CSCB70S - Fundamental Data Structures and Techniques **Spring 1999**

Instructor: Daniel Panario, panario@scar.utoronto.ca

Tutor: Nick Cheng, nick@scar.utoronto.ca

Lectures: Mon, Wed 2:10–3:00 in H216.

Tutorials: there are 4 tutorial sections confirmed (there may be more tutorial sections)

Wed	11:10–12:00	(S358)
Wed	12:10–13:00	(S358)
Tue	12:10–13:00	(R3011)
Fri	10:10–11:00	(S357)

Office hours:

Lecturer: Mondays and Wednesdays 10:10–11:00, and 3:10–4:00 in S-626A.
Tutor: TBA.

Textbook:

No textbook. We will use the "CSC270S/B70S Spring 1999 Readings" notes in the bookstore.

C programming language book:

K.N. King, "*C programming: a modern approach*", Addison-Wesley, 1996.

Other texts for consult:

B.W. Kernighan and D.M. Ritchie, "*The C programming language*", 2nd ed., Prentice-Hall, 1988.

B. Stroustrup, "*The C++ programming language*", 3rd ed., Addison-Wesley, 1998.

Course goals (from the academic calendar): Standard programming methods, with an introduction to C and C++. Use of classes to represent abstract data types, graph representation and graph algorithms. Simulation: Data structures and program organization for event-driven models. Representation of floating-point numbers; introduction to numerical methods, optimization using dynamic programming. Programming assignments stress both the proper use of abstract data types (lists, stacks, trees, heaps) and approaches to writing larger, more complex programs. Prerequisite: CSCA58F/S. Pre or co-requisite: MATA26Y & one A-level science course.

Course Outline (tentative):

1. Numerical Methods & C Programing Language [4 weeks]. C control constructs, data types, I/O, and functions. Finding roots and maxima/minima. Integration methods.

2. Graphs & C++ Programing Language [3 weeks]. C pointers. Object oriented programming. C++ classes & templates. C++ I/O. Graph representations. Breadth-first search. Depth-first search. Topological sort. Maybe shortest path and transitive closure.

3. Simulation & C++ [3 weeks]. Data structures for simulation. Event–driven models. Queueing.

4. Optimization [2 weeks]. Greedy algorithms. Dynamic programming.

5. Number Representation [1 week]. Floating point numbers and their perils.

Home page for the course:

`http://www.cs.toronto.edu/~daniel/teaching/B70/index.html`

Marking Scheme:

4 Assignments at 10%, 10%, 15%, 10%	45%
1 Term Test at 15%	15%
Final Exam at 40%	40%
	100%

To pass the course, you must receive an average of at least 33/100 on the final exam and the term test, and an average of at least 33/100 on the assignments. See the webpage of the course for the policies on plagiarism and lateness.

Additional material will be taught in the tutorials. You are expected to know this material. Graded assignments and test will be handed back by the tutor.

Schedule:

First lecture - Jan 4

Assignment	**Hand-out Date**	**Due Date**	**Contents**	**Worth**
1	Jan 11	Jan 29	Numerical analysis	10%
2	Feb 1	Feb 26	Graphs	10%
3	Mar 1	Mar 19	Simulation	15%
4	Mar 22	Apr 2	Dynamic programming	10%

Midterm test - Feb 10 (2:10 in lecture)

Reading week - Feb 15 - 19

Last date to drop - Mar 5

Last lecture - Apr 9

Final exam - Apr 19 - May 7, 1999.