CmSc 250 DM: Introduction to Design and Analysis of Algorithms

Instructor: Lydia Sinapova
Office: Carver Science 235, Phone: 961-1841 (office), 962-1574 (home)

Course Description
This course provides a survey of useful techniques for solving problems that arise frequently in computer applications. Topics to be covered include advanced data structures, advanced sorting and searching algorithms, graph representations, algorithm design techniques, and the fundamentals of computational complexity and analysis of algorithms. Laboratory exercises will be used to demonstrate problem solving techniques. Programming assignments and laboratory work will be done in Java programming language.

Course objectives
- Understand basic data structures for searching and sorting – hash tables, trees, heaps, and the computational complexity of the searching and sorting algorithms that use these structures
- Understand advanced sorting algorithm and their computational complexity
- Understand basic graph algorithms and their computational complexity
- Understand basic classes of algorithms, such as greedy algorithms, divide-and-conquer algorithms, and backtracking algorithms.
- Understand the concepts of computational complexity and computability and be able to apply in practice the Big-Oh notation technique for complexity evaluation.

Prerequisites: CmSc155 Fundamentals of Computing II, CmSc180 Discrete Mathematics (Same as CmSc 175). Math background: basic math knowledge of polynomial, logarithmic and exponential functions, ability to manipulate mathematical expressions.


Course work, assignments and tests
This course is tough. You have to dedicate a considerable amount of time to prepare your assignments. For each hour spent in class the estimated minimum preparation time is three hours. In other words, plan at least 12 hours per week for studying.

All assignments have due dates. They have to be turned in via Scholar on or before the due date. The names of the folder and all files in the assignment have to start with your first name, e.g. John_HW01.doc.

You will have the right to turn late two assignments without penalty, but not later than 48 hours after the deadline. Use this option in case of unforeseen emergency. No other late assignments will be accepted. If you happen to be seriously ill or some other events prevent you from meeting the above requirements, you have to make personal arrangements with me for a make-up work.

Policy on Late assignments
The penalty for being late is 10% reduction of the grade for each day after the due date.
You will have the option to turn in late (but not later than 24 hours after the due time) two assignments without penalty. Use this option in case of unexpected circumstances.
Homework assignments
You will have two types of homework assignments: problem solving and programming.

- Problem solving assignments are based (but not limited to) on the exercises provided in the textbook. They have to be turned in by the due date as WORD documents.
- Programming assignments may be completed using BlueJ or Eclipse. Details how to submit programs will be given in class.

You are encouraged to send your homework earlier (at least 48 hours before the due time) for feedback. This will not affect your grade.

Laboratory work
During our laboratory sessions we shall do various programs to exemplify the theory and the algorithms taught in class. **At the end of each lab you will have to turn in the source code of the programs you have developed even if they are not completed, sent by email.** Completed programs and a written report describing your lab work have to be sent on or before the posted due date (usually on Thursdays). Programs must run on the computers in the lab. Instructions about the contents of the reports will be given in each lab assignment. Your work in the lab will be graded. You have to do each lab assignment even if you happen to be absent.

Research project
The purpose of the research project is to extend your knowledge beyond the scope of the material discussed in class. You will gain experience in exploring new topics and learning about new methods and approaches in algorithm design. The work involves researching resources, studying some new theory, and developing a program that illustrates a specific approach or a method to solve problems within the scope of the researched topic. I will provide significant support in terms of materials and algorithms/source code if available.

A list of research topics will be posted on the class web page. Research topics will be assigned during the third week of the semester. Progress on the research projects will be recorded in three intermediate reports, draft paper and a final report. Deadlines are given in the class schedule. All research projects will be presented in class during the last week of the semester. The outcome of your work will be a paper and a program to be demonstrated during the presentation.

Exams: There will be three unit tests and a final exam. The final exam is comprehensive.

Instructional Help: I will be generally available for any type of questions and consultations during and outside my office hours. You are encouraged also to send questions by e-mail.

Grading Policies
The final grade will be determined by (but not necessarily equal to) the ratio of the points you have earned and the total number of points. The assignments and tests will be weighted as follows:

<table>
<thead>
<tr>
<th>Component</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laboratory work</td>
<td>10%</td>
</tr>
<tr>
<td>Homework assignments</td>
<td>20%</td>
</tr>
<tr>
<td>Research project</td>
<td>20%</td>
</tr>
<tr>
<td>Unit tests</td>
<td>25%</td>
</tr>
<tr>
<td>Final exam</td>
<td>25%</td>
</tr>
</tbody>
</table>
Necessary conditions to pass this class: The average in each type of course work listed in the table above should be at least 60%

Grades follow a normal distribution as shown in the following table where the percentage is the lowest percentage allowed to obtain that grade.

