Concordia University

GINA CODY
SCHOOL OF ENGINEERING
AND COMPUTER SCIENCE

Department of Computer Science and Software Engineering (CSSE)

Minor in Computer Science (25 credits)

Basic computer science skills are the perfect complement to any degree program at Concordia. The Minor in Computer Science is designed to meet the growing demand for computer-literate professionals, and may offer students more career opportunities after graduation.

This minor must be combined with a major, specialization or honours program. Generally, it is only available to students enrolled in degree programs outside of the Gina Cody School of Engineering and Computer Science.

Admission criteria:

- Students requesting a Minor in Computer Science must have completed MATH 203, 204 and 205 or the equivalent(s) with a minimum grade of C- prior to applying.
- Approval for adding a Minor in Computer Science is based on academic performance. (i.e. an annual/cumulative GPA and grades in relevant subjects.)

Students in Computer, Electrical and Software Engineering are **not eligible to take the Minor in Computer Science*

Program structure

Undergraduate Calendar, section 71.70.5 Minor in Computer Science

Minor in Computer Science: 25 credits		*Credits*
COMP 228	System Hardware	3.00
COMP 232	Mathematics for Computer Science	3.00
COMP 248	Object-Oriented Programming I	3.50
COMP 249	Object-Oriented Programming II	3.50
COMP 352	Data Structures and Algorithms	3.00
	Computer Science Electives (see §71.70.2)	9.00

Students who require any of the above courses (or equivalents) as part of their major should replace these courses with elective courses chosen from the list of Computer Science Electives.

How to apply:

Please complete and submit the **Add/Drop a minor form** with the Student Academic Services Office for the Gina Cody School (EV 2.125, sas-front-desk@encs.concordia.ca).

Course Descriptions

Undergraduate Calendar, section 71.70.10

COMP 228 *System Hardware* (3 credits)

Prerequisite: COMP 248; MATH 203 or Cegep Mathematics 103 or NYA previously or concurrently; MATH 204 or Cegep Mathematics 105 or NYC previously or concurrently. Levels of system abstraction and von Neumann model. Basics of digital logic design. Data representation and manipulation. Instruction set architecture. Processor internals. Assembly language programming. Memory subsystem and cache management. I/O subsystem. Introduction to network organization and architecture. Lectures: three hours per week. Tutorial: two hours per week.

NOTE: Students who have received credit for SOEN 228 may not take this course for credit.

COMP 232 *Mathematics for Computer Science* (3 credits)

Prerequisite: MATH 203 or Cegep Mathematics 103 or NYA; MATH 204 or Cegep Mathematics 105 or NYC. Sets. Propositional logic and predicate calculus. Functions and relations. Elements of number theory. Mathematical reasoning. Proof techniques: direct proof, indirect proof, proof by contradiction, proof by induction. Lectures: three hours per week. Tutorial: two hours per week.

NOTE: Students who have received credit for COMP 238 or COEN 231 may not take this course for credit.

COMP 248 *Object-Oriented Programming I* (3.5 credits)

Prerequisite: MATH 204 or Cegep Mathematics 105 or NYC previously or concurrently. Introduction to programming. Basic data types, variables, expressions, assignments, control flow. Classes, objects, methods. Information hiding, public vs. private visibility, data abstraction and encapsulation. References. Arrays. Lectures: three hours per week. Tutorial: two hours per week. Laboratory: one hour per week.

COMP 249 *Object-Oriented Programming II* (3.5 credits)

Prerequisite: COMP 248; MATH 203 or Cegep Mathematics 103 or NYA; MATH 205 or Cegep Mathematics 203 or NYB previously or concurrently. Design of classes. Inheritance. Polymorphism. Static and dynamic binding. Abstract classes. Exception handling. File I/O. Recursion. Interfaces and inner classes. Graphical user interfaces. Generics. Collections and iterators. Lectures: three hours per week. Tutorial: two hours per week. Laboratory: one hour per week.

COMP 352 *Data Structures and Algorithms* (3 credits)

Prerequisite: COMP 232 previously or concurrently; COMP 249. Abstract data types: stacks and queues, trees, priority queues, dictionaries. Data structures: arrays, linked lists, heaps, hash tables, search trees. Design and analysis of algorithms: asymptotic notation, recursive algorithms, searching and sorting, tree traversal, graph algorithms. Lectures: three hours per week. Tutorial: one hour per week.

NOTE: Students who have received credit for COEN 352 may not take this course for credit.

⚠ All 200-level courses within the program which are prerequisites for other courses must be completed with a C- or higher. A 200-level course in which a student has obtained a D+ or lower must be repeated before attempting a course for which it is a prerequisite.