ESG 575 Tropical Environments and Climate Change 3-3-0
This course attempts to provide an overview of the tropics as a unique environment and one that poses special problems to its human occupants. The working assumption in the course is that the tropics comprise a far too complex and heterogeneous environment for simple generalizations to apply. However, by gaining some understanding of how its component systems work, one can be in a better position to identify the appropriate questions to be asked and experiments to be performed, so that site-specific solutions can be developed for management problems in different parts of the tropical world. The course will provide a review of tropical climatology, soils, and biomes, in addition to discussing more applied issues such as forestry and agriculture.

ESG 577 The Health Impacts of Climate Change 3-3-0
Climate change is expected to affect human health in numerous ways. The most obvious health impacts are those associated with thermal stress and extreme weather events such as floods and hurricanes (premature deaths, infectious diseases; diarrhoeal disease). Global warming will also be associated with a spread of vector-borne diseases (such as malaria, dengue fever, yellow fever, Lyme disease, etc.) and increases in seasonal allergies. The course will examine the overall impact of environmental degradation, displacement and loss of livelihood on the general physical and mental health of populations.

Courses will be offered in the winter semester of each year. Students may take all three courses in one semester, or take courses over a period of several years. The courses will consist of a combination of seminar, lecture, fieldtrip, and laboratory instruction.

Graduate Certificate in Knowledge Mobilization

Program Overview (15 credits) CONKMB

Knowledge Mobilization (KMb) is the process by which we share and uptake information for the benefit of society. The goal of this Graduate Certificate is to develop students' knowledge, skills and values with respect to KMb and build the capacity to select and apply KMb tools and techniques in research and/or applied contexts. The basic entry requirement will be an undergraduate degree in any field from a recognized university with at least a B standing in the final two years of study. There are no specific pre-requisites, though it is highly recommended that students have some background in research methods and scientific concepts. Students in this program will complete 4 courses (15 credits), including a 6-week practicum.

KMB511 Theories of Knowledge Mobilization in Research Settings 3-3-0
This course is an overview of the theories and practices involved in the creation, synthesis, translation and dissemination of knowledge in science and social science research contexts, including areas of knowledge translation and implementation science. We will discuss various contexts in which knowledge is created, ethical and equity principals of what research "should" be mobilized, integrating knowledge mobilization into research design, and how to identify barriers and facilitators researchers face using and sharing knowledge. Across various disciplines, we will identify current tools and techniques to evaluate the success of KMb initiatives. Students in this class will create their own KMb plan for a program of research and will design an evaluation of their KMb project.

KMB515 Theories of Knowledge Mobilization in Applied Settings 3-3-0
This course is an overview of how applied settings such as health, social services, and non-profit sectors, can engage with research at the level of practice, program development, and policy. We will also discuss how these sectors can inform research creation. We will discuss various contexts in which research could/should be applied and weighed in decision making, how to identify audiences for specific areas of knowledge, and how to identify barriers and facilitators to brokers or people in the field. We will discuss how researchers can building partnerships with consumers of their research. Across various sectors, we will identify current tools and techniques to evaluate the success of KMb initiatives. Students in this class will create their own KMb package to inform or raise awareness, and will design an evaluation of their KMb project.

BIO 520 / KMB 520 Science Communication 3-3-0
This course will focus on skill development, writing and communication strategies for online and print media, such as online biogs, and columns in local newspapers, as well as current innovations in communication such as infographics. Students will hone their skills in writing techniques, particularly in communicating complex scientific material to a broader audience.

KMB 530 Knowledge Mobilization Practicum 6-6-0
The practicum will explore the application of knowledge gained from the theoretical courses taken in the first half of the Certificate. Students would be placed in a local organization with a mandate to translate knowledge into action. Examples of projects that could be conducted during a practicum include establishing a KMb strategy for an organization, creating KMb materials that would meet the organization's needs, researching and writing a column in a local newspaper, developing a workshop, or participating in a KMb internship at Bishop's University's Research Office. At the end of the practicum, students would submit a portfolio that would include the KMb projects that they have led and/or supported over its duration, a preliminary evaluation of these projects, as well as a journal outlining how previous study of KMb theories and of science communication informed their practicum experience.
Prerequisites: KMB 520

Master's Degree in Computer Science

Master's Degree Program (45 credits)

I. Thesis Option CONCSC

Entrance Requirements
The minimum requirements for admittance to the Master's program are an undergraduate degree with a major in Computing Science or equivalent, and a high upper-second class standing.

