

Faculty Courses

FES 102 Calculus I (3:3-0)

Prerequisite: None

Topic covered include: functions and their graphs; limits, continuity; differentiation and its applications; chain rule; Hospital's rule; minimum and maximum of a function; introduction to the anti-derivative.

FES 103 Calculus II (3:3-0)

Prerequisite: FES 102

Topic covered include: Integration, techniques of integration with applications such as: area, work, arc length, volumes; polar coordinates and area; improper integrals; series and tests for convergence, partial derivatives of a function of more than one variable.

FES 111 Programming I (3:3-1)

Prerequisite: Grade 12 Mathematics

Introduction to designing, implementing, and understanding computer programs using a modern programming language. Topics include fundamentals of problem solving, program design, algorithms, and programming using a high-level language. Introduction to OOP.

FES 112 Programming II (3:3-1)

Prerequisite: FES 111

This course covers subjects including: More OOP, basic concepts of data types (strings, arrays, records, sets, files); data structures (linked lists, stacks, queues, trees); searching and sorting algorithms.

FES 150 Natural Sciences (3:3-0)

Prerequisite: None

This course aims at providing students with fundamentals of chemistry, physics and biology, and widening their appreciation and conceptual understanding of the application of sciences and technologies for the development of modern life. Topics covered include: physical laws, depleting and renewable sources of energy, earth and the atmosphere, environmental pollution and its sources and consequences; concepts of chemical reactions, chemical compounds and their uses in our life, basics of heredity and gene expression, and the concept of biotechnology and its applications in contemporary life.

FES 202 Introduction to Statistics (3:3-0)

Prerequisite: None

This course is designed to provide students with an introduction to statistics: the science of collecting, organizing and interpreting data. This course enables students to understand important concepts in general statistics and contains basic topics such as descriptive statistics, probability, estimation, hypothesis testing, and linear correlation and regression, etc. The course will focus on the importance of different types of statistics and their application in life as well as drawing inferences about populations.

FES 204 Introduction to Discrete Structures (3:3-0)

Prerequisite: FES 102

Topics to be covered include: set algebra including mappings and relations; algebraic structures including semi-groups and groups; elements of the theory of directed and undirected graphs; Boolean algebra and propositional logic; these structures applied to various areas of computer science.

FES 205 Differential Equations and Linear Algebra (3:3-0)

Prerequisite: FES 103

Complex numbers; matrices and basic matrix operations; system of linear equations; determinants; eigenvalues and eigenvectors; introduction to differential equations such as: first order and higher order differential equations; system of differential equations.

FES 232 Physical Science I (3:3-3)

Prerequisite: FES 102

Topic covered include: Simple harmonic motion; wave motion, sinusoidal waves, phase velocity, Huygens' Principle, resonance, reflection, refraction and interference; sound; the classic Doppler effect; ray and first order matrix optics, total internal reflection and dispersion; the electromagnetic spectrum; optical spectra and electronic structure; de Broglie waves; principles and applications of nuclear structure, nuclear reactions and ionizing radiation.

FES 233 Physical Science II (3:3-3)

Prerequisite: FES 102

Vectors, Review of Kinematics, particle dynamics, curvilinear motion, momentum, angular momentum, energy.

Program Courses

CSC 225

Advanced Data Structures & Algorithm Analysis (3:3-1)

Prerequisite: FES 112

Random access machine model. Time and space complexity, average and worst case analysis, upper and lower bounds. Application of correctness proof techniques. Algorithms: internal searching, merging, sorting, selection, hashing; graphs: traversals, topological sort, transitive closure, strongly connected components, shortest path, minimum spanning tree. The existence of intractable problems, heuristics. Data structures: B-trees, heaps and graphs.

CSC 230

Computer Organization & Assembly Language (3:3-1)

Prerequisite: FES 112, FES 204

Topics covered include: instruction sets, symbolic addressing, bus organization, instruction fetch and execution, read/write cycles, interrupt processing, I/O processing, general microprocessor design. Assembly language programming.

