Computing Science Course Outlines

CMPT 373 - D100 Software Development Methods

Instructor(s): Nick Sumner

Calendar Objective/Description:
Survey of modern software development methodology. Several software development process models will be examined, as well the general principles behind such models. Provides experience with different programming paradigms and their advantages and disadvantages during software development.

Instructor’s Objectives:
This course exposes students to modern software development practices. Several software best practices will be introduced. Students will gain experience with different programming methodologies and their advantages and disadvantages during software development. The course is principally a laboratory course, with lectures, discussions, exercises, and project homework to supplement the laboratory work. Students will work in groups of roughly eight individuals on term projects assigned by the instructor. Projects will be implemented using C++, developed for and using a Linux oriented platform. Weekly laboratory times are for mandatory project group meetings, including meetings and code reviews with the instructor. The primary goal of the laboratory work is to correctly follow and understand the development practices for the project. Students are marked individually depending on their adherence to the model and contribution to the project. Students should expect to participate in class discussions and to give an informal presentation regarding their project or assigned specialty.

Prerequisites:
CMPT 213 and (CMPT 276 or 275).

Topics:
- Best practices: design patterns, refactoring, language-specific issues, programming by contract
- Agile software development: extreme programming, Scrum, test-driven development
- Managing complexity and designing maintainable software
- Software-engineering tools and environments
- Software development process models: component-based development, iterative processes

Grading:
To be discussed the first week of classes. This course involves substantial programming contributions to a group project.

Required Books:

Reference Books:
Working Effectively with Legacy Code, Michael Feathers, Prentice Hall, 2005, 9780131177055

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).