



## COMPUTER SCIENCE GRADUATE STUDIES

### Course Descriptions by Methodology

**MSc Students must complete 4 Graduate Level Courses and cover breadth in 3 Methodologies.**

#### **METHODOLOGY 1 | Analysis and Computation in Discrete Models**

##### **CSC2104/465 Formal Methods in Software Design**

Using mathematics to write error-free programs. Proving each refinement; identifying errors as they are made. Program development to meet specifications; modifications that preserve correctness. Useful for all programming; essential for programs that lives depend on. Basic logic, formal specifications, refinement. Conditional, sequential, parallel, interaction, probabilistic programming, and functional programming.

**Prerequisite:** CSC236H1/CSC240H1/MAT309H1; **Recommended Preparation:** CSC463H1

##### **CSC2125H Topics in Software Engineering: Modeling Methods, Tools and Techniques**

Model-based software engineering (MBSE) is an approach to software development in which software models play a primary and indispensable role. MBSE allows developers to work and reason about software requirements, design, and correctness at higher levels of abstraction, and to generate automatically implementations, deployments, and other artifacts. MBSE has been successfully applied in several industries (automotive, aeronautic, information systems), though typically in an ad hoc basis. This course will look at the state of the art of MBSE and its future research directions.

##### **CSC2221H Introduction to Distributed Computing**

This course studies fundamental models and problems in distributed computing with an emphasis on synchronization and fault tolerance. Algorithms and impossibility results will both be considered.

**Prerequisites:** recommended course in Operating Systems (e.g., CSC369 or equivalent)

##### **CSC2226H Topics in Verification**

We assume foundational knowledge of program verification and synthesis, as introduced in other introductory graduate courses. The focus of this course is to explore the latest topics in program synthesis (and verification by association) to bring students up to date with the state of the art with research in this area. **Recommended preparation: Good background in Logic, Automata Theory, Algorithms and Data structures, some familiarity with program analysis (for example a course on Compilers), Having basic understanding of what program proofs are.**

**Ideally, CSC410 or equivalent.**

##### **CSC2401H Introduction to Computational Complexity**

This is a graduate introductory level course in computational complexity theory. It is appropriate for all graduate students in Computer Science. The main part of the course will be an introduction to complexity theory where we will discuss uniform and non-uniform models of computation, time and space complexity classes, complexity hierarchies, reductions and completeness, randomization in computation, approximation algorithms, and lower bound methods. In the latter part of the course, we will discuss several new results in complexity theory. Such advanced topics may include: Hastad's famous switching lemma and its applications, including lower bounds for constant-depth circuits, better upper bounds for satisfiability, and connections to learning theory; communication complexity and its applications; NP, coNP and propositional proof complexity.

### **CSC2404/438 Computability and Logic**

Computable functions, Church's thesis, unsolvable problems, recursively enumerable sets. Predicate calculus, including the completeness, compactness, and Lowenheim-Skolem theorems. Formal theories and the Gödel Incompleteness Theorem. Ordinarily offered in years alternating with CSC448H1. **Prerequisite: (CSC363H1/CSC463H1)/CSC365H1/CSC373H1/CSC375H1/MAT247H1**

### **CSC2410H Introduction to Graph Theory**

This course covers many of the most important aspects of graph theory, including the development and analysis of algorithms for problems which arise in applications of graph theory. Topics include minimum spanning tree, shortest path problems, network flow with applications to bipartite matching, general matching, planarity testing, graph colorability, etc. Attention will be given to the mathematical theory that supports the algorithms presented. **Recommended preparation: You should have a very basic familiarity with graphs. More importantly, you will need to be strong with mathematical proofs.**

### **CSC2414H Topics in Applied Discrete Mathematics: Discrepancy Theory in Computer Science**

Discrepancy theory is an area of mathematics that studies how well discrete objects can approximate continuous ones. This course will be an introduction to the theory, with a focus on its many applications to computer science. We will start with the basics, and move on into computational issues that arise in discrepancy theory itself. In the final part of the course we will explore applications to computational geometry, complexity theory, the design of approximation algorithms, private data analysis, and communication complexity. Along the way, we will learn about some beautiful and powerful techniques from combinatorics, linear algebra, and convex geometry. **In addition to mathematical maturity, you will need basic knowledge of linear algebra and probability theory.**

### **CSC2415H Advanced Topics in Distributed Computing**

Advanced course on distributed computing, aimed at students who wish to explore in depth a topic in this area. It will be run largely as a research seminar course. Each student will pick a topic from a list suggested by the instructors or a topic of the student's own choosing approved by the instructors. The student will then study the relevant literature, and will eventually present the subject to the class in one or more 2-3 hour sessions. **A course in theory of distributed computing, such as CSC2221. People with good backgrounds in complexity theory are welcome to take the course, provided they are willing to do some background reading.**

