

DEPARTMENT OF COMPUTER SCIENCE

Dr. W. Birmingham, Chair; Dr. Adams, Dr. Gribble, Dr. Jenny, Mr. Otto, Dr. Yeager.

The Computer Science Department seeks to provide its students with a solid foundation in the field of computing in order to prepare them for employment in a demanding and highly competitive industry, or for advanced studies in top-ranked graduate schools. This foundation is built on knowledge of mathematics; programming languages; algorithms and data structures; and theory. Moreover, students explore advanced topics, research projects, and technology projects.

The Department believes that it must transmit more than technical expertise to its students: whenever possible, the curriculum emphasizes the need for students to understand their responsibilities to society and to behave ethically, as well as to strengthen and live their Christian faith and witness to the professional community.

Students who complete one of the Department's three Bachelor of Science majors – Computer Science, Computer Information Systems, or Applied Physics/Computer – are prepared to use their skills in computing in application areas that are demanding and rewarding. Broadly educated persons with computing skills are in great demand.

No matter what career majors pursue, they must possess the ability to locate, evaluate and use information. In addition they must be able to communicate their ideas and conclusions clearly and coherently through the written and spoken word. In the course sequence Computer Science 451 and 452 required for Computer Science majors and in the course Computer Science 341 required for Computer Information Systems majors, instruction and experience is provided in these skills—Writing Intensive (WI), Speaking Intensive (SI), and partial instruction in Information Literacy (IL). One of the significant problems in the computer field today is that of ethics. For this reason all computer majors are required to take Computer Science 305, *Ethics in the Computer Profession*. For both majors, the Information Literacy requirement is also met through taking this course.

The Computer Science Department has formulated the following objectives and specific outcomes to guide us in directing and evaluating our program.

Program Objectives

1. To prepare students with the technical abilities required for successful employment as a computer scientist or participation in computer-science graduate studies, or both.
2. To enable students through a Christian worldview to understand their ethical and professional obligations to society, and to encourage the growth of Christian attitudes and moral convictions.
3. To encourage initiative and creative thinking and an understanding of the importance of life-long learning that will enable graduates to grow in their effectiveness throughout their professional careers.

Program Outcomes

1. An ability to apply knowledge of computing and mathematics appropriate to the discipline;
2. An ability to analyze a problem and identify and define the computing requirements appropriate to its solution;
3. An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs;
4. An ability to function effectively as a member of a team to accomplish a common goal;
5. An understanding of professional, ethical, and social responsibilities that is integrally bound to the understanding of professional and ethical responsibility in a Christian context;
6. An ability to communicate effectively with a range of audience, such as professional colleagues and the scientific community;
7. An ability to analyze the impact of computing on individuals, organizations, and society, including ethical, legal, security, and global policy issues that builds on the following liberal-arts areas important in giving the Christian student a background for making judgments concerning computing solutions: history of civilization, Biblical revelation, philosophy, literature, visual art, music, and modern civilization in international perspective;
8. Recognition of the need for, and an ability to engage in, continuing professional development;
9. An ability to use current techniques, skills and tools necessary for computing practice;
10. An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
11. An ability to apply design and development principles in the construction of software systems of varying complexity; and,
12. An understanding of the character qualities needed to conduct oneself honorably and with distinction in a professional career. For example, these qualities include personal integrity, a strong work ethic, and versatility.

Computing Facility

The Computer Science Lab is housed in the Technological Learning Center and is equipped for use by students enrolled in the Department's courses. This lab contains a variety of hardware and software systems. Hardware includes workstations of various types: Windows, Linux, thin clients with high resolution screens and high-performance graphics. Also a flatbed color scanner and a slide and film scanner are available. Software supporting an Intranet WWW server and video and editing are installed. Also the Microsoft Developer Network software is available. In addition, the lab has a "museum" of classical game consoles. The lab also supports a state-of-the-art amateur radio station, with specialized support for software-defined radio projects.

The Technological Learning Center also houses public access Windows based computers with Microsoft Office and other applications.

Course Requirements for Bachelor of Science Degree in Computer Science (99 hours) (CSCI)

Computer Science Core Requirements (30 hours):

Computer Science 141, 155, 220, 222, 244, 252, 314, 322, 342.

Engineering 402.

Advanced Core Requirements (26 hours):

Computer Science 305, 340, 361, 422, 442, 443, 445, 450, 451, and 452.

Computer Science Electives (12 hours):

Choose four of the following courses:

Computer Science 341, 441, 444, 446 and 447.

Math/Science Core Requirements: (31 hours):

Chemistry 105.

Engineering 274.

Mathematics 161, 162, 213, and 261.

Physics 101 and 102.

