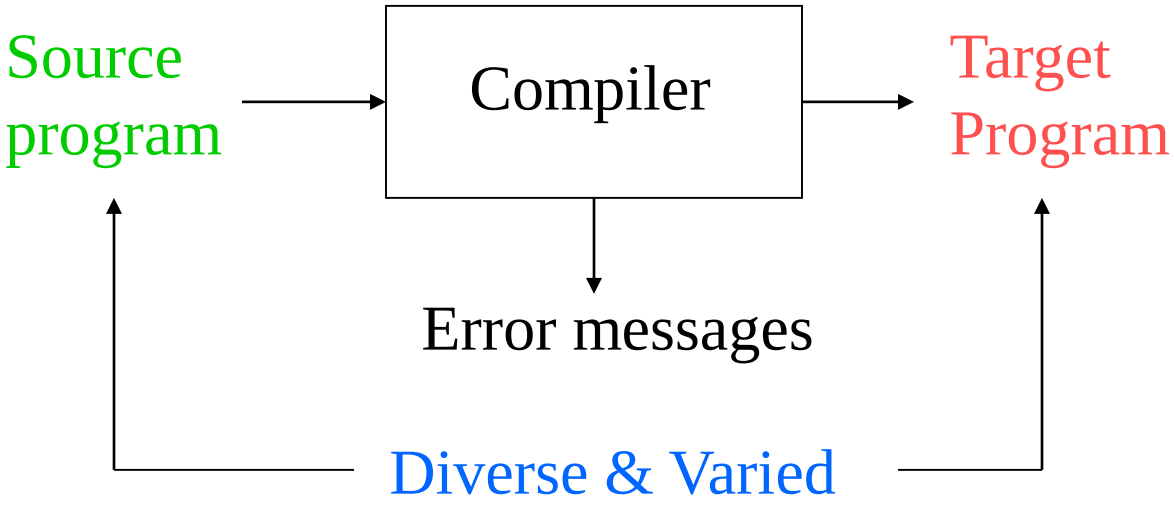
- Upon completion of this lecture, you will be able
- ✓ To understand the basics Compiler
- ✓ To understand analysis and synthesis.
- ✓ To understand use and purpose of compiler

Review…./ Concepts

- What do you mean by Software?
- What do you mean by Operating System?
- What do you mean by system?
- What do you mean by compiler?

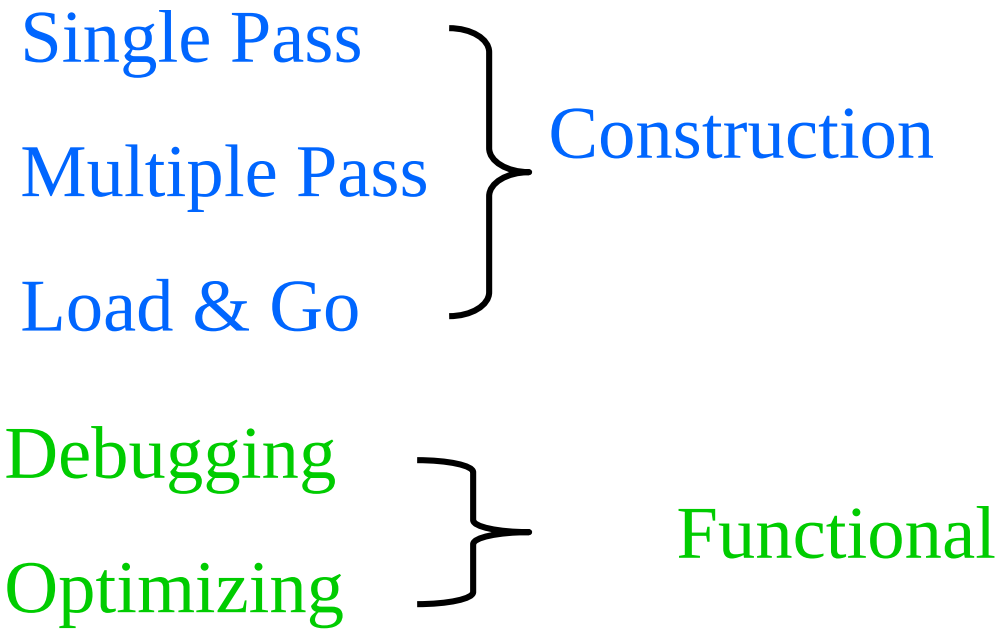
Introduction to Compilers

- As a Discipline, Involves Multiple CS&E Areas
  - Programming Languages and Algorithms
  - Theory of Computing & Software Engineering
  - Computer Architecture & Operating Systems
- Has Deceivingly Simplistic Intent:



Classifications of Compilers

- Compilers Viewed from Many Perspectives



- However, All utilize same basic tasks to accomplish their actions

The Model

- The TWO Fundamental Parts:

  Analysis: Decompose Source into an intermediate representation

  Synthesis: Target program generation from representation

- We Will Discuss Both in This Class, and FOCUS on analysis.

