UNIVERSITY OF MARY WASHINGTON – PROGRAM CHANGE PROPOSAL

Electronically submit this completed form with attachments in one file to the Chair of the College Curriculum Committee.

COLLEGE (check one):	**Arts and Sciences**	X	**Business**		**Education**	
Proposal Submitted By: Jennifer Polack			Date Prepared: 10/24/2014			
Department /Program:	**Computer Science**					

Note: for any program change entailing the addition any new courses, or revisions to existing courses, separate proposal for those course actions must also be submitted.

PROPOSAL TO CHANGE EXISTING PROGRAM (check **no than one** of the following)	
Revise requirements for existing **major**	X
Revise requirements for a concentration within an existing **major**	
Revise requirements for an existing **degree program**	
Revise requirements for existing **certificate** program	
Revise requirements for existing **minor**	
Implementation Date: FALL semester, year:	2015

REQUIRED ATTACHMENTS FOR CHANGES TO EXISTING PROGRAMS:

1. **Rationale statement** (Why is this program change needed? What purposes will it serve?)
2. **Impact Statement** (Provide details about the Library, space, budget, technology, and impacts created by this program change. Supporting statements from the Library, IT Department, etc. evaluating the resource impact and feasibility of the program change are required.)
3. **Catalog Copy** (Provide the *existing* Catalog Description **and** the complete statement of the *proposed* new Catalog description that reflects the program changes)

PROPOSAL TO CREATE NEW PROGRAM *NOT* REQUIRING STATE ACTION (check no more that **one** of the following)		
New concentration within existing major		Name:
New minor		Name:
New Major but NOT a new degree*		Name:
*Use ONLY for interdisciplinary majors that will be grouped as part of the "Special Majors/General Liberal Arts and Sciences" degree (CIP Code 24.0101) or reported as a BLS degree (CIP Code 24.0199)		
Implementation Date (semester and year):		

REQUIRED ATTACHMENTS FOR NEW PROGRAMS *NOT* REQUIRING STATE APPROVAL:

1. **Rationale statement** (Why is this additional program needed? What purposes will it serve?)
2. **Impact Statement** (Provide details about the Library, space, budget, technology, and impacts created by this program change. Supporting statements from the Library, IT Department, etc. evaluating the resource impact and feasibility of adding the new program are required.)
3. **Catalog Copy** (Provide the complete Catalog Description for the proposed new program)

Department Chair Approval: Jennifer Polack **Date:** 10/24/14

CCC Chair Approval: Dawn S. Bowen Digitally signed by Dawn S. Bowen DN: cn=Dawn S. Bowen, o, ou, email=dbowen@umw.edu, c=US Date: 2014.11.20 11:26:46 -05'00' **Date:**

Dean Approval: **Date:**

UCC Chair Approval: **Date:**

***Provost Approval:** **Date:**

**Required only in cases of proposals for new concentrations, new minors, or new majors that do not involve a new degree*

REQUIRED ATTACHMENTS FOR CHANGES TO EXISTING PROGRAMS:

1. Rationale statement (Why is this program change needed? What purposes will it serve?)

The computer science department seeks to update the requirements for the major. These updates have been made after reviewing the Computer Science Curricula 2013 report (CS2013) created by the Joint Task Force on Computing Curricula. This report was sponsored by the two principal academic and professional organizations for our discipline: the Association for Computing Machinery (ACM) and the IEEE Computer Society. It was released in Dec 2013 and its recommended curriculum represents the consensus view of the very best undergraduate computer science programs in the world.

One notable change that CS2013 introduces pertains to the expectations for traditional mathematics coursework. In the CS2013 report, it is noted that "...while we do not specify such requirements, we note that undergraduate CS students need enough mathematical maturity to have the basis on which to then build CS-specific mathematics (for example, as specified in the Discrete Structures Knowledge Area), which, importantly, does not explicitly require any significant college-level coursework in calculus, differential equations, or linear algebra." We concur with these statements. As we have seen our discipline develop, and its application areas evolve, we have recognized that nine hours of coursework in these traditional, theoretically-based mathematical areas is not the best use of CPSC majors' time. These requirements reflect the legacy of our field's roots, but updated pedagogy demands a different approach to the curriculum.

Mathematical reasoning continues to be bedrock for our discipline. However, over the past decade or so, computer science programs (including our own) have increasingly incorporated that reasoning directly into the relevant courses, instead of depending on a second department's coursework (namely, Mathematics) to supply it. This allows the treatment of that material to be more focused and directly applicable to the analytical problems our students will face, rather than being delivered from a more distant "mathematics in general" point of view. Accordingly, we believe that CPSC 125A *Introduction to Discrete Mathematics* covers all of the required content in the Discrete Structures Knowledge Area that CS2013 identifies. And as we move the course from 3 to 4 credits, we will include content in applied linear algebra that will support courses within the major.

CS2013 recommends including the following required core knowledge areas: Discrete Structures, Algorithms and Complexity, Architecture and Organization, Operating Systems, Software Engineering, and Software Development Fundamentals. We have proposed changes to the number of credits associated with the following courses, increasing them from 3 to 4 credits: *CPSC 125A Discrete Mathematics*, *CPSC 326 Theory of Computation*, *CPSC 350 Database Applications*, and *CPSC 405 Operating Systems*. The new four-credit versions will allow additional content coverage and lab practica to better provide students with the knowledge essential for computer science graduates.

