Elements of Programming Languages
Draft Course Syllabus

Dr. Philip E. Cannata
email: cannata@cs.utexas.edu

Course Description and Objectives:
This course teaches the essential skills necessary to program in a variety of currently popular programming languages and programming models including the Object Oriented Model (Java 8 and Python), the Prototype Model (with JavaScript/jQuery), and the Functional Model (with Java 8 and Python). This course will enable students to understand the differences between the programming models and decide to embrace one or the other, or a combination of each. It will be shown that a strong motivating factor in this decision will be how each language and model handles concurrency. In the not too distant future, programmers must have a coherent way of supporting concurrency and selecting the right tools will be critical.

The course will also teach how programming languages are constructed and introduce students to Ply. Ply is a pure-Python implementation of popular compiler construction tools. Although Ply was developed to help build languages compilers, it has many other applications in computer science.

Prerequisites:

• a grade of C or better in cs303e or equivalent
• the class will work in groups of 2 or 3 students all semester, each group needs to have at least one laptop in class each class day.

Required Texts:

• Seven Languages in Seven Weeks: A Pragmatic Guide to Learning Programming Languages. Bruce A. Tate, Pragmatic Bookshelf. (ISBN: 978-1934356593)
• Functional Programming in Java: Harnessing the Power Of Java 8 Lambda Expressions, Venkat Subramaniam, Pragmatic Bookshelf. (ISBN# 978-1937785468)

Recommended Texts:

Class Discussion Tool:

Piazza will be used for this purpose.
- I will post class announcements and information to the Piazza discussion group
- Post your questions about the class to the Piazza discussion group. Do not post solutions or chunks of code > 3 lines on Piazza,

Grading:

- There will be 8 quizzes over the course of the semester each worth 10 points for a total of 80 points.
- There will be 6 lab projects over the course of the semester each worth 20 points for a total of 120 points. Students will work in groups of 2 or 3 on these labs. Groups will be expected to complete the lab projects on your own using the provided code when given. Class examples and examples from the web can be used, but these must be documented. To be clear, each group must do a significant amount of the work on the lab projects by themselves.
- Each group will have to complete a separate project of their choosing (after consulting with Dr. Cannata) which will be worth 50 points.

Grading will be on a straight scale as follows:

- A = 100 - 90% (225-250 pts)
- B = 89 - 80% (200-224 pts)
- C = 79 - 70% (175-199 pts)
- D = 69 - 60% (150-174 pts)
- F = < 60% (below 150 pts)

Course Topics

- Object Oriented Programming
- Class Inheritance
- Prototype Programming
- Prototype Inheritance
- Functional Programming
- Functions
- Static and Dynamic Scoping
- Closures
- Currying
• Immutability
• Message Passing
• Static, Dynamic, and Duck Typing
• Type Inferencing
• Lazy Evaluation
• Metaprogramming
• Pattern Matching
• List Comprehension
• Monads
• Concurrency
• Actors
• Coroutines
• Futures