The role of brewer’s yeast in the brewing process, particularly its fermentation of sugars to produce alcohol, is fairly well known. However, yeast is also responsible for producing dozens, if not hundreds, of chemical compounds as it metabolizes the sugars, amino acids, and other components during fermentation. Many of these compounds contribute significantly to the flavour and aroma of beer. Other microorganisms, such as wild yeast and bacteria, are also potential contributors to the complex chemistry and biochemistry that occurs in the fermenter; sometimes to the benefit of the beer but more often to its detriment. This course will look at all of the microorganisms that are commonly found in the brewery and provide a detailed description of their chemistry and thus their impact on beer flavour and aroma.

Prerequisites: CHM 211 and BIO 201

As a food product, beer is rigorously controlled at both the federal and provincial levels of government. Part of this process is ensuring that a number of analytical parameters are accurately reported (e.g. alcohol by volume). Many other properties of beer are indicators of the efficacy of the brewing process and whether the brewer is producing a quality product. Analysis of the ingredients of beer (water, malt, hops, yeast) is essential to ensure that standards of quality necessary to produce good beer are met. This course will provide students with an in-depth look at the chemical analyses commonly used to analyze beer and its precursors, using the methods database of the American Society of Brewing Chemists. Students will use what they learn to analyze the ingredients and the beer that they use/produce in the co-requisite practicum in brewing.

Prerequisites: CHM 141, CHL 141, CHM 245 and CHL 245

There is a great deal of time and hard work that goes into planning, building, equipping, and running even a small microbrewery. When a microbrewery fails, it is generally because the ownership doesn’t have a particular skill set, whether it be on the brewing side or on the business side. This course will take students through all of the steps necessary to get a microbrewery from the planning to the operation stage, and also introduce them to the business knowledge necessary for running a successful microbrewery.

Note: A Co-op program is offered for all students in programs 2) and 3) above. Please refer to the Co-op section.

5) Master’s Degree Program (see Graduate Programs section)
   I. Thesis Option
   II. Project Option
Undergraduate B.Sc. Degree Programs

Honours in Computer Science (120 credits)

A. Dissertation-based Honours

18 credits: Program prerequisites (please refer to Table II in the Divisional section of the Calendar)
57 CS credits: 39 required: CS 201, CS 211, CS 216, CS 304, CS 310, CS 311, CS 317, CS 321, CS 403, CS 409, CS 455, CS 499
18 electives: must include 12 credits from 400-level courses and above
12 MAT credits: 12 required: MAT 108, MAT 200, MAT 206, MAT 207
3 PHY credits: PHY 101
3 credits: Arts and Science requirement (please refer to the Divisional section of the Calendar)
27 credits of free electives

B. Course-based Honours

The course-based stream does not require a dissertation (i.e. CS 499) but requires 3 more CS courses. It is primarily designed for students wishing a specialization in Computer Science but are not interested in research and do not intend to pursue graduate studies:

18 credits: Program prerequisites (please refer to Table II in the Divisional section of the Calendar)
60 CS credits: 33 required: CS 201, CS 211, CS 216, CS 304, CS 310, CS 311, CS 317, CS 321, CS 403, CS 409, CS 455
27 electives: must include 15 credits from 400-level courses
12 MAT credits: 12 required: MAT 108, MAT 200, MAT 206, MAT 207
3 PHY credits: PHY 101
3 credits: Arts and Science requirement (please refer to the Divisional section of the Calendar)
27 credits of free electives

General Notes for Honours

- after a minimum of 1 semester, a student with a grade of at least 80% in required courses may request entry to the Honours program
- the dissertation stream requires, in addition, departmental permission.
- students must maintain an average of 80% in required courses to stay in the program

Major in Computer Science (120 credits)

18 credits: Program prerequisites (please refer to Table II in the Divisional section of the Calendar)
45 CS credits: 30 required: CS 201, CS 211, CS 216, CS 304, CS 310, CS 311, CS 317, CS 321, CS 403, CS 409, CS 499
15 electives
9 MAT credits: 6 required: MAT 108, MAT 200
3 elective (MAT 19X cannot count as MAT elective)
3 PHY credits: PHY 101
3 credits: Arts and Science requirement (please refer to the Divisional section of the Calendar)
42 credits of free electives

B.A. PROGRAM, MAJOR IN INFORMATION TECHNOLOGY

Information Technology

Information Technology (IT) is defined by the Information Technology Association of America (ITAA), as the study, design, development, implementation, support or management of computer-based information systems, particularly software applications and computer hardware. IT deals with the use of electronic computers and computer software to convert, store, protect, manage, transmit and retrieve data, securely.

