ABET
Self-Study Report

for the

Bachelor of Science in Computer Science Program

at

Fitchburg State University

Fitchburg, Massachusetts

July 1, 2013

CONFIDENTIAL

Table of Contents

BACKGROUND INFORMATION

A. Contact Information

Brady Chen, Department Chair
Computer Science Department
Fitchburg State University
160 Pearl Street
Fitchburg, Massachusetts 01420
xchen@fitchburgstate.edu
1-978-665-3434

Paul Weizer, Associate VP
Academic Affairs
Fitchburg State University
160 Pearl Street
Fitchburg, Massachusetts 01420
pweizer@fitchburgstate.edu
1-978-665-3272

B. Program History

The Fitchburg State Computer Science Department was founded in 1975. The Bachelor of Science in Computer Science (CS) degree was the original program offered by the department. A major in Computer Information Systems (CIS) was introduced in 2001. Both CS and CIS programs have been accredited since 10/1/2008. There have been no major changes in the required courses since then.

C. Options

The program does not offer different tracks or concentrations. Students are required to select five advanced elective courses focusing in areas of their choice.

D. Organizational Structure

The CS Curriculum Committee takes responsibility for negotiating with other departments on behalf of the program, but curricular decisions are made by the department as a whole, and ratified or denied by faculty governance. Most such changes would be reviewed by the university-wide Curriculum Committee and adopted by the All University Committee (AUC).

Day-to-day decisions are made by the faculty member teaching a specific course, possibly with consultation with colleagues. Semester-to-semester course scheduling is the responsibility of the department chair, who reports to the Associate Vice President for Academic Affairs (AVPAA). Conflicts with the AVPAA over numbers of sections and faculty loads can be referred to the Vice President for Academic Affairs (VPAA), and then to the President.

Initial faculty salaries are the province of Academic Affairs. Subsequent raises are governed by the Massachusetts State College Association (MSCA) union contract.

Faculty promotions, sabbaticals, and tenure are guided by the terms of the union contract. Promotions and tenure go through faculty governance, the Vice President of Academic Affairs, the President, and are finally approved by the Board of Trustees.

E. Program Delivery Modes

The primary delivery of instruction is by lectures. Four required courses, Introduction to Electronics, Digital Electronics, Computer Organization, and Microprocessors have dedicated laboratory facilities used in addition to lectures.

F. Program Locations

All courses are taught on campus. In recent years, all have been taught during the day, with the small exception of an occasional section of Systems Programming, which has also been taught as a 'twilight' offering at 4 or 5 PM.

G. Deficiencies, Weaknesses or Concerns from Previous Evaluation(s) and the Actions Taken to Address Them

Our last evaluation team identified:

1. one deficiency, which was an insufficient number of credit-hours for mathematics and science;
2. one weakness in assessment of our program objectives;
3. and four curricular concerns:
    - introductory core courses insufficiently stressed problem analysis and program design,
    - written communication skills insufficiently emphasized,
    - oral communications skills insufficiently emphasized, and
    - social and ethical implications of computing coverage limited.

These were addressed and solutions communicated in our interim report (see Appendix G).

1. We now require an additional math course with a course number greater than 3000 for CS majors;
2. We have continued to work on our assessment plan since the last review, and we will present below some of these changes;
3. We have made a number of tweaks in our presentation of course material to address the concerns.

H. Joint Accreditation

The program had been accredited by ABET Computing Accreditation Commission (CAC) only and currently seeking renewal by the same commission.

GENERAL CRITERIA

CRITERION 1. STUDENTS

A. Student Admissions

Students can declare a major at admission before actually matriculating. During summer academic advising sessions for admitted students, the department chair or a department faculty volunteer consults with incoming students and assists them with course registration.

A few freshmen are 'undeclared' and join our department later. Some students switch to Computer Science from other majors.

All are self-selected, as are most of those who choose to leave.

B. Evaluating Student Performance

Student performance is measured using a variety of instruments depending on the course. Programming assignments, exams, quizzes and formal presentations and reports are all part of the process used to determine student performance. Student progress is monitored through courses passed and grades earned in those courses. Although we assess many courses to measure how each course is meeting program goals, we do not attempt to measure whether each student passing a course successfully meets all of the goals we have set for the course.

A campus-wide database (called 'Web4') tracks each student's progress through the program requirements. Students can access Web4 reports themselves, but the reports are also used during meetings with their assigned department advisors each semester. Students are required to meet with their advisors before they can register for the following term. This is enforced with a 'registration pin', unique to each student and each semester, which they need to register on-line, or by the advisor's signature on a paper registration form. Web4 enforces prerequisites and will only allow students to register for a course if the prerequisite is met or in-progress. The instructor can waive a prerequisite when circumstances warrant it. Under special circumstances, only the department chair can waive program requirements.

C. Transfer Students and Transfer Courses

Fitchburg State has transfer agreements with institutions throughout the state of Massachusetts. A program called "MassTransfer" (www.mass.edu/masstransfer) is in place for students seeking to transfer academic credit among any of the Massachusetts community colleges, state universities and University of Massachusetts campuses. Through MassTransfer, a student who completed an associate degree is eligible to transfer up to 60 credits into a linked MassTransfer program at a Massachusetts state university or University of Massachusetts campus. Qualified students may also receive automatic satisfaction of most or all general education requirements, guaranteed transfer of credits, guaranteed admission, waiver of application fee, and a tuition discount. Students without an associate

degree may still transfer their credits through "MassTransfer Block." Qualified students may transfer a block of courses (34 credits) which satisfy general education, core, or distribution requirements between institutions. However, completing the MassTransfer Block does not guarantee admission. The maximum number of transfer credits that will be accepted by the University for any degree program is 75.

Additional agreements with specific institutions enable transfer students to automatically receive credit for some of our core and elective courses. Information on these additional agreements is also available through the MassTransfer web site. For example, CIS majors may receive credit for Introduction to Programming and economics courses taken at Bunker Hill Community College if they have opted for the Computer Information Systems concentration, taken the appropriate courses, and graduated with the associate degree. Similarly, CS majors may receive credit for calculus, physics, CS 1 and CS 2 taken at Bunker Hill Community College if they have opted for the Computer Science concentration, taken the appropriate courses, and graduated with the associate degree.

Students transferring to the University who choose to change majors or come from institutions with which we have no transfer agreement may graduate from the University with more than the minimum 120 credits. This may happen because some of the courses they transfer in do not directly map to our core. In other cases, the student may transfer in courses in excess of what is required for the major. When no transfer agreement exists course transfers are handled on a case-by-case basis. The department chair may review the transcript of a transfer student and assign credit for one or more of our courses based on the course syllabi and the grades earned by the student. In any circumstance, the student can still apply these courses as general (not CS-related) electives.