CSC 327

Fundamental of Database Systems (3:3-1)

Prerequisite: FES 112

Topics to be covered include: data entities and relationships; data modeling using Entity-Relation Diagrams: hierarchical, network and relational models of databases; query languages; physical representation of data in secondary storage; relational algebra and calculus as applied to the design of databases; security and integrity in the context of concurrent use; and basic ethical issues associated with database design and use.

CSC 330

Computer Architecture (3:3-0)

Prerequisite: CSC 230, FES 202

This course provides an overview of the architectures of computers. The focus is on instructions set driven architectures, computer systems design and approaches. Memory management and control is also presented.

CSC 350

Computer Ethics (3:3-0)

Prerequisite: SWE 265

This course involves social issues, such as access rights, working place monitoring, censorship and junk mail; professional issues such as professional responsibility and code of conduct; legal issues such as legal obligations, data protection, computer misuse and software piracy.

CSC 360

Computer Networks (3:3-1)

Prerequisite: FES 202, CSC 225

Topics Will Include Layered Network Architectures, Packet Switching Networks, Local area networks, protocol design and verification, network security, and applications in distributed computing.

CSC 370

Operating Systems (3:3-1)

Prerequisite: CSC 330, FES 202, CSC 225

An introduction to the major concepts of operating systems and study of the interrelationships between the operating system and the architecture of computer systems. Topics discussed include operating system structures, concurrent programming techniques, CPU scheduling, deadlocks, memory management, file systems and protection.

CSC 371

Web Programming and Technologies (3:3-1)

Prerequisite: CSC 327

The course will focus on the design and development of web based applications using a number of currently popular tools and strategies; also to be explored is the use of databases as data repositories for web applications. Topics to be examined include: introduction to html, server side and client side scripting, database connectivity, scalability, security and other web technologies.

CSC 425

Theory of Computing (3:3-0)

Prerequisite: CSC 225, FES 204

A survey of formal models and results that form the theoretical foundations of computer science; typical topics include finite automata, Turing machines, un-decidable problems, context free languages and computational complexity.

CSC 429

Computer & Network Security (3:3-1)

Prerequisite: FES 205, CSC 225, CSC 360

Topics include an overview of computer and network security, protection, disaster planning, and recovery. Risk analysis and security plans. Basics of cryptography. Public key cryptography and protocols. Security models, kernel design and systems testing. Database, network and Web security. The course discusses applications which need various combinations of confidentiality, availability, integrity and covertness properties; mechanisms to incorporate these properties in systems. Policy and legal issues are also covered.

SWE 265

Introduction to Software Engineering (3:3-0)

Prerequisite: FES 112

Tools and techniques to promote programming productivity and software quality. Topics include specifications, code review and inspection techniques, testing and debugging methods and tools, reusable software components and templates, file system navigation, scripting languages, software configuration management, software tools, environments, and instrumenting and profiling.

SWE 310

Human Computer Interfaces (3:3-1)

Co-requisite: SWE 265, FES 112

Understanding human behavior 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 modeling techniques and their application to designing interfaces. Interface representation and prototyping tools. Cognitive walkthroughs, usability studies and verbal protocol analysis. Case studies of specific user interfaces.

SWE 320

Software Project management (3:3-0)

Prerequisite: SWE 265

The course provides the essential knowledge, processes and tools required for students to successfully direct and oversee a software development project. Topics include planning, leading, organizing, estimating, directing, monitoring and controlling software projects and their teams. Quantitative progress measures, software lifecycles, estimating and risk management are emphasized throughout the course.

SWE 362

Software Design and Architecture (3:3-0)

Prerequisite: SWE 265

The course discusses the analysis, selection, and evaluation of an appropriate architecture; architectural tactics and architectural patterns; documenting a software architecture; identifying and scheduling architectural releases; identifying constraints on and enablers for architectural decisions; and analyzing architectural trade-offs.

