CMPT 383 - D200 Comparative Programming Languages

Instructor(s): Toby Donaldson

SFU Surrey

Calendar Objective/Description:
Various concepts and principles underlying the design and use of modern programming languages are considered in the context of procedural, object-oriented, functional and logic programming languages. Topics include data and control structuring constructs, facilities for modularity and data abstraction, polymorphism, syntax, and formal semantics.

Instructor’s Objectives:
This course explores the various concepts and principles underlying the design and use of modern programming languages. We will take a detailed look at core programming techniques of procedural languages, and also examine alternative programming models such as object-oriented programming, functional programming, declarative programming, and concurrency.

Prerequisites:
CMPT 225, and (MACM 101 or (ENSC 251 and ENSC 252)).

Topics:
- Basic syntax and semantics
- Control flow
- Type systems
- Composite types
- Subroutines
- Objects
- Functional programming
- Declarative programming
- Concurrency

Grading:
The course will contain a number of programming exercises and projects, plus a midterm and final exam. The exact marking scheme will be provided in the first week of class.

Recommended Books:
Concepts of Programming Languages (11th Edition), Robert Sebesta, Addison-Wesley, 2015, 9780133943023, It is okay to use the 10th edition, if you prefer.

Reference Books:
Programming Language Pragmatics, Michael L. Scott, Morgan Kaufmann, 2015, 9780124104099

Academic Honesty Statement:
Academic honesty plays a key role in our efforts to maintain a high standard of academic excellence and integrity. Students are advised that ALL acts of intellectual dishonesty will be handled in accordance with the SFU Academic Honesty and Student Conduct Policies ( http://www.sfu.ca/policies/gazette/student.html ).