

Department Chair: Marc Thomas
Department Office: Science Building I, 417
Telephone: (661) 654-3082
email: computersci@csub.edu
Website: www.cs.csubak.edu
Faculty: M. Danforth, S. Garcia, W. Li, T. Meyer,
D. Meyers, J. Moloney, L. Taylor, M. Thomas,
H. Wang, A. Wani

Program Description

Computer Science is a constantly evolving discipline. To quote the Association for Computing Machinery, "Computer Science is not simply concerned with the design of computing devices-nor is it just the art of numerical calculation. . . . Computer Science is concerned with information in much the same sense that Physics is concerned with energy; it is devoted to the representation, storage, manipulation, and presentation of information in an environment permitting automatic information systems."

The Computer Science major at CSUB has three tracks. The Computer Science track follows the guidelines recommended by the Association for Computing Machinery (ACM) and the Accreditation Board for Engineering and Technology (ABET). The Computer Information Systems track is intended for training application programmers or for those who wish to apply computer science in another discipline. The Hardware track is intended for students who will be working in a hardware or system software environment. Students in the three tracks will take different advanced courses of their choice. A Computer Science minor is also offered.

The Computer Science Department moved into a new building in Fall 2008 together with the Mathematics Department and has received almost a threefold increase in space. The department administers its own local area network which includes multiple Unix/Linux servers, two software programming labs, a walk-in lab, one advanced workstation lab, one digital electronics hardware lab, and a robotics lab. There is also a departmental library available to students. An important goal of the department is to enable students to work much more closely with faculty than they would be able to at larger universities. A detailed description of student learning goals and objectives can be found at http://www.cs.csub.edu/index.php?t=1&p=academic_info/program_info/index.

Requirements for the Bachelor of Science Degrees with a Major in Computer Science

The Bachelor of Science Degrees with a major in Computer Science require a minimum of 180 units which includes courses for the major (and minor, if selected) and courses for the other university-wide graduation requirements: General Education, American Institutions, First-Year Experience, Gender-Race-Ethnicity, Upper Division Writing, and Foreign Language (see pages 56-63).

A. Computer Science Track

This track follows the guidelines of the Association for Computing Machinery (ACM) and the Accreditation Board for Engineering and Technology (ABET). Students in this track will take advanced courses of their choice.

- 1. Introductory courses** (16 units):
 - CMPS 150 (or CMPS 215)
 - CMPS 221
 - CMPS 222
 - CMPS 223
- 2. Intermediate courses** (50 units):
 - CMPS 295
 - CMPS 312
 - CMPS 320
 - CMPS 321
 - CMPS 335
 - CMPS 342
 - CMPS 350

CMPS 356
CMPS 360
CMPS 376

3. Advanced courses (15 units):
CMPS 490

Two courses from the following:

Algorithms and Complexity

CMPS 411

Architecture and Organization

CMPS 420

CMPS 421

CMPS 422

Intelligent Systems

CMPS 456

CMPS 457

Programming Languages

CMPS 410

CMPS 450

Operating Systems & Computer Networks

CMPS 460

CMPS 476

Software Engineering and Database Systems

CMPS 435

CMPS 442

CMPS 465

Visual Computing

CMPS 371

CMPS 471

CMPS 472

CMPS 473

CMPS 477 Special Topics in Computer Science

Depending on topic, this course may count for one of the sub-areas above.

4. The following math/physics courses (37 units):

MATH 201

MATH 202

MATH 203

MATH 330

MATH 340

PHYS 221

PHYS 222

5. Information on General Education Courses

- PHYS 221 will satisfy General Education Areas B2 and B3.
- Any of the required calculus courses satisfy General Education Area B4.
- The General Education Theme I requirement is waived for all Computer Science tracks.
- PHIL 316 (Professional Ethics) must be taken and will satisfy General Education Theme 2 and the Computer Science Ethics requirement.

B. Computer Information Systems Track

This track is intended for training application programmers or for those who wish to apply computer science in another discipline.