The revised major will have the following required courses: CPSC 125A, 220, 225, 240, 305, 326, 340, 350, 405 and 430. Additionally, students will be required to complete one 3-credit Computer Science course numbered 300 or higher and two 3-credit Computer Science courses numbered 400 or higher (except Computer Science 499). We feel confident that these revised requirements emphasize the appropriate content areas within computer science and also provide flexibility for students to pursue interests through departmental electives.

In addition to the content changes noted above, we are reordering the content in two of our existing courses. Students currently take *CPSC 230A Data Structures* prior to taking *CPSC 330 Object-oriented Analysis and Design* (OOA&D). *CPSC 230A* currently serves as a prereq to *CPSC 330* to provide students with additional programming experience prior to taking the *OOA&D* course. We have determined that students would better be served by taking the *OOA&D* course before taking the Data Structures course because the *OOA&D* course is a more natural sequel to *CPSC 220 Programming and Problem Solving*: it builds on the 220 content in a more direct way by letting students combine those techniques to form more complex programs. Additionally, *Data Structures* is a more technically rigorous and complex course, and students would have a greater chance of success with an additional semester of programming. To reflect the new order of the courses in our curriculum, we are proposing to renumber these courses as *CPSC **240** Object-oriented Analysis and Design* and *CPSC **340** Data Structures and Algorithms*. This change in course order results in a series of prerequisite change requests for other courses to make the new prerequisites reflect the appropriate course in terms of content and/or expected programming experience level. The courses with new prerequisites related to the resequencing of Data Structures and OOA&D are: CPSC 326, CPSC 340, CPSC 350, CPSC 401, CSPC 415, CPSC 425, CPSC 440, and CPSC 444. The new course reordering also impacts the

requirements for the CPSC minor.

Additionally, we are proposing to eliminate the concentrations within our major and create one unified program of study for all Computer Science majors. The details of this change are found in a separate CPSC major change request specifically related to the concentrations.

The table below shows a summary of the proposed course changes:

Course Number	**Proposed Title**	**Change from Current Catalog**
CPSC 125A	Introduction to Discrete Mathematics	o Existing course o Proposed to expand from 3-credits to 4-credits, and will include applied linear algebra content.
CPSC 219	Foundations for Data Science	o New 3-credit course
CPSC 220	Computer Programming and Problem Solving	o Existing 4-credit course o Requesting change to prerequisites (was [CPSC 109 or 110 or permission of instructor]; it will become [CPSC 109 or 110 or 219 or permission of instructor])
CPSC 225	Software Development Tools	o New 1-credit course
CPSC 240	Object-Oriented Analysis and Design	o Existing 4-credit course o Currently numbered CPSC 330; renumbering requested to reflect the new position within the major course sequence o Requesting change to prerequisites (was CPSC 230 and will become CPSC 220 & CPSC 225)
CPSC 305	Computer Systems and Architecture	o Existing 4-credit course o Change to prerequisite (was [CPSC 230A and CPSC 125A or MATH 201] and will become [CPSC 220 and CPSC 225] or CPSC 230A)
CPSC 326	Theory of Computation	o Existing course with a proposed new course title (was *Theoretical Foundations of Computing* and will become *Theory of Computation*) o Proposed to expand from 3-credits to 4-credits (to include content in applied algorithms) o Requesting change to prerequisites (was [CPSC 230A and MATH 122] and [either CPSC 125A or MATH 201]; will become CPSC 220 and CPSC 125A)
CPSC 340	Data Structures and Algorithms	o Existing 4-credit course o Currently numbered CPSC 230; renumbering requested to reflect the new position within the major course sequence

		o Requested change to prerequisites (was CPSC 220; will become [CPSC 240 and CPSC 225] or CPSC 230)
CPSC 345	Introduction to Computer Security	o New 3-credit course
CPSC 350	Applications of Databases	o Existing course o Proposed to expand from 3-credits to 4-credits to include additional content on load balancing, model-view controller and defensive programming. o Requesting change to prerequisites (was CPSC 230A; will become [CPSC 225 and a C or better in CPSC 220] or CPSC 230A)
CPSC 370-377	Special Topics in Computer Science	o Existing special topics course o Change course description to remove the reference to CPSC 230A. This reference is not accurate anyway as the prerequisite could be CPSC 220 or CPSC 219.
CPSC 391	Special Projects in Computer Science	o Existing individual project course. o Requesting change of prerequisites (was [CPSC 230A] and will become [permission of the instructor].
CPSC 401	Organization of Programming Languages	o Existing course o Requesting change to prerequisites (was CPSC 326 and 330; will become CPSC 326 and CPSC 340)
CPSC 405	Operating Systems and Systems Programming	o Existing course with a new course title (was *Operating Systems* and will become *Operating Systems and Systems Programming*) o Proposed to expand from 3-credits to 4-credits to include additional content on cluster computing and parallelism. o Requesting change to prerequisites (was CPSC 305 and CPSC 330; will become CPSC 305)
CPSC 414	Network Principles and Applications	o Existing course o Requesting change to prereqs (was [CPSC 125A or Math 201; and CPSC 330] and will become [CPSC 220]
CPSC 415	Artificial Intelligence	o Existing 3-credit course o Requesting change to prerequisites (was CPSC 125 and 230A; will become CPSC 125, and either 230A or 240)
CPSC 419	Data Mining	o Existing 3-credit course o Requesting change to prerequisites (was CPSC 220;

