**Chapter 1 reading**

For this assignment, read chapter 1 in your textbook and answer the questions below. Your answers should be thoughtful and clear.

1. Which of the reasons for studying concepts of programming languages do you find speaks to you the most? Why? How does this relate to your previous experiences?

I think if I understand the theory behind how compilers and interpreters work I can apply it to all languages. I can use this knowledge when learning new languages and it will help me learn faster.

1. What are the important differences between readability, writability, and relaibility?

**Readability** is how easy it is to interpret the language you are looking at. This is useful when working in groups and you need to easily understand someone else’s code.

**Writability** is a measure of how easily a language can be used to create programs for a chosen problem domain. This is useful for solving a problem on the fly.

**Relaibility** is the technical uses of that language. So you might lose readability or writability but you can accomplish a lot more with that type of language.

1. What are two examples of how readability, writability, and reliability can conflict with each other in the context of programming language design?

An example of conflicting criteria that leads directly to design trade-offs, consider the case of APL. APL includes a powerful set of operators for array operands. Because of the large number of operators, a significant number of new symbols had to be included in APL to represent the operators. APL is very writable but not very readable.

The pointers of C++ can be manipulated in a variety of ways, which supports highly flexible addressing of data. Because of the potential reliability problems with pointers, they are not included in Java.

1. What are the relative advantages and disadvantages of compilation vs. interpretation?

Basically, compiled code can be executed directly by the computer's CPU. That is, the executable code is specified in the CPU's "native" language (assembly language). The code of interpreted languages however must be translated at run-time from any format to CPU machine instructions. So interpretation takes a little bit longer.