1. Introductory courses (16 units):

CMPS 150 (or CMPS 215)

CMPS 221

CMPS 222

CMPS 223

2. Intermediate courses (40 units):

CMPS 295

CMPS 312
CMPS 335
CMPS 342
CMPS 350
CMPS 356
CMPS 360
CMPS 376

3. Advanced courses (10 units):

CMPS 435 **OR**
CMPS 442 **OR**
CMPS 465
CMPS 490

4. Required Mathematics courses (10 units):

MATH 140 or MATH 192 or higher level mathematics course
MATH 190/191

5. Electives from (35 units):

CMPS 211
CMPS 215
CMPS 216
CMPS 280
CMPS 294

OR any other 300-400 level computing course taken with the consent of the program advisor. Courses from other departments relevant to CIS (not exceeding 10 units) may be taken with the written consent of the program advisor. A minor in another department can be used to offset some electives upon approval of a Computer Science Department advisor.

6. Information on General Education Courses

- MATH 140 or higher level mathematics course will satisfy General Education Area B4.
- The General Education Theme I requirement is waived for all Computer Science tracks.
- PHIL 316 (Professional Ethics) must be taken and will satisfy General Education Theme 2 and the Computer Science Ethics requirement.

C. Computer Science Hardware Track

This track is intended for computer scientists who will be working in a hardware or system software environment, including, for example, embedded systems, computer controlled instrumentation, device controllers, and operating system code.

1. Introductory courses (16 units):

CMPS 150 (or CMPS 215)
CMPS 221
CMPS 222
CMPS 223

2. Intermediate courses (25 units):

CMPS 224
CMPS 295
CMPS 320
CMPS 321
CMPS 360

3. Advanced courses (15 units):

CMPS 420
CMPS 421
CMPS 490

4. The following math/physics courses (42 units):

MATH 201
MATH 202
MATH 203
MATH 204 or MATH 205, or MATH 206
MATH 330
PHYS 221

PHYS 222

ENGR 207

5. Electives from (15 units):

CMPS 322 Digital Design with VHDL

CMPS 335 Software Engineering (Hardware Track students should choose a hardware-oriented project)

CMPS 350

CMPS 371

CMPS 376

CMPS 422

CMPS 450

CMPS 457

CMPS 471

CMPS 476

OR any other 300-400 level computing courses may be taken as an elective with the written consent of the program advisor.

6. Information on General Education Courses

- PHYS 221 will satisfy General Education Areas B2 and B3.
- Any of the required calculus courses satisfy General Education Area B4.
- The General Education Theme I requirement is waived for all Computer Science tracks.
- PHIL 316 (Professional Ethics) must be taken and will satisfy General Education Theme 2 and the Computer Science Ethics requirement.

Requirements for a Minor in Computer Science

A Minor in Computer Science will require the student to take a total of at least 20 units of 200-level or higher course work as well as satisfy the additional requirements:

a. CMPS 223 (which requires CMPS 221 or the equivalent)

b. One course chosen from the following: CMPS 215 (3 units), CMPS 222, CMPS 224, or CMPS 295.

c. At least 10 units of upper division course work in computer science (normally two courses) chosen with the help of a computer science advisor. MATH 305 may be substituted for one computer science course.

Remarks: Note that CMPS 222 and CMPS 223 require either CMPS 221 or equivalent knowledge of C/C++ procedural programming. CMPS 221 requires either CMPS 150 (1 unit) or CMPS 215 as a corequisite. Students who do not require CMPS 221 in their major (and, hence, are under no university "double-counting" limitations) can count CMPS 221 towards the total of 20 units. Some suggested paths to the Computer Science Minor are shown in the table below with the area of specialization noted, but the student is strongly encouraged to discuss all options with a computer science advisor.

Academic Regulation

A grade of C- is the minimal grade acceptable for progression in the CMPS 221, 222, and 223 sequence.

COURSE DESCRIPTIONS

Lower Division

The Department of Computer Science offers courses on topics of current interest to the community from time to time. Call the Computer Science office, (661) 654-3082, to express interest or inquire concerning offerings.

