Wentworth Institute of Technology

COMP3350 Programming Languages

Homework 1

1. **Why is it useful for a programmer to have some background in language design, even though the programmer may never actually design a programming language? [5]**

It improves the capacity to generate ideas, increase the learning time of new programming languages as well as to choose programming languages when solving a problem and many more.

1. **What is the disadvantage of having too many features in a language?**

**[5]**

Problems with the compiler, gives a hard time to programmer to learn it, can cause weird behaviors if some features don’t work properly.

1. **What does *orthogonality in programming language* mean? Give one example of a lack of orthogonality in the design of C? [5]**

It means that operations don’t affect the others; they change one thing.

1. **Why is type checking the parameters of a subprogram important? [5]**

It is important because without it, errors can escalate bigger. It is easier to determine the errors faster and right away before run time so that we can handle them. Subprograms run at call time – harder to notice the errors before that.

1. **Briefly describe the three fundamental features of an object-oriented** **programming language? [5]**

Abstraction, Encapsulation, and Inheritance. “If you need it once, implement it. If you need it a second time, call it. If you need it a third time, abstract it.” Some computerphile said once. As our programs get bigger and bigger, we can use of reusable code (inheritance). Also, we don’t want others that don’t have access to our project to access it (encapsulation). And, when the program is used in many workspaces, we need to find a way to make it generic so we all can use it no matter where we are.

1. **What is the Von Neumann bottleneck? Why is it important? [5]**

A lot of data transfers, but not enough processing power to process the data. By knowing this, we can determine how to serve the CPU faster by allowing faster memory access.