Courses that count in the CSCI major quality point average (MQPA):

All courses with “COMP” prefix, MATH 213, excluding COMP 102 and COMP 204. A minimum MQPA of 2.00 is required to graduate.

**COMPUTER SCIENCE (CSCI) MAJOR
FOUR-YEAR PLAN**

FRESHMAN YEAR

	Fall	Spring
Mathematics 161-162	4	4
Physics 101-102	4	4
Computer Science 155	3	-
Computer Science 141	-	3
Humanities 101-102	3	3
Physical Education 101-102 (M) or 111-112 (W)	<u>1</u>	<u>1</u>
	15	15

SOPHOMORE YEAR

	Fall	Spring
Mathematics 261	4	-
Engineering 274	-	3
Computer Science 220-222	3	3
Mathematics 213	4	-
Computer Science 244	3	-
Chemistry 105	-	4
Humanities 201-202	3	3
Computer Science 252	<u>-</u>	<u>3</u>
	17	16

JUNIOR YEAR

Computer Science 305	-	2
Computer Science 314	-	3
Computer Science 322	3	-
Computer Science 340	-	3
Computer Science 342	3	-
Computer Science 361	3	-
Computer Science 450	-	3

SENIOR YEAR

Computer Science 422	3	-
Computer Science 451-452	1	2
Computer Science 442	-	3
Computer Science 443	-	3
Computer Science 445	3	-
Computer Science Electives	6	3
Engineering 402	-	3

Computer Science Elective	-	3	Social Science/Gen. Elective	$\frac{3}{16}$	$\frac{3}{17}$
Humanities 301-302	3	3			
Studies in Sci, Faith, Tech (SSFT)	$\frac{3}{15}$	$\frac{-}{17}$			

Course Requirements for Bachelor of Science Degree in Computer Information Systems (CIS) (65 hours)

This interdisciplinary program is to prepare the student to facilitate the development of computer utilization, databases and information systems to satisfy the requirements and needs of organizational management.

Computer Core Requirements (31 hours):

Computer Science 102, 220, 222, 244, 252, 305, 340, 341, 342, and 450.

One 300 or 400-level Computer course.

Business Core Requirements (21 hours):

Accounting 201, 202, and Engineering 402.

Business 201, 203, 301, and 307.

Business/Computer electives (6 hours):

Select two courses from Business 204, 302; Entrepreneurship 309; Computer Science 390 or 400-level Computer Science elective.

Major-Related Requirements (7 hours):

Economics 101.

Mathematics 141.

Courses that count in the CIS major quality point average (MQPA):

All courses with “ACCT,” “BUSA,” and “COMP” prefixes, excluding BUSA 205 and BUSA 206. A minimum MQPA of 2.00 is required to graduate.

Students are expected to contact their advisors for a detailed schedule of courses recommended to meet requirements for a major.

Course Requirements for Applied Physics/Computer Major

These major requirements may be found in the Department of Physics.

Course Requirements for a minor in Computer Science (21 hours):

Computer Science 141, 314, 220, and 222.

Select nine hours of Computer Science courses 200-level and above.

Note: Students pursuing one of the six Business majors or Applied Physics as a first major might prefer Computer Information Systems or Applied Physics/Computer as the first major rather than this minor.

COMPUTER SCIENCE (COMP)

102. INTRODUCTION TO PRODUCTIVITY SOFTWARE. This course is designed to familiarize students with the use of a personal computer. The course involves no programming and does not assume the student has had programming

experience. The course includes terminology for hardware, software, telecommunications, and applications. Hands on applications include Microsoft Office (Word, Excel, Access, Power Point), One Note Microsoft Publisher, and Internet Explorer. *Semester course, three hours.*

141. COMPUTER PROGRAMMING I. This course provides the student with an understanding of hardware and software concepts, structured program design, and programming using C++ in an integrated development environment. Topics include Boolean expressions, iteration, standard library functions, programmer-defined functions, arrays, searching and sorting, multidimensional arrays, string class, vector class, pointers and dynamic memory allocation, programmer-defined classes, and abstract data types. This course, along with Math 161 and 488, fulfills the Information Literacy (IL) requirement for the Mathematics major. *Semester course, three hours.*

155. INTRODUCTION TO COMPUTER SCIENCE. This course provides an introduction to the field of Computer Science. Topics include data representation, gates and circuits, algorithm design, programming languages, abstract data types, operating systems, information systems, artificial intelligence, networks, and the World Wide Web. The course is intended to prepare students for further study in the field and is also appropriate for non-major students who wish to become familiar with computer science. *Fall semester only, three hours.*

204. THE TECHNOLOGIES OF INSTRUCTION. An introduction to educational media with an emphasis on applications of computer technology in education; general models for computer usage in education and educational institutions; and case studies of specific projects in terms of approach, effectiveness, and implications for the future. Emphasis will be on the application of educational media within the K-12 school curriculum. Prerequisite: For education majors only. *Semester course, three hours.*

