



University  
of Regina



Faculty of  
Science

## CS 340-001 – ADVANCED DATA STRUCTURES AND ALGORITHM DESIGN WINTER - 2024

**Course Instructor:** [Malek Mouhoub](#)

**Lectures:** Monday 1:00 PM – 2:15 PM via Zoom (the link is available on [UR Courses](#))

**Office hours:** Monday 2:30 PM – 3:45 PM via Zoom

**Course description:** Fundamental algorithms: depth- and breadth-first traversals, pattern matching, and graph algorithms. Algorithmic strategies: brute-force, greedy, divide-and-conquer, backtracking, branch-and-bound, dynamic programming, and randomized. Algorithm analysis, complexity theory, performance evaluation. Parallelism: fundamentals, algorithms, communication.

**Prerequisites:** CS 210. Advanced algorithmic and OO programming (in C++) skills are required for this course. These include recursion, ADTs Lists, Stacks, Queues, and Trees.

### Learning outcomes:

1. List and contrast standard complexity classes. Explain the use of big O, big omega, and big theta notations to formally give asymptotic bounds on time and space complexity of algorithms. Illustrate, through examples, the time-space trade-offs of algorithms. Solve recurrence relations to determine the time complexity of recursively defined algorithms.
2. Describe the heap property and the use of heaps as an implementation of priority queues.
3. Discuss the runtime and memory efficiency of principal algorithms for sorting, searching, and hashing. Define the lower bound for sorting. Describe bucket sort and radix sort, indirect sorting and external sorting, and order statistics.
4. Solve problems using graph algorithms, including topological sort, shortest path, network flow, minimum spanning trees, depth-first, and breadth-first search.
5. Solve combinatorial problems using greedy algorithms, divide and conquer, dynamic programming, randomized algorithms, backtracking algorithms, and branch and bound. Define the classes P and NP and explain the significance of NP-completeness.
6. For a given program, distinguish between its sequential and parallel execution, and the performance implications thereof. Define the differences between the concepts of Instruction Parallelism, Data Parallelism, Thread Parallelism/Multitasking, and Task/Request Parallelism.

**Textbook:** [Data Structures and Algorithm Analysis in C++ 4th edition by Mark Allen Weiss](#)

## Grading

Assignments (4x5%)	20%
Midterm exam	30%
Final exam	50%
-----	
Total	100%

**Tentative Schedule:** Students are required to watch the video related to each chapter before its scheduled time.

Week #	Date	Chapter	Topic	Assignments/Project
1	Monday, January 8, 2024	1	Review on Alg Analysis & Tree (BT, BST)	A1 posted
2	Monday, January 15, 2024	1	AVL (Addition/Deletion), Splay Trees and B-Trees	
3	Monday, January 22, 2024	2	P Queues	
4	Monday, January 29, 2024	3	Sorting (until Quicksort)	A2 posted/ A1 due
5	Monday, February 5, 2024	3	Sorting (until the General Bound for Sorting)	
6	Monday, February 12, 2024	3	Sorting	
7	Monday, February 19, 2024	Winter Break - No classes		
8	Monday, February 26, 2024	MIDTERM EXAM		A3 posted/ A2 due
9	Monday, March 4, 2024	4	Graphs and Networks (until Net. Flow Problems)	
10	Monday, March 11, 2024	4	Graphs and Networks	
11	Monday, March 18, 2024	5	Advanced Algorithms (until Dynamic Programming)	A4 posted/ A3 due
12	Monday, March 25, 2024	5	Advanced Algorithms	
13	Monday, April 1, 2024	5	Parallel Algorithms	
14	Monday, April 8, 2024	6	Review for the final exam	A4 due
	Wednesday, April 24, 2024	Final Exam		

## Topics

- Review of Algorithm Analysis & Trees:** Asymptotic notations and growth of functions. Overview of general, binary and search trees. AVL trees, Splay trees, B-trees.
- Priority Queues:** Binary heap, d-heap, Min-Max heap, leftist heap, binomial heap.
- Sorting and Order Statistics:** Overview and classification of sorting algorithms, insertion sort, merge sort, heapsort, quicksort, lower bound for sorting, bucket sort and radix sort, indirect sorting and external sorting, Order statistics.
- Graphs and Networks:** Implementation of graphs, topological sort, shortest path, network flow problems, minimum spanning tree, search techniques, introduction to NP-completeness.
- Algorithm Design Techniques:** Combinatorial optimization, greedy algorithms, divide and conquer, dynamic programming, randomized algorithms, backtracking algorithms, branch and bound.
- Advanced Algorithms (as time permits):** Distributed Algorithms. Parallel Algorithms. Advanced algorithmic analysis: Amortized analysis, online and offline algorithms.