<table>
<thead>
<tr>
<th>Grade</th>
<th>B +</th>
<th>87.0%</th>
<th>C +</th>
<th>77.0%</th>
<th>D +</th>
<th>67.0%</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>93.0%</td>
<td></td>
<td>C</td>
<td>73.0%</td>
<td>D</td>
<td>63.0%</td>
</tr>
<tr>
<td>A -</td>
<td>90.0%</td>
<td>B -</td>
<td>80.0%</td>
<td>C -</td>
<td>70.0%</td>
<td>D -</td>
</tr>
</tbody>
</table>

**How to Succeed in This Class**
- Read the lecture notes and the discussed sections in the textbook before and after each class session. Send (by e-mail) questions on specific paragraphs that are difficult for you to understand.
- Start to work early on your homework. In case of difficulties, send me your questions.
- Attend classes and participate in class discussions. Take notes. Spending 15 minutes to read your notes immediately after going home is very much worth the effort.
- Make a weekly schedule of the time you are going to spend studying for this class. Generally you will need 12 hours per week spend in studying, including work on homework assignments.

**Attendance**
I will take attendance as required for financial aid purposes. However the basic reason I want you to be in class is the importance of class and lab attendance for efficient learning. Remember that you are required to do all kinds of assignments no matter whether you are present or absent.

**Inclement Weather**
In case classes are canceled due to inclement weather, I will send you an e-mail with instructions how to proceed with the material scheduled for the canceled class meeting.

**Continuity Plan**
Should the normal instructional activity on the campus be interrupted or shortened by a campus-wide closing, students will receive information from the instructor or other representative of the college about when and if the course might be continued or completed via Internet, telephone, or United States mail.

**Policies on Plagiarism**
All work you turn in should be entirely your own. Discussing a problem with other students is encouraged, however you have to be able to explain and justify each step of your solution. Cheating results in disgracing yourself and failing the class.

**Miscellaneous**
Any issues not discussed above will be solved on a case by case basis.
Schedule CmSc 250 DM: Introduction to Design and Analysis of Algorithms,  
Spring Semester 2011

<table>
<thead>
<tr>
<th>Week</th>
<th>Topics</th>
<th>Assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Week 1</strong> &lt;br&gt; 01/11</td>
<td>Chapter 1: Mathematical Background</td>
<td></td>
</tr>
<tr>
<td><strong>Week 2</strong> &lt;br&gt; 01/18</td>
<td>Chapter 2: Algorithm analysis</td>
<td>HW 01 : due 01/22</td>
</tr>
</tbody>
</table>
| **Week 3** <br> 01/25 | Chapter 3: Lists, Stacks, Queues (overview)  
Laboratory 1: Stacks, Queues (due 01/29) | HW 02 : due 01/29  |
| **Week 4** <br> 02/01 | CANCELED                                                               |                    |
| **Week 5** <br> 02/08 | Chapter 4: Trees, Binary Search Trees, AVL Trees (4.1, 4.2, 4.3, 4.4)  
Laboratory 2: Trees (due 02/12) |                    |
| **Week 6** <br> 02/15 | Chapter 5: Hashing  
Laboratory 3: Hashing (due 02/19) | RP report 1 due 02/19 |
| **Week 7** <br> 02/22 | Unit Test 1A (Big Oh Analysis, Binary Trees)  
Chapter 6: Priority queues  
Laboratory 4: K-select problem (due 02/26) | HW 03 : due 02/26  |
| **Week 8** <br> 03/01 | Chapter 7: Sorting (7.1, 7.2, 7.3, 7.4, 7.5)  
Laboratory 5: Sorting (due 03/05) | RP report 2 due 03/05 |
| **Week 9** <br> 03/08 | Chapter 7: Solving recurrence relations.  
Mergesort, Quicksort (7.6, 7.7)  
Laboratory 6: Sorting (due 03/12) | HW 04 : due 03/12  |
| **Spring break (03/12 – 03/20)** |                                                                         |                    |
| **Week 10** <br> 03/22 | Unit Test 2 (Hashing, Priority queues, Sorting)  
Chapter 9: Graph algorithms (9.1, 9.2)  
Laboratory 7: Topological sorting (due 03/26) | RP report 3 due 03/26 |
| **Week 11** <br> 03/29 | Chapter 9: Graph algorithms (9.3)  
Laboratory 8: Shortest paths (due 04/02) | HW 05 : due 04/02  |
| **Week 12** <br> 04/05 | Chapter 9: Graph algorithms (9.4)  
Laboratory 9: Network flow (due 04/09) | Draft paper due 04/09 |
| **Week 13** <br> 04/12 | Chapter 9: Graph algorithms (9.5)  
Laboratory 10: Spanning trees (due 04/16) | HW 06 : due 04/16  |
| **Week 14** <br> 04/19 | Unit Test 3 (Graphs)  
Chapter 10: Algorithm Design Techniques | Final paper due 04/23 |
| **Week 15** <br> 04/26 | Computational complexity revisited, NP-hard problems  
Research project presentations  
Final Exam review |                    |

*Final Exam Tuesday 05/03*