### **CSC2419H Topics in Cryptography: Algorithms and Complexity in Private Data Analysis**

The practical applicability of data analysis to sciences and decision-making is often limited by individual privacy concerns. Some of the most interesting data sets - medical studies, genetic data, movie reviews, search logs, social network connections - contain sensitive information about people represented in the data set. If individuals do not think their privacy concerns are addressed, they may refuse to participate in data collection, or not answer surveys truthfully. These issues limit the validity of the data analysis results, and make balancing privacy and usefulness essential to many applications of big data. In this course we will study privacy in data analysis from a rigorous theoretical perspective. We will focus on Differential Privacy: a recent approach to achieving strong provable privacy guarantees in the analysis of sensitive data. Informally, a data analysis algorithm is differentially private if changing the data of a single individual changes the output distribution of the algorithm only slightly. This guarantee ensures that the privacy risk to any individual increases only slightly by participating in data collection. Our focus will be on the design of efficient differentially private algorithms, but we will also look at some negative results that show the limits of private data analysis. In the process, we will learn about fascinating connections between differential privacy and game theory, learning theory, and geometry. **This is a theoretical, proof-based course. It will require mathematical maturity and a solid background in probability theory, and the design and analysis of algorithms.**

### **CSC2420H Algorithm Design, Analysis and Theory**

This course serves as a foundational course, appropriate for students in computer science, computer engineering, and mathematics. However, the course should also be of interest to "theory students" looking for research topics. We will begin with topics normally discussed in undergraduate courses such as our CSC373/CSC375 using some less standard examples and also discussing some precise models for standard algorithmic paradigms such as greedy, local search and dynamic programming algorithms. Other possible topics include:

online algorithms, the random order model, randomized algorithms and Markov chains, linear and submodular maximization subject to independence constraints, LP and SDP relaxations, primal dual algorithms, online primal dual algorithms, sublinear time algorithms, embedding techniques, data stream algorithms, MapReduce algorithms, multiplicative weights algorithms, spectral methods, algorithmic aspects of mechanism design, social networks and information retrieval. **Recommended Preparation: CSC373 or equivalent; course in probability.**

### **CSC2426H Fundamentals of Cryptography**

Rigorous definition of security for pseudo-random generators, shared private-key cryptosystems, public-key cryptosystems, and digital signature schemes. Methods (including number-theoretic conjectures) for constructing secure cryptographic primitives. Methods for using secure primitives to construct secure systems.

### **CSC2427H Topics in Graph Theory: Random Constraint Satisfaction Problems**

Produce a random graph by starting with  $n$  vertices, and then adding  $M$  random edges (in the obvious way). How big does  $M$  need to be so that the graph is probably not  $r$ -colourable? This is the oldest unsolved problem in random graph theory. Produce a random instance of  $k$ -SAT by starting with  $n$  variables, and then adding  $M$  random  $k$ -clauses (in the obvious way). How big does  $M$  need to be so that this is probably not satisfiable? These are two important examples of random constraint satisfaction problems. We will study these and several others. One important goal is to determine the values of  $M$  asked about above. Another is to understand some algorithmic issues: when the number of edges is at the upper end of the range where the graph is probably  $r$ -colourable, then it appears to be very challenging (perhaps impossible) to produce an actual  $r$ -colouring efficiently. Most recent successes on these problems have been focussed on some deep insights arising from statistical physics. Physicists have developed a set of remarkable hypotheses which have led to a very strong understanding of these random problems and the reasons that they are algorithmically challenging. Moreover, physicists have used these insights to develop a very impressive heuristic algorithm: Survey Propagation. This is a version of Belief Propagation which performs quite well on random 3-SAT. Much of this course will be focussed on understanding Survey Propagation, and other contributions from statistical physics. **A good introductory level background in graph theory, such as that provided in CSC 2410 or a similar undergraduate course, and a basic background in probability will be assumed.**

### **CSC2429H Topics in the Theory of Computation: Circuit Complexity**

This course presents an introduction to Circuit Complexity, an area of Theoretical Computer Science that aims to prove unconditional lower bounds in combinatorial models of computation (for instance, showing that the  $k$ -clique problem requires monotone circuits of large size). We will begin by reviewing the classic lower bounds for bounded-depth and monotone circuits and formulas (switching lemmas, the polynomial method, gate elimination, matrix rank). We will then focus on a new set of techniques that pin down the average-case complexity of subgraph isomorphism problems on Erdős-Rényi random graphs. Along the way, we will encounter useful tools from the probabilistic method and highlight various open problems. No prerequisites are required beyond basic mathematical maturity. **Prerequisite: CSC2404 (438) or equivalent course in logic**

### **CSC2502/486 Knowledge Representation & Reasoning**

Representing knowledge symbolically in a form suitable for automated reasoning, and associated reasoning methods. Topics from: first-order logic, entailment, the resolution method, Horn clauses, procedural representations, production systems, description logics, inheritance networks, defaults and probabilities, tractable reasoning, abductive explanation, the representation of action, planning. **Prerequisite: CSC384H1, CSC363H1/CSC365H1/CSC373H1/CSC375H1/CSC463H1**

### **CSC2506/412 Probabilistic Learning and Reasoning**

An introduction to probability as a means of representing and reasoning with uncertain knowledge. Qualitative and quantitative specification of probability distributions using probabilistic graphical models. Algorithms for inference and probabilistic reasoning with graphical models. Statistical approaches and algorithms for learning probability models from empirical data. Applications of these models in artificial intelligence and machine learning. **Prerequisite: CSC411H1**