CMPS 120 Computer Skills and Concepts I (5)

Instruction and tutoring in basic computer skills. An overview of computer terminology, hardware and software. Included: storage devices, input/output devices, the internet, operating systems, word

processing, spreadsheets, presentation software, creating web pages and simple databases. Meets for 250 minutes. Prerequisites: None

CMPS 150 Introduction to Unix (1)

Basic Unix commands and programming utilities will be introduced. Students will learn how to use email, a text editor, and manage files and directories. This course is designed for students who have no experience with Unix. Computer Science majors are encouraged to take CMPS 215 in place of this course, if possible.

CMPS/MATH 206 Advanced Engineering Mathematics (5)

Introduction to ordinary differential equations, Fourier Series and Integral, other transforms, and partial differential equations; applications to computer hardware, such as the resonance, wave equation, transmission line equation, and filtering. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: MATH 203.

CMPS 211 Internet Programming and Web Design (5)

Introduction to internet programming using HTML and JavaScript or other scripting language. The course is intended for students with no programming experience. Students will learn the concepts of structured programming and control structures. They will become familiar with HTML interfaces by designing interactive web sites. This course may be used as an elective in the CIS track. Prerequisite: None.

CMPS 215 Unix Programming Environment (3)

This course covers common Unix commands, shell scripting, regular expressions, tools and the applications used in a Unix programming environment. The tools to be introduced include make utility, a debugger, advanced text editing and text processing (vi, sed, tr). Each week lecture meets for 100 minutes and lab meets for 150 minutes. Prerequisite: None.

CMPS 216 Unix System Administration (3)

This course covers the knowledge and skills critical to administering a multi-user, networked Unix system. The course assumes a basic knowledge of Unix commands and an editor (vi or Emacs). Topics include: kernel and network configuration, managing daemons, devices, and critical processes, controlling startup and shutdown events, account management, installing software, security issues, shell scripting. Many concepts will be demonstrated during hands-on labs. Each week lecture meets for 100 minutes and lab meets for 150 minutes. Prerequisite: CMPS 215.

CMPS 221 Programming Fundamentals (5)

Introduces the fundamentals of procedural programming. Topics include: data types, control structures, functions, arrays, and standard and file I/O. The mechanics of compiling, linking, running, debugging and testing within a particular programming environment are covered. Ethical issues and a historical perspective of programming within the context of computer science as a discipline is given. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: Passing score on ELM OR satisfaction of the ELM exemptions AND a passing score on the Pre-Calculus Readiness Test.

CMPS 222 Object-Oriented Programming (5)

Builds on foundation provided by CMPS 221 to introduce the concepts of object-oriented programming. The course focuses on the definition and use of classes and the fundamentals of object-oriented design. Other topics include: an overview of programming language principles, basic searching and sorting techniques, and an introduction to software engineering issues. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 221.

CMPS 223 Data Structures and Algorithms (5)

Builds upon the foundation provided by CMPS 221 to introduce the fundamental concepts of data structures, algorithms, and algorithm analysis. The course focuses on the concepts and implementation of fundamental data structures (linked lists, stacks, queues, trees, hash tables, and graphs). Other topics

include: pointers, recursion, searching and sorting algorithms, and the basics of algorithm analysis. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 221.

CMPS 224 Assembly Language Programming (5)

Introduction to machine architecture and program structure; code, data, and stack segments; programming with an assembly language. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 221.

CMPS 270 Introduction to CAD in Engineering (3)

Use of computer-aided design software, such as AutoCAD, in engineering. CAD concepts including drawing setups, commands and system variables, layers and object properties, 2-dimensional entity creation, coordinate systems, creating objects, drawing with precision, plotting, and editing methods are applied to a variety of engineering applications. Two hours lecture/discussion and three hours laboratory per week.

CMPS 271 Intermediate CAD in Engineering (3)

Intermediate topics in computer-aided design using AutoCAD. Introduction to 3-dimensional drawing and modeling with engineering applications, adding text to drawings, creating dimensions, using blocks and external references, managing content with Autocad Design Center, creating a layout to plot, plotting your drawings, working with raster images, creating compound documents with OLE, and using other file formats. Two hours lecture/discussion and three hours laboratory per week. Prerequisite: CMPS 270

CMPS 277 Topics in Programming Languages (1-5)

A study of programming languages not offered otherwise. Prerequisite: Knowledge of a high-level programming language or permission of the instructor.

