

School of Computer Science

Herzberg Bldg. 542

The School

Director of the School:

J.R. Pugh

Supervisor of Graduate Studies:

S. Dandamudi

The School of Computer Science offers degrees leading to a Master of Computer Science or a Ph.D. in Computer Science through the Ottawa-Carleton Institute for Computer Science. The institute is jointly administered by the school and the Department of Computer Science at the University of Ottawa. For further information, including admission and program requirements, see page 174.

A program leading to the M.Sc. in Information Systems Science is offered in cooperation with the Department of Mathematics and Statistics and the Department of Systems and Computer Engineering, see page 192.

The research expertise of the school faculty is concentrated in the following areas:

- *Algorithms and Complexity*

Computational geometry and algebra, combinatorial optimization, distributed and parallel algorithms, multi-dimensional data structures, stochastic automata, graph theory, partial orders.

- *Intelligent Systems*

Expert systems, knowledge acquisition tools, knowledge based assistants, connectionism and neural networks, natural language understanding, learning and adaptability, robotics, pattern recognition.

- *Object-Oriented Systems*

Visual programming, filing systems, databases, user interfaces, simulation, animation, software engineering, office automation.

- *Distributed Systems*

Operating systems, databases, systolic architectures, tools for performance studies, distributed programming languages, parallel computing, communication complexity, networks.

In addition to its undergraduate laboratories the School maintains three research laboratories, containing PC-AT clones, MacII's, Tektronix and SUN workstations, and laser printers all integrated via a department and campus area network.

Graduate Courses*

The complete list of courses available through the Ottawa-Carleton Institute for Computer Science is given on page 174. The following courses are offered by the School of Computer Science.

- Computer Science 95.501F1 (CSI5113)

Foundations of Programming Languages

This course will study current topics in the theory and practice of programming language design and implementation. Different styles of languages: imperative, applicative, logic, constraints, object-centered, dataflow, production systems. Abstraction mechanisms; primitives; extensibility; procedural versus declarative semantics; interpretation; compilation; program transformation.

Prerequisite: Computer Science 95.207 or the equivalent.

- Computer Science 95.502W1 (CSI5119)

User Interface Facilities

This project oriented course is concerned with the concepts, methodologies and algorithms for the specification, design and implementation of visual User Interface Facilities (UIF). The principal focus is on the software engineering of user interfaces. UIF applications in computer aided instruction, computer-aided design and visual programming are used to illustrate both general and special purpose user interfaces. Current commercial and research approaches are studied from the perspective of the user, the application designer and the systems programmer. The alternative programming metaphors of control flow, data flow, objects and constraints are introduced and their importance is discussed in the context of integrated user interface.

Prerequisite: Computer Science 95.501 or equivalent.

*F,W,S indicates term of offering. Courses offered in the fall and winter will be followed by T.

The number following the letter indicates the credit weight of the course: 1 denotes a half-course credit, 2 denotes a full-course credit, etc.

- Computer Science 95.503F1 (CSI5308)
Principles of Distributed Computing
Formal models; semantics of distributed computations; theoretical issues in design of distributed algorithms; computational complexity; reducibility and equivalence of distributed problems. Related topics: systolic systems and computations, oligarchical systems and control mechanisms.
Prerequisite: Permission of the school.

- Computer Science 95.504W1 (CSI5305)
Topics in Arithmetic Complexity
Most scientific calculations rest on the basic arithmetic operations carried out on numbers, polynomials, and matrices. The course begins by studying the complexity of these operations. It proceeds to examine the related problem of finding the factors of an integer or polynomial, and it discusses the applications of this problem to cryptography and coding theory. The course concludes with a selection of other fundamental problems, such as polynomial evaluation, and the exploitation of parallel hardware.
Prerequisite: Computer Science 95.484 or equivalent.

- Computer Science 95.505F1 (CSI5390)
Automata Models of Learning Systems
This course will introduce students to computerized adaptive learning. Learning models in mathematical psychology will be discussed. Mathematical tools such as Markov chains and the solution of difference equations will be reviewed. The heart of the course will involve deterministic and stochastic automata, operation in random environments, norms of learning, linear and nonlinear learning schemes, convergence problems, and discretized automata with ergodic and non-ergodic properties. Applications of learning automata in file allocation, game playing, path finding, optimization and decision making will be discussed.
Prerequisite: Mathematics 70.260 or 70.350, or Engineering 94.553, or equivalent.

- Computer Science 95.506W1 (CSI5306)
Natural Language Understanding
This course will introduce the students to current research in natural language processing. The emphasis of the course will be on semantic and pragmatic rather than syntactic issues and on analyzing connected discourse rather than single sentences. Several existing natural language analyzers and their applications to text analysis, CAI, knowledge acquisition, database retrieval and intelligent assistants will be described in detail. Topics will include: meaning representation; representation of pragmatic information and speech act theory; flexible parsing; determination of focus and reference; task-oriented dialog systems; dynamic

memory issues. Students will be required to implement a prototype natural language analyzer.
Prerequisite: Computer Science 95.407 or 95.501, or equivalent.

