Classics 295 (3)—Topics in Classical Civilization
Selected subject areas in classical civilization. The topic selected varies from year to year. May be repeated for degree credit with permission and if the topics are different. *Staff.*
Offered when interest is expressed and departmental resources permit.

Classics 300 (History 300) (3)—Seminar in Ancient History
Prerequisites: Junior standing and permission of the instructor. A consideration of the major Greek and Roman historians, the influence of various literary and philosophical conventions on the development of their method, and their approach to selected problems in ancient history evaluated in the light of modern historical research. (HU, GE3: Literature, only if taken as Classics.) *Sanders.*
Winter 2010 and alternate years

Classics 401 (1), 402 (2), 403 (3)—Directed Individual Study
May be repeated for degree credit with permission and if the topics are different.
Offered when interest is expressed and departmental resources permit.

Classics 421 (1), 422 (2), 423 (3)—Directed Individual Research
May be repeated for degree credit with permission and if the topics are different.
Offered when interest is expressed and departmental resources permit.

Classics 473 (3)—Senior Thesis
Prerequisites: Senior standing, major in classics, and permission of the department. The student researches and writes a senior thesis under the direction of a faculty member.
Fall, Winter

Classics 493 (3-3)—Honors Thesis
Prerequisite: Honors candidacy.
Fall-Winter

COMPUTER SCIENCE (CSCI)

PROFESSORS **LAMBERT**, WHALEY
ASSOCIATE PROFESSORS LEVY, NECAISE
ASSISTANT PROFESSOR SPRENKLE

MAJORS

A major in **computer science** leading to a Bachelor of Arts degree requires completion of at least 41 credits, including the following:

1. Computer Science 111, 112, 209, 210, 211, 312, 313; Mathematics 121
2. Either Mathematics 102 or 122
3. Two courses chosen from Computer Science 315 through Computer Science 341
4. Six additional credits in computer science.

A major in **computer science** leading to a Bachelor of Science degree requires completion of at least 50 credits, including the following:

1. Computer Science 111, 112, 209, 210, 211, 312, 313; Mathematics 102, 121, 222
2. Two courses chosen from Computer Science 315 through Computer Science 341
3. Six additional credits in computer science
4. Six additional credits in mathematics at the 200 level or above.

An additional course required as a prerequisite for completion of the above is Mathematics 101.

In order that the discrete mathematics requirement for Computer Science 211, 312, and 313 be completed in a timely fashion, first-year students expecting to major in computer science are encouraged to take Mathematics 121 in their first year.

HONORS: An Honors Program in computer science is offered for qualified students; see the department head for details.

Computer Science 101 (4)—Survey of Computer Science
Not open to students with previous credit in computer science. An overview of the discipline of computer science achieved through an introductory-level survey of a number of major areas of computer science. Topics include algorithms used for computer solutions of important practical problems, computer programming, digital logic applied to computer circuitry, computer architecture, data representation and organization, Web page basics, computer networks, and theoretical limits of computation. Lectures and formal laboratories. (FM, GE5b) *Staff.*
Fall, Winter

Computer Science 111 (4)—Fundamentals of Programming I

A disciplined approach to programming with Python. Emphasis is on problem-solving methods, algorithm development, and object-oriented concepts. Lectures and formal laboratories. (FM, GE5b) *Staff.*

Fall, Winter

Computer Science 112 (4)—Fundamentals of Programming II

Prerequisite: Computer Science 111. A continuation of Computer Science 111. Emphasis is on the use and implementation of data structures, introductory algorithm analysis, and object-oriented design and programming with Python. Laboratory course. (SC, GE5c) *Staff.*

Fall, Winter

Computer Science 121 (4)—Scientific Computing

Not open to students who have taken Computer Science 211 or higher. An introduction to computer programming for scientific applications and a survey of the main methodological areas of scientific computation. The course provides the tools needed for students to use computers effectively in scientific work, whether in physics, chemistry, mathematics, economics, biology, psychology, or any field involving quantitative work. Programming in Matlab, a scientific-computing software package, with a focus on topics relevant to students' major fields of study. Lectures and formal labs. (FM, GE5b) *Levy*.

Winter

Computer Science 196 (3)—Special Topics in Computer Science for Non-Majors

Special topics of current or general interest chosen on the basis of interest to faculty and students. Topics taught vary from year to year and are announced in advance of registration. (SC, GE5c) *Staff.*

Offered when interest is expressed and departmental resources permit.

Computer Science 199 (3)—Special Topics in Computer Science for Non-Majors

Special topics of current or general interest chosen on the basis of interest to faculty and students. Topics taught vary from year to year and are announced in advance of registration. (This course does *not* satisfy FDR or GE requirements.) *Staff.*

Offered when interest is expressed and departmental resources permit.