Candidates with high graduate academic standing in an undergraduate degree other than computer science, who have some computer science background either graduate academic or professional, may be admitted as graduate preparatory students. Preparatory students will be asked to complete up to one year of undergraduate courses to enhance their background. We may at our discretion replace the requirement of preparatory period with a set of preparatory co-requisites. These co-requisites will consist of undergraduate courses to be taken concurrently with the regular graduate courses. In these circumstances the candidate is admitted directly to the graduate program, but should expect a longer residency in the program.

In addition to graduate academic performance, a combination of factors is taken into consideration in assessing the eligibility of a candidate for admission into graduate programs. One important such a factor is the availability of a faculty member competent and

willing to supervise the graduate academic program of studies and research of the candidate; a candidate will not be admitted to the program under any circumstance unless such a faculty member exists.

Other factors graduate the performance of the candidate and the assessment provided by his/her referees as a measure of the likelihood that the candidate can graduate complete the course of studies and research.

Program Requirements

45 CS credits:

15 credits: five 500-level CS courses
6 credits: Graduate Seminar courses (CS 597 and CS 598)
24 credits: Master's Thesis CS 599

Graduate students should familiarize themselves with the University and divisional calendar and regulations. Some of the information herein is adapted from these regulations, but is not intended as a replacement.

Completing the degree normally requires five one-term 500-level courses, registration and participation in the Graduate Seminar courses (CS 597 and CS 598), together with a Master's thesis (CS 599). Courses are chosen by students in consultation with their supervisor. All courses prescribed for a student's approved program of study are designated as primary. Courses additional to the student's approved program are designated as secondary. Failure to attain a minimum of 65% in any of the primary courses may result in the student being required to withdraw from the program.

Under certain circumstances, it is permissible for a student admitted to the program to follow an approved graduate-level credit course at another university. All interested students should consult their supervisor and the chair of their department prior to registration in order to obtain further information on procedures and conditions of eligibility.

A thesis proposal is expected by the end of the third term in the program. Such a proposal is a requirement in CS 597 and so students should plan to register in this course accordingly (and in consultation with their supervisor). A progress report (CS 598) is expected in the term following the proposal and should be provided no later than two terms after the proposal. Failure to register in CS 597 or CS 598 in time as well as a Fail grade in either of these two courses may result in the student being required to withdraw from the program.

Thesis topics are chosen after discussion with potential supervisors. The amount of flexibility allowed in pursuing a particular topic will vary according to the supervisor's needs and interests. Theses are defended before an examining committee consisting of two examiners and one chair. The chair of the examining committee does not have to submit an evaluation of the thesis, but can do so if she/he so wishes. In the event of a difference in opinion between examiners on whether the thesis is acceptable the chair shall have the tie-breaking vote. The supervisor(s) of a candidate cannot be a voting member of that candidate's examining committee, but is expected to participate in the committee's deliberations.

Any candidate (full-time or part-time), after initial registration in a thesis must maintain this registration in all successive terms (including the term in which the student is examined) until his/her thesis is completed, with the exception of possible temporary leaves of absence dully approved by their supervisor as specified in the general regulations for graduate studies in science. Completion means submission of a final grade to the Faculty after modifications, any retyping involved, etc. Students should note that faculty approval to register in the thesis is given on the understanding that the student will be in regular contact with his/her supervisor, and that thesis research will be actively pursued in each term of registration.

Stream change

Students in the Project stream can switch to the Thesis stream only with permission from the department. In addition, they must meet the following conditions: (a) they have taken and passed at least four graduate courses at Bishop's, (b) they have an average grade of 90 or better in the graduate courses taken at Bishop's, and (c) at least one faculty expresses interest in supervising their research toward the Master's thesis. No more than five graduate courses can be counted toward the requirements of the new program..

II. Project Option CONCSP

Entrance Requirements

The minimum requirements for admittance to the Master's program are an undergraduate degree (minimum of a 70%) with a major in any of the following disciplines: Computing Science, Information Technology, Computer Engineering, Electrical Engineering, Mathematics or Physics. Note however that the admission process is competitive and so meeting the minimum requirements does not guarantee admission.

