CSC260S - Introduction to Scientific, Symbolic, and Graphical Computation
Spring 2000

Instructor: Daniel Panario, daniel@cdf.toronto.edu

Home page for the course:

`http://www.cs.toronto.edu/~daniel/teaching/260/index.html`

Lectures: Mondays from 7:10 to 9:00 in MP 103.

Tutorials: Mondays from 6:10 to 7:00; tutors and places are:

Tutor name	Room	Student name
Ovidiu Calin	(MP 103)	A-K
Theo Garefalakis	(SS 1086)	L-Z.

Office hours: Mondays from 5:10 to 6:00 PM in SF2303A.

Textbook:

Eugene Fiume, "*An Introduction to Scientific, Symbolic, and Graphical Computation*", A. K. Peters Ltd., 1995.

The course will use the Maple language on the CDF Unix computers in the Engineering Annex.

Maple programming language books:

Michael Monagan et al., "*Maple V Programming Guide*", Waterloo Maple, 1996.

Heal et al., "*Maple V Learning Guide*", Waterloo Maple, 1996.

Course goals (from the academic calendar): Problems in transforming continuous mathematical models to discrete computational models. Inadequacy of naive computer solutions, and techniques to remedy inadequacies. Symbolic computation, plotting, 3-D graphics, and conventional programming languages. Intended for students from computer science, sciences and mathematics: for computer scientists, introduction to design and implementation of robust algorithms; for scientists, techniques in transforming scientific problems into computational solutions; for mathematicians, insight into differences between mathematical models and computational solutions.

Exclusion: CSC160H as taught in Spring 1992; after taking CSC270H and any 300-/400-level CSC course, you may not take CSC260H without departmental permission in writing.

Prerequisite: CSC108H/computing experience, as from a good high school programming course; CGPA 2.0

Co-requisite: MAT135Y/137Y/157Y, 223H/240H

Purpose of the Course: This course is an introduction to how computers can be used to implement mathematical models that reflect reality or that behave in some desirable way. More specifically, we will study

- Two types of computation: *symbolic* computation, such as the sort of algebraic manipulation that you are probably used to doing with pencil and paper, and *numeric* computation, such as you are used to doing with a calculator.
- Symbolic and numerical methods for solving several types of problems, such as *interpolation* of data points, *finding zeros* of a function, *integration* of functions, and *filtering* of signals.
- How these methods can be implemented in the Maple V programming language, as well as the use of the built-in facilities that Maple provides.
- How such methods can be used to solve problems involving mathematical modeling, especially those that arise in computer graphics.

Course Outline (tentative):

Jan	3	Topics we will cover; examples of symbolic and numeric computation in Maple. *Assignment 1 handed out, due January 17*
	10	Exact, fixed-point, and floating-point representations of numbers; rounding errors.
	17	Approximating functions by Taylor series and other methods; evaluating polynomials.
		Assignment 2 handed out, due February 7
	24	Explicit, implicit, and parametric representations of curves and surfaces.
	31	Rendering lines and curves for computer display.
Feb	7	Solving differential equations; dynamical simulations.
	14	READING WEEK (no classes)
	21	Lagrange and piecewise polynomial interpolation; natural cubic splines; basis functions.
		Assignment 3 handed out, due March 13
	28	Finding zeros of functions symbolically and numerically.
		Midterm test in tutorial
Mar	6	Integrating functions symbolically and numerically.
	13	Multidimensional integration with product rules and by Monte Carlo.
		Assignment 4 handed out, due April 3
	20	Approximation using Fourier series; generating random fractal functions.
	27	Filtering signals and images.
Apr	3	Fourier transforms.

Marking Scheme:

4 Assignments at 5%, 10%, 10%, 10%	35%
1 Term Test at 20%	20%
Final Exam at 45%	45%
	——
	100%

To pass the course, you must receive at least 35% on the final exam.

Schedule (tentative):

First lecture - January 3

Assignment	Hand-out Date	Due Date	Contents
1	January 3	January 17	Maple warm-up
2	January 17	February 7	Taylor approximations
3	February 21	March 13	Interpolation
4	March 13	April 3	Integration

Midterm test - Monday, February 28 (in tutorial)

Last date to drop - Friday, March 3

Last lecture - Monday, April 3

Final exam - April 17 - May 5, 2000.

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

See the webpage of the course for the policies on plagiarism and lateness.