### **CSC2512H Advanced Propositional Reasoning**

Many problems in Computer Science can be represented as instances of propositional reasoning problems. For example any problem in NP can be represented as a SAT problem (Boolean Satisfiability) since SAT is complete for the class NP. For many problems, their representation in SAT is very natural, and more importantly, can often be effectively solved by a general purpose SAT solver. This means that instead of developing and implementing a problem specific algorithm, we can often solve our problems by the much simpler device of encoding them into SAT and then using a SAT solver. Surprisingly, the SAT solver can often outperform problem specific algorithms. In this course you will be introduced to the basic algorithms that are used to SAT and other types of propositional reasoning problems. In addition, we will discuss various encoding techniques for translating various problems into SAT—the encoding used can have a dramatic effect on performance. Besides problems that can be encoded into SAT we often need to deal problems that require some form of optimization, or some form of quantification (e.g., to reason about two person games). Such problems can be encoded as MAXSAT problems, or QBF problems. We will also cover algorithms for solving these kinds of propositional problems. The aim of the course is to provide you with the background needed to exploit modern SAT–MAXSAT–QBF solvers in your own work. **Knowledge of the basics of propositional logic; be familiar and comfortable with programming, data structures and algorithms.**

### **CSC2517H Discrete Mathematical Models of Sentence Structure**

In this year's installment, we'll focus on graphical approaches to derivability in natural language syntax, including categorial proof nets, Petri nets and KLMST decompositions. An introduction to the principal mathematical models of sentence structure used in computational linguistics today. Topics include: string matching and similarity, string and tree transducers, extended context-free formalisms, tree-adjoining grammar, substructural logics, discourse representation calculi, typed feature structures, and topological models. Parsing, algorithmic complexity, algebraic properties, and formal equivalence will be discussed. **A basic knowledge of logic, formal language theory and graph theory is required. Some familiarity with syntactic theory will be helpful, but is not assumed.**

### **CSC2519H Natural Language Semantics**

No description available.

### **CSC2523H Object Modelling & Recognition: Shape Perception in Human and Computer Vision**

The shape of an object is a key feature for object categorization, and is invariant to changes in colour, texture, and illumination: a car with a leopard-skin paint job is not mistaken for a leopard, nor is an elephant painted with stripes mistaken for a zebra. How shape is perceived in the human visual system can both inform and inspire how shape can be represented and recovered in a computer vision system. This interdisciplinary graduate research seminar, suitable for students in human or computer vision, will examine shape perception from this dual perspective. Through a discussion of seminal research papers in both human and computer vision, the course will explore both classical and current models of shape perception, along with the challenges both communities face. Through individual course projects, students will be given the opportunity to explore a shape perception topic of their choice in greater depth. **Recommended Preparation: a prior course in human and/or computer vision would be helpful, but is not required.**

### **CSC2534H Decision Making Under Uncertainty**

A seminar on sequential decision making under uncertainty, covering a number of models and computational methods for their solution. Topics will include: basics of probabilistic inference and decision theory, fully- and-partially-observable Markov decision processes, reinforcement learning. The course may also cover some introductory topics in multiagent decision making (primarily game-theoretic models). An emphasis will be placed on AI-style representation (eg: Bayesian networks) and computational (eg: decompositions, function approximation) techniques.

### **CSC2542H Topics in Knowledge Representation & Reasoning**

This is a seminar-style topics course that explores recent advances in knowledge representation and automated reasoning. The topic for 2017-18 will be posted on the course web page. **Students taking this course should have the equivalent of an undergraduate introductory course in AI, such as CSC384, and reasonable competency as a programmer.**

### **CSC2547H Topics in Machine Learning: Learning Discrete Latent Structure**

In the last few years, new inference methods have allowed us to train learn generative latent-variable models. These models let us generate novel images and text, find meaningful latent representations of data, take advantage of large unlabeled datasets, and even let us do analogical reasoning automatically. However, most generative models such as GANs and variational autoencoders currently have pre-specified model structure, and represent data using fixed-dimensional continuous vectors. This seminar course will develop extensions to these approaches to learn model structure, and represent data using mixed discrete and continuous data structures such as lists of vectors, graphs, or even programs. The class will have a major project component.

**This course is designed to bring students to the current state of the art, so that ideally, their course projects can make a novel contribution. A previous course in machine learning such as CSC321, CSC411, CSC412, STA414, or ECE521 is strongly recommended. However, the only hard requirements are linear algebra, basic multivariate calculus, basics of working with probability, and basic programming skills.**

### **CSC2556H Algorithms for Collective Decision Making**

This course introduces students to the interdisciplinary field of computational social choice. Social choice theory studies collective decision making by groups of individuals. The central problem in the field is to aggregate heterogeneous preferences or opinions of individuals to arrive at consensus collective decisions. This includes fair division dilemmas such as a group of heirs fairly dividing an estate, voting dilemmas such as a group of friends selecting a restaurant for lunch, or matching dilemmas such as assigning students to schools. This field is gaining increasing practical relevance, and its algorithms are deployed in the real world, helping groups of people make decisions in their daily life. The course will cover the entire spectrum from theory to practice, and touch upon ongoing cutting-edge research.