This program provides the necessary skills and knowledge to work/design/participate within organizations that manage large amount of data and provide services to a large number of users. Students will develop skills and knowledge in Information Technologies, Management practices and Organizations, with the required fundamentals of Computer Science.

Note: Students following this degree program are not eligible to add a Business program.

Program prerequisites: (12 credits) Please refer to Table II in the Divisional section of the Calendar.

Core curriculum (30 credits):

- CS 201, CS 211, CS214/CS 325, CS 304, CS 307
- BCS 220, BHR 221, BMA 140, BMA 141, BMG 100

Secondary Core [1] (30 credits)

A minimum of 3 courses in Computer Science.
A minimum of 3 courses in Business, normally chosen from the following list:

- BAC 121, BCS 210, BCS 212, BCS 216, BCS 313, BMG 214, BMK 211, BMK 214, BMK 291, BMK 323, BMK 371, BMS 231, BMS 303, BMS 332

[1] Students are advised to consult the Calendar for prerequisites
Arts and Science requirements (3 credits)
Please refer to the Divisional section of the Calendar

Free electives (45 credits)

Co-Operative Education Program  
B.Sc. Coop

The co-operative Education Program combines a student’s academic program with integrated work experiences through full-time work terms and regular academic sessions. The work terms are designed to present the students with the opportunity to blend theory and practice and to gain relevant work experience.

Each co-operative work term is between 12 and 16 weeks in length, and the student will be registered in a 3-credit Co-operative Placement course (CS 391, CS 392 or CS 393). These course credits count as free electives. Each is graded on a pass/fail basis and this grade is not included in the student’s cumulative average. The evaluation is the responsibility of the Departmental Chair and will be based upon the submission of a work term report and a job performance report submitted by the employer. Normal academic regulations apply to the conduct and evaluation of the courses.

The number of work terms needed depends on the number of credits the student needs to complete upon admission to Bishop’s. Students who have been granted 30 advance credits (or more) will be required to complete two work terms (6 credits). Other students who have been admitted into a regular 120-credit degree program will be required to complete three work terms (9 credits). These credits will be added to the student’s program and do not count as computer science courses, computer science electives, or free electives. All work terms must be completed before the student’s final academic semester and a student’s last semester before graduation cannot be a work term. While every effort will be made to find a suitable placement for all students in the program, no guarantee of placement can be made since the employment process is competitive and subject to market conditions.

Admission to the Co-operative Education Program

Students must submit an application to be admitted to the program. Full-time students in any Honours or Major program in the Computer Science Department who have completed the online application package, who have successfully completed BMG191 and have a minimum cumulative average of 70% upon application are admissible into the Co-op Program. Students in the Co-op must maintain their 70% average and be full-time in order to stay in the program.

Work Term Registration

Once a student has signed the Co-operative Education Agreement, the student may not drop the course associated with the work placement, except for exceptional circumstances. A student who decides to do so will not be able to stay in the Co-operative Education program.

Tuition and Fees

Each work term placement is a 3-credit course and students will pay tuition based upon their fee paying status (Quebec resident, Canadian out-of-province, International).

Work Term Evaluation

Successful completion of the work term is based upon the following:

- The receipt of a satisfactory job performance report from the employer
- The submission of a satisfactory work term report by the student.

The job performance report will be completed by the employer, using guidelines supplied by the Computer Science Department. It is the student’s responsibility to ensure that the employer sends the completed evaluation to the Co-op Coordinator on or before the established deadline. Employer evaluations are confidential and are not reported on the student’s transcript.

MINOR IN COMPUTER SCIENCE  
MINCSC

(24 credits)

- 9 required: CS 201, CS 211, CS 304
- 15 electives from any CS course

CERTIFICATE PROGRAM  
CONCSC

(30 credits)

Description and objectives:
The Certificate Program in Computer Science is designed for individuals who need to acquire the basic understanding of computers and programming and knowledge of the field in order to expand their area of interest and professional expertise. Topics include: Programming, Software Engineering, Web Design, Networks, Graphics, Artificial Intelligence and others. This program will help students to take full advantage of the computer technology available in the workplace.