220. COMPUTER PROGRAMMING II. This is a second course in the C++ language: a review of essential language concepts, structured programming, and top-down design. Object oriented program design principles including inheritance, abstract base classes, virtual functions, and polymorphism are covered. Other topics include operator overloading, templates, linked data structures, and exception handling. Prerequisite: Computer Science 141. *Fall semester only, three hours.*

222. INTRODUCTION TO DATA STRUCTURES AND ALGORITHMS. An advanced course in programming the computer utilizing C++ including the run-time behavior of programs; the design and structure of programs; linear data structures; recursion; binary search trees; Heapsort; and Hash technique for searching. Prerequisite: Computer Science 220. *Semester course, three hours.*

244. DATABASE MANAGEMENT SYSTEMS. An introduction to database management systems emphasizing the relational model. Topics include data manipulation languages (SQL, QBE); database design (intuitive design, normalization,

and E-R design model); three-tier and multi-tier architecture; database security; and database integrity. Prerequisite: Computer Science 141 or knowledge of its content.

Fall semester only, three hours.

252. COMPUTER ARCHITECTURE AND ORGANIZATION. Organization, elementary architectural design and computer instruction sets are examined and used via programming in an assembly language. Students are given an introduction to the manner in which digital computers actually work. Prerequisite: Computer Science 141 or knowledge of its content.

Spring semester course, three hours.

305. ETHICS IN THE COMPUTING PROFESSION. This course is an overview of ethical systems and copyright and intellectual property laws, an examination of IEEE and ACM ethical codes, ethical problem-solving techniques and an examination of ethical cases. This course partially fulfills the Information Literacy (IL) requirement for Computer Science and Computer Information Systems majors.

Spring semester only, two hours.

314. FOUNDATIONS OF COMPUTER SCIENCE. This course is an introduction to computation theory including the topics: finite automata, regular languages, pushdown automata, context-free languages, Turing machines, recursive languages and functions, and computational complexity. Prerequisite: Computer Science 141, Math 213.

Spring semester only, three hours.

318. HIGH-TECHNOLOGY VENTURES. The purpose of this course is three-fold: to introduce students to the process of technological innovation within a business; to learn to work effectively within a multidisciplinary team; and, to design and prototype a product working with a local company. Students experience what it takes to bring a product (or prototype) from concept to market. The class is centered on product development and writing a business plan to support the product. Students will spend time in lecture and laboratory and will make off-site visits to the partner company. The final outcome will be a prototype and a business plan. Prerequisite: junior or senior standing and instructor approval.

Spring semester only, three hours.

322. OBJECT ORIENTED AND ADVANCED PROGRAMMING. Topics include: programming techniques in Standard C++ for large-scale, complex, or high-performance software; encapsulation; automatic memory management; exceptions; generic programming with templates and function objects; standard library algorithms and containers; using single and multiple inheritance and polymorphism for code reuse and extensibility; basic design idioms, patterns, and notation. Prerequisite: Computer Science 222.

Fall semester only, three hours.

340. OPERATING SYSTEMS. A study of the basic principles of operating system design and implementation including types of computer systems, general architecture of several representative computer systems, security, run-time systems, and performance measurement and evaluation. Prerequisite: Computer Science 222.

Spring semester only, three hours.

341. SYSTEMS ANALYSIS. An introduction to the application of the systems development life cycle (SDLC) as applied to a variety of practical software systems. Special emphasis is given to systems requirements determination and analysis of systems by means of team-based projects. This course fulfills the Writing Intensive (WI), Speaking Intensive (SI), and partially fulfills the Information Literacy (IL) requirements for the Computer Information Systems major. Prerequisite: Computer Science 102 or 141, and 244. *Fall semester only, three hours.*

342. DATA COMMUNICATION AND NETWORKING. An introduction to the concepts of data communications used in information networks including equipment utilization in information networks; techniques utilized to transmit signals (e.g., modulation, multiplexing, error detection, and correction); methods of message handling; network configuration; and software utilized in implementing networks. Prerequisite: Computer Science 141. *Fall semester only, three hours.*

361. INTRODUCTION TO COMPUTER GRAPHICS. This course is an introduction to computer graphics covering fundamental concepts such as rasterizing, two-dimensional transformations, windowing, clipping, line and surface detection. Prerequisites: Computer Science 222, Engineering 274, and Mathematics 213. *Fall semester only, three hours.*

390. SELECTED TOPICS IN COMPUTER SCIENCE. An examination of areas of computer systems not fully covered by regular departmental offerings. Subject matter varies each semester. Prerequisite: Computer Science 220 and permission of the department. *Semester course, two or three hours.*

