COMP 426 Fall 2007: Lecture Schedule

Brigitte Pientka
bpientka@cs.mcgill.ca

September 4, 2007

Week 1	Lecture 1	Sep 5	Overview	
	Lecture 2	Sep 7	Natural Deduction	
Week 2	Lecture 1	Sep 10	Tutorial(Maja)	
	Lecture 2	Sep 12	Tutorial(Maja)	
	Lecture 3	Sep 14	Normal Proofs	HW 1 out
Week 3	Lecture 1	Sep 17	Notational definitions	
	Lecture 2	Sep 19	Proofs-as-Programs	
	Lecture 3	Sep 21	Proofs-as-Programs	HW 1 due, HW 2 out
Week 4	Lecture 1	Sep 24	Properties of proof terms	
	Lecture 2	Sep 26	Lambda-calculus	
	Lecture 3	Sep 28	Primitive recursion	HW 2 due, HW 3 out
Week 5	Lecture 1	Oct 1	Data-types	
	Lecture 2	Oct 3	Induction	
	Lecture 3	Oct 5	tba	HW 3 due, HW 4 out
Week 6	Lecture 1	Oct 9	First-order logic (TUESDAY!)	
	Lecture 2	Oct 10	First-order logic	
	Lecture 3	Oct 12	First-order logic	HW 4 due, HW 5 out
Week 7	Lecture 1	Oct 15	Contracting proofs	
	Lecture 2	Oct 17	Reasoning about data	
	Lecture 3	Oct 19	Review	HW 5 due
Week 8	Lecture 1	Oct 22	Midterm	
	Lecture 2	Oct 24	Normal natural deduction	
	Lecture 3	Oct 26	Consistency proof	
Week 9	Lecture 1	Oct 29	Sequent calculus	
	Lecture 2	Oct 31	Meta-theoretic properties	
	Lecture 3	Nov 2	Inversion and focusing	Project Proposal due
Week 10	Lecture 1	Nov 5	Guided theorem prover	
	Lecture 2	Nov 7	Logic programming	
	Lecture 3	Nov 9	Logic programming	
Week 11	Lecture 1	Nov 12	Forward theorem proving	
	Lecture 2	Nov 14	Forward theorem proving	
	Lecture 3	Nov 16	Bi-directional theorem proving	Project milestone
Week 12	Lecture 1	Nov 19	Special topic: Modal logic	
	Lecture 2	Nov 21	Special topic: Dependent types	
	Lecture 3	Nov 23	Special topic: Linear logic	
Week 12	Lecture 1	Nov 26	Presentations	
	Lecture 2	Nov 28	Presentations	
	Lecture 3	Nov 30	Presentations	
Week 13	Lecture 1	Dec 3	Review	

General information about the course

There is a home page for the course

http://www.cs.mcgill.ca/~bpientka/courses/atp/index.html

We will place useful information there. This includes assignments, hints, examples of code, office hours and the lecture schedule in postscript format and pdf format. Some material will be in plain format. You can find the Adobe Acrobat reader (for pdf) or Ghostview (for postscript) available free on the web in order to read and print the files.

In the first half of the course, we will be using the proof tutor, Tutch. Please note that you must install it yourself. Information on how to install Tutch can be found on the course webpage (check under "Resources" and follow the link to the Tutch page). If you have trouble installing it, let us know asap.

Sml should also be running on the department's lab machines. If you are an emacs user the Sml-mode under emacs environment is an excellent programming environment. If you have your own computer you will want to download a free copy of SMLNJ and run it on your own machine. Check the "Resources" tab on the web page for instructions on obtaining Sml.

Method of evaluation

There will be 5 assignments (35 %), a midterm (35 %) and a project or final (30%). The assignments will be a mix of practical and theoretical exercises. You have 2 late days during the semester which you can use for your homeworks. You must notify your instructor before the homework is due, that you want to use one or two late days. If you have used up your late days, we will only accept your late homework under exceptional circumstances.

The midterm will be on **Oct 26** in class and it will be closed book. One sheet of notes will be permitted.

Some possible project topics will be handed out in class, but you are encouraged to suggest your own. The project involves:

- Project proposal (one page) (6%)

- Project milestone (one page) (4%)

- Project presentation (10%)

- Project report (5 - 10 pages) (10%)

Please note that you must discuss your project in person with me before submitting your project proposal.

Your project proposal briefly describes the background, motivation, and goal of your project. Your milestone should give an outline of your project and project report, and concretely list the problem you will attempt to solve and what you have done already. Ideally, it will include a time line.

Your project presentation should provide a summary of a given paper.

Your final project report however must go beyond simply summarizing some results. You can for example provide some simple implementation for a technique described, you can experiment with a given system and reconstruct the described work in practice, or you could expand some of the theoretical results described in the paper.