Prerequisites to programs:
Applicants with insufficient Math background might be required to take an additional 3-credit Math course in their first semester (Math 190 or equivalent).

Program Overview

Certificate in Computer Science

- 12 required credits: CS 201, CS 211, CS 304, CS 321
- 18 credits of CS electives

General Notes/Restrictions:
1. Only one of CS 404, CS 408 or CS 499 may be taken for credit, unless with a special departmental authorization
2. Computer Science courses that are double-listed in Math cannot be counted toward fulfilling the Math electives required for the Computer Science Honours/Major.
3. Students must fulfill their Arts and Science requirements and Humanities requirements outlined in the “Divisional” section of the Calendar.
### List of Courses

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
<th>Units</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS 201</td>
<td>Foundations of Computer Science</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CSL 201</td>
<td>Foundations of Computer Science Laboratory</td>
<td>1-0-3</td>
</tr>
<tr>
<td>CS 203</td>
<td>Interactive Web Page Design</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CSL 203</td>
<td>Interactive Web Page Design Laboratory</td>
<td>1-0-3</td>
</tr>
<tr>
<td>CS 211</td>
<td>Introduction to Programming</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CSL 211</td>
<td>Introduction to Programming Laboratory</td>
<td>1-0-3</td>
</tr>
<tr>
<td>CS 214</td>
<td>Introduction to Networks</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CSL 214</td>
<td>Introduction to Networks Laboratory</td>
<td>1-0-3</td>
</tr>
<tr>
<td>CS 216</td>
<td>System Programming Languages</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CSL 216</td>
<td>System Programming Languages Laboratory</td>
<td>1-0-3</td>
</tr>
<tr>
<td>CS 219</td>
<td>General Topics in Computer Applications</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CSL 284</td>
<td>Unix System Administration Laboratory</td>
<td>1-0-0</td>
</tr>
<tr>
<td>CS 301</td>
<td>Computer Ethics</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CS 304</td>
<td>Data Structures</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CS 306</td>
<td>Functional and Logic Programming</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CS 307</td>
<td>Using and Designing Data Bases</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CS 308</td>
<td>Scientific Programming</td>
<td>3-3-0</td>
</tr>
<tr>
<td>CS 310</td>
<td>Introduction to Software Specifications</td>
<td>3-3-0</td>
</tr>
</tbody>
</table>

Prerequisite: CS 211

Note: Registration priority is given to Science and IT students. CS students must take this course in their first year.

This is the practical laboratory for CS 201

This is the practical laboratory for CS 203

This is the practical laboratory for CS 211

This is the practical laboratory for CS 214

This is the practical laboratory for CS 216

This is the practical laboratory for CS 219

This lab familiarizes students with the Linux and Unix environments covering system administration and user management. Students will start with isolated machines then learn how to interface a Unix system with a network. Advanced topics include the configuration and administration of email and Web servers, as well as techniques for the automation of system administrator tasks via scripting languages. All students will have root and console access to real machines, thus they will gain real networking experience.

Prerequisite: CS 211

Prerequisite or Corequisite: MAT 200
CS 311  Computer Organization and Logic Design  3-3-0
This is a theoretical course on computer organization and architecture. Different computer components and how they function are studied in detail. By the end of the course, students should be able to build (in theory) a small computer without interface. Topics covered are: boolean algebra and gates, combinational circuits (decoders, multiplexers, PLAs), logic design (flip-flops, shifts registers, counters, sequential circuits), the ALU, memory (RAM, ROM, secondary storage), I/O Devices and the control unit (hardwired, microprogrammed). For those interested students, a follow-up course, largely consisting of lab experiments, is CS 312
Prerequisite: CS 201, CS 211

CSL 311  Computer Organization and Logic Design Laboratory  1-0-3
This is the practical laboratory for CS 311

CS 312  Microcomputer Interfacing  3-3-0
This course and integrated laboratory introduces the techniques used to interface a microcomputer to the real world with a robot as the main interface. Students will complete projects of increasing difficulty as they build and program a robot to accomplish a given task and brief the class on their design and findings. Topics will include: interactive programming, analog and digital inputs, use of the bus and registers to control output signals, simple electronic sensors, multiplexing and decoding and practical problem solving.
Prerequisite: CS 311 or equivalent programming and electronic knowledge.

