Csc344 Assignment: Programming Languages I Might Like To Learn

What's It All About?

This assignment is one of a handful of non-programming assignments that you will be tasked with doing this semester. One of the goals for this semester is to broaden your view of the programming language landscape, by which I mean the tangible features of programming languages which are considered in terms of their utility or their aesthetic value. By surveying a range of programming languages, which is what this assignment is all about, you will make some progress on advancing your knowledge of the programming language landscape.

Task 1 The Search

Making good use of Google, search for 6 programming languages, other than Java, Lisp/Scheme/Racket, Prolog, Haskell, or Rust, that you think you might like to learn about, by which I mean, for each language:

1. learn about where it stands in the programming landscape,
2. learn about how it tends to be used (typical applications), and
3. learn to write programs in the language.

Task 2 The Short Texts

For each of the 6 languages that you found which resonate with you to the extent that you think it might be worth some of your time to seriously make their acquaintance:

1. Identify the language by name. Do your best to describe the origin of the language, including a date, something about the human(s) who contributed to the design/implementation of the language, and an account of the languages that influenced its conception, and also of languages that were influenced by the design of the language.
2. An articulation, of some substance indicating why you might be interested in learning this language. Do your best to include 2 or 3 of the more salient reasons why you believe that you would be interested in learning this language.

Task 3 The List Creation

Create an ordered list (1 .. 6) consisting of the short texts that your wrote for the 6 languages in the second tasks, ordering your list by date of origin of the language.

Task 4 The Document Creation

Contextualize your length 6 list within a document which contains three things:

1. A title the title of this assignment.
2. A short abstract, of your own creation, which describes this assignment in just a few simple sentences.
3. The list of length 6 presenting your short texts, ordered according to date of language "origin"
4. **Save this document as a pdf file.**

Task 5 The Posting

The due date of this assignment will be tied to the due date of your initial web work site creation assignment. Both will be due on Thursday, February 2, 2023. The submission of this assignment, like the submission of all of your assignments this semester, simply requires that you reference it in an appropriate manner from your web work site. (You will not be emailing your work to anyone, or posting it to any site other than your very own web work site.)

Appendix A: Document Template

Assignmnet: **Programming Languages I Might Like To Learn**

Abstract

<<Abstract goes here>>

Language 1: <<Name of Language>>

<<Short Text 1>>

Language 2: <<Name of Language>>

<<Short Text 2>>

Language 3: <<Name of Language>>

<<Short Text 3>>

Language 4: <<Name of Language>>

<<Short Text 4>>

Language 5: <<Name of Language>>

<<Short Text 5>>

Language 6: <<Name of Language>>

<<Short Text 6>>

Appendix B: Example Short Text

The Prolog language, along with Java, Lisp/Scheme/Racket, and Rust, is off limits for this assignment. So it makes some sense for me to feature it in an example of the sort of short text that I am asking your to write 6 times, once for each of the programming languages that you determine to feature in your work for this assignment.

The **Prolog** language was developed and implemented in Marseille, France, in 1972 by Alain Colmerauer with Philippe Roussel, based on Robert Kowalski's procedural interpretation of Horn clauses at University of Edinburgh. Kowalski's interpretation of Horn clauses was, in turn based on Alan Robinson's Resolution Principle (JACM 1965). Prolog was directly influenced by Carl Hewitt's **Planner** (MIT). Prolog influenced **Clojure**, **Erlang**, and many other languages. ((Wikipedia))

Prolog would be worth learning for any number of reasons, including the following:

1. Programming in logic has long been a dream of computer scientists since the inception of the field more than 7 decades ago, and computing in logic has been a dream of mathematicians and philosophers for hundreds of years. Since Prolog is by far the most significant logic programming language, it would be good to learn about its design and implementation.
2. Prolog has a powerful pattern matching mechanism that has influenced a range of languages, including Erlang, Haskell, and Rust. Thus, it would be good to experience Prolog's pattern matching mechanism first hand.
3. The motivating application for Prolog was natural language processing, and Prolog has been used in many natural language processing projects. Since I am interested in natural language processing, I would like to learn about Prolog's "definite clause grammar", and experience how the language might be used to do NLP work.