Computer Science Course Descriptions

CS 1115  Computer Science I Laboratory  (1 semester hour) Optional laboratory course for CS 1315. This course teaches basic computer literacy/programming skills: disk operating system (DOS) commands (to format disks and to create, manipulate, and remove directories and files), the authoring of ASCII text files, compiler usage in converting source programs into executable form, printer commands. CS 1315 students without prior computer programming experience should enroll in this laboratory. (0-2) S

CS 1315 (COSC 1315)  Computer Science I  (3 semester hours) Computer programming in a high-level, block structured language. Algorithmic thinking and the history and utility of machines which automate it. Basic data types and variables, memory usage, control structures (sequential, selection, repetition), functions and parameter passing, recursion, console and file input/output. Prerequisite: Basic computer literacy/programming skills (see CS 1115 description) or concurrent enrollment in CS 1115. (3-0) S

CS 2305 (MATH 2305)  Discrete Mathematics for Computing I  (3 semester hours) Principles of counting. Boolean operations. Propositional calculus. Sets, relations, functions, strings, languages, partial orders, and lattices. Prerequisite: MATH 1326 or MATH 2417 or consent of the instructor. (3-0) S

CS 2315 (COSC 2315)  Computer Science II  (3 semester hours) Advanced programming techniques, including an introduction to object-oriented programming. Classes, inheritance, dynamic function binding, strings, stacks, queues, lists, and trees. Dynamic memory allocation/management. Prerequisite: CS 1315. (3-0) S

CS 2325 (COSC 2325)  Computer Organization  (3 semester hours) The composition of central processing units and its impact on low-level programming. The study of computers as symbolic processors; the nature and manipulation of the symbols. Assembly language programming. Prerequisite: CS 2315. (3-0) S

CS/SE 2V95  Individual Instruction in Computer Science  (1-6 semester hours) Individual study under a faculty member's direction. May be repeated for credit. Consent of instructor required. ([1-6]-0) R

CS/SE 3195  Special Topics in Computer Science/Software Engineering  (1 semester hour) May be repeated for credit (4 hours maximum). Must be taken Credit/No Credit. Consent of instructor required. (1-0) R

CS 3305  Discrete Mathematics for Computing II  (3 semester hours) Topics in enumeration; principle of inclusion and exclusion. Algorithmic complexity; recurrence relations. Graph theory. Prerequisite: CS 2305. (3-0) S

CS 3333  Data Structures  (3 semester hours) Programming with basic data structures (arrays, stacks, queues, lists, and trees) and their associated algorithms. Various sorting and searching techniques. Fundamental graph algorithms. This course covers much of the same material as CS 3345 without requiring the analysis of algorithms. Computer Science majors may NOT take this course; an individual transferring to Computer Science who has already completed this course may substitute this course for CS 2315 in the Computer Science degree plan. This course may not be taken for degree credit by students who have completed CS 2315 (C/C++). Prerequisite: CS 1315 (C/C++) or CS 3335 or equivalent programming experience, including knowledge of C. Corequisite: It is recommended that students with minimal prior programming experience also enroll in CS 3133. (3-0) Y

CS 3335  C and C++  (3 semester hours) Numerous programming projects in both C and C++. All fundamentals of C, with special emphasis on use of pointers. Use of C++ extensions to create and extend (by inheritance) abstract data types. The use/advantages of virtual functions (dynamic polymorphism). This course may not be taken for degree credit by students who have completed CS 2315 (C/C++) or CS 3333. Prerequisite: CS 2315 (in a language other than C/C++) or equivalent programming experience. (3-0) S

CS 3336  Programming in JAVA  (3 semester hours) Overview of object-oriented programming. Implementation of object-oriented designs using the JAVA programming environment. Emphasis on various aspects of interactive and internet programming. Prerequisite: CS 2315 (C/C++) or CS 3333 or CS 3335 or equivalent programming experience. (3-0) T

exponential and Poisson random variables. Illustrative examples and simulation exercises from queuing, reliability, and program analysis disciplines. Elements of parameter (point) estimation. Prerequisites: MATH 1326 or MATH 2419, and CS 2305 (3-0) S

**CS/SE 3345 Algorithm Analysis and Data Structures** (3 semester hours) Metrics for performance evaluation of algorithms. Formal treatment of basic data structures such as arrays, stacks, queues, lists, trees. Various sorting and searching techniques. Fundamental graph algorithms. Prerequisites: CS 2315 and CS 3305. (3-0) S

**CS/SE 3354 Software Engineering** (3 semester hours) Introduction to software life cycle models. Software requirements engineering, formal specification and validation. Techniques for software design and testing. Cost estimation models. Issues in software quality assurance and software maintenance. Prerequisites: CS 2315 or CS 3333, and CS 2305 (3-0) S

**CS 3385 Ethics, Law, Society, and Computing** (3 semester hours) Issues of professional ethics; computer crime; wiretapping and encryption; protecting software and other intellectual property; privacy and information; careers and computers; reliability and safety; constitutional issues. Broader issues on the impact and control of computers. (3-0) S

**CS/SE 3390 Technical Writing** (3 semester hours) This course trains students to develop technical communications skills required by computer professionals and computer science researchers. The course satisfies the Advanced Writing component of core curriculum requirements. (3-0) S

**CS/SE 3V95 Undergraduate Topics in Computer Science/Software Engineering** (2-9 semester hours) Subject matter will vary from semester to semester. May be repeated for credit (9 hours maximum). ([2-9]-0) S