CSL 312  Microcomputer Interfacing Laboratory  1-0-3
This is the practical laboratory for CS 312

CS 315  Data Communications  3-3-0
This is a theoretical course on Data Communications. It covers the basic and physical aspects involved when data is transmitted from one point to another, such as analog vs. digital transmission, various forms of encoding analog and digital data into appropriate signals, error detection techniques, multiplexing, etc... As well as an introduction is given on networking techniques, differences between circuit and packet switching, routing techniques, and Local Area Networks. The course is of mathematical and physical nature.
Prerequisites: CS 211 or CS 216
Note: See PHY 365. Students may not take this course for credit if they have received credit for PHY 365.

CS 316  Artificial Intelligence  3-3-0
A course aiming to introduce students to the basic concepts and techniques of Artificial Intelligence. Topics will include: Search strategies; knowledge representation; AI languages; Rule-based inference systems, expert systems; computer vision; planning and problem solving; natural language understanding.
Prerequisite: CS 304, CS 306 or CS 403

CS 317  Design and Analysis of Algorithms  3-3-0
This course is intended to make students familiar with most of the existing techniques for problem solving. It starts with an introduction to algorithms efficiency, solving recurrence relations and basic data structures. Then different techniques for algorithms design are discussed; the divide-and-conquer technique, the greedy technique and its applications to graph algorithms, dynamic programming, backtracking and branch and bound algorithms. With every technique presented, examples from different domains are studied and their algorithms analyzed. At the end, students are briefly introduced to the vast area of "difficult" problems, or NP-complete.
Prerequisite: CS 304 and MAT 200

CS 318  Advanced C++ Programming  3-3-0
The C++ language has become an industry standard as an implementation language. The course aims at introducing the student to intermediate and advanced programming using C++, with particular emphasis on systems software and the use of the C++ object-oriented extensions in software engineering. C++ Programming basics (loops and decisions, arrays, structures, functions, pointers) Objects and Classes, Inheritance, Virtual functions, files and stream, I/O Structure and design of Class libraries, Standard Template Library, OOP Design basics. Although no prior experience in C is required, it is assumed that the student is already fluent in some other programming language and in the programming of data structures.
Prerequisite: CS 304

CS 321  Advanced Programming Techniques  3-3-0
The course is intended to be a sequel to introductory programming with emphasis placed on the architecture of software. It will go in depth into object-oriented techniques, reusability, data abstraction, class design, and implementation, design and structure of class libraries. Topics to be covered include: polymorphism, encapsulation, overloading, inheritance and delegation, types of inheritance (Inheritance for Extension, Specialization and Specification), composition, aggregation and design of collections. Static and dynamic types, downcasting, exception handling. The second half of the course will be devoted to software design patterns, with particular emphasis on the observer, iterator, visitor and selected creationlal patterns. Course work will involve significant programming projects. The teaching language will be Java.
Prerequisite: CS 304 Allow concurrent

CSL 321  Advanced Programming Techniques Laboratory  1-0-3
This is the practical laboratory for CS 321

CS 325  Computer & Network Security  3-3-0
This course provides an introduction to security and privacy issues in various aspects of computing, including cryptography, software, operating systems, networks, databases, and Internet applications. It examines causes of security and privacy breaches, and gives methods to help prevent them.
Prerequisite: CS 216

CS 330  Programming Mobile Apps  3-3-0
This course will cover mobile application development for the Android operating system using Android Studio. The programming language is Java. Setting up and using Android Studio IDE will be covered in the introduction. Students will learn how to design and develop Android applications using best practices to account for the limited screen size and memory of mobile devices. Topics to be covered include layout design/management, communication between apps, Google Maps, 2D graphics, and mobile app specific software engineering patterns. The course concludes with monetization (ads, in-app purchases, etc) and app store optimization strategies. The goal is for each student to develop and release an app by the end of the course.
Prerequisite: CS 211

CSL 330  Advanced Programming for Mobile Apps Laboratory  1-0-3
This is a practical laboratory for CS 330
Co-requisite: CS 330

CS 375  Numerical Methods  3-3-0
Prerequisite: CS 211, Mathematics 108 and 207.
Note: See Mat 325 and Phy 375. Students may not take this course for credit if they have received credit for MAT 325 or for PHY 375

CS 379  Electric Circuits and Electronics  3-3-3
Review of D.C. circuits, Kirchhoff’s laws, network theorems. Network analysis for A.C. circuits, phasors. Diode circuits and filters. The physical basis of semiconductor devices including semiconductor diodes, junction transistors, and field-effect transistors. The operation of transistor amplifiers, digital electronics and integrated circuits will also be covered.
Note: See PHY 319. Students may not take this course for credit if they have received credit for PHY 319