		will become CPSC 219 or 220) o Updating catalog description to reflect current course topics and correct a typo.
CPSC 420	Modeling and Simulation	o Existing 3-credit course o Requesting change to prerequisites (was CPSC 220; will become CPSC 219 or 220)
CPSC 425	Parallel Processing	o Existing 3-credit course o Requesting change to prerequisites (was CPSC 230A; will become CPSC 340) o Small change to course description to correct errors and better reflect course content.
CPSC 430	Software Engineering	o Existing 4-credit course o Requesting change to prerequisites (was CPSC 330; will become CPSC 330 or both CPSC 340 and 350) o Updating catalog description to reflect current course topics
CPSC 440	Game Programming	o Existing 3-credit course o Change prerequisite from CPSC 230A to CPSC 34-
CPSC 444	Three Dimensional Computer Graphics	o Existing 3-credt course o Change prerequisite from CPSC 230A to CPSC 340
CPSC 470-477	*Selected Topics in Computer Science*	o Existing varirable credit special topics course. o Requesting update to catalog copy to reflect recent course offerings in this category.

2. Impact Statement (Provide details about the Library, space, budget, technology, and impacts created by this program change. Supporting statements from the Library, IT Department, etc. evaluating the resource impact and feasibility of the program change are required.)

The proposed revisions to the Computer Science major, if approved, will not require any additional lab or teaching space, faculty positions, technology, or budget. The current Computer Science Department faculty have the necessary expertise to offer all of the courses required for the revised major.

The current CPSC major requirements vary from 41-44 credits and also include between 6-9 additional hidden prerequisite credits (depending on concentration). The proposed major will require 46 credits; however these 46 credits represent *all* courses and prerequisites required for the major, which we feel is more forthright for students contemplating the major.

We anticipate that the demand for the major will remain constant.

3. Catalog Copy (Provide the *existing* Catalog Description and the complete statement of the *proposed* new Catalog description that reflects the program changes)

Existing Catalog Description

The Computer Science Program

Computer Science provides the kind of dynamic, interactive work environment few fields can match. Software developers and analysts invariably work in teams to tackle cutting-edge projects. We join forces with scientists, doctors, military commanders, social psychologists, and others to produce solutions that are bigger than the sum of their parts. We enjoy the satisfaction of working with our peers to make a common vision become reality.

Computer software is probably the most malleable medium invented for human artistry. Unlike the gadgets of the industrial revolution, which were hardwired to a single purpose, computer software essentially mimics the fluidity of the human mind, making it extensible in any direction the author sees fit. Because of this, in Computer Science, we don't just *study* what "is," but we define what is. Our goal is to invent, create, and solve problems in exciting new ways.

We offer three choices for students majoring in Computer Science: the Traditional Program, the Computer Information Systems Concentration, and the Geographic Information Systems Concentration. Each is tailor-made for a challenging and practical course of study and paves the way for a dynamic career path and advanced study in the major. We also offer a minor in Computer Science for students majoring in other disciplines and offer courses in support of the interdisciplinary Data Science minor. Students have opportunities for individual study, undergraduate research, and internships at technical firms, government offices, or software development agencies.

The academic and research programs in Computer Science are supported by a network of computer systems that employ Solaris, Linux, and Windows operating systems.

Students who complete any of the undergraduate programs in Computer Science are well prepared for a variety of entry-level positions and for continued academic work in graduate school.

Computer Science Major Requirements

Students who major in Computer Science may choose from three concentrations: the Traditional Program, the Computer Information Systems Concentration, or the Geographic Information Systems Concentration.

1. The Traditional Program.
Students study the classic discipline of Computer Science, with emphasis on the theoretical foundation and practical applications of computers and computer software. Courses explore such subjects as system architecture, object-oriented design, and computational theory, in addition to the rich mathematical underpinnings that support these topics. Graduates are well-equipped to solve problems in a broad spectrum of application areas and begin satisfying careers as software engineers, system architects, or application developers.

Computer Science: Traditional Program 41 credits as follows:
A. The following required courses: CPSC 220, 230A, 305, 326, 330, 350, 405, and 430.
B. One course chosen from MATH 300 or 312 or 351.
C. Two courses, minimum three credits each, in CPSC numbered 400 or higher (except Computer Science 499) that were not used to satisfy any of the preceding requirements. CPSC 491 fulfills this requirement if said course is at least three credits.
D. One course, minimum three credits, in CPSC or MATH 300 or higher, CPSC 391, 491, or 499.
A maximum of 3 credits of CPSC 499 can be counted toward the Computer Science major. Note that MATH 122 (Calculus II) is a prerequisite for CPSC 326 and should be taken before

the junior year. Also, note that CPSC 125A is a prerequisite for CPSC 305 and 326 and should be taken before the junior year.

2. The Computer Information Systems Concentration in Computer Science.
This combines the foundations of Computer Science with an applied approach to application development and computing in a business environment. It offers our students an avenue to consider the use of computing as it applies to problems in business and related fields. Together with courses in economics, decision support, and accounting, this alternative focuses on how technology applies in an organizational setting. Graduates are well-positioned for careers as software developers, systems analysts, network and systems administrators, project leaders, database administrators, or business analysts.

Computer Science: CIS Concentration 44 credits as follows:
A. The following required courses: CPSC 220, 230A, 310, 330, 350, 430; BUAD 132; ECON 200 or higher; and MATH 200.
B. One course chosen from BUAD 300 or 353.
C. One course chosen from CPSC 414 or 448.
D. One course, minimum three credits, in CPSC numbered 400 or higher (except CPSC 499) that was not used to satisfy any of the preceding requirements. CPSC 491 fulfills this requirement if said course is at least three credits.
E. One course, minimum three credits, in CPSC, BUAD or MATH numbered 300 or higher that was not used to satisfy any of the preceding requirements. CPSC 391, 491, or 499 fulfills this requirement if said course is at least three credits.
A maximum of 3 credits of Computer Science 499 can be counted toward the Computer Science major. Note that Computer Science 125A is a prerequisite to Computer Science 414 and should be taken before the junior year. Business Administration 131 is a prerequisite for Business Administration 132.

