Michael Beaver

Dr. Patricia Roden

CS 410W

26 August 2014

A Proposal to Research the Ada Programming Language

I propose to research the Ada programming language. This language has a long, influential history and is actively maintained today [1]. The Ada language has been used extensively in industries such as aviation, data communications, banking and finance, and medicine, among others [2]. Jean Ichbiah designed Ada “with three overriding concerns: program reliability and maintenance, programming as a human activity, and efficiency,” and newer revisions of Ada by Tucker Taft have improved on these original goals by implementing features such as standardized packages [1].

The focuses on reliability, maintenance, humanity, and efficiency are appealing. As any programmer may attest, developing reliable programs that are also efficient and maintainable is a difficult task, and any language that aids in that task is preferable. We cannot overlook or underemphasize Ichbiah’s and Taft’s emphasis on programming as a human activity. At present, humans are necessary for producing efficient and maintainable programs. Hence, any programming language that recognizes and embraces this fact surely will see widespread use; this may partially explain Ada’s popularity. Indeed, after a cursory examination, Ada source code is fairly readable and somewhat similar to other popular programming languages, which may help reduce the learning curve. It is also beneficial that there is a wealth of Ada-related resources available digitally and in print.

The Ada programming language is impressively robust, which is perhaps unsurprising given its long-term development. For example, Ada has the ability to interface with other languages, including C/C++, COBOL, and Fortran [1]. Modern software development is steadily making the transition to multi-language development, and this feature helps strengthen Ada’s legitimacy for large-scale, industrial software development. Ada also has extensive support for various data types, arrays, and records. Other essential, modern features include support for subprograms, packages, scope, and exceptions [1]. Ada’s adaptability to modern programming language trends is impressive considering it has been in development for over thirty years.

The Ada programming language represents a critical milestone in the development of programming languages. The original language has experienced continual development and is still popular over thirty years since its inception. The design goals of reliability, maintainability, emphasis on humans, and efficiency seem superficially to be more ideals rather than attainable, concrete features of a practical programming language. This research project is an excellent opportunity to study how Jean Ichbiah achieved these goals and how Tucker Taft and his colleagues continue to improve upon them. This project is an opportunity to discover what has earned Ada its reputation and place in the canon of programming languages.

Alternate choices:

Eiffel

Haskell

References

[1] AXE Consultants. (2012). *Ada reference manual, ISO/IEC 8652:2012(E): Language and standard libraries*. Retrieved from http://www.ada-auth.org/standards/12rm/

RM-Final.pdf

[2] Feldman, M. B. (2012). Who’s using Ada? Retrieved from http://www.seas.gwu.edu/

~mfeldman/ada-project-summary.html