湖北工業大學
HUBEI UNIVERSITY OF TECHNOLOGY

Course Title	Fundamental Algorithms
Course Code	CMPT 3491
Semester	Summer 2026
Course Length	4 Weeks, 60 Contact Hours
Credits	4
Instructor	TBA
Office	TBA
Email	TBA
Prerequisite	CMPT 1011 Introduction to Computer Science CMPT 2141 Data Structures and Algorithms MATH 2331 Discrete Mathematics

Course Description:

This course introduces fundamental techniques for algorithm development and computational problem solving. Emphasis is placed on algorithmic thinking, asymptotic efficiency analysis, and the selection of appropriate data structures for practical and theoretical applications. Students examine classical algorithms for sorting, searching, graph traversal, optimization, and dynamic programming, while learning formal methods for evaluating time and space complexity.

The course also discusses algorithmic paradigms including divide-and-conquer, greedy strategies, and dynamic programming, together with selected topics in graph algorithms and computational complexity. Students are expected to develop the ability to compare alternative algorithmic solutions and justify design decisions based on correctness and efficiency considerations.

Course Goals:

Students who successfully complete this course will demonstrate competency in the following general education core goals:

- **Critical Thinking Skills** – Students will engage in analytical thinking, demonstrating the ability to critically evaluate, synthesize, and apply knowledge to complex problems, and construct well-reasoned solutions and arguments.
- **Independent Research and Inquiry** – Students will conduct independent research, utilizing academic resources to explore relevant topics, formulating research questions, analyzing data, and presenting findings in a coherent, scholarly manner.
- **Problem-Solving and Application** – Students will apply theoretical concepts and methodologies learned in the course to real-world problems, demonstrating the ability to develop practical solutions informed by academic inquiry.

- **Global and Cultural Awareness** – Students will gain awareness of the global and cultural contexts relevant to the course, appreciating diverse perspectives and considering the implications of their studies in a broader, international context.

Student Learning Outcomes:

Upon completion of this course, students will be able to:

- Analyze the efficiency of algorithms using asymptotic notation;
- Apply standard algorithm design paradigms to computational problems;
- Construct and evaluate recursive and iterative algorithms;
- Implement and compare classical sorting and searching methods;
- Solve optimization problems using greedy and dynamic programming techniques;
- Design and analyze graph traversal and shortest-path algorithms;
- Explain foundational concepts in computational complexity and tractability;
- Communicate algorithmic reasoning clearly using formal and technical methods.

Textbooks/Supplies/Materials/Equipment/ Technology or Technical Requirements:

Textbook:

Introduction to Algorithms (4th edition) by Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein. The MIT Press. 2022.

Technology Requirements:

Students may use any standard programming language approved by the instructor, such as: C++/Java/Python.

Programming assignments may be completed on Windows, Linux, or macOS platforms.

Course Requirements:

Problem Set Assignments (20%)

Students complete several written assignments. The assignments focus on asymptotic analysis, recurrence solving, correctness arguments, greedy-choice reasoning, and dynamic programming formulation.

Programming Exercises (15%)

Students complete several short programming exercises involving practical implementation of classical algorithms and data structures. Topics may include sorting algorithms, heap operations, graph traversal techniques, and shortest-path computation. Emphasis is placed on correctness, efficiency, code organization, and testing practices.

Midterm Examination (25%)

The midterm examination assesses students' understanding of foundational algorithmic techniques covered during the first half of the course. The exam may include complexity analysis, divide-and-conquer recurrences, sorting algorithms, order statistics, greedy strategies, and introductory dynamic programming problems.

Applied Algorithm Analysis Report (10%)

Students prepare an individual report analyzing a selected algorithmic problem or design paradigm. The report should compare alternative algorithmic approaches in terms of efficiency, scalability, implementation considerations, and theoretical limitations. Possible themes include graph algorithms, approximation methods, randomized algorithms, or computational complexity topics introduced near the end of the semester.

Final Examination (30%)
The final examination is cumulative and evaluates students' comprehensive understanding of algorithm design and analysis. The exam covers graph algorithms, shortest-path methods, spanning trees, complexity theory, approximation techniques, and previously introduced analytical foundations. Students are expected to demonstrate the ability to apply algorithmic paradigms to unfamiliar computational problems.

Assessments: Activity	**Percent Contribution**
Problem Set Assignments	20%
Programming Exercises	15%
Midterm Examination	25%
Applied Algorithm Analysis Report	10%
Final Examination	30%

Grading:

Final grades will be based on the sum of all possible course points as noted above.

Grade	**Percentage of available points**
A	94-100
A-	90-93
B+	87-89
B	84-86
B-	80-83
C+	77-79
C	74-76
C-	70-73
D	64-69
D-	60-63
F	0-59

Course Schedule:
The schedule of activities is subject to change at the reasonable discretion of the instructor. Minor changes will be announced in class, major ones provided in writing.