For students who score 3, 4, or 5 on the advanced placement (AP) examination, the university grants advanced placement status and credit in the area tested. Students entering our program with AP credits in Computer Science may choose to skip Computer Science 1. Similarly, students with AP credits in Calculus may choose to skip Calculus 1 and are not required to take the math placement test.

D. Advising and Career Guidance

Each full-time faculty member is required to schedule three hours weekly to meet with students for both academic and course advising. During these meetings students can receive additional one-on-one instruction outside of the classroom.

Each student in the major is assigned an advisor from the department faculty. Each advisor has both Computer Science and Computer Information Systems majors as advisees. All of the faculty members are familiar with the graduation requirements for both majors. In addition, they use web4 (a campus-wide academic database) for reports of graduation readiness. The department has prepared a 4-year plan of study and a checklist for each major to assist advisors in determining whether students are on track. Additional office hours are set aside for a three-week period before registration begins to assist the advisees with course selection for the coming semester. Students are forced to meet with their advisors during this period by requiring them to obtain their 'registration pin' from the advisor, which is needed for registering on-line. The registration pin is unique to each student, and it changes each semester.

Some career advising happens during these semi-yearly sessions, and some occurs in or after class. We bring alumni to campus to talk to students about the real world from time to time. Often the speaker will leave with several resumes. The university maintains a Career Services Center that presents workshops to help students prepare for employment.

E. Work in Lieu of Courses

Students entering Fitchburg State University with college-level training or experience can be tested on college course material to earn credit toward their degree. The Fitchburg State University examination program is especially valuable for individuals who have had learning experiences outside the college classroom (employment experience, life experience, independent study, etc.) which may come to bear upon their formal academic training. It is possible to gain up to 60 college credits through the program.

On the College-Level Examination Program (CLEP), Fitchburg State University adheres to the standards established by the American Council on Education granting credit for tests on which a score of 50 has been achieved. This credit is awarded only to students enrolled in degree programs at Fitchburg State University.

In addition, the University has a Life Experience Credit Program (LECAP) to earn credit for courses based on their life experience. A student who takes this option is assigned a faculty member to evaluate a portfolio prepared by the student describing the work. The student pays full tuition for the credit-hours, and the faculty member assigns a grade based on the submitted portfolio.

F. Graduation Requirements

Requirements for the major in Computer Science include the following courses:

CSC/ MATH	1900	Discrete Mathematics
CSC	1500	Computer Science I
CSC	1550	Computer Science II
CSC	1600	Introduction to Electronics
CSC	1650	Digital Electronics
CSC	2560	Systems Programming
CSC	2600	Computer Organization
CSC	3100	Operating Systems
CSC	3200	Programming Languages
CSC	3600	Microprocessors
CSC	3700	Algorithms and Data Structures
CSC	4400	Software Engineering
MATH	1300	Pre-Calculus (if needed)
MATH	1800	Business Statistics
MATH	2300	Calculus I
MATH	2400	Calculus II
MATH	2600	Linear Algebra

One additional Math course at or above 3000 level

PHYS	2300	General Physics I
PHYS	2400	General Physics II

Five additional CS electives at or above 3000 level

In addition, there are university-wide Liberal Arts and Science (LA&S) requirements which apply to all majors. Some of those requirements are automatically met by the CS requirements; for example, the computer literacy and mathematics requirements are met and exceeded by the CS degree requirements.

Students and their advisors can use the online Web4 database system to obtain a report that lists which requirements have been met and which remain. The registrar uses the same system to check the transcripts of students who have filed a graduation form indicating their intention to graduate at the end of the semester.

The department provides a Four-year Study Plan (see Appendix E) for the program which ensures a timely completion of degree requirements (120 credits). However, many students graduate with more than 120 credits for a number of reasons such as: additional major/minor, switching majors, personal interest in other areas (including from within our department), and transfer credits that do not fit the overall curriculum.

G. Transcripts of Recent Graduates

Six randomly selected transcripts of recent graduates are provided in Appendix H. Each transcript provides the total credits earned from Fitchburg State University, credits transferred, and GPA based on the credits earned from the university.

CRITERION 2. PROGRAM EDUCATIONAL OBJECTIVES

A. Mission Statement

The Fitchburg State Mission Statement

Fitchburg State University is committed to excellence in teaching and learning and blends liberal arts and sciences and professional programs within a small university environment.

Our comprehensive public university prepares students to lead, serve, and succeed by fostering lifelong learning and civic and global responsibility. A Fitchburg State education extends beyond our classrooms to include residential, professional, and co-curricular opportunities. As a community resource, we provide leadership and support for the economic, environmental, social, and cultural needs of North Central Massachusetts and the Commonwealth.

The Department Mission Statement

The Computer Science department at Fitchburg State College strives to provide quality education, sound academic advising and student support in the areas of Computer Science and Computer Information Systems. Our primary focus is to provide students with a foundation that prepares them to continue learning after graduation. The department is committed to providing high quality professional programs and life-long learning opportunities at the graduate and undergraduate levels and in continuing education. The department seeks to help the local hi-tech/business community by providing professional education and a skilled workforce in Computer Science and Computer Information Systems.

The Program Mission Statement

The Computer Science Program offers the opportunity for students to develop a unique blend of knowledge and skills in the areas of computer software and computer hardware. The program provides sufficient theoretical background for continued learning. It also provides practical skills to prepare the students for professional careers in the rapidly changing field of computer science. Students develop good communication skills and the ability for teamwork and leadership roles in their professional careers.

B. CS Program Educational Objectives (CSPEO)

CSPEO-1 Apply what they have learned about computer hardware, programming languages and the fundamental mathematical and engineering principles to computer system functions and software development.

CSPEO-2 Apply what they have learned about the theoretical foundations of computing machinery to understand and use new technologies as they are

introduced.

CSPEO-3 Apply the skills they have learned to achieve and maintain professional careers in computer science.

CSPEO-4 Apply what they have learned to behave ethically, professionally and provide leadership in their professional careers.

CSPEO-5 Apply communication skills including oral and written presentations.

CSPEO-6 Collaborate in group-projects.

CSPEO-7 Continue learning after graduation.

C. Consistency of the Program Educational Objectives with the Mission of the Institution

	Mission Alignment Strength (0-2)		
Program Educational Objectives	**University**	**Department**	**Program**
CSPEO-1 Apply what they have learned about computer hardware, programming languages and the fundamental mathematical and engineering principles to computer system functions and software development.	1	2	2
CSPEO-2 Apply what they have learned about the theoretical foundations of computing machinery to understand and use new technologies as they are introduced.	1	2	2
CSPEO-3 Apply the skills they have learned to achieve and maintain professional careers in computer science.	1	2	2
CSPEO-4 Apply what they have learned to behave ethically, professionally and provide leadership in their professional careers.	2	1	2
CSPEO-5 Apply communication skills including oral and written presentations.	1	1	2
CSPEO-6 Collaborate in group-projects.	0	1	2
CSPEO-7 Continue learning after graduation.	2	2	2

D. Program Constituencies

Program constituencies include …

1. the students in the program, who hope to be prepared with the knowledge and skills for satisfying careers;
2. the students' potential customers and employers, both here in North-Central Massachusetts, and throughout the world, who expect effective workers and prospective leaders;
3. the students' families, who are often making a big investment;
4. the faculty, whose interactions with the students as they grow and learn can be extremely rewarding;
5. the administration and staff, whose primary goal is to assist the students and faculty in the logistics of running the institution.

