CPSC390A: Theory of Computation

Fall 2014
Dr. Anil M. Shende
Trexler 365A
Email: shende at roanoke dot edu
Tel: 540-375-2341
Office Hours: MT 1:00-2:30pm; F 12:30-1:30pm
and by appointment.

Syllabus

Course Description
This course deals with the study of formal models of computation. Topics include formal languages, automata theory, Turing machines, undecidability, and an introduction to computational complexity.

Text

An Introduction to the Theory of Computer Science, Languages and Machines, 3rd edition by Thomas A. Sudkamp

Prerequisites

CPSC 170 and MATH 131.

Intended Learning Outcomes

By the end of the course, successful students will have the following abilities:

1. Students will be able to use various proof techniques, e.g., proof by contradiction and mathematical induction, in proving properties of formal languages, e.g., regular and context-free.
2. Students will understand the formalisms of finite state automata and pushdown automata and their relationship with regular languages and context-free languages, respectively. Students will be able to construct, when possible, such automata to prove a given language to be of a certain type (regular, context-free).
3. Students will understand the formalism of a Turing machine and be able to construct such machines for Turing computable problems.
4. Students will understand the notion of undecidability and be able to prove problems to be undecidable.

Class Attendance

Regular attendance in class is highly recommended. Regardless of attendance, students are responsible for all material covered or assigned in class.

Mechanics
The course will meet in class for 3 hours during the week. There will be three tests in class during the semester, and a final exam on Tuesday, December 9, 2014, from 8:30am to 11:30am.

Make-up tests will be available by pre-arrangement only in case of scheduling conflicts. After the test, make-ups will be available only in case of documented medical emergencies.

Besides the exams, there will be quizzes in class, regular homework assignments, and a co-curricular requirement.

**Quizzes:** Quizzes will be in class and will be announced one class period before the quiz.

**Homework:** Homework will be assigned on a regular basis and posted at the course website. All homeworks are due at the beginning of class on the posted due date. It is highly recommended that you hand in printouts of your homework typeset in LaTeX. The course website has a tutorial on LaTeX. **Late home works will not be accepted.**

**Co-curricular Requirement:** The Mathematics, Computer Science and Physics department offers a series of discussions that appeal to a broad range of interests related to these fields of study. These co-curricular sessions will engage the community to think about ongoing research, novel applications and other issues that face these disciplines. Each student is required to attend at least three of these sessions, and turn in a short paper describing the contents of the session, and his/her critical reflections about the topic and content. These papers are due in class within a week of the session. A paper submitted beyond a week from the event being discussed in the paper will not be accepted. The **MCSP Conversation Series website** has the schedule of talks in the series.

**Grading**

The final grade will be computed based on the grades in the quizzes, tests, the final exam, home works, and co-curricular activities according to the following weights.

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Co-curricular</td>
<td>4%</td>
</tr>
<tr>
<td>Homeworks</td>
<td>20%</td>
</tr>
<tr>
<td>Quizzes</td>
<td>16%</td>
</tr>
<tr>
<td>Tests (3)</td>
<td>36%</td>
</tr>
<tr>
<td>Final Exam</td>
<td>24%</td>
</tr>
</tbody>
</table>

The final course grade will be calculated as follows:

<table>
<thead>
<tr>
<th>&lt; 60</th>
<th>60-62</th>
<th>63-65</th>
<th>66-69</th>
<th>70-72</th>
<th>73-75</th>
<th>76-79</th>
<th>80-82</th>
<th>83-85</th>
<th>86-89</th>
<th>90-92</th>
<th>&gt; 92</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>D-</td>
<td>D</td>
<td>D+</td>
<td>C-</td>
<td>C</td>
<td>C+</td>
<td>B-</td>
<td>B</td>
<td>B+</td>
<td>A-</td>
<td>A</td>
</tr>
</tbody>
</table>

**Academic Integrity**

Students are expected to adhere to the Academic Integrity policies of Roanoke College. All work submitted for a grade is to be strictly the work of the student unless otherwise specified by the instructor. The policies as outlined in the Academic Integrity handbook will be enforced in the course.

Graded programs are subject to the Roanoke College Academic Integrity policies. Copying a program or a portion of a program (even a single line) or reading another person’s program to obtain ideas for solving a problem is plagiarism. Other examples of integrity violation include writing code for someone else, using code written by someone else, telling someone else how to solve a problem or having someone tell you how to solve a problem (and using their method). These cases apply to any work that is handed in for a grade under the instructor's assumption that the work is your own. Unless specified...
otherwise by the instructor, discussion among students should be limited to general discussion of concepts and language details, not specific aspects of a solution to the assigned problem.