Language Project*

Presentation due December 14th

Report due Thursday, December 20th

1 Overview

You will be assigned a programming language to study from the following list:

1. Julia
2. Elm
3. Go
4. Liquid Haskell
5. Rust

You will research your language and prepare a short presentation and report on its features.

2 Topics

In your presentation and paper you should address the following topics:

- Overview:
  - What are the strengths and weaknesses of your language?
  - For what type(s) of problems/domains is it particularly well suited?
  - Which programming paradigms is the language commonly used in (functional programming, verification, imperative, declarative, object-oriented)?
- Language properties
  - Type system
    * Is the language statically or dynamically typed?
    * Is there support for type inference?
    * What is the type system like? Are there any unusual features?
    * Give an example of a type judgment rule in the notation of the language.
  - State and evaluation
    * Are there restrictions on mutation?
    * What evaluation strategy does the language use?
- Language features
  - Are functions first class?
  - What sort of control flow constructs does it have?
  - What are the core abstractions of the language?
  - What core features are there that make your language unique?
- Usability
  - What is difficult about learning to use this language?
  - What form does the programmer interaction take (compiled/interpreted, virtual machine, interactive shell or work from files)?

---

*Thanks to Valerie Barr for developing the original version of this project.

- Taxomony
  - What is the history of the language? When was it developed and by whom?
  - What other programming languages is the language related to, if any?
  - What uses does the language have today (and if it is not used today, why not)?
- Special topic:
  - Julia: multiple dispatch
  - Elm: persistent data structures
  - Go: goroutines and channels
  - Liquid Haskell: refinement types
  - Rust: ownership and borrowing

In addition to these questions, you should show at least two programs: (1) a version of factorial written in a style that you feel is representative of the language; and (2) a more complicated program of your choice, which showcases the kind of problem that the language is well-suited for.

3 Presentation

During your presentation, you will demo features of the language and describe its properties. You should explain the practicalities of the language (how to install, compile/interpret, and execute programs in the language). You should discuss its core abstractions and unique features, including the special topic I have given you for your particular language. You may also find it useful to make comparisons with more familiar programming languages.

Your demo must include at least two programs:
(1) a version of factorial written in a style that you feel is representative of the language; and
(2) a more complicated program of your choice, which showcases the kind of problem that the language is well-suited for.

Your presentation should last 10 minutes, and you will have 2 minutes to take questions from the class at the end. **All presentations will be in class on December 14th.**

4 Report

Your report must address all of the topics listed above and should go beyond a simple tutorial. You will have to do some research into the language, including its history and origins.

Your report should be about 8 pages in length (11pt font, single-spaced) and describe aspects of the language in greater depth than your presentation, though the writing should still be concise. The tone can be fairly informal— think of it as an introduction to the language for an advanced undergraduate student who has taken a programming languages class (like you!).

Your write-up is due by 10pm on December 20th.

5 Attribution

It is very important that you follow Wellesley's Honor Code in your presentation and write-up. **In particular, you must attribute any code that you do not write yourself**. You may present example programs that you find on the internet, but you must cite the original source. Any factual information obtained from reference sources must also be properly attributed as well, of course.