Welcome to CSci 5421!

In this course, we will discuss a variety of techniques for the design and analysis of efficient computer algorithms and data structures. Topics that we plan to cover include divide-and-conquer, dynamic programming, the greedy method, matroid theory, balanced search trees, augmented data structures, techniques for amortized analysis, design of amortized-efficient data structures, geometric problems, graph problems, approximation algorithms for NP-complete problems, and some randomized algorithms. We will explore the theoretical underpinnings of these methods and will illustrate their uses with examples drawn from a variety of problem domains.

To succeed in this course, you must be familiar with elementary data structures (lists, stacks, queues, heaps, trees, etc.), basic algorithms (sorting, binary search, etc.), standard analysis techniques (summations, recurrences, big-O notation, etc.), some basic probability theory (random variables, expected value, etc.), and standard proof methods (contradiction, induction, etc.). Such background can generally be obtained by taking, for instance, CSci 4041 and CSci 2011 and by reviewing Ch. 1–3, 5.1–5.3, and Appendices A, B, C.1–C.4 in the text. I will assume that everyone is familiar with these concepts and will not review them in the course. If you are unsure about your background for the course, please talk to me right away. Additionally, you should read ahead for class, attend the lectures regularly and participate actively in them, and work out as many problems as possible from the text. The pace of coverage will be somewhat brisk and assignments may require you to delve deeper into some topics introduced briefly in lecture. Please do avail of the assistance offered by yours truly and by the Teaching Assistant. We’re here to help you do your best.

The rest of this document describes the course in more detail. In particular, please pay attention to the section on class policies. These policies are designed to ensure that the course proceeds smoothly and they will be implemented from the very beginning.

Once again, welcome to CSci 5421 and best wishes for a successful semester. I hope that you enjoy the course as much as I enjoy teaching it.
Detailed information

When/Where/etc.: MW 8:15–9:30 a.m. in 3–111 Keller Hall. Credits: 3. Prerequisite: CSci 4041 or instructor’s consent.

Teaching assistant: Mr. Jie Xue (xuexx193@umn.edu). Office hours: TTh 2:00–3:00 p.m. in 2–209 Keller Hall; (612)-626-7512.


Coursework: Five homework assignments (no programming), two midterm exams, and a final exam. (See schedule on the following pages.)

The exams will be in the regular classroom and are closed book. The final exam will be cumulative, i.e., it will cover all of the material discussed during the semester.

Evaluation: Grades will be based on a weighted average of the assignments (45%; weighted equally), midterm exams (30%; weighted equally), and final exam (25%).

Grades will be assigned on the following scale:

- $A \geq 90\%$
- $A- \geq 85\%$
- $B+ \geq 80\%$
- $B \geq 75\%$
- $B- \geq 70\%$
- $C+ \geq 65\%$
- $C \geq 60\%$
- $C- = S \geq 55\%$
- $D+ \geq 50\%$
- $D \geq 45\%$
- $F < 45\%$

Note that the weighted average is not rounded up when determining the course grade.

Important class policies—Please read carefully!

Assignments and exams: To qualify for full credit, a legible hardcopy of each homework assignment must be submitted in its entirety at the beginning of class on the due date. Work submitted after the start of class (i.e., 8:15 a.m.) is considered late. Late work will be accepted in the instructor’s office until 10:00 a.m. on the due date but will lose 10% of the total points that the assignment is worth. No work will be accepted after that.

Do not email your homework to the instructor or TA, or leave it in the department mailbox. Such submissions are difficult to track and could get lost or overlooked.

Please keep the exam dates in mind when planning your Thanksgiving and Christmas Break travel. Exams must be taken as scheduled.

Any exceptions to the above policies on timely submission of assignments and taking of exams will be at the instructor’s discretion and will be made in a manner that is consistent with University policy (https://policy.umn.edu/education/makeupwork).

Incomplete: An ‘Incomplete’ is given very rarely and will be considered only if a student is doing well but is unable to complete a small part of the course due to an emergency. In particular, an ‘Incomplete’ will not be given for grade improvement purposes.

Re-grade requests: Requests for re-grading questions on an assignment/midterm will be considered if submitted in writing within one week from the time the work is returned in class. (For logistical reasons, it will not be possible to consider re-grade requests for the last homework or the final exam.) Note that the score may change in either direction as a result
of a re-grade. The instructor reserves the right to limit the number and scope of re-grades requested by a student.