E. Process for Review of the Program Educational Objectives

The Program Educational Objectives are reviewed and revised in department faculty meetings. For example, we added a CS life-long learning objective during the October 2 and 9 meetings in Fall 2012. Our discussion was inspired by our Industrial Advisory Board and our University mission statement. ABET student outcomes clearly state the need for CS professionals to continue to stay current as the field evolves.

CRITERION 3. STUDENT OUTCOMES

A. Student Outcomes

The most recent publication of CS Student Outcomes (CSSO) to the rest of the campus occurred in the 2007 CS self-study.

The department documents our assessment intentions in a multi-sheet spreadsheet. We keep current and historical versions in a shared folder on the web, accessible to the entire department faculty, and essentially no one else. From that document, we present the following table:

	PEO Allignment Strength (0-2)						
Program Student Outcomes	**PEO-1**	**PEO-2**	**PEO-3**	**PEO-4**	**PEO-5**	**PEO-6**	**PEO-7**
CSSO-1 - Demonstrate proficiency in relevant aspects of mathematics and concepts from physics and electrical circuits.	2	2	1	0	1	0	1
CSSO-2 - Demonstrate proficiency with logic, discrete mathematics, algorithms and data structures.	2	2	1	0	1	1	1
CSSO-3 - Demonstrate the ability to design and implement digital logic circuits and apply this knowledge to the understanding of a computer's organization and architecture.	2	2	1	0	1	1	1
CSSO-4 - Demonstrate the development of both hardware and software interfaces between computers and digital devices.	2	1	1	0	1	1	1
CSSO-5 - Demonstrate proficiency with computer programming languages and different programming paradigms.	2	1	2	0	1	0	2
CSSO-6 - Demonstrate understanding of the principles underlying the design of operating systems and proficiency using operating systems.	2	2	1	0	0	0	1
CSSO-7 - Demonstrate proficiency in software design and development methods.	2	1	2	2	2	2	2
CSSO-8 - Demonstrate the ability to communicate in both oral and written forms and to work in teams.	1	1	2	2	2	2	1
CSSO-9 - Demonstrate the ability to learn after leaving the university.	1	2	2	1	2	1	2
Student Outcomes:							
These are statements that describe what students are expected to know and/or be able to do by the time of graduation. If students have achieved these outcomes, it is anticipated that they will be able to achieve the educational objectives after graduation.							

B. Relationship of Student Outcomes to Program Educational Objectives

The table above shows how strongly each Student Outcome aligns with each of the Program Educational Objectives. The table uses a scale from 0 to 2 where 0 indicates no association and 2 indicates a very strong alignment. For example, each of the student outcomes contributes to some degree with PEO-7, Life-Long Learning.

C. Process for the Establishment and Revision of the Student Outcomes

The process of setting formal goals began in a task force the department established to work toward accreditation in the late 90's. The process continues in a weekly series of department meetings, which also address some other business (for example, election of the department chair) but are primarily devoted to curriculum and assessment issues. In Fall 2012 we added the Program Objective, Life-Long Learning, which is also part of the university mission statement.

D. Enabled Student Characteristics

The goal of our assessment process is to ensure that our program enables the ABET Student Characteristics (a) through (k). The table below shows the mapping to our CS Student Outcomes. Since the CSSO's are assessed the characteristics are enabled.

	Correlation with CSSO								
General Criteria	1	2	3	4	5	6	7	8	9
(a) An ability to apply knowledge of computing and mathematics appropriate to the discipline	X	X	X	X	X	X	X		X
(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution		X	X	X	X	X	X		X
(c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs		X	X	X	X	X	X	X	X
(d) An ability to function effectively on teams to accomplish a common goal				X			X	X	X
(e) An understanding of professional, ethical, legal, security and social issues and responsibilities		X				X	X		X
(f) An ability to communicate effectively with a range of audiences							X	X	X
(g) An ability to analyze the local and global impact of computing on individuals, organizations, and society							X		X
(h) Recognition of the need for and an ability to engage in continuing professional development		X	X	X	X	X	X		X
(i) An ability to use current techniques, skills, and tools necessary for computing practice.	X	X	X	X	X	X	X		X
Specific Criteria: CS									
(j) 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.	X	X	X		X	X	X		X
(k) An ability to apply design and development principles in the construction of software systems of varying complexity.		X			X	X	X		X

Computer Science Program Learning Objectives

CSSO-1 Demonstrate proficiency in relevant aspects of mathematics and concepts from physics and electrical circuits.

CSSO-2 Demonstrate proficiency with logic, discrete mathematics, algorithms and data structures.

CSSO-3 Demonstrate the ability to design and implement digital logic circuits and apply this knowledge to the understanding of a computer's organization and architecture.

CSSO-4 Demonstrate the development of both hardware and software interfaces

between computers and digital devices.

CSSO-5 Demonstrate proficiency with computer programming languages and different programming paradigms.

CSSO-6 Demonstrate understanding of the principles underlying the design of operating systems and proficiency using operating systems.

CSSO-7 Demonstrate proficiency in software design and development methods.

CSSO-8 Demonstrate the ability to communicate in both oral and written forms and to work in teams.

CSSO-9 Demonstrate the ability to learn after leaving the university.

CRITERION 4. CONTINUOUS IMPROVEMENT

A. Student Outcomes

Eleven key courses were used for assessment purposes. Instructors for the 11 key courses gather assessment data every other year according to the schedule shown below. This schedule provides a complete program assessment every two years. Thus, since fall of 2009 we have completed two assessment cycles. For cycle 1 (Fall 2009 through Spring 2011) assessments were obtained for 9 of 11 key courses. A faculty member responsible for two courses retired and did not leave sufficient student data for assessment purposes (C2600 and C3600) during cycle 1. For cycle 2 (Fall 2011 through Spring 2013), assessment data was gathered for all 11 courses. Assessment tools align with course objectives and the number of objectives varies from 5 to 9 depending on the course. Student performance related to each objective is assessed by various tools embedded within each key course. The tools used to assess student learning of any given course objective may consist of quizzes (Q), exams (E), tests (T), homework (H), assignments (A), final exam questions (F), projects (P), lab exercises (L), final presentations (FP) or a combination of these. Student grades on each tool associated with each objective for each of the key courses are used to compute a score for each objective. A percentile rank of students (generally 70% to 80%) scoring above a particular threshold score (generally 70%) is used to identify areas requiring improvement. It is important to note that our class sizes are often small (24 maximum; many classes have enrollments less than 18). Smaller classes may have difficulty meeting an 80 percentile criteria for every course objective especially in classes below the 3000-level where students may still be unsure about continuing with the computer science major.