CMPS 280 X-Windows (3)

This course is an introduction to the use of an X-Windowing environment. The course is designed more for the end user than for X11 programmers. Its goal is to familiarize the applications user with the standard X11 productivity tools as well as explain the underlying principles, configuration questions, and security considerations involved in working or administering an X-Workstation with Internet access. Each week lecture meets for 100 minutes and lab meets for 150 minutes.

CMPS 281 Problem Solving in Compute Science (1)

This workshop is designed for students in the Louis Stokes alliance for Minority Participation Program (LSAMP), but is open to other students as well. It covers topics from CMPS 221. Typically students work during the meeting on problems related to their class, being helped by a facilitator.

CMPS 295 Discrete Structures (5)

Discrete structures and applications in computer science. Proof techniques, induction, predicate logic, functions, relations and sets, asymptotics, counting techniques, recurrence relations, graph theory and trees. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 221.

Upper Division

CMPS 305 Numerical Analysis (5)

Number representation and basic concepts of error; numerical solutions of nonlinear equations and systems of equations; interpolation and extrapolation; numerical differentiation and integration; numerical solution of ordinary differential equations; approximation by spline functions. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisites: CMPS 221 and MATH 203 or permission of instructor. Cross-listed with MATH 305: Numerical Analysis.

CMPS 311 Server Scripting Languages (5)

CMPS 312 Algorithm Analysis and Design (5)

Algorithm analysis, asymptotic notation, hashing, hash tables, scatter tables, and AVL and B-trees, brute-force and greedy algorithms, divide-and-conquer algorithms, dynamic programming, randomized algorithms, graphs and graph algorithms, and distributed algorithms. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 295 or CMPS 300 and CMPS 223.

CMPS 320 Digital Circuits (5)

An introduction to the logical design of digital computers including the analysis and synthesis of combinatorial and sequential circuits, and the use of such circuits in building processor components and memory. The course will apply the circuit theory to the design of an elementary processor with a small instruction set with absolute addressing and a hard-wired control unit. An assembly language for this processor will also be developed. This course includes a laboratory which will cover a mix of actual circuit work together with circuit synthesis and testing using software. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: One course in programming or permission of the instructor.

CMPS 321 Computer Architecture (5)

This course follows the Digital Logic Design course and focuses on the design of the CPU and computer system at the architectural (or functional) level: CPU instruction sets and functional units, data types, control unit design, interrupt handling and DMA, I/O support, memory hierarchy, virtual memory, and buses and bus timing. In contrast, the Digital logic Design course is primarily concerned with implementation; that is, the combinatorial and sequential circuits which are the building blocks of the functional units. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 322 Digital Design with VHDL (5)

This course uses a hardware description language (HDL) to design application-specific integrated circuits. The continuation of CMPS 320 includes modern digital design technology, in-depth treatment of algorithms and architectures for digital machines and comprehensive treatment of behavioral modeling in advanced digital design. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 320.

CMPS 335 Software Engineering (5)

A general introduction to Software Engineering. Deals with the specification, development, management, and evolution of complex software systems. Shows how to cost-effectively apply the methods and theory from Computer Science to solve difficult problems. The course presents a broad perspective on software and system engineering and surveys a wide spectrum of tools and techniques. Students are required to complete a project as part of a small software engineering team. Students may choose system projects involving software and hardware integration. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 340 Introduction to Digital Forensics (5)

Investigative techniques, evidence handling procedures, forensics tools, digital crime reconstruction, and legal guidelines. Case studies cover a range of hardware and software platforms. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: None (CMPS 215 or a good working knowledge of Unix is recommended)

CMPS 342 Database Systems (5)

Basic issues in data modeling, database application software design and implementation. File organizations, relational model, relational database management systems, and query languages are addressed in detail. Two-tier architecture, three-tier architecture and development tools are covered. Meets for 200 minutes of lecture and 150 minutes of lab. Prerequisite: CMPS 295.