Candidates with insufficient academic background in Computer Science may be admitted as graduate preparatory students. Preparatory students will be asked to complete up to one year of undergraduate courses to enhance their background. We may at our discretion replace the requirement of preparatory period with a set of preparatory co-requisites. These co-requisites will consist of undergraduate courses to be taken concurrently with the regular graduate courses. In these circumstances the candidate is admitted directly to the graduate program, but should expect a longer residency in the program.

Program Requirements

45 CS credits:

36 credits: twelve 500-level CS courses
9 credits: CS 590 (Master's Project)

CS Graduate students should familiarize themselves with the University and divisional calendar and regulations. Some of the information herein is adapted from these regulations, but is not intended as a replacement.

Completing the degree normally requires twelve one-term 500-level courses, together with a Master's project (CS 590).

The twelve one-term courses (36 credits) must be chosen subject to the following restrictions: a minimum of three courses (9 credits) must come from the Data Science block, a minimum of three courses (9 credits) must be taken from the Systems and Applications block, and a minimum of one course (3 credits) must come from the Theory block. The blocks are defined as follows:

1. Data Science (at least 9 credits): CS 503, CS 504, CS 505, CS 507, CS 509, CS 550, CS 566.

2. Systems and Applications (at least 9 credits): CS 501, CS 512, CS 516, CS 520, CS 526, CS 556, CS 557, CS 560, CS 563, CS 564, CS 565.

3. Theory (at least 3 credits): CS 502, CS 506, CS 510, CS 515, CS 555, CS 561, CS 567, CS 571, CS 562.
Most courses are available during the regular semesters (Fall and Winter). The Master's project is normally available only during the Spring/Summer semester.

Students whose cumulative average falls under 65% will be restricted to 3 course per semester until their average is brought back to 65%. Failure to maintain a minimum of a 65% cumulative average may result in the student being required to withdraw from the program.

Stream change

Students in the Project stream can switch to the Thesis stream as long as they meet the following conditions: (a) they have taken and passed at least four graduate courses at Bishop's, (b) they have an average grade of 75 or better in the graduate courses taken at Bishop's, and (c) at least one faculty expresses interest in supervising their research toward the Master's thesis. No more than five graduate courses can be counted toward the requirements of the new program.

List of Graduate Courses

CS 500 Project I 3-3-0
This course can only be taken by M.Sc.-course-based (CONCSP) students. The goal is to pursue a research project under the supervision of a faculty member or opt for an internship at a public institution or a private company. The project or internship must be approved in advance by the department. It is also the responsibility of the student to find the internship and a faculty member of the department willing to supervise the proposed project. Students will be expected to submit a written report and to make at least one presentation on the project.
Prerequisites: Permission of the department and availability of a supervisor.

CS 501 The Internet of Things 3-3-0
How can companies deal with the vast amount of data coming from a variety of different devices? In the 'Internet of Things' there are many different devices, sensors and data logs. How can a computer scientist take this data and turn it into a readable or graphical form (dashboard) for people to make sense of. The course will consist of looking at how devices such as the 'Fitbit', smartphones, in house security systems send data over the Internet to a server and how this data can be interpreted into something that large corporations can use.

CS 502 Computational Topology 3-3-0
Computational topology uses topological concepts with efficient algorithms to analyze data and solve problems in many fields, including computer graphics and image analysis, sensor networks, clustering, robotics, and others. This course will present an introductory, self-contained overview of computational topology. This course has no formal prerequisites, but a basic mathematical knowledge in calculus and algebra at the senior undergraduate level and some familiarity with the use of computer packages (e.g., Matlab, R, C++, etc.) are expected. We will cover basic concepts from a number of areas of mathematics, such as abstract algebra, algebraic topology, and optimization. We will also look at algorithms and data structures, and efficient software for analyzing the topology of point sets and shapes – termed topological data analysis, or TDA.

CS 503 Data Visualization 3-3-0
The course explores analytical methods paired with appropriate visualizations for automated and human-assisted analysis for data sets. Several visualization techniques allowing to present data to an observer in a way that yields insight and understanding will be investigated. These big data analysis and visualization techniques are applied to data sets from a wide variety of scientific domains such as biology, physics, engineering, and medicine. The analysis and visualization methods will be illustrated through concrete examples.

CS 504 Programming Languages for Data Analysis 3-3-0
In this course students will be introduces to the most popular languages and software environments used in statistical computing and visualization. The course will involve significant programming projects in SAS, Weka, R and Python.