Assessments occur over a two year cycle. During this period 11 key courses contribute to the assessment. One of these courses is offered in the Mathematics Department. The key courses used for assessment are:

CSC 1600 Introduction to Electronics
CSC 1650 Digital Electronics
CSC 1900 Discrete Math
CSC 2560 Systems Programming
CSC 2600 Computer Organization
CSC 3100 Operating Systems
CSC 3200 Programming Languages
CSC 3600 Microprocessors
CSC 3700 Algorithms and Data Structures
CSC 4400 Software Engineering
MATH 2600 Linear Algebra

These 11 courses cover all of our student outcomes. Most items are assessed in more than one course as shown in the table below.

CS Student Outcomes	C3100	C3200	C2560	C3700	C4400	C1600	C1650	C2600	C3600	C1900	M2600
1 - Proficiency in relevant math, physics and electronics concepts.						X					X
2 - Proficiency in discrete math, algorithms and data structures.				X						X	
3 - Proficiency in logic circuits and computer architecture.							X	X			
4 - Hardware and software interfaces with digital devices.									X		
5 - Computer programming and programming paradigms.		X	X	X							
6 - Principles underlying the design of operating systems.	X										
7 - Proficiency in software design and develoment methods.					X						
8 - Oral and written communications and team work.		X			X	X	X				
9 - Lifelong learning					X						

The schedule of course assessments is shown in the next table (below).

	2008		2009		2010		2011		2012		2013	
CS Outcomes Assessed	Spring	Fall	Spring	Fall	Spring	Fall	Spring	Fall	Spring	Fall	Spring	Fall
1. Math, physics and electronics.				M2600		C1600		M2600		C1600		
2. DM, algorithms & data structures.					C3700		C1900		C3700		C1900	
3. Logic circuits and computer org.					C1650	C2600			C1650	C2600		
4. Hardware and software interfaces.						C2600	C3600			C2600	C3600	
5. Programming languages & paradigms.				C3200		C2560		C3200		C2560		
6. Operating systems.							C3100				C3100	
7. Software engineering.					C4400				C4400			
8. Communications skills and team work.					C4400				C4400			
9. Lifelong learning.					C4400				C4400			

Assessment Data Tables (Fall, 2009 to Spring, 2013)

Data for all course objectives (assessment cycle 1)

Cycle 1: Fall 2009 - Spring 2011	How Measured			Target %tile scoring better than 70%	Actual %tile	Action taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
Solve systems of linear equations	Fall1	M2600	T1;F	80%	75%	
Finite dimensional vector spaces	Fall1	M2600	T1;F	80%	75%	
Perform linear transformations	Fall1	M2600	T2;	80%	79%	
Bases and linear independence	Fall1	M2600		80%		Was not assessed by the math teacher.
Eigenvalues and eigenvectors	Fall1	M2600	T2;F	80%	79%	
Functional, logical & procedural paradigms	Fall1	C3200	T1	80%	70%	
Programming paradigm strengths and weaknesses	Fall1	C3200	T2	80%	70%	
Implementation structures for the paradigms	Fall1	C3200	A1;T3;F	80%	90%	
Functional programming using Scheme	Fall1	C3200	A2,3;T4;F	80%	80%	
Logic programming using Prolog	Fall1	C3200	A4;F	80%	100%	
Sorting Algorithms	Sp1	C3700	P1;T1	80%	70%	Develop handouts with programming hints
Graph Algorithms	Sp1	C3700	P2;T2	80%	90%	
Cryptographic Algorithms	Sp1	C3700	P3;T3	80%	100%	
Dynamic data structures	Sp1	C3700	P4;T4	80%	70%	Develop practice test
Complexity theory	Sp1	C3700	T5	80%	90%	
Design paradigms	Sp1	C3700	T5;F	80%	100%	
Digital representation of data	Sp1	C1650	F	70%	30%	Many students have mathematical difficulty (PI-1 & 2). Combinational

Cycle 1: Fall 2009 - Spring 2011	How Measured			Target %tile scoring better than 70%	Actual %tile	Action taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
Boolean algebra	Sp1	C1650	F	70%	50%	implementation difficulty on exam is puzzling as it differs from lab and sequential design results. This may be a recency effect. Actions: Change assessment of PI-1,2 and 4 to a different tool. Spend more time on Boolean simplification. (10 students)
Combinational circuit analysis	Sp1	C1650	F	70%	70%	
Combinational circuit implementation	Sp1	C1650	F	70%	40%	
Sequential circuit analysis	Sp1	C1650	F	70%	70%	
Sequential logic design	Sp1	C1650	F	70%	80%	
Program a CPLD	Sp1	C1650	F	70%		
Plan and implement a logical design	Sp1	C1650	L4	70%	90%	
Professional and ethical responsibilities	Sp1	C4400	A1	80%	100%	
Software Engineering processes and CASE tools.	Sp1	C4400	P2	80%	100%	
Requirements Analysis and documentation.	Sp1	C4400	P1;FD	80%	84%	
Architecture design and documentation.	Sp1	C4400	FD	80%	78%	
Database design and documentation.	Sp1	C4400	P3	80%	100%	
User interface design and documentation.	Sp1	C4400	P4	80%	100%	
Object oriented analysis, design and documentation.	Sp1	C4400	A2;FD	80%	73%	
Implementing verification, validation and testing.	Sp1	C4400	FP;FD	80%	78%	

Cycle 1: Fall 2009 - Spring 2011	How Measured			Target %tile scoring better than 70%	Actual %tile	Action taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
Working in groups to complete a software project.	Sp1	C4400	P4;FD;FP	80%	85%	
Voltage, current, power and energy	Fall2	C1600	F1-29	80%	53%	Many students had poor numerical skills and struggled. This slowed progress on more advanced topics. Took actions to emphasize numeracy in subsequent offerings. Developed a lab assignment to cover arithmetic. (15 students)
DC circuit analysis	Fall2	C1600	F35-39,47,48	80%	67%	
Build and test electronic circuits	Fall2	C1600	Lab	80%	73%	
Electrical signals	Fall2	C1600	F30-34	80%	87%	
RC circuit analysis	Fall2	C1600	F40-43	80%	40%	
Operational Amplifier circuits	Fall2	C1600	F49-58	80%	27%	
Discrete semiconductor circuits	Fall2	C1600	F59-62	80%	27%	
Data representations / Digital logic design	Fall2	C2600		80%		Professor retired w/o leaving sufficient student data to establish assessment of each course objective.
Register transfer language (RTL)	Fall2	C2600		80%		
Hardwired controller design and implementation	Fall2	C2600		80%		
Microprogram-med controller design	Fall2	C2600		80%		
Instruction set architecture, processing, assembly and pipelining	Fall2	C2600		80%		
Memory hierarchy, cache techniques and virtual memory	Fall2	C2600		80%		
I/O methods, interrupts, raid techniques, data compression	Fall2	C2600		80%		
RISC versus CISC machines	Fall2	C2600		80%		

