

Homework 6: Essay on a new language

Experiment and learn a new (to you) programming language; examples might include (but are NOT limited to) Matlab, Mathematica, Icon, Ruby, C#, Fortran, Pascal, Perl, Ada, BASIC, Cobol, Lisp, Miranda, Magma, Processing, PHP, SML, scheme, smalltalk, Erlang, Io, Clojure, Squeak, OCamel, etc. (Note: Some of these are available freely, but others are not; some are on hopper, and some are not. So you might want to look around for a bit before deciding.) The only languages specifically forbidden for this assignment are C, C++, Java, Python, Prolog and Haskell, since you are either quite familiar with those after the classes in our major here or will see them in some detail later in this course. I'll put you on your honor to not choose one you already know, since that defeats the purpose of the assignment! I hope you'll see this as an opportunity to learn a language that you've been curious about or think would be helpful to know for future projects.

Investigate the features of the language, both through the documentation and writing your own code. Is the language compiled or interpreted? Is it functional or imperative or something else? Does the language use lexical or dynamic scoping? Can scopes nest, and are they open or closed? Does scope encompass an entire block where it is declared? How are recursive subroutines declared and used? Investigate order of evaluation - is it left to right or right to left? Does the language have short circuit boolean evaluation, or a true boolean type? Is the language dynamically or statically type-checked, and is it strong or weak type-checking? Does the language use any coercion or type-casting functionality, and how is polymorphism incorporated (if at all)? What type of object oriented structure does it include, if any? How are subroutines and exceptions handled in the language?

Note that you don't have to address all of these - some aren't even relevant for all languages. You're also welcome to describe other interesting or unusual structural features of the language. My goal is simply to get you to investigate the issues we covered in the first half of the course in more depth. **But:** Be sure to back up your answers, either through relevant and legitimate sources or through examples you devise to test this yourself. So your claims or answers to those questions above should either have a citation or example attached.

For this assignment, you will write a longer paper; I expect about 4-5 pages for most will cover the material requested. I will expect at least 3 **reputable** sources beyond your textbook, with proper citations for relevant facts from those references. Of course, for this assignment, you are welcome to use web or book resources of any type; just be sure to use at least 3 reputable sources and include them in your bibliography. (So you can use wikipedia or reddit if you must, but don't expect me to count it as reputable unless you include code to back up what they claim on it!) For this assignment, I am also expecting you will likely need to include some brief sample code to justify your answers to some of the questions; please be sure to cite any references you use for this code, but you are also responsible for making sure they actually run and are correct. You are also welcome to have an appendix with longer code examples, but I expect the main 4-5 pages to be separately coherent.