Jon Bowen

David Faller

Adam Lee

In-Class Assignment 2/6/2018

Chapter 1

Problem 3, page 31

What arguments can you make for the idea of a single language for all programming domains?

* Everyone would be focused on learning the same language so there would be a common knowledge base.
* Teams would need to spend less time getting up to speed to work together.
* More code would be portable and reusable across projects.

Problem 4, 32

What arguments can you make against the idea of a single language for all programming domains?

* Demand for CS majors would be reduced since there would be less specialization.
* A language that isn’t specialized will generally be less efficient for any given task.
* The needed complexity will probably make a single generalized langue more difficult to learn.
* Less specialized languages are less expressive than generalized languages.

Problem 9, page 32

Explain the different aspects of the cost of a programming language.

* The cost of training programmers depends on language simplicity and orthogonality.
* The cost of writing programs (initial development) depends on the writability.
* The time and energy required to compile programs is also a concern, though much less so in modern day compared to the past.
* The choice of language affects the cost (e.g. time, energy, and hardware) for running programs.
* Reliability of a language affects costs through the frequency of failure. Software that fails and crashed the space shuttle has tremendous costs indeed, while JavaScript errors that require a web page to be reloaded are a small annoyance.
* Finally, and perhaps most important today, is the cost of maintenance. The readability of a language is very impactful on maintenance costs, which are often the highest component of total cost in modern software systems.

Problem 10, page 32

What are the arguments for writing efficient programs even though hardware is relatively inexpensive?

* Just because system resources are inexpensive does not mean they should be wasted. The operating systems where your program is writing will often be executing other programs. If your program claims an unnecessarily high proportion of CPU cycles, memory, or other resources, it has less value to the user.
* In addition, writing efficient programs generally result in programs that are generally easier to read, easier to maintain, and more organized.

Problem 12, page 32

In your opinion, what major features would a perfect programming language include?

* Built in common high level data structures such as linked lists, hash tables, and queues.
* Medium orthogonality to balance expressiveness with readability.
* Unary operators for often used constructs such as incrementing and operating then assigning.
* Functions are first class objects that can be passed like data types.

Problem 14, page 32

Describe some advantages and disadvantages of some programming environment you have used.

IntelliJ Environment

* Real time syntax checking
* Debugger with breakpoints and stepping through code line by line
* Type checking
* Syntax coloring
* Autocompletion
* Templating can speed up prototyping

Disadvantage

* Templating can force project structure and overcomplicate simple projects.
* The learning curve for all the operations can be steep given the complication.
* Configuring the IDE across different OS’s can be tricky
* You forget how to code without the aid of the IDE

Chapter 2

Problem 14, page 106

What are the arguments for and against the idea of a typeless language?

Arguments for

* Convenience
* Adds to writability.
* Makes the language more flexible
* Writing short programs are very fast

Arguments against

* Compile time errors are converted to run time errors
* Extra operations required to convert data types as necessary
* Encourages sloppy coding
* Deceases readability since the use of variables can change