CS 391  Co-operative Placement I  3-0-0
Students will integrate theory and practice through a related work placement
Prerequisite: admission to the Co-op Education Program

CS 392  Co-operative Placement II  3-0-0
Students will integrate theory and practice through related work placement
Prerequisite: CS 391

CS 393  Co-operative Placement III  3-0-0
Students will integrate theory and practice through related work placement
Prerequisite: CS 392

CS 394  Stage in Bioinformatics  6-0-0
Students will integrate theory and practice through a related stage.
Prerequisite: CS 372 / BCH 342

CS 400  Independent Studies  3-0-0
Individual study and research under the guidance of an advisor and Department staff.
Prerequisite: Permission of the department
CS 401 Simulation Techniques 3-0-0
Computer simulation is defined and put into the context of other simulation methods. Two main techniques are studied, one involving automated spreadsheets (financial modelling) and the other queuing theory. A term project involving the simulation of an actual system is part of the course.
Prerequisite: CS 304, PHY 101 (or equivalent)
Note: Students may not take this course for credit if they received credit for BMS 343. This course will be offered in alternate years.

CS 402 Computer Graphics 3-0-0
This is an introductory course to the principles of interactive raster graphics. Topics include an introduction to basic graphics concepts, scan conversion techniques, 2-D and 3-D modeling and transformations, viewing transformations, projections, rendering techniques, graphical software packages and graphics systems. Students will use OpenGL graphics API to reinforce concepts and study fundamental computer graphics techniques.
Prerequisites: CS 304, MAT 108

CSL 402 Computer Graphics Laboratory 1-0-0
This is the practical laboratory for CS 402

CS 403 Principles of Programming Languages 3-3-0
The objective of this course is to introduce, analyze and evaluate, on a comparative basis, the concepts on which programming languages and their implementations are based. Topics to be covered: definition of languages, syntax and semantics; Compilation techniques, top-down parsing, creating a parser; variables and binding, expressions, statements; data types; procedures scope, and run-time considerations; coroutines; implementation of block-structured languages; modularity and abstractions; concurrency exception handling and program correctness; functional programming object-oriented programming languages; logic programming and constraint languages.
The languages ML, Eiffel, Lisp, Scheme, Prolog, Haskell, 02, Java and Smalltalk will be used to illustrate the above concepts.
Prerequisites: CS 304 and CS 310

CS 404 Project 3-0-3
This course is normally taken by CS students in their final year. The project must be approved in advance by the department. Students will be expected to submit a written report and to make a presentation.
Prerequisite: approval of the dept., 80% in CS courses

CS 405 Data Mining 3-3-0
Data is now created faster than humans are able to understand it and use it. There may be patterns hiding within this data with potentially useful information. This course will teach students, how to discover these patterns for the purpose of solving problems, gaining knowledge, and making predictions. Topics covered in this course include data preparation, clustering, classification, association rules for mining and models combination. This course includes assignments and a final project where the students are required to perform mining on real datasets.
Prerequisites: PHY 101 (or equivalent)
See PHY 374
Students may not take this course for credit if they received credit for PHY 374.

CS 406 Compilers and Interpreters 3-3-0
This course is intended as an introduction to the fundamentals of language translation and compiler construction. Topics will include language theory and syntax; grammars, finite state machines, non-deterministic push-down automata; a thorough treatment of parsing methods covering top-down, bottom-up and precedence parsers; Syntax directed translation; Run-time environments; optimization and error recovery; code generation. Students will be required to construct a working interpreter of a Pascal-like language.
Prerequisite: CS 310
This course will be offered on alternate years.

CS 408 Project II 3-0-0
This course is normally taken in the final year of studies and may involve work on a theoretical topic or a practical implementation of a sizable software project. The topic must be approved in advance by the department. Students are expected to attend bi-weekly project meetings where they present and discuss their work. In addition, they will make a final presentation at the end of term and submit a report.
Prerequisite: approval of the dept. 80% in CS courses.

