Computer Science (COMP) 674

Theory of Computation (Revision 3)

Status:	Replaced with new revision, see the **course listing** for the current revision
Delivery mode:	**Individualized study online**. Delivered via Brightspace.
Credits:	3
Area of study:	Information Systems
Prerequisites:	You should have a strong background in discrete mathematics, data structures, and algorithms. If you are concerned you may not meet the prerequisites for this course, contact the course coordinator before registering.

Precluded:	None
Faculty:	**Faculty of Science and Technology**
Notes:	This is a graduate-level course, and you must apply and be approved to one of the graduate programs or as a non-program **School of Computing and Information Systems** graduate student in order to take this course. The minimum admission requirements must be met. Undergraduate students who do not meet the admission requirements will not normally be permitted to take this course. Workload: In a 16-week study schedule, approximately 5 to 25 hours of coursework per week, depending on your level of mathematical knowledge and skills.
Coordinator:	**Dr. Dunwei (Grant) Wen**

Overview

Computer Science 674: Theory of Computation is a graduate course on the theoretical aspects of computing. The course will introduce you to the mathematical foundations of computation, including automata, formal languages and grammars, algorithms, decidability, computability, and complexity.

Why study theory when the current focus of computer science—and even more so of information systems—is on technology and the pragmatic knowledge required for the development and management of computer information systems? The reasons are manifold. Theory provides a simple, elegant view of the complex machine that we call a computer. Theory also has high permanence and stability, in contrast with the ever-changing paradigms of the technology, development, and management of computer systems.

Further, parts of the theory have direct bearing on practice. For example, automata are applicable to digital systems design, compiler design, search algorithms, and artificial intelligence. Formal languages and grammars are used in programming and human language processing. Complexity has direct applications in cryptography. And optimization and learning problems are used in manufacturing, business, and management.

COMP 674 is also recommended for research-oriented students, who will make good use of the theory studied in this course.

Outline

Unit 0: Formation of Preliminary Concepts

- Automata, computability, and complexity
- Mathematical tools
- Definitions, theorems, and proofs
- Types of proofs

Unit 1: Regular Languages

- Finite automata
- Nondeterminism
- Regular expressions
- Nonregular languages

Unit 2: Context-Free Languages

- Context-free grammars
- Pushdown automata
- Non-context-free languages

Unit 3: The Church–Turing Thesis

- Turing machines
- Variants of Turing machines
- Definition of *algorithm*

Unit 4: Decidability

- Decidable languages
- The halting problem

Unit 5: Reducibility

- Undecidable problems in language theory
- A simple undecidable problem
- Mapping reducibility

Unit 6: Advanced Topics in Computability

- The recursion theorem
- Decidability of logical theories
- Turing reducibility
- A definition of *information*

Unit 7: Time Complexity

- Measuring complexity
- The class P
- The class NP
- NP-completeness
- NP-complete problems

Learning outcomes

Upon successful completion of this course, you should be able to

- discuss key notions of computation—such as algorithm, computability, decidability, reducibility, and complexity—through problem-solving.
- explain the models of computation—including formal languages, grammars, and automata—and their connections with each other.
- state and explain the Church–Turing thesis and its significance.
- analyze and design finite automata, pushdown automata, Turing machines, formal languages, and grammars.
- solve computational problems to determine their computability and complexity and prove the basic results of the theory of computation.

Evaluation

To **receive credit** for COMP 674, you must achieve a course composite grade of at least **B– (70 percent)**, an average grade of at least 60 percent on the three assignments, and a grade of at least 60 percent on the final assessment.

The weighting of the composite grade is as follows:

Activity	Weight
Assignment 1	20%
Assignment 2	20%

Activity	Weight
Assignment 3	20%
Final assessment	40%
Total	**100%**

Materials

Physical course materials

The following course materials are included in a course package that will be shipped to your home prior to your course's start date:

Sipser, M. (2006). *Introduction to the theory of computation* (2nd ed.). Thompson Course Technology.

Important links

- Future Course Offerings
- Important Dates and Deadlines
- MSc CIS Contact Information

Opened in Revision 3, May 9, 2025

Updated June 16, 2026

View **previous revision**