CMPS 350 Programming Languages (5)

An examination of underlying concepts in high level programming languages and techniques for the implementation of a representative sample of such languages with regard to considerations such as

typing, block structure, scope, recursion, procedures invocation, context, binding, and modularity. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 356 Artificial Intelligence (5)

This course is intended to teach the fundamentals of artificial intelligence which include topics such as expert systems, artificial neural networks, fuzzy logic, inductive learning and evolutionary algorithms. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 360 Operating Systems (5)

A study of the introductory concepts in operating systems: historical development of batch, multiprogrammed, and interactive systems; file, memory, device, process, and thread management; interrupt and trap handlers, abstraction layer, message passing; kernel tasks and kernel design issues; signals and interprocess communication; synchronization, concurrency, and deadlock problems. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 371 Computer Graphics (5)

Introduction to computer graphics hardware, animation, two-dimensional transformations, basic concepts of computer graphics, theory and implementation. Use of graphics API's such as DirectX or OpenGL. Developing 2D graphics applications software. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 376 Computer Networks (5)

A study of computer networks focusing on the TCP/IP Internet protocols and covering in detail the four layers: physical, data link, network, and transport. This course includes a laboratory in which students will cover important network utilities, debugging tools, process and thread control as it relates to network programming, and the coding of programs which do interprocess communication over sockets. The typical Internet client program which accesses a TCP network server daemon will be covered in detail. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 377 Special Topics in Computer Science (3-5)

This course will be used to supplement other courses with additional work at the intermediate level. Prerequisite: Permission of instructor.

CMPS 394 Java: Client, Server and Internet Programming (5)

This course will use Java's OOP features and libraries to explore client-side and server-side programming techniques (applet, servlet, dynamic HTML, XML, style sheet and database access). Multithreading and synchronization concepts, networking programming, socket and/or Remote Method Invocation techniques will be introduced and used to implement Internet chatting/gaming, file transfer, and 2-tier and 3-tier database applications. Through those implementations, students will learn the common programming skills used in common server programs such as web, mail http, ftp and database servers. Meets for 200 minutes of lecture and 150 minutes of lab. Prerequisites: CMPS 222 and CMPS 223.

CMPS 410 Theory of Language Translation (5)

A study of techniques relevant to the theory of language translation including finite state machines, formal languages, grammars, lexical and syntactic analysis. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 350.

CMPS 411 Theory of Automata (5)

A study of algorithms as they relate to nonlinear data structures and external files. Time and space analysis of several popular algorithms, and a discussion of NP-hard and NP-complete problems. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 420 Embedded Systems (5)

Built on logic designs, using assembly and C languages to study embedded systems with regard to their software, hardware, theories and implementation methodology. Various embedded system development tools, such as assemblers, debuggers and cross compilers, will be introduced and used in the course. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisites: CMPS 224 and CMPS 320.

CMPS 421 Advanced Computer Architecture (5)

Continuation of CMPS 321 including speed-up arithmetic algorithms, vector and parallel processing, organization of memory for high performance processors, and a comparative study of supercomputer architectures. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 321.

CMPS 422 Digital Signal Processing (5)

Introduction to principles of Digital Signal Processing (DSP) including sampling theory, aliasing effects, frequency response, Finite Impulse Response filters, Infinite Impulse Response filters, spectrum analysis, Z transforms, Discrete Fourier Transform and Fast Fourier Transform. Emphasis on hardware design to achieve high-speed real and complex multiplications and additions. Pipelining, Harvard, and modified Harvard architectures. Overviews of modern DSP applications such as modems, speech processing, audio and video compression and expansion, and cellular protocols. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisites: PHYS 221, MATH 203, and CMPS 320.

CMPS 432 Instrumentation, Control, and Data Acquisition (5)

CMPS 435 Advanced Software Engineering (5)

Continuation of study of the software lifecycle. Methods and tools for the implementation, integration, testing and maintenance of large, complex software systems. Program development and test environments. Group laboratory project. Technical presentation methods and practice. Ethical and societal issues in software engineering. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 335.

CMPS 442 Advanced Database Systems (5)

A wide range of topics such as query processing and optimization, object-oriented database systems, distributed database systems, database warehousing and data mining will be discussed. The course will also be used to introduce emerging issues related to database systems. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 342.