Cycle 1: Fall 2009 - Spring 2011	How Measured			Target %tile scoring better than 70%	Actual %tile	Action taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
Structured programming with C	Fall2	C2560	A1;T1	80%	73%	Develop handout
Dynamic arrays and linked lists	Fall2	C2560	A3;T2;F	80%	90%	
Trees and pointer arithmetic	Fall2	C2560	A2;T2;F	80%	73%	Develop practice test
Pass by value versus pass by reference	Fall2	C2560	A2;T3;F	80%	100%	
File manipulation and IO methods	Fall2	C2560	A4	80%	90%	
Problem Analysis and Design	Fall2	C2560	A2-4;F	80%	100%	
UNIX systems and programming	Fall2	C2560	T4	80%	90%	
Boolean expressions and Truth tables	Sp2	C1900	T1;F	80%	87%	
Proof techniques	Sp2	C1900	T2;F	80%	94%	
Boolean techniques in digital electronics	Sp2	C1900	T3;F	80%	87%	
Basic Set theory	Sp2	C1900	T4;F	80%	100%	
Basic Number theory	Sp2	C1900	T5;F	80%	81%	
Basic counting principles	Sp2	C1900	T6;F	80%	100%	
Graphs and trees	Sp2	C1900	T7;F	80%	95%	
Basic Computational theory	Sp2	C1900	T8;F	80%	100%	
Machine Architecture	Sp2	C3600		80%		Professor retired w/o leaving sufficient student data to establish assessment of each course objective.
Assembly Language	Sp2	C3600		80%		
CPU Hardware	Sp2	C3600		80%		
Memory Interfacing	Sp2	C3600		80%		
I/O Interfacing	Sp2	C3600		80%		
Interrupts	Sp2	C3600		80%		
DMA	Sp2	C3600		80%		

Cycle 1: Fall 2009 - Spring 2011	How Measured			Target %tile scoring better than 70%	Actual %tile	Action taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
Principles and components of an Operating System	Sp2	C3100	H0,1;T1	80%	83%	
Processes	Sp2	C3100	H2,3;T2;P1	80%	67%	Use homework material in quizzes
CPU scheduling, deadlock detection and deadlock avoidance.	Sp2	C3100	H2,3;T2;P1	80%	67%	
Memory management	Sp2	C3100	H4;T2	80%	75%	
File systems	Sp2	C3100	H5;F	80%	75%	
Operating system security issues.	Sp2	C3100		80%		

Data for all course objectives (assessment cycle 2)

Cycle 2: Fall 2011 - Spring 2013	How Measured			Target %tile scoring better than 70%	Actual %tile	Action to be taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
Solve systems of linear equations	Fall1	M2600	T1;T2;F	80%	58%	Coordinate with Math department to establish more focused assessments. Eigenvalues topic was not assessed or covered by Math instructor. Instructor was not the same as in previous cycle.
Finite dimensional vector spaces	Fall1	M2600	T2;F	80%	55%	
Perform linear transformations	Fall1	M2600	T2;F	80%	55%	
Bases and linear independence	Fall1	M2600	T2;F	80%	55%	
Eigenvalues and eigenvectors	Fall1	M2600		80%		
Functional, logical & procedural paradigms	Fall1	C3200	H1,3,6;T1	80%	75%	Switch textbook. Spend more time on programming in scheme.
Programming paradigm strengths and weaknesses	Fall1	C3200	H0,1,2,3	80%	50%	
Implementation structures for the paradigms	Fall1	C3200	H0,2,4,5,6, 7,8.9;F	80%	38%	
Functional programming using Scheme	Fall1	C3200	T2,3	80%	19%	
Logic programming using Prolog	Fall1	C3200	H4;T1	80%	69%	
Sorting Algorithms	Sp1	C3700	P1;T1	80%	76%	
Graph Algorithms	Sp1	C3700	P2;T2	80%	92%	
Cryptographic Algorithms	Sp1	C3700	P3;T3	80%	100%	
Dynamic data structures	Sp1	C3700	P4;T4	80%	76%	
Complexity theory	Sp1	C3700	T5	80%	92%	
Design paradigms	Sp1	C3700	T5;F	80%	100%	
Digital representation of data	Sp1	C1650	F	70%	21%	Results improved over last cycle mathematical difficulties are still significant. Difficulties shown with representations and Boolean algebra. Action: use tool (Quiz/Test) from earlier in semester for assessment. Did
Boolean algebra	Sp1	C1650	F	70%	57%	
Combinational circuit analysis	Sp1	C1650	F	70%	93%	
Combinational circuit implementation	Sp1	C1650	F	70%	79%	
Sequential circuit analysis	Sp1	C1650	F	70%	71%	
Sequential logic design	Sp1	C1650	F	70%	86%	
Program a CPLD	Sp1	C1650	F	70%		
Plan and implement a logical design	Sp1	C1650	L4	70%	86%	

Cycle 2: Fall 2011 - Spring 2013	How Measured			Target %tile scoring better than 70%	Actual %tile	Action to be taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
						not provide significant coverage of CPLDs.
Professional and ethical responsibilities	Sp1	C4400	A5	80%	94%	
Software Engineering processes and CASE tools.	Sp1	C4400	A3	80%	100%	
Requirements Analysis and documentation.	Sp1	C4400	A1,4;P1-3	80%	95%	
Architecture design and documentation.	Sp1	C4400	A1;P1-3	80%	97%	
Database design and documentation.	Sp1	C4400	A2	80%	100%	
User interface design and documentation.	Sp1	C4400	A4	80%	94%	
Object oriented analysis, design and documentation.	Sp1	C4400	P1-3	80%	96%	
Implementing verification, validation and testing.	Sp1	C4400	P1-3;FP	80%	97%	
Working in groups to complete a software project.	Sp1	C4400	P1-3;FP	80%	97%	
Voltage, current, power and energy	Fall2	C1600	F1-29	80%	67%	Arithmetic exercises/feedback has helped. RC circuit problems require exponential functions, applying initial conditions and approximating when appropriate. Restructure the RC circuits lab to emphasize calculation instead of measurement. (15 students)
DC circuit analysis	Fall2	C1600	F35-39,47,48	80%	67%	
Build and test electronic circuits	Fall2	C1600	Labs	80%	80%	
Electrical signals	Fall2	C1600	F30-34	80%	93%	
RC circuit analysis	Fall2	C1600	F40-43	80%	27%	
Operational Amplifier circuits	Fall2	C1600	F49-58	80%	67%	
Discrete semiconductor circuits	Fall2	C1600	F59-62	80%	53%	
Data representations / Digital logic design	Fall2	C2600	H1-3	80%	83%	
Register transfer language (RTL)	Fall2	C2600	H4;T1	80%	100%	