### **CSC2601H Topics in Analysis and Computation in Discrete Models**

No Description

## METHODOLOGY 2 | Analysis and Computation in Continuous Models

### **CSC2206H System Modelling and Analysis**

The emphasis of the course is on models for systems with uncertainty. We study the properties of various models and discuss how they can be applied to analyze algorithm and system performance. This course is useful for students interested in the analysis of algorithms and systems with uncertainty that arise in computer systems, artificial intelligence, social networks, online markets etc. Concepts covered include Poisson, renewal, and Markov processes. **Solid knowledge of basic probability theory.**

### **CSC2302H Numerical Solution of Initial Value Problems for Ordinary Differential Equations**

Issues involved in the numerical solution of initial value problems in ODEs. Error propagation and the design of robust numerical methods. Methods designed for stiff and non-stiff problems will be reviewed and the significant difficulties arising in each area identified. State-of-the-art software will be surveyed and critically evaluated. Difficulties associated with implicit equations, algebraic constraints and delay terms will also be considered.

### **CSC2305/466 Numerical Methods for Optimization Problems**

Numerical methods for unconstrained optimization problems, in particular line search methods and trust region methods. Topics include steepest descent, Newton's method, quasi-Newton methods, conjugate gradient methods and techniques for large problems. This course will normally be offered every other year.

**Prerequisite: CSC336H1, MAT221H1/MAT223H1/MAT240H1, MAT235Y1/MAT237Y1/MAT257Y1**

### **CSC2321H Matrix Calculations**

Review the basic concepts in the numerical solution of linear systems, introduce state-of-the-art developments in numerical linear algebra/PDEs, develop and study efficient linear solvers, implement the above solvers as software, use existing software (routines and higher level environments), study the performance of methods and software. Topics among basic iterative methods, acceleration, preconditioning, multigrid, domain decomposition, fast Fourier transform linear solvers. **Prerequisite: calculus, numerical linear algebra, interpolation, some knowledge of PDEs, proficiency in a programming language, preferably MATLAB or FORTRAN**

### **CSC2322H Numerical Solution of Boundary Value Problems for Ordinary Differential Equations**

The topics include a survey of numerical methods for initial value problems, shooting methods, collocation methods and finite difference methods. A presentation of one or more particular areas of difficulty, such as singular perturbation problems, bifurcation problems or parameter fitting, will be included. Issues related to the implementation and effective use of current numerical software will be emphasized.

### **CSC2421H Topics in Algorithms: Fast Algorithms via Continuous Methods**

This course will focus on research topics in modern algorithm design, including tools, techniques, and their applications to the design of algorithms for fundamental problems in theoretical computer science. In this offering, the focus will be on methods from continuous optimization and analysis, and their applications to the design of fast algorithms for fundamental problems in theoretical computer science and numerical linear algebra. **Prerequisites/Recommended Preparation: Algorithms and Data Structures, Linear Algebra, Probability, preferably Vector Calculus.**

### **CSC2322H Numerical Solution of Boundary Value Problems for Ordinary Differential Equations**

The topics include a survey of numerical methods for initial value problems, shooting methods, collocation methods and finite difference methods. A presentation of one or more particular areas of difficulty, such as singular perturbation problems, bifurcation problems or parameter fitting, will be included. Issues related to the implementation and effective use of current numerical software will be emphasized.

### **CSC2421H Topics in Algorithms: Fast Algorithms via Continuous Methods**

This course will focus on research topics in modern algorithm design, including tools, techniques, and their applications to the design of algorithms for fundamental problems in theoretical computer science. In this offering, the focus will be on methods from continuous optimization and analysis, and their applications to the design of fast algorithms for fundamental problems in theoretical computer science and numerical linear algebra. **Prerequisites/Recommended Preparation: Algorithms and Data Structures, Linear Algebra, Probability, preferably Vector Calculus.**

### **CSC2429H Topics in the Theory of Computation: Proof Complexity, Semi-definite Hierarchies and Extended Formulations**

Proof complexity, the study of the lengths of proofs in propositional logic, is an area of study that is fundamentally connected both to major open questions of computational complexity theory and to practical properties of automated theorem provers and reasoning systems. In the last twenty years, there have been a number of significant advances in proof complexity. Moreover, new connections between proof complexity and circuit complexity have been uncovered, and the interplay between these two areas has become quite rich. Concepts and techniques in proof complexity have borrowed from and contributed to several other areas of computer science, including: cryptography, automated reasoning and AI, complexity of search classes within NP, the analysis of heuristics and algorithms for NP-hard problems, and limitations of LP and SDP algorithms. In this course, we will cover the fundamentals of proof complexity, and some of the connections with these other areas. **Prerequisite: CSC2404 (438) or equivalent course in logic**

### **CSC2503H Foundations of Computer Vision**

Introduction to vision, visual processes, and image understanding. Scene lighting and reflectance models. Camera system geometry and image acquisition. The robust estimation of edges, lines, and regions. Perceptual organization. View-based image models. Image matching and the estimation of motion in image sequences. Multi-view geometry. Projective and metric reconstructions. Markov random fields. Object recognition. The main course web site has a calendar that specifies when specific topics will be covered. **Prerequisite: MAT235 and CSC324 or equivalents. Comfortable with elementary probability and statistics, linear algebra, elementary geometry, and vector calculus (including partial differentiation). It is also assumed that the student is comfortable programming, and will be expected to rapidly learn to use Matlab (or analogous python-based tools).**