3. The Geographic Information Systems Concentration in Computer Science.
The GIS Concentration combines the foundations of Computer Science with specialized study of geospatial information systems. This program is designed to address the need for customized desktop and web-based applications related to business, geospatial intelligence, education, health care, and numerous other employment fields. The field encompasses integrated hardware, software, and database systems that are capable of capturing, storing, analyzing, and displaying geographic information.

Computer Science: GIS Concentration
41 credits as follows:
A. The following required courses: CPSC 220, 230A, 330, 350; GISC 351 and 450.
B. One course selected from: GISC 200 or GEOG 250 or EESC/GEOL 205.
C. One course selected from: GISC 440 or 460 or 471 or 491.
D. One course, minimum three credits, in Computer Science numbered 400 or higher (except Computer Science 499) that was not used to satisfy any of the preceding requirements. Computer Science 491 fulfills this requirement if said course is at least three credits.
E. Two courses, minimum six credits, in Computer Science numbered 300 or higher that were not used to satisfy any preceding requirements. Computer Science 391, 491, or 499 fulfills this requirement if said course is at least three credits.
F. One course, minimum 3 credits, chosen from CPSC 430 (4 credits), 491, or 499. A maximum of 3 credits of CPSC 499 can be counted toward the CPSC major.
Computer Science Minor Requirements (20 credits): Any Computer Science course, of at least 3 credits, numbered 100 or higher; CPSC 220 and 230A; any three (3) Science courses numbered 300 or above, for a total of at least 9 credits.

Computer Science Course Offerings

Computer Science course offerings will be found under the 4 letter code of CPSC in the course listings.

104 – The Internet: Technology, Information, and Issues (3)
A survey of the technology and issues underlying the use of the Internet for communication, resource discovery, research, and dissemination of information in multimedia formats. Topics include an introduction to Internet protocols, Internet history and development, electronic mail, use and functions of a Web browser, accessing Internet services and resources, using the Internet for research, Website design and implementation, and social, legal, and ethical issues related to using the Internet.

105 – Problem-solving with Databases (3)
Introduces relational databases as a common problem-solving tool for business, scientific, and personal applications. Covers issues of information representation, abstraction, redundancy and inconsistency. Introduces database schema design based on the relational model, including semantic modeling, integrity constraints, and language-based queries. Students build sample database solutions based on real-world problem domains. May not be taken after passing CPSC 350.

106 – Digital Storytelling (3)
People have been telling stories since the beginning of time, but how is storytelling evolving in the digital age? This course explores how computers are being used to tell stories. We'll study text-based technologies – blogging, the web – and how those models have changed the way we publish and disseminate narratives. We'll also study the roles of audio, video, and images in narrative: computer animation, the ethics of altering digital images, and the Story Corps project. Students will use technology including blogs, virtual worlds, and computer games to create and tell their own stories. No previous computer experience is necessary.

109 – Introduction to Modeling and Simulation (3)
This course introduces students to the concepts of modeling and simulation as tools for solving problems in the sciences. Students will be introduced to several modeling and simulation tools and will learn how to decompose problems so they can be represented and solved with the tools. Agent models and system models will be introduced. Example problems to demonstrate the modeling and simulation techniques and tools drawn from a number of scientific fields and will introduce basic problems that will not require depth of knowledge in any particular field of science. Examples of these problems include forest fires, predatory problems, transmission of diseases, chemical reactions, and elementary particle simulations. Students completing the course will be able to model complex systems and have attained programming skills equivalent to those learned in CPSC 110. Successful completion of this course is sufficient to continue on to CPSC 220. No previous programming experience or computer background is expected.

110 – Introduction to Computer Science (3)
This course provides a foundation in computer science for a student who does not have prior programming experience. It provides sufficient support to permit a student to continue in the major program. Topics include an introduction to the algorithm and program development process using a high-level structured programming language and the department's computing facilities. Supervised hands-on experience provided. May not be taken for graded credit after passing any Computer Science course numbered 220 or higher.

125A – Introduction to Discrete Mathematics (3)
Designed to prepare beginning Computer Science majors for advanced study by emphasizing the components of Discrete Mathematics especially related to Computer Science. An introduction to topics which may include number systems, logic, methods of

proof, counting techniques, mathematical induction, sets, relations, functions, matrices, graphs and trees.

220 – Computer Programming and Problem Solving (4)
Prerequisite: CPSC 109 or 110 or permission of instructor. Continued coverage of disciplined problem-solving and algorithmic development including emphasis on procedural and data abstraction. Topics include elementary data structures such as arrays, files, and classes. The notions of data modeling and the linking of data type definitions with their associated operations is introduced. Study of program design, coding, debugging, testing, and documentation in a higher level language that supports the object-oriented paradigm. Intended for students who have had previous programming experience.

230A – Data Structures and Algorithms (4)
Prerequisite: Grade of C or better in CPSC 220. Continued study of data modeling and incorporation of abstract data types including linked lists, stacks, queues, heaps, trees, and graphs. Study of advanced sorting and searching techniques. Provides experience in the use of algorithm analysis. Continued study of program design, coding, debugging, testing, and documentation in an object-oriented higher level language.

