Michigan Technological University courses taken (STEM)
Joseph Halcombe

Computer Science

CS 1121 - Introduction to Programming I
Starting point of the computer science programs. A high-level, object-oriented programming language is introduced as a problem-solving tool. Topics include design, coding, documentation, debugging, and testing of programs. Programming assignments are given in both a closed lab setting and as homework.
Credits: 3.0

CS 1122 - Introduction to Programming II
Continuation of CS 1121. Topics include data abstraction, class hierarchies and polymorphism, list, stack, queue and tree data structures, complexity-based algorithm and data structure choices, and recursion. Homework programming assignments are given.
Credits: 3.0

CS 1142 - Programming at the Hardware Software Interface
Programming in assembly language and C for students with prior experience in Java. Topics include binary number encodings, instruction set architecture, assembly language programming, and instruction encodings. C programming topics include program structure, preprocessor, arrays, structures, pointers, input/output, dynamic memory management, and linked data structures.
Credits: 3.0

CS 2311 - Discrete Structures
Presents fundamental concepts in discrete structures that are used in computer science. Topics include sets, trees, graphs, functions, relations, recurrences, proof techniques, logic, combinatorics, and probability.
Credits: 3.0

CS 2321 - Data Structures
Presents fundamental concepts in data structures. Topics include abstract data types (priority queues, dictionaries and graphs) and their implementations, algorithm analysis, sorting, text processing, and object-oriented design. A significant programming project is assigned.
Credits: 3.0

CS 3000 - Ethical and Social Aspects of Computing
An examination of social and ethical issues associated with computing. Topics include: ethical theories and decision making, intellectual property, freedom of expression, privacy, security, and professional responsibility.
Credits: 3.0

CS 3141 - Team Software Project

This course focuses on software development as a team. It covers software design models emphasizing process activities including Agile methodologies and Secure Software Development Life Cycle practices. Key topics include version control, automated testing, and documentation. Students will develop skills in communication, teamwork.
Credits: 3.0

CS 3311 - Formal Models of Computation
Introduction to the theory of formal languages and computation. Topics include regular languages and finite automata, context-free languages and push-down automata, Turing-acceptable languages, Turing machines and the halting problem. Proof techniques and applications, such as parsing, are also treated.
Credits: 3.0

CS 3331 - Concurrent Computing
Concepts and techniques in concurrent computing. Topics include: processes and threads, mutual exclusion, semaphores, monitors and condition synchronization, deadlock, safety and liveness, message passing, and concurrent architectures.
Credits: 3.0

CS 3411 - Systems Programming
Development of robust programs that provide efficient services to system software developers. Topics include: file I/O, process creation and management, linking and libraries, interprocess communication, performance measurement, and socket programming.
Credits: 3.0

CS 3421 - Computer Organization
Introduction to the logical structure of computers, including the fundamentals of logic design, information storage and manipulation, control, and input/output. Topics include a review of current hardware technology, combinational and sequential logic, arithmetic, datapaths, hard-wired control, interrupts, caches, virtual memory, and an introduction to pipelining.
Credits: 3.0

CS 3425 - Introduction to Database Systems
This course provides an introduction to database systems including database design, query, and programming. Topics include goals of database management; data definition; data models; data normalization; data retrieval and manipulation with relational algebra and SQL; data security and integrity; database and Web programming; and languages for representing semi-structured data.
Credits: 3.0

CS 3712 - Software Quality Assurance
This course concentrates on ensuring quality through the software process including definition, analysis, and measurement of quality attributes. Topics are software testing, static analysis,

code review, process improvement and security engineering emphasizing derivation of test cases from requirements specifications and writing test plans.
Credits: 3.0

CS 4099 - Directed Study in Computer Science
Students study one or more special topics in computer science under the direction of one or more faculty members.
Credits: 6.0 (2 Semesters of research)

CS 4121 - Programming Languages
A discussion of the concepts underlying programming languages. Topics include programming paradigms; language theory and properties (including syntax, semantics, run-time behavior, and implementation issues); data, procedure, functional and control abstraction; functional programming, logic programming, and language security.
Credits: 3.0

CS 4321 - Introduction to Algorithms
Fundamental topics in algorithm design, analysis, and implementation. Analysis fundamentals include asymptotic notation, analysis of control structures, solving recurrences, and amortized analysis. Design and implementation topics include sorting, searching, and graph algorithms. Design paradigms include greedy algorithms, divide-and-conquer algorithms, and dynamic programming.
Credits: 3.0

CS 4711 - Software Processes and Management
Focuses on the software development process and related management issues. Topics include software process models, the Capability Maturity Model, process tools, use of standards, software maintenance, configuration management, project planning and tracking, team management, and measurement and estimation.
Credits: 3.0

CS 4760 - User Interface Design and Implementation
Principles of user interfaces (UI) design and implementation. Topics include: UI theory, design principles, evaluation, and tools. Requires completion of a group project implementing and evaluating a UI.
Credits: 3.0

Mathematics

MA 1032 - Precalculus
This course examines the behavior of linear, polynomial, rational, exponential, logarithmic and trigonometric functions.
Credits: 4.0

MA 1160 - Calculus with Technology I

An introduction to single-variable calculus, which includes a computer laboratory. Topics include trigonometric, exponential, and logarithmic functions, differentiation and its uses, and basic integration. Integrates symbolic tools, graphical concepts, data and numerical calculations.
Credits: 4.0

MA 2330 - Introduction to Linear Algebra
An introduction to linear algebra and how it can be used, including basic mathematical proofs. Topics include systems of equations, vectors, matrices, orthogonality, subspaces, and the eigenvalue problem.
Credits: 3.0

MA 3710 - Engineering Statistics (*Taken at Gogebic Community College)
Introduction to the design, conduct, and analysis of statistical studies aimed at solving engineering problems. Topics include methods of data collection, descriptive and graphical methods, probability and probability models, statistical inference, control charts, linear regression, design of experiments.
Credits: 3.0

MA 3720 - Probability
Introduction to probabilistic methods. Topics include probability laws, counting rules, discrete and continuous random variables, expectation, joint distributions, and limit theorems.
Credits: 3.0

Technical Electives

SAT 3812 - Cyber Security I
The evolution of information security into cybersecurity and its relationship to nations, organizations, society, and individuals. Exposure to multiple cybersecurity technologies, processes, and procedures; analyzing threats, vulnerabilities and risks present; and developing appropriate strategies to mitigate potential cybersecurity issues. Applied lab to develop cyber security offensive attributes and learn how to prevent and/or mitigate threats.
Credits: 3.0