Cycle 2: Fall 2011 - Spring 2013	How Measured			Target %tile scoring better than 70%	Actual %tile	Action to be taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
Hardwired controller design and implementation	Fall2	C2600	H5-7	80%	92%	
Microprogrammed controller design	Fall2	C2600	FP;F	80%	100%	
Instruction set architecture, processing, assembly and pipelining	Fall2	C2600	H5,8;FP;F	80%	100%	
Memory hierarchy, cache techniques and virtual memory	Fall2	C2600	FP;F	80%	100%	
I/O methods, interrupts, raid techniques, data compression	Fall2	C2600	T2	80%	25%	Spend more lecture time on these topics
RISC versus CISC machines	Fall2	C2600	H9	80%	83%	
Structured programming with C	Fall2	C2560	P1;T1	80%	83%	
Dynamic arrays and linked lists	Fall2	C2560	P3;T2;F	80%	93%	
Trees and pointer arithmetic	Fall2	C2560	P2;T2;F	80%	83%	
Pass by value versus pass by reference	Fall2	C2560	P2;T3;F	80%	93%	
File manipulation and IO methods	Fall2	C2560	P4	80%	90%	
Problem Analysis and Design	Fall2	C2560	P2-4;F	80%	93%	
UNIX systems and programming	Fall2	C2560	T4	80%	90%	
Boolean expressions and Truth tables	Sp2	C1900	T1;F	80%	90%	
Proof techniques	Sp2	C1900	T2;F	80%	85%	
Boolean techniques in digital electronics	Sp2	C1900	T3;F	80%	87%	
Basic Set theory	Sp2	C1900	T4;F	80%	100%	
Basic Number theory	Sp2	C1900	T5;F	80%	90%	
Basic counting principles	Sp2	C1900	T6;F	80%	100%	
Graphs and trees	Sp2	C1900	T7;F	80%	100%	
Basic Computational theory	Sp2	C1900	T8;F	80%	100%	
Machine Architecture	Sp2	C3600	F;L4	80%	86%	Many snow days and lab issues resulted in
Assembly Language	Sp2	C3600	F;L2	80%	79%	

Cycle 2: Fall 2011 - Spring 2013	How Measured			Target %tile scoring better than 70%	Actual %tile	Action to be taken
Performance Indicators (Course Objectives)	**Term**	**Course**	**Embedded Tool(s)**			
CPU Hardware	Sp2	C3600	F;L1	80%	100%	DMA not being covered. Memory interfacing covered earlier in semester was not retained for final exam. Actions: use an earlier assignment for assessment of memory interfacing. Provide handout comparing I/O and RAM interfaces.
Memory Interfacing	Sp2	C3600	F;L5	80%	36%	
I/O Interfacing	Sp2	C3600	F;L6	80%	86%	
Interrupts	Sp2	C3600	F;L7	80%	79%	
DMA	Sp2	C3600		80%		
Principles and components of an Operating System	Sp2	C3100	H1,2; P1;T1;F	80%	74%	
Processes	Sp2	C3100	H3,4,5, 6 ;P2, F	80%	52%	Use homework material in quizzes
CPU scheduling, deadlock detection and deadlock avoidance.	Sp2	C3100	H5,6;T2, F	80%	70%	
Memory management	Sp2	C3100	H7,8; T 2; F	80%	65%	
File systems	Sp2	C3100	H9, F	80%	61%	
Operating system security issues.	Sp2	C3100		80%		Not covered this term

B. Continuous Improvement

Since our last interim report in 2009, we have modified our assessment process and adapted a program-wide embedded assessment model. The assessment tools and targets are set by the individual instructors based on course objectives, course level, course complexity and previous assessment results.

Each course outline listed in the "Course Syllabi" section (Appendix A) contains a table showing how each course objective aligns with program student outcomes. The student outcomes are mapped to program educational objectives as given in the table under Criterion 3 section A.

For assessing courses in our department we use a percentile above a threshold grade. This measure tells us the proportion of students meeting the threshold criteria and gives an indication of how well the student population performs with respect to each course objective (currently 80% or 70% depending on course).

During the first assessment cycle (Fall 2009 through Spring 2011), 58 course objectives were assessed and 26 (45%) did not meet the target percentile. However, 16 of the 26 were within 10% of meeting criteria. The remaining 10 objectives were associated with 3 courses: C1650 (3/8), C1600 (4/7) and C3100 (3/6). Issues with C1600 and C1650 (Introduction to Electronics and Digital Electronics) were related to math preparedness. Corrective actions focused on encouraging numeracy early in the semester and adjusting assessment tools to compensate for long-term forgetfulness. Actions taken with C3100 (Operating Systems) included quizzing students on material from homework.

During the second assessment cycle (Fall 2011 through Spring 2013), 72 course objectives were assessed and 28 (39%) did not meet the target percentile. However, 10 of the 28 were within 10% of meeting criteria. The remaining 18 were distributed across 7 courses: M2600 (4/5), C3200 (4/5), C1650 (2/8), C1600 (2/7), C2600 (1/8), C3600 (1/7) and C3100 (4/6). Of these, C1600, C1650 and C3100 had all been flagged for action in the previous cycle. Assessment data for C1600 and C1650 (Introduction to Electronics and Digital Electronics) showed improvement when compared with the first assessment cycle, suggesting that actions taken in these courses had a positive impact. During the next cycle, calculation over measurement will be emphasized in C1600 and Boolean simplification will get additional attention in C1650. Assessment of C3100 (Operating Systems) during the second cycle demonstrated that the actions taken as a result of the first cycle were ineffective. The action to be taken for the next cycle is to add in-class exercises in processes, deadlock, memory management and file systems to improve student engagement with these topics. C2600 and C3600 (Computer Organization and Microprocessors) failed to meet assessment criteria for only one course objective each. New instructors are in place for these classes due to faculty retirement. Thus, teaching methods and textbooks will change during the next assessment cycle. The instructor for M2600 (Linear Algebra, taught by mathematics department faculty) during the second cycle differed from the one teaching this course successfully during the first cycle. Increased coordination between mathematics and computer science faculty on M2600 will be emphasized during the next assessment cycle.

Our assessment methods continue to be a work in progress. With each cycle, we learn more about how to fairly assess student learning and make changes to improve our program. The goal of course assessment is to take effective action to improve learning when student performance related to a course objective does not meet the criteria we establish.

The factors associated with student performance measured by any particular tool are numerous and can be viewed from multiple perspectives. In every course offering, faculty measure student performance throughout the semester using a variety of tools to determine the grade assigned to each student. Our embedded assessment methodology uses student performance on some of these tools to also attempt a determination of the percentage of students that demonstrate understanding/competence for each course objective. These tools, however, often serve a tertiary purpose: they provide student feedback throughout the semester to motivate further learning. This means that students may not actually gain understanding of a particular course objective until after they have failed an assignment used as an assessment tool.