SWE 421

Software Engineering Requirements & Specifications (3:3-0)

Prerequisite: SWE 265, FES 204

Combines a range of topics integral to the design, implementation, and testing of a medium-scale software system with the practical experience of implementing such a project as a member of a programmer team. Introduces formal methods, requirements engineering, specifications, software life cycle models.

SWE 425

Software Testing and Measurement (3:3-0)

Prerequisite: SWE 421

This course is concerned with the principles underlying the development of software that needs to meet specific external goals, where these goals need to be expressible in measurable terms. It covers the principles of software measurement, and the ways in which they are used in planning software projects, and in monitoring how well projects are being carried out. It also examines the methods used for software testing and discusses the application of software measurement techniques to these testing methods.

SWE 426

Software Quality Control(3:3-0)

Prerequisite: SWE 421

The students will learn how to choose appropriate quality goals and select, plan, and execute quality assurance activities throughout development, evolution and testing to predictably meet quality and schedule goals. They will learn how quality assurance can be incorporated into process improvement feedback loops that amplify the ability of an organization to cost-effectively prevent and detect faults.

SWE 498, SWE 499

Capstone Project-1 and 2 (3:3-1)

Prerequisite: 4th Year standing in Faculty

The student is required to pursue an independent project under the supervision of a faculty member, to prepare a written report and present a seminar describing the work. This is the first semester of a 2-semester effort by the students.

ELECTIVES

CSC 436

Advanced Programming (3:3-1)

Prerequisite: 4th Year standing in Faculty

This course is designed to introduce students to the advanced concepts in application development. Topics include: system level programming, multithreading and multiprocessing, network programming, advanced web programming, and enterprise computing.

CSC 455

Computer Graphics (3:3-1)

Prerequisite: 4th Year standing in Faculty

The fundamental algorithms and data structures used in generative computer graphics. Topics discussed include structure of interactive graphics programs, raster algorithms, colour, two-dimensional and three-dimensional geometric transformations, animation, parallel and perspective projection, hidden line and hidden surface algorithms, cubic curves and surfaces, and shading models. Students will use high resolution raster display workstations, and other graphical devices.

CSC 460

Programming Languages & Compilers (3:3-1)

Prerequisite: 4th Year standing in Faculty

The core requirement for the course is the construction of a compiler for a simple but nontrivial language of the ALGOL family with nested scope and heap-allocated records, supplemented by written exercises, and exams. Important techniques such as: abstract syntax trees to avoid dangling syntax with semantics, separation of instruction selection from register allocation, sophisticated copy propagation to allow greater flexibility to earlier phases of the compiler, and careful containment of target-machine dependencies to one module. Some additional concepts to be covered include: garbage collection, compilation of object-oriented and functional languages, dataflow analysis, loop optimizations, parser error recovery, code-generator generators, byte-code interpreters, static single-assignment form, instruction scheduling and software pipelining, parallelization techniques, and cache-locality optimizations such as pre-fetching, blocking, instruction-cache layout, and branch prediction.

CSC 461

Object Oriented Programming & C++ (3:3-1)

Prerequisite: 4th Year standing in Faculty

This course is an in-depth look at Object-oriented programming using C++ as the method for learning. The course reviews methods of programming, OOP and related concepts, C++ and OO within C++, and relations between C++ and C.

CSC 462

Object Oriented Design Patterns (3:3-1)

Prerequisite: 4th Year standing in Faculty

This course provides a number of distinct object oriented design patterns which help students formulate programs and concepts in programming. These design patterns fit into the software engineering process to create reliable, consistent and maintainable software.