270 – Introductory Special Topics in Computer Science (3)
Prerequisite: Specified by Instructor. Lecture or seminar class dealing with a topic accessible without extensive Computer Science experience. Most recently this has included topics such as JavaScript Programming. May be repeated for credit with a change in topic.

302 – Computer Ethics (3)
Prerequisite: CPSC 110 or 220. An examination of issues and events related to ethics, professional conduct and social responsibility as they apply to the field of Computer Science. Includes study of ethical responsibilities and behaviors appropriate for computer scientists.

305 – Computer Systems and Architecture (4)
Prerequisites: (1) CPSC 125A or MATH 201, and (2) CPSC 230A. This course examines the basic operation of computing systems. It takes a bottom-up approach covering each major component of such systems including hardware, logic circuit design, CPU instruction sets, assemblers, and compilers. Students will gain experience programming in assembly language and C.

310 – Computer Information Systems (4)
Prerequisite: CPSC 220. This course introduces the student to the use and implications of information technology in the business environment. This course covers such topics as data management, networks, analysis and design, computer hardware and software, decision support systems, database management systems, transaction processing systems, executive information systems, and expert systems. It also provides activity with computer-based and non-computer-based problems/cases and includes real-world programming projects that are implemented using a high-level programming language.

320 – Statistics for Computer Scientists (3)
Prerequisites: MATH 121 and CPSC 220. A calculus-based mathematical statistics course with special emphasis on applications in Computer Science. Topics include probability, mass/density functions, discrete/continuous random variables, and general data analysis using a statistical software package.

326 – Theoretical Foundations of Computing (3)
Prerequisites: (1) CPSC 125A or MATH 201, and (2) CPSC 230A and MATH 122. Emphasis will be on structures and concepts relating to the theory of computation and the different types of

theoretical machines that are mathematical models for actual physical processes. Topics may include automata theory, Turing machine theory, formal language theory, and the halting problem.

330 – Object-Oriented Analysis and Design (4)
Prerequisite: CPSC 230A. Theory and practice of the object-oriented software development paradigm. Focus is on major design principles such as abstraction, encapsulation, inheritance, polymorphism, aggregation, and visibility. Modeling notations for capturing and critiquing designs. Introduction to the concept of design patterns, and coverage of a catalog of common patterns. Students work in team projects to develop collaborative software solutions in an object-oriented language.

348 – Web Application Development (3)
Prerequisite: CPSC 220. This course provides an introduction to topics related to developing Web applications including: overview of Web technology; introduction to networks and the Internet; popular scripting languages such as PHP, JSP, and JavaScript; electronic payment systems; databases; and security.

350 – Applications of Databases (3)
Prerequisite: CPSC 230A. Presents logical database modeling and design, emphasizing the construction and analysis of relational schemas. Covers semantics, integrity constraints, functional dependencies, and table normalization. Practical use of relational algebraic operators, the tuple relational calculus, and their expression in declarative query languages. Introduces basic data mining concepts. May involve student team projects to develop small but representative data collection and analysis applications.

370 through 377 – Special Topics in Computer Science (1–4)
Prerequisites: Specified by Instructor. Lecture or seminar class dealing with a topic accessible after completing CPSC 230A. Most recently this has included topics such as Data Science, and Computational Linguistics. May be repeated for credit with a change in topic.

391 – Special Projects in Computer Science (1-4)
Prerequisite: CPSC 230A or higher. Intensive individual investigation of significant research problem under the direction of a faculty member. GPA and course prerequisites apply.

401 – Organization of Programming Languages (3)
Prerequisites: CPSC 326 and 330. A course in programming language construction and design emphasizing the run-time behavior of programs. Alternative implementations of programming language constructs are considered. Techniques for language definition may also be discussed.

405 – Operating Systems (3)
Prerequisites: CPSC 305 and 330. This course examines the abstractions above the hardware that make a computer usable to both programmers and users. These abstractions include concurrent programming, virtual addressing, transactions, and virtualization. Many of these abstractions are useful not just for operating system kernel development but for development of any large scale programming project. Students apply these concepts by working with a real operating system kernel. Programming intensive.

414 – Network Principles and Applications (3)
Prerequisite: CPSC 125A or MATH 201; and CPSC 330. This course provides an introduction to the basic principles of networking. Topics covered in the course include: network topologies, protocols, the OSI Model, methods of data transmission, error detection and correction, TCP/IP, network security and other topics as time permits. This course is

theoretical and concept-oriented rather than consisting of the details of specific network packages.

415 – Artificial Intelligence (3)
Prerequisites: CPSC 125A and 230A. A survey of current artificial intelligence topics including informed search, knowledge representation, knowledge-based systems, and machine learning. Other topics such as image processing, robotics, and language processing, may also be covered. Artificial intelligence programming projects are required.

419 – Data Mining (3)
Prerequisites: CPSC 220. Practical knowledge of data mining and information retrieval. Students will examine the theoretical foundations of a variety of techniques, gain experience with these techniques using open source software, and learn how to apply them to real-world problems. Topics include decision trees, Naïve Bayes, Probabilistic retrieval models, clustering, support vector machines and approaches to web mining.

420 – Modeling and Simulation (3)
Prerequisite: CPSC 220. A robust introduction to techniques of mathematical modeling and computational simulation applied to practical problems. Topics include system dynamics approaches, discrete-event simulation, and agent-based models. Students complete small projects on topics as diverse as population growth, epidemic transmission, queuing theory, and forest fire outbreaks.