We continue to work toward making our assessment tools a genuine reflection of student learning so we may have a fair assessment of our program. This work may include

changing assessment tools, time of execution, weights given to individual tools or assessment criteria (threshold grade, target percentile or both). In addition, we may revise course objectives or the way content is delivered, and we may modify student exercises to emphasize particular course objectives.

There are many other aspects out of our control. In particular, student preparedness in mathematics seems to vary greatly. Although we would like to see every student choosing to major in computer science complete the program, many students move to other majors (often, to our CIS program) during the second year because they do not have the prerequisite mathematical interest/skill necessary for success. These students may be included in the assessment data for some courses.

In conclusion, we believe our assessment procedures and data demonstrate our commitment to maintaining the quality of our program. Data obtained during the first assessment cycle resulted in modifications to our teaching methods. Two of the three courses flagged for action during the first cycle showed improvement during the second cycle. Course assessments for the second cycle flagged four additional courses. Our review of this most recent data has resulted in a number of actions aimed at improving student learning in our program. Despite our best efforts to provide consistency in course delivery, we see variability in our assessment results between cycles. This variability may reflect differences between student cohorts. We will continue to adjust our teaching methods and/or assessment tools working toward consistency in our assessment data and improving student learning.

C. Additional Information

Assessment instrument examples will be available at the time of the site visit.

CRITERION 5. CURRICULUM

A. Program Curriculum

The required Table 5-1 is shown below.

Course	*Required, Elective or a Selected Elective by an R, an E or an SE.*[1]	*Subject Area (Credit Hours)*				*Last Two Offerings*	*Section Enrollment: Student # (sections)*
		Math & Sciences	*Computing Topics **F**undamental **A**dvanced*	*General Education*	*Other*		
Fall Freshman Year (16)							
ENGL 1100 Writing 1	R			3		*S13, F12*	*130(6), 664(30)*
CSC 1500 Computer Science I	R		3F			*S13, S12*	*32(2), 29 (2)*
MATH 1300 Precalculus *(if needed)*	R	3				*S13, F12*	*72(5), 111(5)*
PHYS 2300 General Physics I (SMT)	R	4				*S13, F12*	*37(2), 49(3)*
History LA&S elective (CTW)	SE			3			
Spring Freshman Year (16)							
ENGL 1200 Writing II	R			3		*S13, F12*	*683(31), 138(6)*
CSC 1550 Computer Science II	R		3F			*S13, S12*	*22(2), 27(2)*
CSC 1900 Discrete Math (SMT)	R	3				*S13, S12*	*22, 25*
PHYS 2400 General Physics II (SMT)	R	4				*S13, F12*	*26(2), 40(3)*
EXSS 1000 LA&S elective (SMT)	R	3					

Course	*Required, Elective or a Selected Elective by an R, an E or an SE.*[1]	*Subject Area (Credit Hours)*				*Last Two Offerings*	*Section Enrollment: Student # (sections)*
		Math & Sciences	*Computing Topics **F**undamental **A**dvanced*	*General Education*	*Other*		
Fall Sophomore Year (17)							
CSC 1600 Intro. To Electronics	R	4				*F12, F11*	*15(3), 18(3)*
CSC 2560 Systems Programming	R		3F			*S13, F12*	*15, 21*
MATH 2300 Calculus I	R	4				*S13, F12*	*16, 29(3)*
Literature LA&S elective (ARTS)	SE			3			
Literature LA&S elective (CTW)	SE			3			
Spring Sophomore Year (17)							
CSC 1650 Digital Electronics	R		4F			*S13, S12*	*14(2), 14(3)*
CSC 3700 Algorithms & Data Structure	R		3A			*S13, S12*	*17, 15*
MATH 2400 Calculus II	R	4				*S13, F12*	*23(2), 11*
SPCH 1000 Speech (ARTS)	R			3		*S13, F12*	*91(4), 135(6)*
Behavior LA&S elective (CTW)	SE			3			
Fall Junior Year (16)							
CSC 2600 Computer Organization	R		4F			*F12, F11*	*12, 12*
CSC 3200 Programming Languages	R		3A			*F12, F11*	*25, 16*
MATH 1800 Business Statistics	R	3				*S13, S12*	*57(2), 42(2)*
CSC 3XXX CSC Elective	E		3A				
Free Elective					3		
Spring Junior Year (14)							
CSC 3600 Microprocessors	R		4A			*S13, S12*	*14, 11*
CSC 3100 Operating Systems	R		3A			*S13, S12*	*23, 5*
MATH 2600 Linear Algebra	R	3				*S13, F12*	*23, 9*

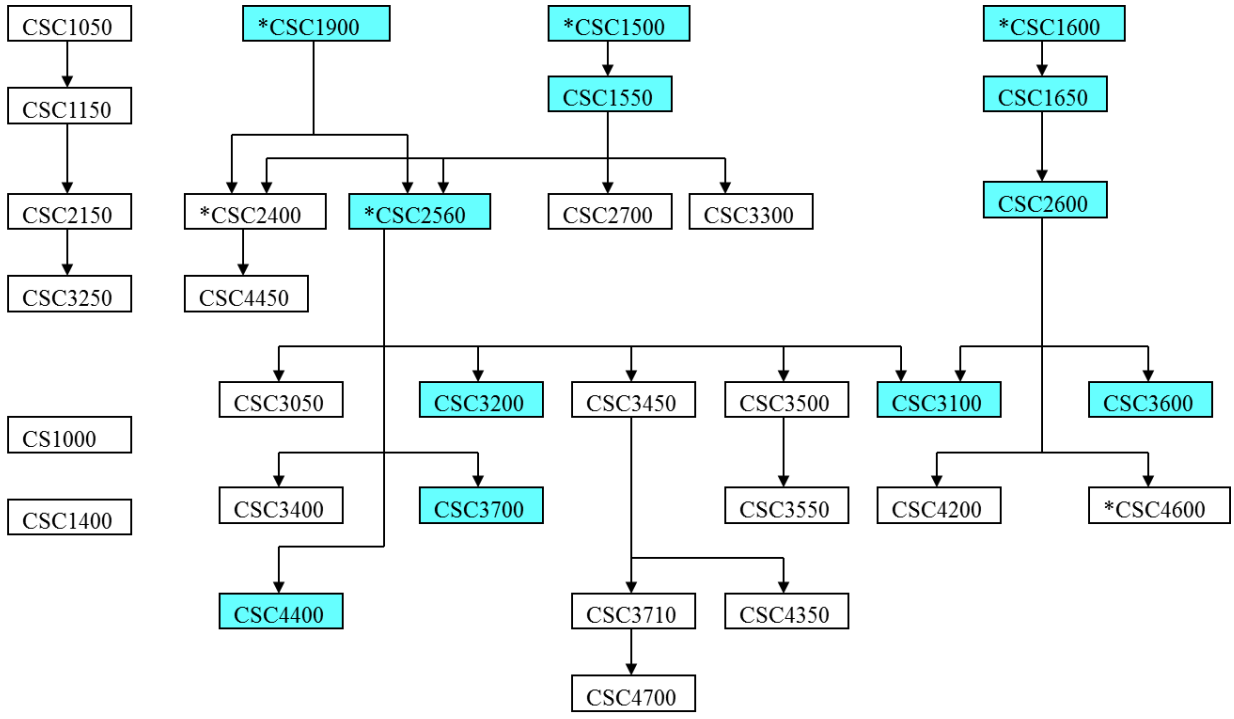
Course	*Required, Elective or a Selected Elective by an R, an E or an SE.*[1]	*Subject Area (Credit Hours)*				*Last Two Offerings*	*Section Enrollment: Student # (sections)*
		Math & Sciences	*Computing Topics **F**undamental **A**dvanced*	*General Education*	*Other*		
CSC 3XXX CSC Elective	E		3A				
Free Elective					1		
Fall Senior Year (12)							
CSC 3XXX CSC Elective	E		3A				
CSC 3XXX CSC Elective	E		3A				
Art/Music LA&S elective	SE			3			
Free Elective	SE				3		
Spring Senior Year (12)							
CSC 4400 Software Engineering	R		3A			*S13, S12*	*16, 17*
MATH 3XXX Math Elective	SE	3					
CSC 3XXX CSC Elective	E		3A				
Free Elective	E				3		
Add rows as needed to show all courses in the curriculum.							
TOTALS-ABET BASIC-LEVEL REQUIREMENTS		38	17F+31A =48	24	10		
OVERALL TOTAL CREDIT HOURS FOR COMPLETION OF PROGRAM	120						