Computer Science 209 (3)—Software Development

Prerequisite: Computer Science 112. An examination of the theories and design techniques used in software development. Topics include the software life cycle, design patterns, the Unified Modeling Language, unit testing, refactoring, rapid prototyping, and program documentation. *Lambert.*

Fall

Computer Science 210 (3)—Computer Organization

Prerequisite: Computer Science 111. Multilevel machine organization studied at the levels of digital logic, microprogramming, conventional machine, operating system, and assembly language. *Staff.*

Fall

Computer Science 211 (3)—Algorithm Analysis

Prerequisites: Computer Science 112 and Mathematics 121. Computer representations of data structures as derived from mathematical models: stacks, queues, and linked lists. Access methods into data structures, such as hashing and tree searching. Storage allocation and deallocation procedures. Algorithms for manipulating such structures are introduced, analyzed, and implemented by the student. *Staff.*

Winter

Computer Science 250 (3)—Introduction to Robotics

Prerequisite: Computer Science 111 or 121 or permission of the instructor. This course combines readings from the contemporary robotics literature with hands–on lab experience building robots with the popular Lego Mindstorms toolkit (provided). The lab experience culminates with a peer–judged competition of robot projects proposed and built during the second half of the course. (SC, GE5c) *Levy.*

Offered when interest is expressed and departmental resources permit.

Computer Science 252 (3)—Neural Networks and Graphical Models

Prerequisite: Computer Science 112. A survey of the major developments in neural and belief networks, from the early perceptron models of the 1940s through the probabilistic Bayesian networks that are a "hot topic" in artificial intelligence today. Topics include the back-propagation algorithm, simple recurrent networks, Hopfield nets, Kohonen's Self-Organizing Map, learning in Bayesian networks, and Dynamic Bayesian Networks, with readings from both popular textbooks and the scholarly literature. A major focus of the course is on writing programs to implement and apply these algorithms. *Levy.*

Offered when interest is expressed and departmental resources permit.

Computer Science 253 (3)—Genetic Algorithms

Prerequisite: Computer Science 112. A survey of the major developments in genetic/evolutionary algorithms, from the Simple Genetic Algorithm through modern multiobjective optimization methods. Topics include fitness landscapes, the Schema and Building Block Hypotheses, learning and the Baldwin Effect, and genetic programming, with readings from both popular textbooks and the scholarly literature. A major focus of the course is on writing programs to implement and apply these algorithms. *Levy.*

Offered when interest is expressed and departmental resources permit.

Computer Science 295 (1-3)—Language Laboratory

Prerequisite: Permission of the instructor. Introduction to a computer language, which will be chosen according to needs of students and of other computer science courses. Typical languages include Smalltalk, LISP, PROLOG. May be repeated once for degree credit with permission and if the languages are different. May only be used once toward the major requirements. *Staff.*

Offered when interest is expressed and departmental resources permit.

Computer Science 312 (3)—Programming Language Design

Prerequisites: Computer Science 112 and 210, Mathematics 121. Formal language description tools; semantic concepts and syntactic constructs appropriate to diverse applications. Comparison of several high-level languages, such as Scheme, Java, ML, and PROLOG, and their implementations of these syntactic and semantic elements. Students learn the Scheme programming language and how to use it to write interpreters for other programming paradigms (object-oriented, logic-oriented, and type-inferencing). *Staff.*

Fall

Computer Science 313 (3)—Theory of Computation

Prerequisites: Computer Science 112, Mathematics 121, and either Mathematics 102 or 122. A study of the principles of computer science embodied in formal languages, automata, computability, and computational complexity. Topics include context-free grammars, Turing machines, and the halting problem. *Staff.*

Winter

Computer Science 315 (3)—Artificial Intelligence

Prerequisite: Computer Science 209. Basic concepts of heuristic search, game playing, natural language processing, and intelligent systems, with a focus on writing programs in these areas. Course combines a discussion of philosophical issues with hands-on problem solving. *Levy.*

Offered when interest is expressed and departmental resources permit.

Computer Science 317 (3)—Database Management

Prerequisite: Computer Science 209. Database design with the entity-relationship model, the relational database model including normal forms and functional dependencies, SQL database query language, server-side scripting for Web access to databases. A major project to design and implement a database using a commercial package. *Whaley.*

Offered when interest is expressed and departmental resources permit.

Computer Science 320 (3)—Parallel Computing

Prerequisites: Computer Science 209 and 210. A survey of parallel computing including hardware, parallel algorithms, and parallel programming. The programming projects emphasize the message-passing paradigm. *Necaise.*

Offered when interest is expressed and departmental resources permit.