425 – Parallel Computing (3)
Prerequisite: CPSC 230A. This course provides an introduction to parallel computing, covering topics including parallel Architectures, programming techniques and libraries, the study of existing parallel computing systems, and performance analysis. Students will use a variety of hardware to explore current libraries and methods used for parallel programming and complete a project using the material they learn.

430 – Software Engineering (4)
Prerequisite: CPSC 330. Techniques for modeling, designing, implementing, and managing large scale computer programs are studied. Studies include Object-Oriented Analysis and Design, modeling using UML, and application development with a CASE tool. Students work in groups and apply the techniques studied to semester-long projects.

440 – Game Programming (3)
Prerequisite: CPSC 230A. Students will design, develop, and implement computer games that involve real-time, event-driven, and multimedia programming techniques. Students learn the history of computer games and the elements of video game design and architecture.

444 – Three Dimensional Computer Graphics (3)
Prerequisites: CPSC 230A. The study of three-dimensional modeling involving the use of light, color, texture and transformation; visible surface detection; parallel and perspective projections; clipping algorithms.

448 – Advanced Web Application Development (3)
Prerequisite: CPSC 348 or 350 or permission of the instructor. An examination and application of contemporary software technologies focused on providing Web-based services and applications. Students work in teams to design and develop leading-edge projects. In the recent past this has included developing advanced data-driven applications employing AJAX, PHP, and a database management system.

470 through 477 – Selected Topics in Computer Science (1-4)

Prerequisite: Specified by Instructor. Treatment of selected topics in Computer Science. Most recently this has included topics such as E-Commerce and Programming with AJAX. May be repeated for credit with a change in topic.

491 –Individual Study in Computer Science (1-4)
Prerequisite: Departmental permission. Individual study under the direction of a member of the department. GPA and course prerequisites apply. May be repeated for credit with a change in topic. Two semesters of 3 credits of study required for graduation with Departmental Honors.

499 – Internship (Credits variable)
Supervised off-campus experience with an academic component, developed in consultation with the department. Only 3 credits may count toward the major and minimum GPA and course prerequisites apply.

Proposed Catalog Description

We include the entire updated catalog listing for Computer Science below. This includes proposed changes to the major, minor, and individual courses. Additional change request forms provide details and justifications for the proposed changes to individual courses and the minor.

The Computer Science Program

Computer Science provides the kind of dynamic, interactive work environment few fields can match. Software developers and analysts invariably work in teams to tackle cutting-edge projects. We join forces with scientists, doctors, military commanders, social psychologists, and others to produce solutions that are bigger than the sum of their parts. We enjoy the satisfaction of working with our peers to make a common vision become reality.

Computer software is probably the most malleable medium invented for human artistry. Unlike the gadgets of the industrial revolution, which were hardwired to a single purpose, computer software essentially mimics the fluidity of the human mind, making it extensible in any direction the author sees fit. Because of this, in Computer Science, we don't just *study* what "is," but we define what is. Our goal is to invent, create, and solve problems in exciting new ways.

The major is tailor-made for a challenging and practical course of study and paves the way for a dynamic career path and advanced study in the computing and related fields. We also offer a minor in Computer Science for students majoring in other disciplines and offer courses in support of the interdisciplinary Data Science minor. Students have opportunities for

individual study, undergraduate research, and internships at technical firms, government offices, or software development agencies.

Students considering a career with the federal government should be aware that the US Department of Operations and Personnel Management standards require a minimum of 15 credit hours of mathematics in order for employees to be classified as a "Computer Scientist." Students interested in federal employment in this classification are encouraged to take MATH 121, MATH 122, MATH 200, and any additional MATH course numbered 300 or higher. A minor in Applied Mathematics would also provide appropriate preparation.

Students who are considering a career in IT or project management or who are interested in pursuing an MBA should pursue the Computer Science major and a minor in Business Administration.

Students who are interested in a career in Geospatial Systems should consider completing the requirements for Computer Science major and the Geographic Information Science certificate.

Computer Science Major Requirements 46 *credits as follows:*

A. The following required courses: CPSC 125A, 220, 225, 240, 305, 326, 340, 350, 405, and 430.

B. Two courses, minimum three credits each, in CPSC numbered 400 or higher (except Computer Science 499) that were not used to satisfy any of the preceding requirements. CPSC 491 fulfills this requirement if said course is at least three credits.

C. One course, minimum three credits, in CPSC or MATH numbered 300 or higher, including CPSC 391, 491, or 499 provided the course is at least 3 credits.

A maximum of 3 credits of CPSC 499 can be counted toward the Computer Science major.

Computer Science Minor Requirements (20 credits): Any Computer Science course, of at least 3 credits, numbered 100 or higher; CPSC 220 and 240; any three (3) Computer Science courses numbered 300 or above, for a total of at least 9 credits.

Computer Science Course Offerings

Computer Science course offerings will be found under the 4 letter code of CPSC in the course listings.

104 – The Internet: Technology, Information, and Issues (3)
A survey of the technology and issues underlying the use of the Internet for communication, resource discovery, research, and dissemination of information in multimedia formats. Topics include an introduction to Internet protocols, Internet history and development, electronic mail, use and functions of a Web browser, accessing Internet services and resources, using the Internet for research, Website design and implementation, and social, legal, and ethical issues related to using the Internet.

105 – Problem-solving with Databases (3)
Introduces relational databases as a common problem-solving tool for business, scientific, and personal applications. Covers issues of information representation, abstraction, redundancy and inconsistency. Introduces database schema design based on the relational model, including semantic modeling, integrity constraints, and language-based queries. Students build sample database solutions based on real-world problem domains. May not be taken after passing CPSC 350.