### **CSC2504/418 Computer Graphics**

Identification and characterization of the objects manipulated in computer graphics, the operations possible on these objects, efficient algorithms to perform these operations, and interfaces to transform one type of object to another. Display devices, display data structures and procedures, graphical input, object modelling, transformations, illumination models, primary and secondary light effects; graphics packages and systems. Students, individually or in teams, implement graphical algorithms or entire graphics systems. **Prerequisite: CSC336H1/CSC350H1/CSC351H1/CSC373H1, (MAT135H1, MAT136H1)/MAT137Y1/MAT157Y1, CSC209H1/proficiency in C or C++; Prerequisite for Engineering students only: ECE345H1 or ECE352H1**

### **CSC2511/401 Natural Language Computing**

Introduction to techniques involving natural language and speech in applications such as information retrieval, extraction, and filtering; intelligent Web searching; spelling and grammar checking; speech recognition and synthesis; and multi-lingual systems including machine translation. N-grams, POS-tagging, semantic distance metrics, indexing, on-line lexicons and thesauri, markup languages, collections of on-line documents, corpus analysis. PERL and other software. **Prerequisite: CSC207H1/CSC209H1; STA247H1/STA255H1/STA257H1**

### **CSC2515/411 Machine Learning and Data Mining**

An introduction to methods for automated learning of relationships on the basis of empirical data. Classification and regression using nearest neighbour methods, decision trees, linear models, and neural networks. Clustering algorithms. Problems of overfitting and of assessing accuracy. Problems with handling large databases. Emphasis on statistical approaches. **Prerequisite: CSC263H1/CSC265H1, MAT(135H1,136H1)/MAT137Y1/MAT137Y1/MAT157Y1, STA247H1/STA255H1/STA257H1**

### **CSC2521H Topics in Computer Graphics: Computational Design & Fabrication**

This course is designed to serve four purposes: (1) to teach students fundamental representations of 3D shapes such as mesh-based and voxel-based parameterizations; (2) to introduce students to the basic tools of computational design including numerical simulation and optimization; (3) to introduce students to common fabrication hardware such as laser cutters and 3D printers; and (4) to read, understand and present state of the art research papers in computational design and fabrication.

### **CSC2521H Topics in Computer Graphics: Geometry Processing**

The class is aimed at preparing students for working with geometric data via understanding fundamental theoretical concepts. Extending traditional signal processing, geometry processing interprets three-dimensional curves and surfaces as signals. Just as audio and image signal data can be filtered, denoised and decomposed spectrally, so can the geometry of a three-dimensional curve or surface. In this course, we study the algorithms and mathematics behind fundamental operations for interpreting and manipulating geometric data. These essential tools enable: geometric modeling for computer aided design, life-like animations for computer graphics, reliable physical simulations, and robust scene representations for computer vision. Topics include: discrete curves and surfaces, curvature computation, surface reconstruction from point clouds, surface smoothing and denoising, mesh simplification, parameterization, symmetry detection, shape deformation and animation. **Recommended preparation: Students should have a background in Linear Algebra and Computer Programming. Previous experience with Numerical Methods, Differential Equations, and Differential Geometry is appreciated but not required.**

### **CSC2522H Advanced Image Synthesis**

This course will focus on advanced algorithms and architecture for image synthesis. Algorithms for higher level modeling primitives, shading models, antialiasing, and stochastic modeling will be studied. Designs for high performance display systems, particularly raster displays, will also be studied. Various high-performance raster systems will be used as case studies, and the course will include implementation projects on one or more of them.

### **CSC2529H Character Animation**

The primary focus of this course is on kinematic and dynamic techniques for character animation. Topics include physical modeling and simulation, motion planning, control and learning algorithms, locomotion, motion trajectory optimization, scripting languages, motion capture, and motion deiting. Students will implement algorithms and interactive animation tools and then use these to produce motion for animations.



### **CSC2530H Computer Vision for Advanced Digital Photography**

The ubiquity of digital cameras and the internet, coupled with advances in computer vision and graphics, are bringing about a qualitative change in the way photographs can be captured, manipulated, and organized. Indeed, even our traditional concepts of a “camera” and a “photo” are now being challenged by newer, more flexible alternatives in which computation plays a much more central role. In this seminar course we will review recent work that tightly integrates computation, sensing, optics, and/or the internet in order to enhance the photography experience. The course will cover four general topics: (1) internet photography; (2) photo reconstruction and restoration; (3) mathematical models of light and light transport; and (4) advanced photography systems & techniques.

### **CSC2535H Advanced Machine Learning**

No description available. **CSC2515: Introduction to Machine Learning (or equivalent -- ask instructor).**

### **CSC2539H Topics in Computer Vision**

Selected research topics in Computer Vision. Specific topics will vary.

### **CSC2541H Topics in Machine Learning: Scalable and Flexible Models of Uncertainty**

One of the biggest challenges in machine learning is for algorithms to know what they don't know. This course will cover modern Bayesian methods in machine learning and ways to achieve both scalability and flexibility. Methods will include Markov chain Monte Carlo, variational inference, natural gradient, Gaussian processes, and Bayesian neural networks. We will discuss the use of uncertainty in guiding exploration and in defending against adversarially crafted inputs.

### **CSC2545H Kernel Methods & Support Vector Machines**

The introduction of Support Vector Machines (SVMs) in the 1990s led to an explosion of applications and deepening theoretical analysis that have established SVMs as one of the standard tools for machine learning and data mining. This course provides a comprehensive introduction to SVMs and other kernel methods, including theory, algorithms and applications. Topics covered will be selected from the following: support vector classification and regression; novelty detection and feature extraction; non-linear dimensionality reduction; reproducing kernel maps; regularization; statistical learning theory and robust estimation; convex optimization and implementation; kernel design and applications. There will be 2 hours of lectures per week and 3 homework assignments. The assignments will be a mix of theory and programming.