- Computer Science 94/95.507F1 (CSI5307)
Expert Systems
This course will include: survey of some landmark expert systems; types of architecture and knowledge representation; inferencing techniques; approximate reasoning; truth maintenance; explanation facilities; knowledge acquisition. A project to implement a small expert system will be assigned.
Prerequisite: Computer Science 95.407 or 95.501 or permission of the school.

- Computer Science 95.508W1 (CSI5164)
Computational Geometry
This course will study the design and analysis of algorithms for solving geometrical problems. These algorithms have applications in such areas as computer graphics, pattern recognition and robotics. Topics will include: visibility problems, hidden line removal, classes of polygons, testing polygons for structural properties, convex hull problems, movability of objects through a set of obstacles, point inclusion in polygons, decomposition of objects into "meaningful" components, triangulation and guard problems.
Prerequisite: Computer Science 95.384 or equivalent.

- Computer Science 95.509F1 (CSI5141)
Associative Data Structures and Advanced Databases
This course addresses concepts and advanced topics in the design, implementation and analysis of physical storage schemes with emphasis on their application to specialized database and information retrieval systems. The topics covered include associative searching techniques; multidimensional storage structures; design and analysis of algorithms for spatial data modeling; formulation and optimization of database queries; parallel hardware and distributed approaches for physical data organization and information retrieval. Some case studies of their applications to geographic information systems, object bases and multimedia databases are also discussed.
Prerequisites: Computer Science 95.305 and 95.384, or equivalent.

- Computer Science 95.510W1 (CSI5180)

Topics in Artificial Intelligence

A programming oriented introduction to selected topics in Artificial Intelligence (A.I.). Topics for consideration include: A.I. programming techniques, pattern matching systems, natural language systems, rule based systems, constraint systems, learning systems, and cognitive systems. Assignments will be both (a) programming oriented requiring implementations and/or extensions of prototypes in Lisp and/or Prolog and (b) research oriented requiring readings of special topics in current A.I. journals.

Prerequisite: Computer Science 95.501 or equivalent.

- Computer Science 95.511F1 (CSI5311)

Distributed Databases and Transaction

Processing Systems

The course addresses the principles involved in the design and implementation of distributed databases and transaction processing systems. Topics covered include: distributed system architectures and models; distributed file systems; atomicity of distributed transaction; design of stable storage; synchronization mechanisms; concurrency control algorithms in distributed systems. Further topics cover issues of multiple copy updates; applications and protocols; error recovery; reliability and protection in distributed systems.

Prerequisites: Computer Science 95.305, 5.401, and 95.403 or equivalent.

- Computer Science 95.512W1 (CSI5312)

Distributed Operating Systems

A course emphasizing the design issues of advanced multiprocessor distributed operating systems: multiprocessor system architectures; the process model; the object model; synchronization and message passing primitives; memory architectures and management; distributed filesystems; protection and security; distributed concurrency control; deadlock and recovery; remote tasking; dynamic reconfiguration; performance measurement, modeling, and system tuning.

Prerequisite: Computer Science 95.300 or equivalent.

- Computer Science 95.513F1 (CSI5313)

Cryptography

Classical cryptosystems: substitution ciphers, homophonic ciphers, product ciphers, DES. Public key schemes: RSA, Knapsack codes. Digital signatures, fair communication protocols, key management.

Prerequisite: Permission of the school.

- Computer Science 95.514W1 (CSI5314)

Object-Oriented Systems

An examination of advanced topics and current re-

search in object-oriented programming systems, languages and applications. Potential topics include: object-oriented design; comparative evaluation of object-oriented systems; compiled versus interpretive systems; manifest versus latent types; prototypes versus classes; inheritance mechanisms; persistent objects; concurrency; distributed objects; reflective architectures.

Prerequisite: Computer Science 95.501 or equivalent.

- Computer Science 95.515W1 (CSI5132)

Parallel Processing Systems

The aim of this course is to provide an introduction to the issues involved in designing and using parallel processing systems. Topics will be selected from the following: taxonomy and applications of parallel systems; SIMD systems; multiprocessor systems; multicomputer systems; computation versus communication issues in parallel processing; scheduling parallel systems; spinning versus blocking; interconnection networks; hot-spot contention.

Prerequisite: Permission of the school.

- Computer Science 95.516W1 (CSI5123)

Languages for Parallel Computing

This course will survey the major language paradigms for parallel computing: sequential imperative languages (i.e. automatically parallelizing conventional sequential languages), parallel imperative languages, functional languages (reduction and dataflow), communicating sequential processes (CSP), object and message-passing based languages, logic languages, and massive data-level parallelism. The course will cover the fundamental language issues in parallel computing systems, such as parallelism detection, determinism, data partitioning, task scheduling, task granularity, synchronization methods, resource management, and debugging for each paradigm. The course will study actual languages and systems, both past and present, and cover implementation issues as time permits.