CS 409 Principles of Operating Systems 3-3-0
Basic concepts of computer hardware; program translation linking and loading; cooperating sequential processes; critical section problem, process synchronization primitives, parallel programming; introduction to multiprogramming; operating system nucleus; file systems; reliability and protection; system performance, measurement and evaluation. Memory Management. Paging and Virtual memory. Unix. Using and programming the Unix Shell, Unix implementation. Examination of the implementation of Unix clones Minix, Linux, Survey of state-of-the-art operating systems. Distributed Systems, Communication and synchronization in distributed systems. Theoretical issues and implementation.
Prerequisites: CS 304

CS 410 Software Engineering 3-3-0
Software is an engineered product that requires planning, analysis, design, implementation, testing and maintenance. This course is a presentation of the techniques used in each step of the software product process. Topics: software requirements analysis and specifications; software design process, object oriented design; testing, reliability and maintenance. Students will be expected to work jointly on several large software projects.
Prerequisites: CS 304, CS 310, CS 321, CS 403 (allow concurrent)

CS 411 Advanced Computer Architecture 3-3-0
The focus in this course is on basic principles, current practice, and issues in computer architecture and organization. At the end of the course students will have gained an understanding of how a computing system is organized, as well as why it is organized this way. The relation between hardware and the software that runs on it is emphasized, leading to an intuitive understanding of how the behavior of applications influences computer organization and design. Topics covered typically include (but are not limited to): instruction set design, micro-programmed versus hardwired processors, pipelining and superscalar processors, memory organization (cache, primary, virtual), I/O and interrupts, multiprocessors. Comparative critical and quantitative analyses of various systems that currently exist are presented.
Prerequisites: CS 311 or instructor's permission.

CS 412 Computer Games Design 3-3-0
This course will explore the theory and practice of video game design and programming. Students will learn the basic concepts and techniques for the design and development of digital games. The topics covered in this course will include the history and taxonomy of video games, the basic building blocks of a game, computer graphics and programming, use interface and interaction design, and the software architecture for video games. It is assumed that students have taken courses in programming (best if it includes C or C++ and data structures. A good background in algorithms and basic mathematics (matrix algebra, trigonometry, linear algebra, vector calculus) is an asset for this course.
Prerequisite: CS 304

All special topics courses require CS 211 and CS 304 as a prerequisite and/or any other course as specified by the instructor.

CS 415 Special Topics in Communications 3-3-0
The course will present topics of current interest or research directions in Computer Communications Networking and network programming. The course content is expected to vary to reflect the current interests of students and faculty. It will be offered by arrangement with the department.

CS 416 Special Topics in Software 3-3-0
The course will present topics of current interest or research directions in Software Science. The course content is expected to vary to reflect the current interests of students and faculty. It will be offered by arrangement with the department.

CS 417 Special Topics in Computer Applications 3-3-0
The course will present topics of current interest or research directions in Computer Applications. The course content is expected to vary to reflect the current interests of students and faculty. It will be offered by arrangement with the department.

CS 418 Topics in Computer Science 3-3-0
The course will present topics of current interest or research directions in Computer Science. The course content is expected to vary reflecting the interests of the students and the faculty. It will be offered by arrangement with the department.
CS 426 Computer-Assisted Interventions 3-3-0
This course introduces students to the fundamentals of computer-aided intervention (CAI) in medicine. The use of computing technology before, during, and after interventions will be examined. Specifically, this course will teach students about tracking devices, coordinate systems, spatial transformations, rigid and non-rigid registrations (feature-based & intensity-based), calibration, digitization and imaging. Clinical applications will also be discussed. Basic knowledge of either C++, python or matlab is an asset.
Prerequisites: Instructor’s permission
Students cannot receive credits for both CS426 and CS526.

CS 454 Complements in Data Structures and Algorithms 3-3-0
The aim of this course is to cover many concepts in Data Structures, Algorithms, and Programming to make up deficiencies in Computer Science background for entering graduate students.
This course cannot be taken for credits by undergraduate students

CS 455 Theoretical Aspects of Computer Science 3-3-0
The course will include several of the following topics: Computational models, Computational complexity; Finite-state machines; Context-free languages; Pushdown automata; Turing machines; Undecidable problems.
Prerequisite: CS 211, MAT 200

CS 457 Database Software Design 3-3-0
This course covers how one can implement a Database Management system. Major topics are storage management, Query processing, and Transaction management. As a basic assumption, data will not all fit in main memory, so algorithms and data structures appropriate for effective disk storage and quick access must be used. For example, one may use index structures such as B-trees or hash tables. We cover parsing of queries and optimizing of query plans. Finally, we cover durability of transactions using logging, and concurrency control for isolation of transactions. Additional topics in distributed databases are also presented.
Prerequisite: CS 307

