Introduction

Monday, February 1, 2021 8:24 AM

Humans -> PL -> HLL

PL-> Binary (machine) language
Process of translation is done by a program (**compiler**/interpreter)

Source program (PL) -> Compiler -> Target program (assembly/machine laguage program)

Translated as a whole (C, C++)

Assembler language program -> assembler -> machine language

Input -> Target program -> output

Interpreter does not generate target program (line by line translation), Python, Ruby, PHP

Source program + input -> Interpreter -> output

Source program -> Java Compiler -> Intermediate representation (Bytecode machine independent format) + input -> JVM (interpreter) -> output

Software systems required to compile a program:

- Preprocessor (in C): preprocessing in source file
  - #define MAX 10

  MAX is replaced by 10

Translation:
Source Program -> Preprocessor -> Compiler -> Assembly language -> Assembler ->relocatable object code -> linker -> loader

Code must be relocatable:

- If program swapped out and in of memory, it may be placed in a different position in the primary memory.
- No absolute code. Program can be dynamically placed and properly executed anywhere in the memory (jump statements and their address references should be valid)
- Library codes should be linked to the relocatable object code: **linker**
- Final object code to be loaded into the primary memory for execution: **loader**

Translation divided into multiple **phases**. Each phase has a simple, well-defined task to do. With each step, the program becomes more machine-specific (decrease in levels of abstraction).

HLL: operands, operators, data types, statements
LLL: Registers, memory, operands, jumps, labels

Analysis + Synthesis = Compilation

Analysis: front-end

1. Lexical analysis(scanning):
   - read and group characters, find and check words that are acceptable in the language
   - (**LEX: lexical analyzer generator;** given a specification for a language, generates a lexical analyzer that analyses an input and finds if it is valid according to the language)
   - Output: stream of words in the language
2. Syntax analysis: identifies whether a given input belongs to the language syntactically (grammar)

Synthesis: back-end