Prerequisite: Computer Science 95.501.

- Computer Science 95.520 (CSI5182)

Cerebral Computations

Cerebral computation is concerned with computational models of the human brain. It is a programming course devoted to the design and implementation of aspects of the brain viewed as a cerebral computer. It includes such topics as neural models, neural architectures, pre-attentional vision processing, audio and touch processing, hand-eye coordination, mental imagery, map modeling, world modeling, attentional mechanisms, associative mechanisms, affect pro-

cessing, motor control, high-level planning, and models of simpler organisms. A fundamental aim is the investigation of mechanisms that exhibit plasticity and adaptability; i.e., the ability to change and improve over time.

Prerequisite: Permission of the school.

- Computer Science 95.522 (CSI5172)

Network Reliability

This course adopts a graph theoretic model of a communications network and addresses the problem of assessing the network's reliability when components are prone to failure. Topics include: graph theoretic models of computer networks, the complexity of computing reliability, combinatorial algorithms for bounding the reliability, Monte Carlo methods, and applications such as optimal facility location in unreliable networks. This course will draw on results from graph theory, complexity theory, combinatorics, and statistics.

Prerequisite: Permission of the school.

- Computer Science 95.573F1 (CSI5163)

Algorithm Analysis and Design

Topics of current interest in the analysis and design of sequential and parallel algorithms for non-numerical, algebraic and graph computations. Lower bounds on efficiency of algorithms. Complexity classes.

Prerequisite: Permission of the school.

- Computer Science 95.574W1 (CSI5131)

Parallel Algorithms and their VLSI

Implementation

Introduction: models of computation, levels of parallelism, performance measures for parallel algorithms, need for parallel algorithms. Parallel algorithms: techniques in matrix multiplication, solution of linear equations, transforms and differential equations, systolic arrays for the implementation of parallel algorithms in the areas of matrix arithmetic, transforms and relational database operations. VLSI implementations: VLSI and parallel computing structures, mapping of high-level computations into VLSI structures.

Prerequisite: Computer Science 95.484 or equivalent.

- Computer Science 95.582W1

Introduction to Information and Systems Science

An introduction to the process of applying computers in problem solving. Emphasis is placed on the design and analysis of efficient computer algorithms for large, complex problems. Applications in a number of areas are presented: data manipulation, databases, computer networks, queuing systems, optimization.

(Also listed as Mathematics 70.582, Engineering 94.582, Information and Systems Science 93.582)

- Computer Science 95/70.587F1 (CSI5104)

Formal Language and Syntax Analysis

Computability, unsolvable and NP-hard problems. Formal languages, classes of languages, automata. Principles of compiler design, syntax analysis, parsing (top-down, bottom-up), ambiguity, operator precedence, automatic construction of efficient parsers, LR, LR(O), LR(k), SLR, LL(k); syntax directed translation.

Prerequisite: Computer Science 95.302 or Mathematics 70.485 or 70.565, or equivalent.

- Computer Science 95.590F1, W1, S1 (CSI5140)

Selected Topics in Computer Science

Selected topics, not covered by other graduate courses, will be offered. Details will be available at the time of registration.

- Computer Science 95.591F1, W1, S1 (CSI5901)

Directed Studies (M.C.S.)

A course of independent study under the supervision of a member of the School of Computer Science.

- Computer Science 95.592F1, W1, S1 (CSI5900)

Graduate Project (M.C.S.)

- Computer Science 70/94/95.595F, W, S (CSI7999)

M.C.S. Thesis

- Computer Science 70/94/95.598F, W, S

M.Sc. Thesis in Information and Systems Science

- Computer Science 95.610F1 (CSI7131)

Advanced Parallel and Systolic Algorithms

This course is a continuation of 95.574.

Prerequisite: Computer Science 95.574.

- Computer Science 95.661F1, W1, S1 (CSI7160)

Advanced Topics in the Theory of Computing

- Computer Science 95.662F1, W1, S1 (CSI7170)

Advanced Topics in Distributed Computing

- Computer Science 95.663F1, W1, S1 (CSI7161)

Advanced Topics in Programming Systems and Languages

- Computer Science 95.664F1, W1, S1 (CSI7162)

Advanced Topics in Computer Applications

- Computer Science 95.665F1, W1, S1 (CSI7163)
Advanced Topics in Computer Systems
- Computer Science 95.691F1, W1, S1 (CSI7901)
Directed Studies (Ph.D.)
- Computer Science 95.692F1, W1, S1 (CSI7900)
Graduate Project (Ph.D.)
- Computer Science 95.699F, W, S
(CSI9999)
Ph.D. Thesis