CS 462 Image Processing 3-3-0
This course will introduce the area of Image Processing and present classical tools and algorithms in the field including: image perception, image acquisition and display, histogram techniques, image restoration, image enhancement, primitive operations for image analysis, segmentation, image transforms, and pattern and object recognition.
Some examples of industrial applications of image processing and some important developments in image processing research will be also addressed.
Prerequisites: CS 304, MAT 192, PHY 101 (or equivalent)

CS 463 Computer Vision 3-3-0
This course is concerned with the computer acquisition and analysis of image data. Computer vision is the construction of explicit, meaningful descriptions of a physical object from images. Emphasis will be placed on: camera models and calibration, image representation, pattern recognition concepts, filtering and enhancing, segmentation, texture, motion from image sequences, deformable models, matching, stereovision, perceiving 3D from 2D images and tracking with dynamic models. The programming projects assigned in this course will make substantial use of the C and C++ programming languages
Prerequisites: CS 304, CS 318, MAT 192, PHY 101 (or equivalent)

CS 464 Network Programming 3-3-0
This course presents computer networks at a functional level, with strong emphasis on programming distributed applications over a network. Discussion will be based on open networking and application standards such as the TCP/IP protocol suite and the Portable Operating System Interface (POSIX). Topics normally covered are TCP/IP architecture and programming, the client-server model, network file systems, streaming, tunnelling. Programming distributed applications in C or C++ is an integral part of the course.
Prerequisite: CS 216

CS 467 Special Topics in Algorithms 3-3-0
The course builds on the techniques covered in CS 317 to present some specialized algorithms in several areas, including Bioinformatics, Computational Geometry, and Network Flow.
Prerequisite: CS 317 or permission of the instructor

CS 469 Special Topics in Computer Science 3-3-0
The course will present topics of current interest or research directions in Computer Science. The course content is expected to vary from year to year to reflect the current interests of students and faculty. It will be offered by arrangement with the department.

CS 471 Graph Theory 3-3-0
An introduction to the combinatorial, algorithmic and algebraic aspects of graph theory.
Prerequisite: CS 304, MAT 200
Note: See MAT 421. Students may not take this course for credit if they have received credit for MAT 421.

CS 499F Honours Dissertation 6-0-0
The student is required to complete a theoretical or applied project. The subject is arranged with the student’s supervisor during the first four weeks of term. A written dissertation is required as well as two seminar presentations.
Note: This course is open only to final year Computer Science Honour Students in the dissertation stream, and only by permission of the department

Mathematics

Faculty
Madjid Allili, B.Sc.(Algiers), M.Sc., Ph.D.(Sherbrooke); Professor
Thomas Brüstle, B.Sc., (Ludwig-Maximilians), M.Sc., Ph.D. (Zurich), Professor, Maurice-Auslander Research Chair
François Huard, B.Sc., M.Sc., Ph.D. (Sherbrooke); Professor, Chair of the Department
Trevor H. Jones, B.Sc.H. (Acadia), M.Sc. (Dalhousie), Ph.D. (University of New Brunswick); Senior Instructor
Scosha Merovitz, B.Sc.(Bishop’s), M.Sc.(Dalhousie); Coordinator, Math/Stats Help Centre
David Smith, B.Sc., M.Sc., Ph.D. (Sherbrooke); Adjunct Professor
N. Brad Willms, B.Math. M.M., Ph.D. (Waterloo); Associate Professor

Program Overview
Mathematics is the language of the sciences, a language which allows scientists to quantify, model, understand and predict behaviour in an enormously diverse range of phenomena of interest. Simultaneously, Mathematics is often regarded as an art, as it is the creative study of patterns and of problem solving. Mathematics covers a wide range of disciplines including algebra, analysis, combinatorics and discrete mathematics, and differential equations. In first-year courses, mathematics students are joined by other science students, particularly from Physics and Computer Science. In the advanced courses, classes are very small, and some are given on an individual or tutorial basis.

The highest level of specialization is Honours, and Honours programs prepare students for direct entry into graduate work leading to a Master’s or Ph.D. degree. All honours mathematics students have an opportunity to study independently and thus develop their reading and problem solving skills, and there is some chance to pursue special interests. The Majors programs provide students with an excellent general preparation for the career world,