CS 505 Data Mining 3-3-0
Cross-listed with CS 405. 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. Students are expected to perform a substantial analysis of the data set, or prepare a research paper.

CS 506 Parallel Models and Algorithms 3-3-0
This course provides an introduction to the design and analysis of parallel algorithms and to the various models of parallel computation. The course will discuss parallel algorithms for problems such as: basic arithmetic, sorting, searching, selection, graph theory, matrix computations, combinatorial enumeration, optimization, computational geometry, and numerical analysis. Parallel computational models and their properties will be presented. Other typical topics include: complexity classes, and the parallel computation thesis.

CS 507 Statistical Learning 3-3-0
Statistical learning is concerned with modelling and understanding vast and complex datasets using methods rooted in statistics. The main objective is for the students to master how and when to apply statistical learning techniques in real world applications. Topics covered include linear regression, classification, linear discriminant analysis, tree based methods, support vector machines, graphical models, random forests and boosting. Projects illustrating how to implement each of the statistical learning methods are carried out using a statistical software package.

CS 508 Project II 3-3-0
This course can only be taken by M.Sc.-course-based (CONCSP) students who have already completed Project I course (CS 500). The goal is to further a study undertaken in Project I (CS 500). The project must be approved in advance by the department. It is also the responsibility of the student to find a faculty member of the department willing to supervise the proposed project or the internship. Students will be expected to submit a written report and to make at least one presentation on the project.
Prerequisites: CS 500, permission of the department, and availability of a supervisor.

CS 509 Pattern Recognition 3-3-0
This course addresses the fundamental theory and techniques of pattern and features classification in numerical data. Pattern recognition methods can be useful in diverse real world applications such as medical data processing, data mining, information retrieval, computer vision, handwriting and speech recognition, and more. The course topics include Bayesian decision theory, statistical classification, maximum likelihood estimation, nonparametric techniques, stochastic methods and unsupervised learning.

CS 510 Model-Based Testing of Reactive Systems 3-3-0
The course provides an in-depth exposure to the area of formal methods called model-based testing. Various testing models will be presented, including traces, may- and must-testing, refusals, and failure traces. Relations to related specification and verification techniques such as temporal logic and model checking will also be investigated. Students are expected to participate in the presentation of the lecture material and perform independent research.

CS 512 Computer Games Design 3-3-0
Cross-listed with CS 412. 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, user interface and interaction design, and the software architecture for video games. Students are expected to prepare a research paper during the course, or pursue a larger applied project.

CS 515 Concurrent & Real-Time Systems 3-3-0
This course provides an introduction to a process algebra such as CSP. It then uses this language for the specification, analysis, and verification of concurrent and real-time systems. Finally, the course presents the use of such a process algebra as a formal method for concurrency at different stages in the development process.

CS 516 Volumetric Image Analysis & Visualization 3-3-0
Digital volumetric images are stacks of two dimensional image slices produced for instance by tomographic scanner. The goal of this course is to study the different techniques and algorithms for the analysis of volumetric images including a discussion about some sources of volumetric images, especially those occurring in medical imaging with different modalities (Radiology, Computed Tomography, Magnetic Resonance Imaging, Nuclear Medicine, Ultrasound, Positron Emission Tomography). The course will also address the different techniques used to display and visualize volumetric images including volume slicing, surface rendering, and volume rendering.

CS 520 Advanced Topics in Software 3-3-0
The course will present topics of current interest or research directions in software and related areas. The course content is expected to vary to reflect the current interests of students and faculty. Students are expected to participate in the presentation of the lecture material and engage in independent research.

CS 536 Web Mining 3-3-0
This course on web mining delves into the multidisciplinary realm at the forefront of analyzing and harnessing the abundant and diverse data accessible on the World Wide Web. The course delivers a comprehensive introduction to the fundamental principles, techniques, and practical applications of web mining, equipping students with the expertise to extract valuable insights from web-based data. Covered topics include, but are not limited to, association rules, machine learning, information retrieval and web search, social network analysis, web crawling, structured data extraction, information integration, opinion mining, sentiment analysis, and web usage mining. Furthermore, students will have the opportunity to participate in substantial projects, gaining hands-on experience with web mining tools. Upon completing the course, students will be well-prepared to proficiently navigate the intricate web data landscape and apply their skills in diverse real-world scenarios.
Note: Cross-listed with CS 436. Students may not take this course for credit if they have received credit for CS 436.