106 – Digital Storytelling (3)
People have been telling stories since the beginning of time, but how is storytelling evolving in the digital age? This course explores how computers are being used to tell stories. We'll study text-based technologies – blogging, the web – and how those models have changed the way we publish and disseminate narratives. We'll also study the roles of audio, video, and images in narrative: computer animation, the ethics of altering digital images, and the Story Corps project. Students will use technology including blogs, virtual worlds, and computer games to create and tell their own stories. No previous computer experience is necessary.

109 – Introduction to Modeling and Simulation (3)
This course introduces students to the concepts of modeling and simulation as tools for solving problems in the sciences. Students will be introduced to several modeling and simulation tools and will learn how to decompose problems so they can be represented and solved with the tools. Agent models and system models will be introduced. Example problems to demonstrate the modeling and simulation techniques and tools drawn from a number of scientific fields and will introduce basic problems that will not require depth of knowledge in any particular field of science. Examples of these problems include forest fires, predatory problems, transmission of diseases, chemical reactions, and elementary particle simulations. Students completing the course will be able to model complex systems and have attained programming skills equivalent to those learned in CPSC 110. Successful completion of this course is

sufficient to continue on to CPSC 220. No previous programming experience or computer background is expected.

110 – Introduction to Computer Science (3)
This course provides a foundation in computer science for a student who does not have prior programming experience. It provides sufficient support to permit a student to continue in the major program. Topics include an introduction to the algorithm and program development process using a high-level structured programming language and the department's computing facilities. Supervised hands-on experience provided. May not be taken for graded credit after passing any Computer Science course numbered 220 or higher.

125A – Introduction to Discrete Mathematics (4)
Designed to prepare beginning Computer Science majors for advanced study by emphasizing the components of Discrete Mathematics especially related to Computer Science. Topics include number systems, logic, methods of proof, counting techniques, mathematical induction, sets, relations, functions, vectors, matrices, graphs and trees.

219 – Foundations for Data Science (3)
Skills and tools in acquiring, parsing, manipulating, and preparing data for statistical analysis.

220 – Computer Programming and Problem Solving (4)
Prerequisite: CPSC 109 or 110 or 219 or permission of instructor.
Continued coverage of disciplined problem-solving and algorithmic development including emphasis on procedural and data abstraction. Topics include elementary data structures such as arrays, files, and classes. The notions of data modeling and the linking of data type definitions with their associated operations is introduced. Study of program design, coding, debugging, testing, and documentation in a higher level language that supports the object-oriented paradigm. Intended for students who have had previous programming experience.

240 - Object-Oriented Analysis and Design (4)
Prerequisite: (1) Grade of C or better in CPSC 220 and (2) CPSC 225.
Theory and practice of the object-oriented software development paradigm. Focus is on major design principles such as abstraction, encapsulation, inheritance, polymorphism, aggregation, and visibility. Modeling notations for capturing and critiquing designs. Introduction to the concept of design patterns, and coverage of a catalog of common patterns. Students work in team projects to develop collaborative software solutions in an object-oriented language.

270 – Introductory Special Topics in Computer Science (3)

Prerequisite: Specified by Instructor. Lecture or seminar class dealing with a topic accessible without extensive Computer Science experience. Most recently this has included topics such as JavaScript Programming. May be repeated for credit with a change in topic.

302 – Computer Ethics (3)
Prerequisite: CPSC 110 or 220. An examination of issues and events related to ethics, professional conduct and social responsibility as they apply to the field of Computer Science. Includes study of ethical responsibilities and behaviors appropriate for computer scientists.

305 – Computer Systems and Architecture (4)
Prerequisites: (1) CPSC 220 and CPSC 225 or (2) CPSC 340. This course examines the basic operation of computing systems. It takes a bottom-up approach covering each major component of such systems including hardware, logic circuit design, CPU instruction sets, assemblers, and compilers. Students will gain experience programming in assembly language and C.

310 – Computer Information Systems (4)
Prerequisite: CPSC 220. This course introduces the student to the use and implications of information technology in the business environment. This course covers such topics as data management, networks, analysis and design, computer hardware and software, decision support systems, database management systems, transaction processing systems, executive information systems, and expert systems. It also provides activity with computer-based and non-computer-based problems/cases and includes real-world programming projects that are implemented using a high-level programming language.

320 – Statistics for Computer Scientists (3)
Prerequisites: CPSC 220. A calculus-based mathematical statistics course with special emphasis on applications in Computer Science. Topics include probability, mass/density functions, discrete/continuous random variables, and general data analysis using a statistical software package.

326 – Theory of Computation (4)
Prerequisites: CPSC 125A and CPSC 220. Covers structures and concepts relating to the underlying theory of computation and mathematical models of actual physical processes. Also covers a repertoire of advanced algorithms for data processing, and the asymptotic analysis of those algorithms to describe their running time and space requirements. Topics may include formal languages, automata theory, Turing machines, the halting problem, NP-completeness, searching and traversal algorithms, dynamic programming, compression algorithms, and random number generation.

340 – Data Structures and Algorithms (4)
Prerequisite: (1) CPSC 225 and (2) Grade of C or better in CPSC 240. Continued study of data modeling and incorporation of abstract data types including linked lists, stacks, queues, heaps, trees, and graphs. Study of advanced sorting and searching techniques. Provides experience in the use of algorithm analysis. Continued study of program design, coding, debugging, testing, and documentation in an object-oriented higher level language.

