CS 462/662—Formal Languages and Parsing

Fall 2009, University of Waterloo

Course website: http://www.student.cs.uwaterloo.ca/~cs462/

Classes: 02:30PM-03:50PM, Tuesdays and Thursdays, MC 4064.

Instructor: Meng He
Office: DC2332, Phone: x37824, Email: mhe [at] uwaterloo.ca
Office hours: 11:00AM-noon, Mondays and Thursdays. In addition, I'm typically available after lectures. You can also make an appointment, or stop by my office when the door is open.

Teaching Assistant: Margareta Ackerman
Office: DC2515, Phone: x33435, Email: mackerma [at] uwaterloo.ca
Office hours: 2:00PM-3:00PM, Fridays

Course Description:
This course introduces formal languages and automata theory, which are fundamental in computer science. Formal languages allow a computer language to be formally defined, so that it can be processed by a computer in a consistent manner. Models of computation such as the Turing machine are so powerful that it is often very difficult to prove theorems about what they are able to compute. The study of simpler models of computation such as finite automata and pushdown automata allow researchers to obtain more useful results for many problems. Formal languages and automata theory have many applications, and the most important practical application is in compiling and parsing programming languages.

Building on CS 360/365, this course covers more advanced topics in formal languages and automata theory, including classical solutions, recent advancements and practical applications. Topics include:

- *Combinatorics on words*: The properties of finite and infinite strings of symbols over a finite alphabet, such as properties on prefixes/suffixes, power, and repetitions in strings.
- *Advanced topics on finite automata and regular languages*: These topics include the generalizations of the finite automaton model such as Moore and Mealy machines and transducers, advanced closure properties of regular languages, characterization of regular languages and minimization of finite automata.
- *Advanced topics on Context-free grammars and languages*: We will study topics such as advanced closure properties of context-free languages, theorems that can be used to prove whether a language is context-free and interesting subclasses of CFLs.
- *Parsing and recognition*: Parsing is an essential feature of a compiler, and we will study how to parse a context-free grammar. We will also study how to perform faster parsing for certain classes of grammars, which is important because the source codes of modern software are often large.
- *Grammar-based compression*: This is a new and interesting practical application of formal languages to data compression. The basic idea is to find a small context-free grammar that generates a language containing only the text string to be compressed.
- *Advanced topics on Turing machines and other languages classes*: Some topics are unrestricted grammars, Kolmogorov complexity and context-sensitive languages.
- *Communication Complexity*: Initially introduced to address the complexity of distributed and parallel computations, the communication complexity has become one of the most intensively studied complexity measures. Some topics are the application of communication complexity to the complexity measures related to finite automata and Turing machines. This theory can also be used to prove lower bounds for data structure problems.

Course Prerequisites: CS 360 or CS 365 or an equivalent introductory course on theory of computing.

Textbook and Handouts:
There is a required textbook: Jeffrey Shallit, *A Second Course in Formal Languages and Automata Theory*, Cambridge University Press, 2009. This text book covers most of the course materials excluding grammar-based compression and communication complexity. Please also check this page regularly for handouts: http://www.student.cs.uwaterloo.ca/~cs462/lectures.html

Grading:
CS462: 5 written assignments (60%); a take-home final exam (40%); participation in in-class problem-solving sessions (5% *bonus* marks). **CS662:** 5 written assignments (50%); a course project (15%); a take-home final exam (35%); participation in in-class problem-solving sessions (5% *bonus* marks).

Note: 1. You will be given several days to work on the take-home final exam.
2. There will be an in-class problem-solving session every one or two lectures. In each session, students will normally be given 5-10 minutes to work on a problem, and then a volunteer will be invited to show his/her solution to the class. Volunteers will receive bonus marks for their final grades. Note that to receive ALL the 5 bonus marks, you need to present a correct solution to the class, or present near-perfect solutions to two or more different problems.

Assignments:
Each of the five assignments will be due on Tuesdays in class. The solutions will be handed out in class (at the end of the lectures) on the Thursdays following the due dates, and will be made available as course reserves in the DC library after that. Each student is allowed only one late assignment, which is due in class on the Thursdays following the assignment due dates in the same weeks. No other late assignments will be accepted, and no assignments that are handed in after the solutions are given out will be accepted. If there are any valid reasons such as serious illness, please see the instructor. The assignments will be handed out and due as follows:

Assignments Number	Handed Out	Due
1	September 22	October 6
2	October 6	October 20
3	October 20	November 3
4	November 3	November 17
5	November 17	December 1

An additional assignment, assignment 6, will be made available simultaneously with its solutions. It will cover the last few lectures that cannot be covered by assignment 5 due to academic deadlines and regulations. This assignment will not be marked, nor do you need to hand in your solutions. You are encouraged to work on it to benefit the most from this course and to get ready for the final exam.

Project Information (CS662 Only):
Basically, to complete the project, you need to choose a problem related to this course, read several papers on this problem, discuss, improve, extend or apply the results you have found, and write a report of 5-10 pages to present your survey and findings. To complete the project, you are not required to obtain new results, but if you are able to extend or improve previous results, you are encouraged to report them. A brief description of the project is due on November 10, and the report is due on December 8. More detailed guidelines, including suggested topics, will be given.