422. INTRODUCTION TO ALGORITHMS. Topics include: fundamental techniques for designing efficient algorithms and basic mathematical methods for analyzing their performance; paradigms for algorithm design; divide-and-conquer, greedy methods, graph search techniques, dynamic programming; design of efficient data structures, and analysis of the running time and space requirements of algorithms in the worst and average cases. Prerequisite: Computer Science 222 and 314. *Fall semester only, three hours.*

441. COMPUTER GAME DESIGN AND DEVELOPMENT. This course covers concepts and methods for the design and development of computer games. Topics include: graphics and animation, sprites, software design, game design, user interfaces, game development environments. Prerequisite: Computer Science 222. *Alternate years, fall semester only, three hours.*

442. WEB PROGRAMMING TECHNOLOGIES. This course prepares students with the fundamentals needed to program on the Internet. It offers a survey of programming concepts that yield visible or audible results in Web pages and Web-based applications. The course covers effective Web-page design, various markup languages, several

scripting languages, Web servers, and databases to provide all the skills and tools needed to create dynamic Web-based applications. Prerequisite: Computer Science 244.

Alternate years, spring semester only, three hours.

443. OBJECT-ORIENTED PROGRAMMING LANGUAGES (OOPL) AND EVENT-DRIVEN PROGRAMMING. In this course detailed case studies of object-oriented programming languages such as Java, C++, Smalltalk, and C#, and of event-driven programming environments built around those languages are investigated. Examples are the Java Swing library, MFC library, Visual Studio Net, and the original Smalltalk environment. Study of component-based programming using Java beans and/or COM-based models is also included. Prerequisite: Computer Science 222.

Spring semester only, three hours.

444. INTRODUCTION TO INFORMATION RETRIEVAL. The technological underpinnings of search engines is explored in this class. Topics covered include organizing and accessing information, metrics of performance (relevance and precision), query methods, multimedia retrieval, and digital libraries. Prerequisite: Computer Science 222.

Alternate years, three hours.

445. INTRODUCTION TO ARTIFICIAL INTELLIGENCE. Artificial intelligence topics included in this class are: predicate calculus, state space search, knowledge representation, expert systems, reasoning in uncertain situations, and machine learning. Prerequisite: Computer Science 222.

Alternate years, fall semester only, three hours.

446. ADVANCED COMPUTER GAME DESIGN AND DEVELOPMENT. This course is a continuation of Computer Science 441 and is focused on the development of 3D games and other advanced game programming techniques. Prerequisite: Computer Science 441.

Alternate years, spring only, three hours.

447. CONSOLE GAME DESIGN AND DEVELOPMENT. This course is a continuation of Computer Science 441 and is focused on the development of console games, with emphasis on both hardware and software design issues. The course will explore sophisticated programming techniques and advanced algorithms. Prerequisites: Computer science 441, 446, and permission of instructor.

Alternate years, fall only, three hours.

450. SOFTWARE ENGINEERING. This course introduces software-engineering methodology, covering such topics as development cycles, testing, design, requirements gathering and analysis, and project management. Students work in teams on a semester-long project. Software Engineering is designed to fulfill the Writing Intensive (WI), Speaking Intensive (SI), and Information Literacy (IL) requirements for the Computer Information Systems major. Prerequisite: Junior standing.

Spring semester only, three hours.

451. SENIOR PROJECT I. This course is part of the capstone design experience, and is based on applying software engineering to a two-semester long project. This course

focuses on requirements gathering, early prototyping, and design. Students will write reports and make presentations. This course fulfills the Writing Intensive (WI), Speaking Intensive (SI), and Information Literacy (IL) requirements for the Computer Science major.
Fall semester only, one hour.

452. SENIOR PROJECT II. This course is a continuation of Computer 451 and is focused on the development of a working, tested system delivered to a customer. Students will write reports, make presentations, and deliver a working software system. This course fulfills the Writing Intensive (WI), Speaking Intensive (SI), and Information Literacy (IL) requirements for the Computer Science major. Prerequisite: Computer Science 451.
Spring semester only, two hours.

460. INDEPENDENT STUDY. An advanced course for qualified students that provides an opportunity for further computer programming and analysis experience on an individual basis. Prerequisite: Permission of the department.
Semester course, one, two or three hours.

480. INTERNSHIP IN COMPUTER SCIENCE. Students earn academic credit for field experience that allows them to use their computer skills under the supervision of a cooperating entity. A maximum of six credit hours may be applied toward the major. Prerequisites: Junior standing and consent of the department chairman.
Semester course, one to six hours.

499. HONORS IN COMPUTER SCIENCE. A course for qualified junior or senior students who are interested in advanced systems and programming experience. Practical programming assignments are based on the student's interest. Prerequisite: Permission of the department.
Semester course, one, two or three hours.