## Policies

**Course and Contact Information:** Class material (including recorded lectures) is available online through UR Courses. News and announcements are posted on UR Courses, including UR Courses Email. It is the responsibility of students to regularly check their emails. Students should turn on their UR Courses Email notifications. Students should use UR Courses Email to contact the instructor and the marker.

**Attendance policy:** Attendance is mandatory in Zoom scheduled lectures. Little time is available to assist those who have missed these scheduled classes. UR Courses should be used for synchronous Zoom-based lectures/discussions/office hours, asynchronous video lectures, and other class materials. Essential computer equipment is required for remote learning. Students will need a computer or laptop, microphone, speakers or headset. Using a smartphone for UR Courses is possible but not recommended. Please visit the [technical resource page](#) for more details on technology requirements.

**Assignments policy:** All assignments are available online on UR Courses and must be submitted electronically on [UR Courses](#). It is the responsibility of students to make sure that the submitted material has been successfully uploaded to UR Courses before the assignment's due date. Submission status can be checked by viewing the uploaded files. Email and Hardcopy submissions are not accepted. Partially completed assignments are acceptable but LATE assignments are not accepted for any reason and will receive 0 points, except for extensions granted to the entire class. If you are unable to complete and submit an assignment in time due to health problems, [a self-declaration of illness](#) is required (this applies to other unusual cases). In this situation, the grade you get in the final exam will be assigned to your missed assignment grade. All your programming assignments must compile and run with CC on Hercules/Titan. Submitted programs that do not properly compile and run on Hercules/Titan will receive 0 points.

Marking will also be done on UR Courses. If a student does not agree with the marking, the marker must be notified within one week after the assignment grade is posted. Any request made more than one week after the grade posted date will not be considered.

**Midterm exam policy:** The midterm exam is closed book and will be given online via UR Courses during the regular lecture meeting time and using the Proctortrack platform. If you missed the midterm exam (due to health problems or any other unusual cases), you must provide sufficient evidence justifying why you missed the midterm exam. In this situation, the grade you get in the final exam will be assigned to your midterm exam grade.

**Final exam policy:** The final exam is cumulative and closed book. It will be conducted online via UR Courses using the Proctortrack platform. Deferred final examinations can only be granted by the Associate Dean (Academic) (for Faculty of Science students), or by the Deans (and/or Associate/Assistant Deans) of other Faculties or Federated Colleges. Deferred final examinations cannot be granted by the course instructor.

**Academic integrity:** Academic integrity requires students to be honest. Assignments and exams are to help students learn; grades show how fully this goal is attained. Thus, all work and grades should result from a student's own understanding and effort.

Acts of academic misconduct violate academic integrity and are considered serious offenses by the University. Examples include, but are not limited to, cheating on tests or exams, plagiarizing, copying from others, falsifying lab results, etc. Instances of academic misconduct will be reported to the Associate Dean Academic for investigation. Full details are provided in the [Undergraduate academic calendar](#). Students are encouraged to understand their obligations as a student, as well as their rights.

**Accommodations:** The Centre for Student Accessibility upholds the University's commitment to a diverse and inclusive learning environment by providing services and support for students based on disability, religion, family status, and gender identity. Students who require these services are encouraged to contact the [Center for Student Accessibility](#) to discuss the possibility of academic accommodations and other supports as early as possible. For further information, please email [accessibility@uregina.ca](mailto:accessibility@uregina.ca).

### **Taking your Exam with Proctortrack**

1. Proctortrack is a remote proctoring tool that is integrated into UR Courses and provides for student identity verification and the monitoring of students while taking examinations remotely. This remote proctoring option allows University of Regina students to continue with remote learning in the current environment. When using Proctortrack remote proctoring service, your personal information is being collected and will be used for the purposes of creating a student profile account, verifying your identity, and proctoring your exam.
2. When you sign into Proctortrack, you will be asked to provide your consent, agreement and acknowledgment to allow Proctortrack to collect, create, process and store personal information. This personal information may include: U of R student card, live image captured via a webcam, first and last name, institution name, student number, and real-time audio, video and on-screen activity to prevent unauthorized viewing of content during an exam.
3. The personal information collected by Proctortrack will be used by the University of Regina for the purposes of identity verification and exam proctoring. Any video records of you created by Proctortrack will be kept by Proctortrack and shared as necessary with the University of Regina for assessment of possible academic integrity infractions. Non-relevant recordings are destroyed after 180 days. All personal information collected and stored by Proctortrack within your student profile account will be permanently deleted if the account has not been used after one year.
4. For more information, please reference the [remote proctoring FAQ document](#).