CMPT 125 - D100 Intro.Cmpt.Sci/Programming II

Instructor(s): Anne Lavergne

Calendar Objective/Description:
Intro.Cmpt.Sci/Programming II

Instructor's Objectives:
This course is a rigorous introduction to computing science, intended primarily for students who have already taken CMPT 120 as an introduction to algorithms and programming. Students will learn fundamental concepts of computing science and basic principles of algorithm design and software development. Most case studies will be presented using C. Students will also be introduced to object-oriented software development using C++.

Prerequisites:
see go.sfu.ca

Topics:
- Brief review of problem solving and programming; introduction to C.
- Performance measurements; algorithm design and analysis.
- Recursion: simple recursion; divide and conquer algorithms.
- Reasoning about programs: assertions, invariants, and correctness.
- Good programming style; defensive coding practices; testing.
- The memory model: addresses, dynamic data types, safe initialization, safe cleanup and safe arrays.
- Encodings of basic types: int, unsigned, float, char, pointer.
- Elementary data structures; abstract data types; information hiding.
- Introduction to object-oriented software development; basic class design in C++.

Grading:
To be discussed the first week of classes

Students must attain an overall passing grade on the weighted average of exams in the course in order to obtain a clear pass (C- or better).

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).