Computer Science 321 (3)—Computer Networks

Prerequisite: Computer Science 209. Intended as a first course in communication networks for upper-level students. Covers concepts and protocols underlying modern computer networks. Topics include network architecture and layering, routing and switching, the TCP/IP protocol and network applications. Theory and programming. *Necaise.*

Offered when interest is expressed and departmental resources permit.

Computer Science 330 (3)—Operating Systems

Prerequisites: Computer Science 209 and 210. Procedure initiation, environment construction, reentrancy, kernel functions, resource management, input/output, file structures, security, process control, semaphores and deadlock, and recovery procedures. The laboratory includes the opportunity to examine and modify the internals of an operating system. *Necaise.*

Offered when interest is expressed and departmental resources permit.

Computer Science 332 (3)—Compiler Construction

Prerequisites: Computer Science 210, 211, 312, and 313. Lexical analysis, parsing, context dependence, translation techniques, optimization. Students are expected to produce a compiler for a suitably restricted language. *Staff.*

Offered when interest is expressed and departmental resources permit.

Computer Science 340 (3)—Interactive Computer Graphics

Prerequisite: Computer Science 209. A study of the underlying techniques and algorithms used in the display and manipulation of graphics images in an interactive environment. Topics include programming with Motif and the X-window system, two- and three-dimensional viewing and transformations, color models, and realistic rendering of images. A major component of the course is the development and programming of a variety of projects which demonstrate the topics discussed in class. *Necaise.*

Offered when interest is expressed and departmental resources permit.

Computer Science 341 (3)—Digital Image Processing

Prerequisite: Computer Science 209. Digital-image processing has become a familiar term recognized by a large percentage of the general public. The techniques of digital image processing can be seen in movies, commercials, local weather coverage, images from the Hubble Space Telescope, and even Web browsers. This course will introduce some of the basic filters and techniques of image processing used to manipulate and analyze overlaying, spatial filters, and histogramming. *Necaise.*

Offered when interest is expressed and departmental resources permit.

Computer Science 395 (1), 396 (2), 397 (3)—Seminar
Prerequisite: Permission of the instructor. Readings and conferences for a student or students on topics agreed upon with the directing staff. May be repeated for degree credit with permission and if the topics are different. A maximum of six credits may be used toward the major requirements. *Staff.*
Offered when interest is expressed and departmental resources permit.

Computer Science 401 (1), 402 (2), 403 (3)—Directed Individual Study
Prerequisite: Permission of the department. Individual conferences. May be repeated for degree credit with permission and if the topics are different. *Staff.*
Offered when interest is expressed and departmental resources permit.

Computer Science 493 (3-3)—Honors Thesis
Prerequisites: Senior standing and honors candidacy. Staff.
Fall-Winter

DANCE (DANC)

(Department of Theater)

VISITING ASSISTANT PROFESSOR DAVIES

Dance 110 (1-3)—University Dance
Prerequisite: Permission of the instructor. Participation in a university dance production for a minimum of 24 hours of rehearsal and performance. A journal recording the rehearsal/performance process is required. May be repeated for up to nine degree credits. *Staff.*
Fall, Winter, Spring

Dance 120 (3)—Introduction to Contemporary Modern Dance
Prerequisite: Permission of the instructor. This course combines the exploration of individual and ensemble artistic expression in contemporary modern dance with the study of the history of modern dance. The course culminates in a performance presentation. (HA, GE4a) *Staff.*
Fall

Dance 130 (3)—Contemporary Dance Observation and Analysis
Prerequisite: Permission of the instructor. The observation and analysis of live and recorded contemporary dance focusing on the work of emerging and established choreographers. Exploration of methods for describing the moving body in space. Emphasis is placed on the written and verbal critique of contemporary dance in performance. (HA, GE4a) *Davies.*
Fall 2008 and when departmental resources permit.

Dance 220 (3)—Dance Composition
Prerequisite: Dance 120 or permission of the instructor. A studio course exploring the craft and art of creating dance performances in a variety of styles and contexts. Images, text, music, improvisation and the elements of time, space and energy are examined as sources for dance material leading to group choreography. This course focuses on creating a finished performance piece for presentation. (HA, GE4a) *Staff.*
Fall, Winter

Dance 225 (3)—Intermediate Contemporary Modern Dance
Prerequisite: Dance 120 or permission of the instructor. An intermediate studio course devoted to refining effort/shape values and pursuing performance quality phrasing and style in "Horton" modern dance technique. Students investigate self–directed reverse combinations, deconstruct movement phrases into sequential elements, and learn methods for written and oral analysis of dance. Students practice listening to the body by connecting movement phrases with kinesthetic experiences. The class culminates in a performance presentation. (HA, GE4a) *Davies.*
Winter