**Prerequisites: Linear algebra, calculus (including partial derivatives), basic probability, and a willingness to program in Matlab and/or Python. A previous course in machine learning is not required but would certainly help. Mathematical maturity will be assumed.**

### **CSC2548H Machine Learning in Computer Vision**

In recent years, Deep Learning has become a dominant Machine Learning tool for a wide variety of domains. One of its biggest successes has been in Computer Vision where the performance in problems such as object and action recognition has been improved dramatically. In this course, we will be reading up on various Computer Vision problems, The state-of-the-art techniques involving different neural architectures and brainstorming about promising new directions. **Prerequisites/Recommended Preparation: CSC411 or equivalent introductory course in machine learning.**

### **CSC2602H Topics in Analysis and Computation in Continuous Models: Computational Models of Climate Change**

No description available.

## METHODOLOGY 3 | Building Software and Hardware Artifacts

### **CSC2107/488 Compilers and Interpreters**

The structure of compilers, Programming language processing. Scanning based on regular expressions, Parsing using context free grammars, Semantic analysis (type and usage checking), Compiler dictionaries and tables. Runtime organization and storage allocation, code generation, optimization. Use of modern compiler building tools. Course project involves building a complete compiler. **Prerequisite: CSC258H1, CSC324H1, CSC263H1/CSC265H1**

### **CSC2203H Packet Switch and Network Architectures**

This is an MSc/PhD level course on high-performance packet switching computer networks. The course introduces the theory and practice of designing packet switches, such as Internet routers, Ethernet switches, and ATM switches. It consists of two parts which will be presented in an interleaved fashion. The first part will develop basic tools from queueing theory, stochastic analysis, and algorithms. The second part of the course will focus on packet switch architectures: the evolution of switches and routers, and practical issues in this area. **Basic undergraduate courses in algorithms, networking, and in probability theory are strongly recommended.**

### **CSC2208/496 Advanced Operating Systems**

Principles of operating systems. The operating system as a control program and a resource allocator. Comparison of OS types: monolithic, microkernel, etc., as well as virtualization methods. Covers advanced topics in synchronization, mutual exclusion, and deadlock. Additional topics include performance analysis, process scheduling in a multiprocessor context, advanced memory management, file systems design and crash recovery, and protection. Fault tolerance will also be covered in the context of a distributed system. **Requires advanced C systems programming skills and strong intro OS knowledge.**

### **CSC2209/458 Computer Networking Systems**

Computer networks with an emphasis on network programming and applications. An overview of networking basics: layering, naming, and addressing, packet switching fundamentals, socket programming, protocols, congestion control, routing, network security, wireless networks, multimedia, web 2.0, software-defined networking, and online social networks. **Prerequisite: CSC209H1, CSC258H1, CSC263H1/CSC265H1, STA247H1/STA255H1/STA257H1/EC0227Y1**

### **CSC2227H Topics in the Design and Implementation of Operating Systems**

Seminar and discussion of a number of topics in operating system design and implementation, based on a collection of papers as class reading. Emphasis on multiprocessor and distributed operating systems. Case studies of contemporary operating systems. An independent course project and an oral student presentation is required. **Prerequisites: introductory Operating Systems (CSC369 or equivalent), CSC2208/CSC469 recommended.**

### **CSC2228H Advanced Topics in Mobile and Pervasive Computing: Cloud Computing**

The emergence of mobile and cloud computing are two of the most important technological developments of the last few years. By shifting the hardware and staffing costs of managing computational infrastructure to third parties such as Google, Microsoft, or Amazon, cloud computing has made it possible for small organizations and individuals to deploy world-scale services; all they need to pay is the marginal cost of actual resource usage. At the same time, the deployment of 4G networks, the rapid adoption of feature-rich smartphones, and the growing integration of computation into consumer products such as cars and home appliances, have brought mobile and pervasive computing into the mainstream. This course will provide an overview of research areas in these two fields and their intersection. **Prerequisite: Permission of the instructor**

### **CSC2229H Topics in Computer Networks: Software Defined Networking**

This is a MSc/PhD level course introducing Software-Defined Networking (SDN). SDN is an emerging paradigm in computer networks that facilitates change and innovation in infrastructure and network applications by pushing network control functionalities to a logically centralized controller. We will cover the fundamentals of SDN, and focus on various research problems in the context of SDN such as: controller and switch design;

network architecture; reliability, efficiency and scalability; network programming, correctness, and debugging; network security; new abstractions, and state management; network services, network function virtualization; and network optimization. **Basic undergraduate courses in algorithms, networking, and in probability theory are strongly recommended.**

### **CSC2231H Topics in Computer Systems: Parallel Computer Architecture and Programming**

The goal of this course is to build a strong understanding of the fundamentals of the architecture of parallel computers and efficient programming for them. We will examine how architectures are designed to exploit and extract different types of parallelism. The focus will be on fundamentals, tradeoffs in parallel architecture design, and cutting-edge research. Architectures studied may include parallel microprocessors, GPUs and FPGAs.

