Chapter 1 - Introduction
Exercises

February 16, 2018

Exercises

1.1 Language Processors

Exercise 1.1.1

The difference between a compiler and an interpreter is that a compiler translates source code to target code, whereas an interpreter executes the source code directly.

Exercise 1.1.2

(a.) The code output by a compiler will be much faster in execution than the performance of the interpreter.
(b.) Since the interpreter executes statements one at a time, it is easier to debug an interpreted program since you will know on exactly which command the program fails.

Exercise 1.1.3

The advantage to producing assembly code versus machine code after compilation is that assembly code is still readable and debug-able by a human being, as well as being easier for the compiler to produce.

Exercise 1.1.4

The advantages to using C as a compiler's target language is that C is a high-level language with many low-level capabilities, making it easy for another compiler to translate C code into assembly or machine code. Another advantage is that C is straightforward and terse, making it easy for a compiler to write uniform and correct translations from other sources.

Exercise 1.1.5

An assembler needs to translate to translate assembly code into machine code. It also needs to write that machine code into standalone, relocatable modules of machine code such that a linker and loader can take that code and use it to create an executable.

1.2 The Structure of a Compiler

Exercise 1.2.1

Exercise 1.2.2

Exercise 1.2.3