Important Notes

- Today: There are many Software Tools for helping with the Analysis Part. This Wasn't the Case in Early Days. (some) analysis is also important in:
  - Structure / Syntax directed editors: Force "syntactically" correct code to be entered
  - Pretty Printers: Standardized version for program structure (i.e., blank space, indenting, etc.)
  - Static Checkers: A "quick" compilation to detect rudimentary errors
  - Interpreters: "real" time execution of code a "line-at-a-time"

Important Notes

- Compilation Is **Not** Limited to Programming Language Applications
  - Text Formatters
    - LATEX & TROFF Are Languages Whose Commands Format Text
  - Silicon Compilers
    - Textual / Graphical: Take Input and Generate Circuit Design
  - Database Query Processors
    - Database Query Languages Are Also a Programming Language
    - Input is compiled Into a Set of Operations for Accessing the Database

Overview and History (1)

- Cause
  - Software for early computers was written in assembly language
  - The benefits of reusing software on different CPUs started to become significantly greater than the cost of writing a compiler

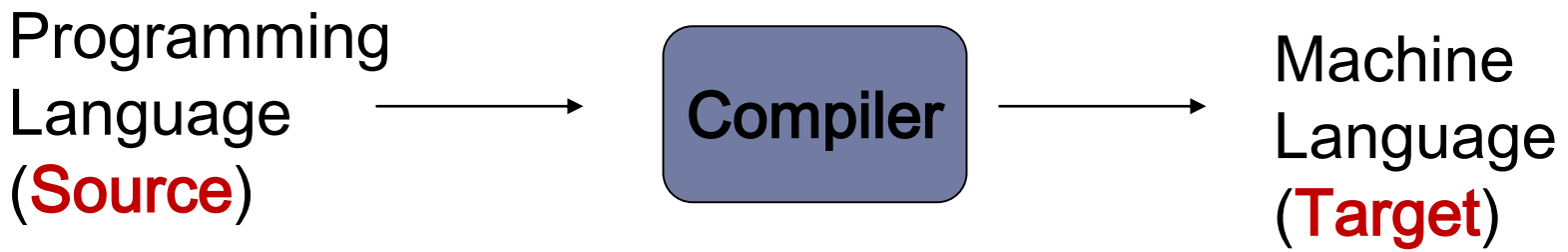
- The first real compiler
  - FORTRAN compilers of the late 1950s
  - 18 person-years to build

Overview and History (2)

- Compiler technology
  - is more broadly applicable and has been employed in rather unexpected areas.
    - Text-formatting languages,
      like nroff and troff; preprocessor packages like eqn, tbl, pic
    - Silicon compiler for the creation of VLSI circuits
    - Command languages of OS
    - Query languages of Database systems

What Do Compilers Do (1)

- A compiler acts as a translator, transforming human-oriented programming languages into computer-oriented machine languages.
- Ignore machine-dependent details for programmer



What Do Compilers Do (2)

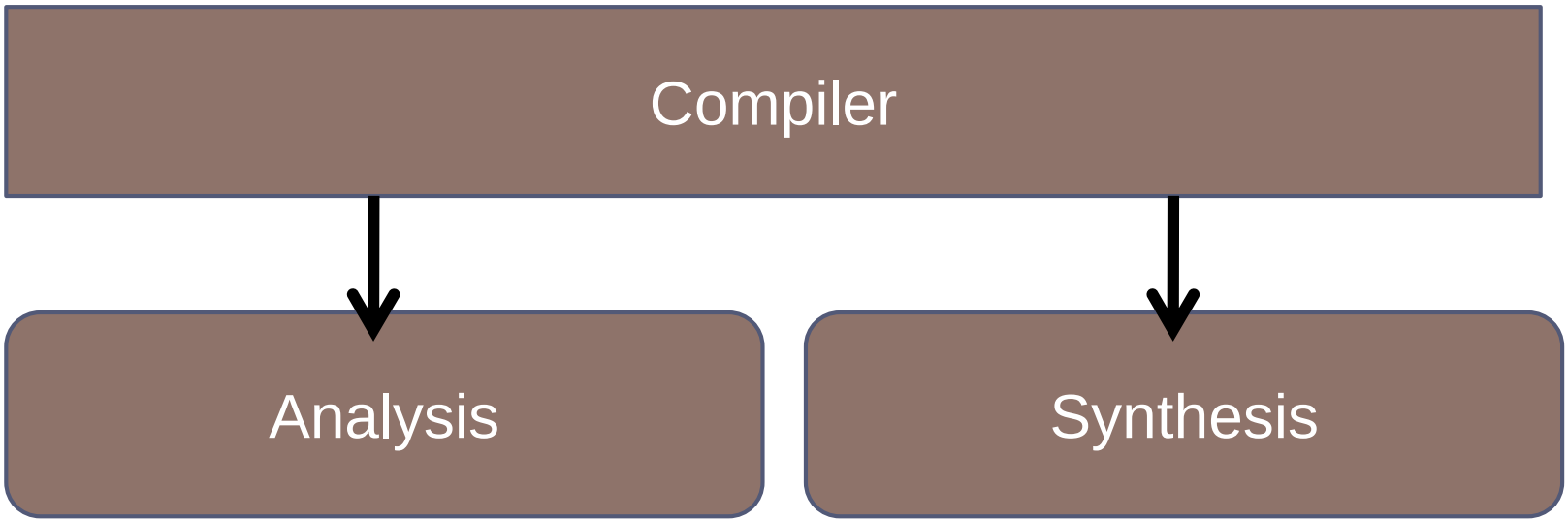
- Compilers may generate three types of code:
  - Pure Machine Code
    - Machine instruction set without assuming the existence of any operating system or library.
    - Mostly being OS or embedded applications.
  - Augmented Machine Code
    - Code with OS routines and runtime support routines.
    - More often
  - Virtual Machine Code
    - Virtual instructions, can be run on any architecture with a virtual machine interpreter or a just-in-time compiler
    - Ex. Java

What Do Compilers Do (3)

- Another way that compilers differ from one another is in the format of the target machine code they generate:
  - Assembly or other source format
  - Relocatable binary
    - Relative address
    - A linkage step is required
  - Absolute binary
    - Absolute address
    - Can be executed directly

The Structure of a Compiler (1)

- Any compiler must perform two major tasks



  - ***Analysis*** of the source program
  - ***Synthesis*** of a machine-language program

Video

1. Introduction to Compiler.

Questions..

1. **Define Compiler?**

Homework..

1. What is Analysis and synthesis?