### **CSC2232H Topics in Computer System Performance and Reliability**

Selected research topics in the performance and/or reliability of computer systems. Specific topics will vary.

### **CSC2233H Topics in Storage Systems**

Over 90% of all new information produced in the world is being stored on digital storage systems, including the crown jewels jewels of enterprises (ranging from Microsoft's source code to the sales database of every e-commerce site) and personal records, such as photos, correspondence, and movies. Storage systems often dominate the performance of a system, and their components' inner workings require amazing feats of engineering. The success of many young and cool companies, such as Google, Facebook and Youtube, is derived from their ability to efficiently and reliably store large amounts of data. The exponentially growing need for data capacity and access speed is making storage systems one of the fields with the most robust demand for bright people, both in industry and in academia. In this class, you will learn about the latest and greatest in storage systems, studying novel ideas from academia and learning about the best ""real-world"" systems. You will also perform a cutting-edge mini-research project, where you design, build, and/or evaluate a system.

### **CSC2299H Special Reading Course in Computer Systems**

No description available.

### **CSC2508H Advanced Database Management Systems**

Over the last few years a plethora of new data management systems and architectures have become mainstream. Such systems have vastly diverse application focus, architectures and differ significantly from traditional transactional database management systems and data warehouses. The goal of this course is to explore these systems, broadly characterized as noSQL/newSQL systems and understand their strengths and limitations. We will also explore new trends in data management fueled by application needs, such as support for advanced analytics, stream processing systems and main memory data processing.

### **CSC2525H Research Topics in Database Management**

Advanced graduate reading course in data management research. Past topics have included Data Mining, Data Integration, and Data Management in Peer-to-Peer Systems. See course web page for information on current topic. This is an advanced graduate seminar. Students should be ready to undertake novel research in data management.

### **CSC2531H Advanced Topics in Database Management Systems**

This course is a reading seminar with a focus on the intersection of database engines and modern hardware. Topics include non-traditional database engine designs, cache- and multicore- aware algorithms, hardware acceleration for database operations (e.g. FPGA and GPU), as well as flash and other storage new technologies.

**Prerequisite: Graduate level data management course.**

### **CSC2603H Topics in Building Software and Hardware Artifacts**

No description available.

## METHODOLOGY 4 | Human-Centered and Interdisciplinary Computing

### **CSC2130H Empirical Research Methods in Software Engineering**

This course will explore the role of empiricism in computer science research, and will prepare students for advanced research by examining how to plan, conduct and report on empirical investigations. The course will cover techniques applicable to each of the steps of a research project, including formulating research questions, theory building, data analysis (using both qualitative and quantitative methods), building evidence, assessing validity, and publishing. The course will cover the principal research methods used to study human interaction with computer technology: controlled experiment, case studies, surveys, archival analysis, action research and ethnographies. We will relate these methods to relevant metatheories in the philosophy and sociology of science, and critically review published examples of work that used each of the principle methods in computer science.

#### **No prerequisite**

### **CSC2417H Algorithms for Genome Sequence Analysis**

In this course we will explore the computational problems that have emerged from genome sequencing. The topics will include the string algorithms used for the classic sequence alignment and genome assembly problems, algorithms for comparing genomes, sequence classification using probabilistic models and algorithms for analyzing gene expression data. Throughout the course a special emphasis will be made on efficient algorithms designed to meet the challenges of rapidly growing data sets in the current era of high-throughput genome sequencing. The course is intended for Computer Science graduate students, and all of the required biology will be explained in the class. Students in biological and related sciences with a strong computational background are encouraged to participate.

### **CSC2431H Topics in Computational Molecular Biology: Computational Methods in Medicine**

This course will introduce students to the variety of ways that computational methods are utilized in modern medicine. The topics will include the use of probabilistic and deterministic techniques in the analysis of genomic data, machine learning and NLP to analyze medical texts, HCI considerations for medical software, as well as other topics. While we will not cover any particular CS research area in depth, the students are expected to have a broad computational background to be able to read research literature in a variety of subject areas, as well as the ability to do independent exploration of topics in medicine.

### **CSC2501/485 Computational Linguistics**

Computational linguistics and the processing of language by computer. Topics include: context-free grammars; chart parsing, statistical parsing; semantics and semantic interpretation; ambiguity resolution techniques; reference resolution. Emphasis on statistical learning methods for lexical, syntactic, and semantic knowledge.

**Prerequisite: STA247H1/STA255H1/STA257H1 or familiarity with basic probability theory, including Bayes's theorem; CSC207H1/CSC209H1 or proficiency in Python and software development.**

### **CSC2514/428 Human-Computer Interaction**

Understanding human behaviour as it applies to user interfaces: work activity analysis, observational techniques, questionnaire administration, and unobtrusive measures. Operating parameters of the human cognitive system, task analysis and cognitive modelling techniques and their application to designing interfaces. Interface representations and prototyping tools. Cognitive walkthroughs, usability studies and verbal protocol analysis. Case studies of specific user interfaces. **Prerequisite: CSC318H1; STA247H1/STA255H1/STA257H1; CSC209H1/proficiency C++ or Java Recommended Preparation: A course in PSY; CSC209H1; (STA248H1/STA250H1/STA261H1)/(PSY201H1)**

### **CSC2518H Spoken Language Processing**

This is a graduate course broadly on topics of speech processing by machine including digital signal processing, automatic speech recognition, and speech synthesis. The theme this year is **Speech in healthcare and assistive technologies** which will include automatic dictation of speech for medical records, analysis of speech in language pathologies (e.g., in cerebral palsy, Parkinson's disease, and Alzheimer's disease), and assistive technologies such as text-to-speech (with and without brain-computer interfaces) for people with limited speech ability.