CS 550 Big Data Management and Analytics 3-3-0
In this course, students will learn the fundamental theory and techniques of Big Data management and analytics, and to apply them to resolve problems in real-world applications. The principle is to learn the strategic extraction and usage of information from large datasets. In fact, the students will exploit recent concepts and trends to manage and analyze Big Data. For that purpose, recently designed algorithmic approaches and technologies will be covered to help the students to manage and analyze large datasets. For that aim, the covered topics will include, but will not be limited to streaming algorithms, distributed file-system architecture, resilient distributed datasets, similarity search, recommendation systems, link analysis, edge computing, and federated machine learning. The students will work on large projects to practice the concepts presented in this course. In addition, the students will have to submit a paper on a topic related to their research.
Note: Students may not take this course for credit if they have received credit for CS 450.

CS 551 Deep Learning 3-3-0
Deep Learning, a cutting-edge subfield of machine learning, revolutionizes our approach to resolve complex artificial intelligence problems. This course provides students with a comprehensive understanding of deep learning techniques, including deep neural networks, Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Generative Adversarial Networks (GANs), Transformer, Generative pretrained Transformer (GPTs), and deep reinforcement learning. The course bridges theoretical foundations and practical implementation, equipping students to design, train, and evaluate deep neural networks for applications in image recognition, speech processing, natural language processing, bioinformatics, autonomous systems, and more. Throughout the semester, students engage in significant practical projects, applying deep learning to real-world artificial intelligence challenges and enhancing their skills for research and industry.

CS 555 Automata Theory & Computational Complexity 3-3-0
Cross-listed with CS 455. The course will address finite-state machines, context-free languages and pushdown automata, computability. A systematic study of the known relations between the most important resource bounded complexity classes, reductions, separation results and translation techniques is also included. Students are expected to prepare a research paper during the course.

CS 556 Compilers and Interpreters 3-3-0
Cross-listed with CS 406. 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 are expected to implement complex semantic analysis and a complex compiler back-end. In particular they are expected to implement various aspects of machine code optimization.

CS 557 Database Software Design 3-3-0
Cross-listed with CS 457. 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. Students are expected to prepare a research paper during the course, or pursue a larger applied project.

CS 560 Software Engineering 3-3-0
Cross-listed with CS 410. 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; automated design tools, programming environments. Students are expected to prepare a research paper during the course, and work on large applied projects.

CS 561 Discrete Structures and Computational Statistics 3-3-0
Sets and functions, Propositional logic, predicates and quantifiers, logical inference, mathematical induction, sequences, summations, recurrence relations, algorithms design and complexity analysis. In depth review of the basic concepts of probability and statistics, simple and multiple linear regressions and applications, analysis of variance. Classification Models: Overview of classification, linear methods, nearest neighbor classification, Bayes classification, logistic regression, linear discriminant analysis. Clustering.
Note: Students cannot receive credits for both CS 561 and MAT 529.

CS 562 Mathematical Models in Image Processing 3-3-0
Cross-listed with CS 462. Image processing is a rapidly growing field. As such, it requires and necessitates a number of mathematical models and domains to achieve efficient processing algorithms. Designing a successful processing technique invariably relies on having a successful model for images themselves. The mathematical techniques needed could range from Partial differential equations, Differential geometry, Morse theory, Topology, Algebraic topology, Wavelets, Statistical techniques, Calculus of variations, Numerical methods, Graph theory, and Optimization. The objective of this course is to discuss in depth a number of selected mathematical topics (and their use in image processing) that are of interest to the students at the moment the course is given.

CS 563 Image Analysis 3-3-0
Cross-listed with CS 463. Image analysis is concerned with the development of machine algorithms in order mimic the biological organism's ability to see and understand images and videos. The course content include: camera models and calibration, image enhancement, features extraction and representation, shape from shading, stereo and texture, optical flow, motion analysis, high level vision and case studies.

CS 564 Network Programming and Distributed Algorithms 3-3-0
Cross-listed with CS 464. The 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). The concept of distributed algorithms together with the associated challenges and examples are then presented. Programming distributed applications (in C or C++) is an integral part of the course. Students are expected to work on a large, distributed, and practically meaningful application as part of the course.

CS 565 Advanced Topics in Computer Applications 3-3-0
The course will present advanced topics of current interest or research directions in Computer Applications. The course content is expected to vary according to the interests of students and faculty. Students are expected to prepare a research paper during the course, or pursue an applied project.