Notre Dame University
Computer Science Department

CSC 311 **Theory of Computation** **Spring 2012**

Instructor

Hikmat Farhat
Room: S 312
Ext: 2081
Email: hfarhat@ndu.edu.lb
Office Hours: MWF 1:00-2:00

1 Course Information

Lecture Hours: MWF 11:00-12:00
This three credit hours course with no lab component (3.0cr) is required for all students majoring in Computer Science. The prerequisite for this course is CSC 213 Program Design and Data Abstraction II.

2 Description

This course is a study of models of computation and their limitations. It will examine the problems that can/cannot be solved by each model. Different models include finite state automata, pushdown automata and Turing machines. The concept of regular, context-free and recursively enumerable languages is studied and their relation to the different models are analyzed. Connection with pattern matching, programming language theory and compiler design is emphasized. The Turing-Church thesis, halting problem and computability in general are also covered. Other models of computation, such as the λ calculus, are also discussed.

3 Textbook& References

- **Required textbook**: Introduction to Automata Theory, Languages, and Computation
  Hopcroft, Motwani and Ullman
  Third Edition
  Addison Wesley
- Supplementary course notes are accessible at Malik's.

4 Grading

- Two Exams: 50%
- Final: 35 %
- Assignments & Participation: 15%

5 Course Learning Outcomes

Below is a table relating the learning outcome of the course to the CS program outcomes.

CS Program Outcome	Course Outcome	Assessment Method
1,9	Have a thorough understanding of finite automata	Construct NFA and DFA that recognize a given language.
1,9	Have good knowledge about regular expressions	Construct a regular expression that generate a given language
1,9	Using different proof techniques to explore the properties of regular languages	Prove that a language is regular or not by using the closure properties or the pumping lemma.
1,9	Have a good grasp of context-free grammars	Construct context-free grammars and equivalent pushdown automata that recognize/generate a given language.
1,9	Have in depth knowledge of CFG	Prove that a language is context free or not by using the closure properties or the pumping lemma for CFL.
1,9	Be familiar with Turing Machines and their relation to algorithms	Construct a Turing machine that recognize a given language/solve a problem.
1,9	Understand undecidability and intractability	Identify some of the well known undecidable problems
1,9	Understand the limit of computations	Prove that a given problem is undecidable/decidable.
1,9	Have some knowledge of intractability	Identify some of the well known NP hard problems.

6 CS Program Learning Outcomes

1. Demonstrate their knowledge of discrete and continuous mathematics and their ability to apply logic and mathematical proof techniques to computing problems.
2. Demonstrate their knowledge of, and ability to apply, programming fundamentals in at least two programming languages.
3. Demonstrate their knowledge and understanding of, and their ability to apply, the concepts, design principles and fundamentals in the core areas of computer science.
4. Exhibit good knowledge in the areas of computer architecture and organization, computer operating systems and computer networks and security.
5. Demonstrate knowledge of fundamental concepts, principles and techniques in software engineering and their ability to apply the best-practices in software development processes, methods and tools.
6. Recognize the need for, and demonstrate an ability to engage in, continuing professional development.
7. Demonstrate their ability to communicate effectively with a range of audiences and to function effectively on teams to accomplish a common goal.
8. Demonstrate their understanding of professional, ethical, legal and security issues and responsibilities.
9. Demonstrate their ability to engage in a graduate program in Computer Science.

7 Schedule

Below is the lecture schedule for this semester together with the corresponding reading material from the textbook.

Week	Topic	Textbook Reading
Week 1	Introduction	1.3,1.4,1.5
Week 2	Finite Automata and DFA	2.1,2.2
Week 3	Nondeterministic automata	2.3,2.5
Week 4	Regular expressions	3.1,3.2
Week 5	Properties of Regular languages	4.2,4.3
Week 6	Minimization & Nonregular languages	4.1,4.4
	Exam 1	
Week 7	Context-free Languages	5.1,5.2,5.4
Week 8	Pushdown automata	6.1,6.2,6.3
Week 9	Properties of context-free languages	7.1,7.2
Week 10	Properties of CFL II	7.3,7.4
Week 11	Turing Machines	8.1,8.2
	Exam 2	
Week 12	Variants of TM and other models	8.4,8.5
Week 13	Undecidability	9.1,9.2
Week 14	Undecidability II	9.3,9.4,9.5
Week 15	Intractability	10.1,10.2

8 Make up Policy

- No make up exam is given unless there was a valid reason for missing the exam (see below). The grade of the missed exam will be counted as zero in the final average of the course.
- A reason for missing the exam is deemed valid if an excuse can be obtained from the SAO.
- When a make up exam is given 10% are automatically deducted from the grade of the exam.

9 Attendance Policy

a) The maximum number of absences for classes that meet on MWF is six, the maximum number for classes that meet on TTH is four. Any student whose absences exceed the maximum limits shall automatically fail the course unless the student withdraws.
The word "absences" above includes authorized (excused) as well as unauthorized (unexcused) absences. Even below the maximum number of absences, a pattern of absences, whether authorized or not, may alter ones grade substantially.
Students who wish to validate their absences have to do so through the SAO, which alone authorizes absences. Finally, no absence absolves a student from the responsibility of acting upon the material presented during his/her absence.

b) Students registering during the late registration day shall be responsible for all work assigned from the beginning of the semester or the session. They shall be also subject to the requirements of the attendance policy (see a) above) as of the first day of classes.

c) Attendance policy applies as of day one of the semester. For students having joined their classes after the drop/add or the late registration period, the attendance policy applies as of the official time of registration in the course, not as of the students first appearance in class.

d) A student having exceeded the maximum number of allowed absences may ask permission from his/her instructor to keep attending classes. However, he/she shall not be allowed to take any remaining quiz or final exam or to have any graded assignment, and he/she will be assigned an automatic UW. Even if the number of allowed absences has been exceeded after the date of official withdrawal from classes, the student will still get an automatic UW, and will not be admitted in any remaining quiz or final exam.

10 Use of Original Textbooks

- The University does not allow photocopied textbooks in the classroom.
- Students with photocopied textbooks in a classroom are given one weeks notice to use a printed copy.
- After the one-week notice students with photocopied textbooks are not accepted to be in the classroom.
- Students who are not accepted in the classroom are reported as absent and the absence policy (see Attendance policy above) is applied to them for that particular course.

11 Smoking policy

NDU prohibits smoking in all indoor facilities. This includes the FNAS premises. Consequently, students are asked to refrain from smoking when entering the FNAS building, and not only the classroom. Disciplinary measures against contravening students may be taken starting from a written warning up to and including freezing student facilities and having a written remark on the students transcript of records. (Excerpts from Student Handbook 2007-08, pp. 65-66)

12 Policy on Academic Integrity

Students are urgently requested to read this policy on pages 43-44 of the Student Handbook 2007-08 and to abide by its standards. Students will be held accountable for any academic dishonesty or misconduct violating this policy.