### **CSC2524H Topics in Interactive Computing: Information Visualization**

In this course we will study techniques and algorithms for creating effective visualizations based on principles from graphic design, visual art, perceptual psychology, and cognitive science. The course is targeted both towards students interested in using visualization in their own work, as well as students interested in building better visualization tools and systems. Students will conduct weekly readings, with two student-led presentations and discussions each week. A final project will make up most of the grade, while student presentation skills and participation in class and in readings will also be emphasized.

### **CSC2526H Topics in Ubiquitous Computing: Assistive Technology**

Lecture-style course focusing on ubiquitous computing concepts and how they could be applied to increase, maintain, or improve the functional capabilities of persons with disabilities. Students work individually or in pairs on design projects that could be potentially useful to specific user populations.

### **CSC2526H Topics in Ubiquitous Computing: Inclusive Computing**

The benefits of computing have barely reached billions of underprivileged lives around the world that are fighting with extreme poverty, illiteracy, violence, disasters, and various other exploitations and marginalizations. In this course, we will study the limits and politics of and around computing, and design technologies to address different challenges with global impact. The students will learn various social and political theories to critically analyze a technology, and design novel technologies through alternative, critical, persuasive, ludic, speculative, and other design strategies. There will be weekly readings, design sessions, discussions, two small projects, and one final project.

### **CSC2527/454 The Business of Software**

The software and internet industries; principles of operation for successful software enterprises; innovation and entrepreneurship; software business definition and planning; business models, market and product planning; product development, marketing, sales, and support; financial management and financing of high-technology ventures; management, leadership, and partnerships. Students will all write business plans in teams.

### **CSC2528H Advanced Computational Linguistics**

This is an advanced seminar with a significant term (research) paper required of all students and several presentations during the term. Each term, four current research topics in computational linguistics will be chosen for deep investigation, exploring the research and secondary literature. **Prerequisites: A prior course in computational linguistics or natural language processing (such as CSC2511 or CSC2501), or permission of the instructor.**

### **CSC2546H Computational Neuroscience**

This course offers an introduction to current topics and methods in computational neuroscience. Theoretical analysis and computational methods are tools for characterizing what nervous systems do and determining how and why they do it. Neuroscience encompasses approaches ranging from molecular studies to human psychophysics. Computational neuroscience encourages cross-talk between the many levels of this broad field by constructing compact descriptions of what has been learned at various levels, building bridges between these descriptions, and identifying potential unifying concepts and principles. This course will cover the basic methods used for these purposes and discuss examples in which computational approaches have yielded insight into brain function.

### **CSC2552H Topics in Computational Social Science**

Computational social science, the emerging discipline in which computational tools and thinking are applied to social science questions, is one of the most rapidly growing interdisciplinary fields in computer science. In this course, the most recent developments in this quickly-changing field will be discussed. We will cover information diffusion, big data ethics and fairness, inherent unpredictability of cultural and computational phenomena, incentive design for online systems, and other topics. **Prerequisites/Recommended Preparation: A senior course on statistics, a senior course on algorithms.**

## **CSC2604H Topics in Human Centered and Interdisciplinary Computing: Computational Models of Semantic Change**

Words are core components of language, but their meanings tend to vary over time, e.g., *face* ('body part' --> 'expression'), *grasp* ('action' --> 'understanding'), *gay* ('happy' --> 'homosexual'). Meaning changes like these allow the lexicon to express emerging ideas with existing words, but they present potential challenges for computers to learn word meanings and their extensions. This seminar course explores theoretical and computational approaches to semantic change. Main topics include word representation, detection and modeling of meaning change, semantic chaining, and metaphor. **Proficiency in Python is assumed. Some knowledge of natural language processing, probability, and information theory will be helpful.**

## **CSC2720H Systems Thinking for Global Problems**

This course offers an introduction to systems thinking and its relationship to complexity science, cybernetics, and the modelling of complex non-linear dynamical systems. The course uses a series of interactive classroom games to illustrate key concepts in systems thinking, including stock and flow analysis, emergent behaviour, feedback loops and tipping points. These are then used as building blocks to explore how we think about and attempt to solve global challenges of the twenty-first century, including climate change, conflict, democracy, health & wellbeing, and food security. The course is suitable for graduate students from any discipline, and makes no assumptions about prior knowledge of computational techniques.

## **RESEARCH AREAS FOR THESE COURSES MAY VARY DEPENDING ON CONTENT**

### **CSC2601H**

*No description available*

### **CSC2602H Topics in Analysis and Computation in Continuous Models: Computational Models of Climate Change**

*No description available*

### **CSC2603H**

*No description available*

### **CSC2604H Topics in Human-Centred & Interdisciplinary Computing**

*No description available*

## **NO BREADTH COURSES**

### **CSC2699H Special Reading Course in Computer Science: Academic Leadership in Computer Science**

*No description available*

### **CSC2600 Capstone Design Project**