CMPS 450 Compiler Construction (5)

An introduction to the construction of compilers, including lexical and syntactic analysis, code generation, and error detection. This course includes a 2 1/2 hours per week laboratory in which students will implement a compiler for a given programming language. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 350.

CMPS 456 Advanced Artificial Intelligence (5)

Continuation of CMPS 356. This course is intended to teach about advances in artificial intelligence. It includes advanced topics on artificial neural networks such as distributed and synergistic neural network models, hybrid artificial intelligence techniques such as neuro-fuzzy models, advanced machine learning techniques and meta-heuristic evolutionary algorithms. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisites: CMPS 356.

CMPS 457 Robotics (5)

The course will provide an opportunity for students to understand intelligent robot system architecture and to design algorithms and programs for control and planning of intelligent robot systems based on analytical modeling and behavior modeling. Students will use simulation software (Webots) and hardware test-bed (Kheraper II) to verify their algorithm and program performance during their project work. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 460 Advanced Operating Systems (5)

Continuation of CMPS 360. Various topics in popular operating systems. Real-time and distributed operating systems will be addressed. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 360.

CMPS 465 Advanced System Analysis and Design (5)

Design and construction of sizeable software products. Technical management of software development teams. Software development process models, software design, documentation, quality assurance during development, software unit and integration testing, CASE tools, development environments, test tools, configuration management. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 335.

CMPS 471 Advanced Computer Graphics (5)

Continuation of CMPS 371. 3D graphics transformations, multi-resolution model building and rendering. Advanced computer graphics concepts - theory and implementation. Advanced animation techniques in a 3D environment. This course includes a laboratory. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisites: CMPS 371.

CMPS 472 AI Agents in Virtual Environments (5)

Continuation of CMPS 471. This course is about creating and interacting with intelligent three-dimensional virtual environments. Topics covered will include hierarchical architecture of three-dimensional virtual environments, and a framework of incorporating intelligent agents within the virtual environment. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 371.

CMPS 473 Computer Game Design (5)

The course will cover fundamental concepts behind designing a game engine. The concepts, theories, and programming aspects of physics engine, graphics engine, and control engine will be covered. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisite: CMPS 223.

CMPS 476 Advanced Computer Networks and Computer Security (5)

Continuation of CMPS 376. Various advanced topics in computer networks and computer security will be addressed. Each week lecture meets for 200 minutes and lab meets for 150 minutes. Prerequisites: CMPS 376

CMPS 477 Special Topics in Computer Science (1-5)

This course will often be used to supplement other courses with additional work at a more advanced level. Prerequisite: permission of instructor.

CMPS 489 Experiential Prior Learning (1-5)

Majors in Computer Science with significant prior experience in computers may have some of their experience count for academic credit toward their degree. In order to be considered for experiential learning credit the student must have completed CMPS 223 and have the approval of the department.

CMPS 490 Senior Project (5)

After consultation with the instructor and investigation of relevant literature, the student shall prepare a substantial project with significance in Computer Science. During the latter part of the quarter, the student will present a project report to the entire class, explaining the nature of the work, the finished product, and its relationship to the field. Prerequisites: major in Computer Science and completion of at least two 400-level courses in Computer Science.

CMPS 496 Internship in Computer Science (1-5)

Internships may be arranged by the department with various agencies, businesses, or industries. The assignments and coordination of work projects with conferences and reading, as well as course credits, evaluation, and grading are the responsibility of the faculty liaison (or course instructor), working with the

field supervisor. Offered on a credit, no-credit basis only. The department will determine credits and application of credit.

CMPS 497 Cooperative Education (5)

The Cooperative Education program offers a sponsored learning experience in a work setting, integrated with a field analysis seminar. The field experience is contracted by the Cooperative Education office on an individual basis, subject to approval by the department. The field experience, including the seminar and reading assignments, is supervised by the cooperative education coordinator and the faculty liaison (or course instructor), working with the field supervisor. Students are expected to enroll in the course for at least two quarters. The determination of course credits, evaluation, and grading are the responsibility of the departmental faculty. Offered on a credit, no-credit basis only. The department will determine application of credit.