**Academic integrity:** All coursework must be done independently. You may discuss an assignment problem in general terms with your friends, but the final answer must be your own. Copying or interfering with the work of another student, plagiarizing from another source (including the Internet), or any other misrepresentation of your work constitutes cheating. Penalties will range from a zero for the entire assignment/exam in question to an “F” for the course. The case will also be referred to the student’s department and/or research advisor and to the Office for Community Standards. Students have been known to lose their assistantships as a result of cheating. You are urged to read the CS&E Department’s official policy on academic conduct.

**Web page:** Throughout the semester, information about the course, including schedule updates, homework assignments, and late-breaking class news, will be posted on the class web page. The page also links to an online forum, where students can discuss material related to the course. Please check the class web page regularly.

**Streaming video access:** For the benefit of UNITE students, class lectures will be available in real-time on the Internet, via streaming video. As a courtesy, UNITE will offer streaming video access to in-class students, too. Per UNITE policy, lectures will be available to in-class students with a ten-day delay. Note that this feature is intended to supplement in-class lectures. It is *not* a substitute for attending classes, so please do not abuse this privilege. The link for this feature can be found on the class web page.

### Schedule

Topics will be covered in roughly the following order; however, schedule, content, and depth of coverage of certain topics are subject to change. Readings are indicated in square brackets, either as chapters from the 3rd edition of the text or as papers posted on the class web page. You should augment these with notes you take in class as discussions in class will often range beyond what is found in the readings.

**Divide & conquer:** Principles; Master Theorem; Applications to integer and matrix multiplication, closest pair; Worst-case and randomized algorithms for order-statistics. [Ch. 4.2, 4.5, 9.2, 9.3, 33.4]

**Dynamic programming:** Principles; Applications to matrix-chain multiplication, longest common subsequence, optimal binary search trees, all-pairs shortest paths. [Ch. 15.2–15.5, 25.2]

**The greedy method:** Principles; Applications to activity-selection, minimum spanning trees; Comparison to dynamic programming. [Ch. 16.1, 16.2, 23]

**Matroid theory:** Principles; Relationship to the greedy method; Application to minimum spanning trees and to task scheduling. [Ch. 16.4, 16.5]

**Data structures for searching:** Binary search trees; Red-black trees; Augmented search structures (OS-trees, general augmentation theorem, Interval trees); Sweepline technique. [Ch. 12, 13, 14]
**Amortized analysis:** Principles; Analysis techniques. [Ch. 17.1–17.3]

**Design of amortized-efficient data structures:** Dynamic tables; Persistent search trees and applications to geometric search; Fibonacci heaps and applications to shortest paths; Splay trees. [Ch. 17.4, 19, papers]

**Approximation algorithms for NP-complete problems:** Principles; Approximation algorithms for vertex cover, traveling salesperson, and set cover; Use of randomization and Linear Programming. [Ch. 35.1–35.4]

**Geometric algorithms:** Geometric primitives; Convex hulls; Segment intersection. [33.1–33.3]

**Homework and exam dates:**

*Note:* The first class meeting is on Wed., 9/6 and the last one is on Wed., 12/13.

<table>
<thead>
<tr>
<th>Out</th>
<th>Due</th>
<th>EXAMS</th>
</tr>
</thead>
<tbody>
<tr>
<td>9/6</td>
<td>9/20</td>
<td>MIDTERM 1: Mon., 10/9, 8:15 a.m.-9:30 a.m.</td>
</tr>
<tr>
<td>9/20</td>
<td>10/4</td>
<td>MIDTERM 2: Mon., 11/20, 8:15 a.m.-9:30 a.m.</td>
</tr>
<tr>
<td>10/11</td>
<td>10/30</td>
<td>FINAL: Sat., 12/16, 10:30 a.m.-12:30 p.m.</td>
</tr>
<tr>
<td>10/30</td>
<td>11/15</td>
<td></td>
</tr>
<tr>
<td>11/22</td>
<td>12/11</td>
<td></td>
</tr>
</tbody>
</table>

**Extra lectures:** Depending on our progress, I may need to add up to two extra lectures towards the end of the semester. These will be on Friday, 12/1 and Friday, 12/8, from 8:15–9:30 a.m. in 3–125 Keller Hall. It will be helpful for you to attend these lectures but, in any case, they will be recorded and made available immediately, without the usual ten-day delay.