345 – Introduction to Computer Security (3)
Prerequisite: CPSC 220. Provides an introduction to computer security. The focus is on providing the students a wide overview of current computer security. Topics covered include, but are not limited to, basic cryptography, network security, system security, wireless security and mobile security. In addition, course labs provide a more hands-on, in-depth exploration of specific topics.

348 – Web Application Development (3)
Prerequisite: CPSC 220. This course provides an introduction to topics related to developing Web applications including: overview of Web technology; introduction to networks and the Internet; popular scripting languages such as PHP, JSP, and JavaScript; electronic payment systems; databases; and security.

350 – Applications of Databases (4)
Prerequisite: (1) CPSC 225 and Grade of C or better in CPSC 220 or (2) CPSC 340. Presents basic techniques for the design and implementation of database-driven web applications. Topics include the design of relational and NoSQL databases and scaling techniques such as the use of load balancing and distributed systems. Programming intensive using a dynamic high-level general-purpose language.

370 through 377 – Special Topics in Computer Science (1–4)
Prerequisites: Specified by Instructor. Lecture or seminar class. Most recently this has included topics such as Information Visualization and Computational Linguistics. May be repeated for credit with a change in topic.

391 – Special Projects in Computer Science (1-4)
Prerequisite: Permission of the instructor. Intensive individual investigation of significant research problem under the direction of a faculty member. GPA and course prerequisites apply.

401 – Organization of Programming Languages (3)

Prerequisites: CPSC 326, and 340. A course in programming language construction and design emphasizing the run-time behavior of programs. Alternative implementations of programming language constructs are considered. Techniques for language definition may also be discussed.

405 – Operating Systems and Systems Programming (4)
Prerequisites: CPSC 305. This course examines the abstractions above the hardware that make a computer usable to both programmers and users. These abstractions include *processes, context switching*, concurrent programming, *semaphores*, virtual addressing, transactions, *access control*, and virtualization. Many of these abstractions are *the foundation for operating system kernel development. The abstractions are also applicable to any lar*ge scale programming project. Programming intensive.

414 – Network Principles and Applications (3)
Prerequisite: CPSC 220. This course provides an introduction to the basic principles of networking. Topics covered in the course include: network topologies, protocols, the OSI Model, methods of data transmission, error detection and correction, TCP/IP, network security and other topics as time permits. This course is theoretical and concept-oriented rather than consisting of the details of specific network packages.

415 – Artificial Intelligence (3)
Prerequisites: CPSC 125A, and 240. A survey of current artificial intelligence topics including informed search, knowledge representation, knowledge-based systems, and machine learning. Other topics such as image processing, robotics, and language processing, may also be covered. Artificial intelligence programming projects are required.

419 – Data Mining (3)
Prerequisites: CPSC 219 or 220. Practical knowledge of data mining, machine learning and information retrieval. Students will examine the theoretical foundations of a variety of techniques, gain experience with these techniques using open source software, and learn how to apply them to real-world problems. Topics include decision trees, Naïve Bayes, probabilistic retrieval models, clustering, support vector machines, approaches to web mining, and scalable machine learning applications.

420 – Modeling and Simulation (3)
Prerequisite: CPSC 219 or 220. A robust introduction to techniques of mathematical modeling and computational simulation applied to practical problems. Topics include system dynamics approaches, discrete-event simulation, and agent-based models. Students complete small projects on

topics as diverse as population growth, epidemic transmission, queuing theory, and forest fire outbreaks.

425 – Parallel Computing (3)
Prerequisite: CPSC340. This course provides an introduction to parallel computing, covering topics including parallel architectures, programming techniques and libraries, the study of existing parallel computing systems, and performance analysis. Students will use a variety of hardware to explore current libraries and methods used for parallel programming.

430 – Software Engineering (4)
Prerequisite: both CPSC 340 and 350. Techniques for modeling, designing, implementing, and managing large scale computer programs are studied. Studies include software process models, modeling using UML, and application development with a CASE tool. Students work in groups and apply the techniques studied to semester-long projects.

440 – Game Programming (3)
Prerequisite: CPSC 340. Students will design, develop, and implement computer games that involve real-time, event-driven, and multimedia programming techniques. Students learn the history of computer games and the elements of video game design and architecture.

444 – Three Dimensional Computer Graphics (3)
Prerequisites: CPSC 340. The study of three-dimensional modeling involving the use of light, color, texture and transformation; visible surface detection; parallel and perspective projections; clipping algorithms.

448 – Advanced Web Application Development (3)
Prerequisite: CPSC 348 or 350 or permission of the instructor. An examination and application of contemporary software technologies focused on providing Web-based services and applications. Students work in teams to design and develop leading-edge projects. In the recent past this has included developing advanced data-driven applications employing AJAX, PHP, and a database management system.

470 through 477 – Selected Topics in Computer Science (1-4)
Prerequisite: Specified by Instructor. Treatment of selected topics in Computer Science. Most recently this has included topics such as Cloud Computing, Animation, and Information System Security. May be repeated for credit with a change in topic.

491 –Individual Study in Computer Science (1-4)
Prerequisite: Departmental permission. Individual study under the direction of a member of the department. GPA and course prerequisites

apply. May be repeated for credit with a change in topic. Two semesters of 3 credits of study required for graduation with Departmental Honors.

499 – Internship (Credits variable)
Supervised off-campus experience with an academic component, developed in consultation with the department. Only 3 credits may count toward the major and minimum GPA and course prerequisites apply.