Computer Science

Cliburn, Collins, Wahl.

Major: Computer Science courses – 121; 122; 223; 234; 335; 336; 461 or 471 (culminating experience); three others above 300 level.
Cognate courses – Mat 220; Mat 343; Mat 122 or Mat 224
Comprehensive evaluation, with passing grade. Total of 10 major courses, plus 3 cognates = 13.

Minor: Computer Science courses – 121; 223 or 234; 336 or 345; two other CS units, or Mat 220 and one other CS unit. CS 205 may be taken twice (with different languages) for a unit towards the minor. CS 111 may count as an elective toward the minor. Total of 5 minor courses.

CS 111. Fundamental Concepts of Computer Science. An introductory survey of computer science topics, including computer organization and architecture, operating systems, formal language theory, program design and verification, visualization, and artificial intelligence. Includes a laboratory component involving significant hands-on experience with both software and hardware. Suitable for non-majors. Satisfies General Degree Requirement III.C, and the Abstraction and Formal Reasoning LADR. Not open to students with prior credit in CS 121.

CS 121. Programming I. Preliminary survey of selected areas within computer science, with emphasis on problem solving and programming in a commonly used programming language.

CS 122. Programming II. The design, implementation, verification, and validation of programs; includes software design, requirements and specification analysis, efficiency of algorithms, input-output analysis, documentation techniques, searching and sorting, basic complex data structures, recursive techniques, and testing strategies. Prerequisite: 121.

CS 160. Special Topics.

CS 205. Programming Practicum. Study and use of a common programming language other than that used in the introductory sequence. May be repeated (with different languages) for credit. .50 unit.

CS 223. Data Structures and Algorithm Analysis. Implementation of various advanced data structures; analysis of the algorithms associated with the various implementations, algorithm complexity, and time-space trade-offs. Prerequisites: 122 and Mat 220.


CS 260. Special Topics.

CS 322. Object-Oriented Software Design. Team planning and execution of a large programming project. Prerequisite: 122.

CS 329. Database Systems. Introduction to fundamental concepts of database management systems, including database design, data definition languages, data manipulation languages, and database system implementation. Particular emphasis will be placed on the now widely used relational database model and SQL language. Prerequisite: 122.

CS 330. Human Factors. Application of fundamental principles of human mental and behavioral processes to the design of more useful and comfortable equipment, computers, and workspaces. Includes design exercises focusing on human factors. Identical to Psy 330. Prerequisites: Psy 111 and completion of General Degree Requirement III.A, or permission of the instructor.

CS 335. Theory of Computation. Finite automata, Turing machines, regular and context-free
languages, computability and solvability, the halting problem, Church’s thesis. Prerequisite: 122.

**CS 336. Programming Languages.** A comparison of design features for selected procedural, functional, logical, or string oriented languages, syntax and semantics, assignment, control structures, data types, procedures and parameter passing, nesting and scoping. Prerequisite: 122.

**CS 345. Operating Systems.** Theory of operating systems, including memory management, concurrent processes, scheduling algorithms, device drivers, and file systems. Prerequisite: 234.

**CS 346. Compiler Design.** Design and implementation of a high-level language compiler, including syntactic analysis, symbol table generation, parsing, error recovery, code generation, code optimization. Prerequisite: 223.

**CS 348. Artificial Intelligence.** Knowledge representation, natural language processing, models of reasoning and learning, heuristic search methods, connectionism, and robotics. Prerequisite: 223.

**CS 349. Computer Graphics.** A study of algorithms used to control computer displays, including algorithms for drawing lines and circles and for rendering surfaces. Prerequisite: 223.

**CS 357. Internship.** Off-campus supervised experience in computer science.

**CS 360. Special Topics.**

**CS 370. Directed Study.**

**CS 461. Senior Seminar.** Seminar on selected research topics.

**CS 471. Independent Study.**