CPSC 490 - Problem Solving in Computer Science

1. Basic Informations

Coordinators:	Robert Tseng, Christopher Head
Email Address:	cs490@ugrad.cs.ubc.ca
Home Page:	TBA
Prerequisite:	CPSC 320 (or equivalent)
Faculty Sponsor:	Dr. Will Evans
Newsgroup:	ubc.courses.cpsc.490

2. Course Overview

The main focus of the seminar is to provide both theoretical and practical perspectives on algorithms presented by other CPSC classes (primarily CPSC 320 and 420). The theoretical part will be presented through class discussions, pseudocode, and diagrams; the practical part will involve students actually implementing these algorithms in C, C++, or Java to solve problems. Unlike other theory courses offered in the department, the practical part is as important as the theory.

Note the this is a Student Directed Seminar and not a regular 3 credit course. This means that there is no professor in this class. The seminar will be student driven - which means that we can decide as a class how the seminar will proceed (including changing this syllabus). For more information, refer to the "Section 6 - Student Directed Seminars".

3. Course Structure

The course will cover several important topic in algorithms (for more detail, refer to "Section 5 - Tentative Schedule"). Each topic will span roughly 2 weeks, consisting of Introductory Lectures and Sample Problem Presentations. Each topic will be assigned to students in groups of 2 or 3, who will be giving the Sample Problem Presentations for the topic. These presenters will also be given the task of creating the homework assignment for the topic as well making the problem for the semester-end quiz (for more detail, refer to "Section 4 - Marking Scheme").

You are free to choose your own group and which topic to present, but sign up quickly as we will assign topics on a first-come-first-served basis. Note that each person should be assigned EXACTLY one topic.

3-1. Introductory Lecture

These lectures are meant to provide a high level theory for the topics. These lectures will be given by us.

3-2. Sample Problem Presentation

At the beginning of each topic, we will post a few sample problems for the topic. After the Introductory Lectures, the presenters assigned the topic will present pseudocode-level solutions to the sample problems. The presenters are also responsible for preparing a few sample problems on their own and leading class discussions on tackling these problems.

3-3. Guest Lectures

There will be a few guest lectures sprinkled throughout the semester. We will invite faculty doing research in algorithms to present some of their work.

4. Marking Scheme

The tentative marking scheme is: assignments (40%), presentation (30%), and quiz (30%).

4-1. Assignments

- There is an assignment for each topic introduced, consisting of 3 problems of varying difficulty
- The assignment will be created by the presenters for the topic. Every students other than the presenters (including us) needs to complete the assignment. Since every student presents exactly one topic, every student will be exempt from exactly one homework.
- Assignment solutions will be marked by an automated judge machine. A solution that passes all tests will be awarded full marks. A solution that fails some tests may be submitted along with an explanation to the presenters for partial marks.

4-2. Presentation

- The presenters for each topic will be peer evaluated on their performance leading the Sample Problem Presentation.
- Please be fair when you evaluate others. We, and faculty sponsor if necessary, will investigate any allegations of bias.

4-3. Quiz

- At the end of the semester, there will be a quiz to test students on the topics presented.
- The presenters for each topic will be responsible for creating the question on that topic on the quiz. The problems are to be submitted directly to the faculty sponsor, Dr. Will Evans.
- Dr. Evans will compile the problems and provide the quiz to the class. EVERYONE (including us) must complete the quiz, except for the question they created.
- The quiz will be marked together as a class (each group of presenters mark the question they created).

5. Tentative Schedule

- Introduction
    - Input/Output in Java, C++
- Graph Theory
    - Shortest Path
    - Euler Path/Cycle
    - Minimum Spanning Tree
- Maximum Flow
    - Ford Fulkerson method
    - Min Cut Max Flow theorem
    - Minimum Cost flow
    - Matching
- Backtracking
    - Branch and bound
    - Bidirectional search
- Dynamic Programming
    - Bitmask DP
    - Memoization vs Iterative
- Number Theory

    - GCD, primes, Euler totient function
    - Modular inverses, Chinese Remainder Theorem
- Miscellaneous (if time allows)
    - KMP string matching
    - Simplex method

6. Student Directed Seminars

This page has been inserted into the course syllabus (outline) at the request of the advisory committee that oversees the program for Student Directed Seminars.

The program of Student Directed Seminars is intended to provide senior undergraduate students with added opportunities to learn in small, collaborative, group-oriented experiences. It is also the program's goal to ensure participants, as members of a self-directed group, have a high degree of control over their own learning experience. The UBC program is modeled on an established student-directed seminar program at the University of California at Berkeley.

The program works as follows. A student (or group of students) in their third or fourth year of undergraduate study, proposes a course not currently offered at UBC. Proposals go to an Advisory Committee for review and if the proposal looks feasible, the committee encourages further development. The student proceeds to develop a course outline under the guidance of their faculty sponsor (or in some cases, multiple faculty sponsors). Student coordinators also have the benefit of a preparation workshop conducted by the UBC Centre for Teaching and Academic Growth. The Student Directed Seminar Advisory Committee considers course outlines for final approval. If approved, the student-initiated course is advertised to the general student body. All upper-level students are eligible to participate, but applicants are subject to a selection process. Normally the minimum enrollment for each class is eight, the maximum fifteen.

The Student Coordinator is not an instructor. The coordinator's role is that of a facilitator. S/he is responsible for organizing the learning resources, such as guest lectures, reading materials, and films to be used in the class. The Student Coordinator also sets the parameters of course content, structure, and evaluation procedures in conjunction with a Faculty Sponsor. The participants have an important role in refining the details of all of these elements during the first classes of the term.

The entire class is responsible to one another for ensuring that the learning experience has a quality and richness that benefits everyone. Ultimately the faculty sponsor is responsible for the grades that are submitted for this course.

This course is subject to the normal rules and regulations, as appropriate, which apply to all UBC courses.

More details are available at the following URL:

http://leap.ubc.ca/get_ahead/student_directed_seminars/