CSC 463

Introduction to OOP Using JAVA (3:3-1)

Prerequisite: 4th Year standing in Faculty

This course provides an introduction to Object-Oriented Programming and Software Development with Java. Subjects covered include: Numerical Data, Classes, Selection Statements, Repetition Statements, Exceptions and Assertions, Characters and Strings, Arrays, Sorting and Searching, File Input and Output, Inheritance and Polymorphism, GUI and Event-Driven Programming, Recursive Algorithms

CSC 475

Analysis of Algorithms (3:3-1)

Prerequisite: 4th Year standing in Faculty

General techniques for designing and analyzing algorithms; an in-depth examination of several problems and algorithms with respect to their time and space requirements; advanced data structures; sorting and searching; graph algorithms; backtracking; NP-complete problems; approximation algorithms.

CSC 480

Numerical Methods (3:3-1)

Co-requisite: 4th Year standing in Faculty

The course will cover computer-oriented methods for solving numerical problems in science and engineering, including finding roots (including iterative solutions such as Newton's method), interpolation and approximation (including curve fitting), numerical integration and differentiation (including Simpson's rule), solving linear equations (including linear algebra), solving differential equations (including Euler's method), and, possibly, optimization and simulation. It will also cover numerical analysis, i.e., the mathematics and theory behind the numerical methods (including floating point arithmetic, errors, stability, convergence, Taylor's series, explicit and implicit methods, and possibly, finite differences), plus a short discussion of history leading to the current interest in scientific computing.

SWE 451

Multi-Media Applications (3:3-1)

Prerequisite: 4th Year standing in Faculty

Topics include: digital video representations; three-dimensional images; physical interfaces; computational tools and media that help people learn new things in new ways (tele-learning); knowledge representation; machine interpretation of sensory data.

SWE 462

Distributed Systems (3:3-1)

Prerequisite: 4th Year standing in Faculty

The exponential growth of Internet services demonstrates the importance and potential of large-scale distributed systems. Concepts covered include: Content delivery networks can potentially speed these services by cleverly caching Web pages; Peer-to-peer applications allow sharing of content; Multimedia services provide streaming delivery of audio and video. The new classes of distributed applications that are becoming ubiquitous seems endless: cluster computing, grid computing, game services, pervasive computing, etc. A fundamental challenge is to provide scalable, secure and robust services in the presence of best-effort communication and unreliable nodes.

SWE 471

Software Evolution (3:3-0)

Prerequisite: 4th Year standing in Faculty

Topics include software engineering techniques for programming-in-the-large, programming-in-the-many, legacy software systems, software architecture, software evolution, software maintenance, reverse engineering, program understanding, software visualization, advanced issues in object-oriented programming, design patterns, antipatterns, and client-server computing. This course culminates in a team project.

SWE 475

Software Systems Development (3:3-0)

Prerequisite: 4th Year standing in Faculty

Software engineering, or software systems development, is concerned with problems relating to the application of sound engineering principles to the production of quality software. This course involves a broad range of topics including: theory and examples, history of software development, current methods, human issues, and computer aided software engineering (CASE), as well as modern approaches such as object-oriented development, metrics and cost estimation, process considerations, and new applications such as software reuse.

SWE 484

Artificial Intelligence (3:3-1)

Prerequisite: 4th Year standing in Faculty

The focus of this course is on an intelligent agent acting in an environment. Introductory concepts such as simple agents acting in simple, static environments and gradually increase the power of the agents to cope with more challenging worlds. For simple agents in simple worlds we start with a highly restricted simple logic. Then as the agent/environment requires, the logical power of the formalism is increased. LISP or Prolog, can be used to support these implementations.

SWE 485

Pattern Recognition (3:3-1)

Prerequisite: 4th Year standing in Faculty

Parallel and sequential recognition methods. Bayesian decision procedures, perceptrons, statistical and syntactic approaches, recognition grammars. Applications to be considered include: feature extraction and selection, scene analysis, and optical character recognition.

SWE 486

Robotics (3:3-1)

Prerequisite: 4th Year standing in Faculty

This course provides a good introduction to all issues in mobile robotics, from locomotion to robot perception, hardware, sensors, planning, reasoning and poise. The focus is on the computational aspects of robotics.