**CS/SE 4140 Computer Architecture Laboratory** (1 semester hour) Laboratory for CS 4340. Must be taken concurrently with CS 4340. Must be taken Credit/No Credit. (0-1) S

**CS 4334 Numerical Analysis** (3 semester hours) Solution of linear equations, roots of polynomial equations, interpolation and approximation, numerical differentiation and integration, solution of ordinary differential equations, computer arithmetic, and error analysis. Prerequisites: CS 1315, MATH 2418, MATH 2419. (Same as MATH 4334.) (3-0) Y

**CS 4337 Organization of Programming Languages** (3 semester hours) Language definition structure, data types and structures, control structures and data flow, run-time considerations. Interpretive languages; functional programming. Prerequisites: CS 2315 or CS 3333, and CS 2305. (3-0) S

**CS 4340 Computer Architecture** (3 semester hours) Boolean algebra and logic circuits; register transfer operations; design of a small computer; input, output, and interrupt organization; powerful addressing modes, instruction formats, and their hardware structures; microprogram control. Must be taken concurrently with CS 4140. Prerequisites: CS 2305 and CS 2325. (3-0) S

**CS/SE 4347 Database Systems** (3 semester hours) This course emphasizes the concepts and structures necessary for the design and implementation of database management systems. Topics include data models, data normalization, data description languages, query facilities, file organization, index organization, file security, data integrity, and reliability. Prerequisite: CS 3345. (3-0) Y

**CS/SE 4348 Operating Systems Concepts** (3 semester hours) An introduction to fundamental concepts in operating systems: their design, implementation, and usage. Topics include process management, main memory management, virtual memory, I/O and device drivers, file systems, secondary storage management, and an introduction to critical sections and deadlocks. Prerequisites: CS 3345, CS 4340, and a working knowledge of C and UNIX. (3-0) S

**CS 4349 Advanced Data Structures and Algorithms** (3 semester hours) Height balanced trees, B-trees, and other techniques for efficient storage and retrieval of information. Algorithm design techniques such as greedy method, dynamic programming, and divide-and-conquer. Issues from computational complexity. Prerequisite: CS 3345. (3-0) S

**CS 4361 Computer Graphics** (3 semester hours) Review of graphic display architecture and graphic input devices. Two- and three-dimensional transformations, matrix formulations, and concatenation. Clipping and windowing. Data structures for graphics systems, segmented display files, rings, etc. Hidden line and surface elimination. Shading. Graphics packages and applications. Prerequisites: linear algebra, CS 2315, and CS 3345. (3-0) Y

**CS 4365 Artificial Intelligence** (3 semester hours) Basic concepts and techniques that enable computers to perform intelligent tasks. Examples are taken from areas such as natural language understanding, computer vision, machine learning, search strategies and control, logic, and theorem proving. Prerequisites: CS 2315 and CS 3345. (3-0) Y

**CS 4375 Principles of UNIX** (3 semester hours) Design and history of the UNIX operating system. Detailed study of process and file system data structures. Shell programming in UNIX. Use of process-forking functionality of UNIX to simplify complex problems. Interprocess communication and coordination. Device drivers and streams as interfaces to hardware
TCP/IP and other UNIX inter-machine communication facilities. Prerequisite: CS 2315 (C/C++) or CS 3333 or CS 3335 or equivalent programming experience, including knowledge of C. (3-0) S

**CS/SE 4376 Object-Oriented Programming Systems** (3 semester hours) In-depth study of the features/advantages of object-oriented approach to problem solving. Special emphasis on issues of object-oriented analysis, design, implementation, and testing. Review of basic concepts of object-oriented technology (abstraction, inheritance, and polymorphism). Object-oriented programming languages, databases, and productivity tools. Prerequisite: CS 2315 (C/C++) or CS 3333 or CS 3335 or equivalent programming experience, including knowledge of C++. (3-0) S

**CS 4380 Senior Design Project** (3 semester hours) Detailed design, implementation, and testing of a system or component under the guidance of a faculty member. Specific technical requirements will be specified by the individual faculty member teaching/supervising the course. All students must submit a written report and make an oral presentation at the culmination of the project. May be repeated for credit (6 hours maximum) to complete a two-semester project. Prerequisite: senior standing. (3-0) Y

**CS 4384 Automata Theory** (3 semester hours) A review of the abstract notions encountered in machine computation. Topics include finite automata, regular expressions, PDAs, and context-free languages. Prerequisite: CS 3305. (3-0) S

**CS 4386 Compiler Design** (3 semester hours) Basic phases of a compiler and their design principles. Topics include lexical analysis, basic parsing techniques such as LR(K) and LL(K) grammars. Prerequisites: CS 3345 and CS 4384. (3-0) T

**CS/TE 4390 Computer Networks** (3 semester hours) The design and analysis of computer networks. Topics include: the ISO reference model, transmission media, medium-access protocols, LANs, data link protocols, routing, congestion control, internetworking, and connection management. Prerequisite: CS/TE 3345. (Same as TE 4390) (3-0) S

**CS/SE 4V95 Undergraduate Topics in Computer Science/Software Engineering** (1-9 semester hours) Subject matter will vary from semester to semester. May be used as CS Guided Elective on CS degree plans. May be repeated for credit (9 hours maximum). ([1-9]-0) R

**CS/SE 4399 Senior Honors in Computer Science/Software Engineering** (3 semester hours) For students conducting independent research for honors theses or projects. (3-0) R