CMPT 3491 Schedule		
Lecture	Topic	Readings
L1	Introduction to algorithms; role of algorithms in computing	Ch.1
L2	Insertion sort and basic algorithm analysis	Ch.2.1-2.2
L3	Asymptotic notation and growth of functions	Ch.3.1-3.3
L4	Divide-and-conquer; merge sort	Ch.2.3, Ch.4
L5	Solving recurrences: substitution, recursion-tree, master	Ch.4.3-4.5

	method	
L6	Heaps and heap sort	Ch.6.1-6.4
L7	Priority queues; quicksort and randomized quicksort *Programming Exercise 1	Ch.6.5, Ch.7.1-7.4
L8	Lower bounds for sorting; counting sort and radix sort	Ch.8.1-8.3
L9	Selection algorithms and order statistics	Ch.9.1-9.3
L10	Probabilistic analysis and randomized algorithms	Ch.5.1-5.3
L11	Greedy algorithms: activity selection and Huffman coding *Programming Exercise 2	Ch.15.1-15.3
L12	Introduction to dynamic programming; rod cutting	Ch.14.1-14.3
L13	Longest common subsequence and matrix-chain multiplication	Ch.14.2, Ch.14.4
L14	Binary search trees and balanced search trees overview	Ch.12.1-12.3, Ch.13.1-13.3
L15	**Midterm Examination**	
L16	Elementary graph representations and breadth-first search	Ch.20.1-20.2
L17	Depth-first search and edge classification *Programming Exercise 3	Ch.20.3
L18	Topological sorting and strongly connected components	Ch.20.4-20.5
L19	Minimum spanning trees: Kruskal and Prim algorithms	Ch.21.1-21.2
L20	Single-source shortest paths: Bellman-Ford and DAG shortest paths	Ch.22.1-22.2
L21	Dijkstra's algorithm and shortest-path properties *Report Assigned	Ch.22.3-22.5
L22	All-pairs shortest paths and Floyd-Warshall algorithm	Ch.23.1-23.2
L23	Disjoint sets and union-find structures *Programming Exercise 4	Ch.19.1-19.3
L24	Polynomial time, NP-completeness, and reducibility	Ch.34.1-34.4
L25	Approximation algorithms; course review *Report Due	Ch.35.1-35.3
---	**Final Examination**	

Accommodation Statement:

Academic accommodations may be made for any student who notifies the instructor of the need for an accommodation. It is imperative that you take the initiative to bring such needs to the instructor's attention, as he/she is not legally permitted to inquire. Students who may require assistance in emergency evacuations should contact the instructor as to the most appropriate procedures to follow.

Academic Integrity Statement

Each student is expected to maintain the highest standards of honesty and integrity in academic and professional matters. The University reserves the right to take disciplinary action, up to and including dismissal, against any student who is found guilty of academic dishonesty or otherwise fails to meet the standards. Any student judged to have engaged in academic dishonesty in coursework may receive a reduced or failing grade for the work in question and/or for the course.

Academic dishonesty includes, but is not limited to, dishonesty in quizzes, tests, or assignments; claiming credit for work not done or done by others; hindering the academic work of other students; misrepresenting academic or professional

qualifications within or without the University; and nondisclosure or misrepresentation in filling out applications or other University records.

Other Items:

Attendance and Expectations
All students are required to attend every class, except in cases of illness, serious family concerns, or other major problems. We expect that students will arrive on time, be prepared to listen and participate as appropriate, and stay for the duration of a meeting rather than drift in or out casually. In short, we anticipate that students will show professors and fellow students maximum consideration by minimizing the disturbances that cause interruptions in the learning process. This means that punctuality is a must, that cellular phones be turned off, and that courtesy is the guiding principle in all exchanges among students and faculty. You will be responsible for the materials and ideas presented in the lecture.

Assignment Due Dates
All written assignments must be turned in at the time specified. Late assignments will not be accepted unless prior information has been obtained from the instructor. If you believe you have extenuating circumstances, please contact the instructor as soon as possible.

Make-Up Work
The instructor will not provide students with class information or make-up assignments/quizzes/exams missed due to an unexcused absence. Absences will be excused and assignments/quizzes/exams may be made up only with written documentation of an authorized absence. Every effort should be made to avoid scheduling appointments during class. An excused student is responsible for requesting any missed information from the instructor and setting up any necessary appointments outside of class.

Access, Special Needs, and Disabilities
Please notify the instructor at the start of the semester if you have any documented disabilities, a medical issue, or any special circumstances that require attention, and the school will be happy to assist.