Table 5-1 shows the breakdown of program credit-hours based on the general topic areas of math and sciences (38), computing fundamentals (17), advanced computing (31), general education (24) and others (10).

Each course outline listed in the "Course Syllabi" section (Appendix A) contains a table showing how each course objective aligns with student outcomes. The student outcomes are mapped to program educational objectives as given in the table under Criterion 3 section A.

Cooperative Education, or internship, is not a required part of our program, but it is a highly valued elective. Each student in an internship has an assigned faculty supervisor, who will provide a grade at the end of the term. The student's employer submits a report, and the student prepares a short paper, describing the work performed and how it relates to her academic program. The grade is based upon both, but since in general employer reports are glowing, a good report from the employer carries less weight than a good description by the student.

The flowchart of course prerequisites is shown below.



Other requirements:

Math1300, Math1800, Math2300, Math2400, Math2600, Phys2300, Phys2400

Five electives at or above 3000 level.

*Additional prerequisites:

CSC1500: Passing the math placement exam or Math0200 for CSC1500
CSC1600: Math1300
CSC1900: Math1250 or 1300 or equivalent knowledge
CSC2400 and CSC2560: Could replace the prerequisite CSC1900 with Math1900
CSC4600: Math2600

Course displays will be provided by the department for each of our courses. Course displays consist of:

- a course binder, which contains the syllabus, all course handouts, assessment tools (for key courses) and samples of student work
- textbooks
- an electronic version of the course binder

B. Course Syllabi

Course Syllabi are provided in Appendix A.

CRITERION 6. FACULTY

A. Faculty Qualifications

Table 6-1. Faculty Qualifications
Computer Science Department

Faculty Name	Highest Degree Earned- Field and Year	Rank	Type of Academic Appointment[2] T, TT, NTT	FT or PT[3]	Years of Experience			Level of Activity[4] H, M, or L		
					Govt./Ind. Practice	Teaching	This Institution	Professional Organizations	Professional Development	Consulting/ summer work in industry
Kevin Austin	Ph. D. Biomedical Engineering 1987	P	T	FT	10	15	13	M	H	L
Brady Chen	Ph. D. Applied Mathematics 1995	P	T	FT	6	15	11	M	H	L
Natasha Kurtonina	Ph. D. Computer Science 1995	ASC	T	FT		15	13	M	H	L
Frits Lander	Academy Engineer Mechanical Engineering 1968	ASC	T	FT	14	32	31	L	H	M
Nadimpalli Mahadev	Ph. D. Algorithms 1984	P	T	FT		29	14	L	H	L
Ricky Sethi	Ph.D. Computer Science 2009	AST	TT	FT		4	1	M	H	L
Stephen Taylor	Ph.D. Computer Science 1995	p	T	FT	23	21	11	M	M	L

1. Code: P = Professor ASC = Associate Professor AST = Assistant Professor I = Instructor A = Adjunct O = Other
2. Code: T = Tenured TT = Tenure Track NTT = Non Tenure Track
3. Code: FT = Full-time PT = Part-time Appointment at the institution.
4. The level of activity (high, medium or low) should reflect an average over the year prior to the visit plus the two previous years

B. Faculty Workload

Most faculty members teach in both CS and CIS programs. Thus, the table reflects the proportion of time spent teaching CS versus CIS majors. Some courses are evenly populated with CS and CIS majors while others are predominantly occupied by students of one particular major. In the table below (6-2), the column labeled "percentage of time devoted to the program" considers these commitments as well as graduate teaching, advising, scholarship and administrative responsibilities.

Table 6-2. Faculty Workload Summary

Computer Science Department

Faculty Member (name)	PT or FT[1]	Classes Taught (Course No./Credit Hrs.) Term and Year[2]	Program Activity Distribution[3]			% of Time Devoted to the Program[5]
			Teaching	Research or Scholarship	Other[4]	
Kevin Austin	FT	Sabbatical leave in Spring 2015	50%	25%	25%	80%
		Intro to Electronics Section 1 (CSC 1600/4 credits) Fall 2014				
		Intro to Electronics Section 2 (CSC 1600) Fall 2014				
		Intro to Electronics Section 3 (CSC 1600) Fall 2014				
Brady Chen	FT	Computer Science Basics (CSC 1010/3 credits) Spring 2015	50%	20%	30%	70%
		Topics: GPU Programming (CSC 3004/3 credits) Spring 2015				
		Computer Information Systems (CSC 1400/ 3 credits) Fall 2014				
		Computer Science II Section 1 (CSC 1550/3 credits) Fall 2014				
		Computer Science II Section 2 (CSC 1550) Fall 2014				
Natasha Kurtonina	FT	Discrete Math Section 1 (CSC 1900/3 credits) Spring 2015	70%	25%	5%	50%
		Discrete Math Section 2 (CSC 1900) Spring 2015				
		Systems Programming (CSC 2560/3 credits) Spring 2015				
		Algorithms Data Structure (CSC 3700/3 credits) Spring 2015				
		Ethical Issues in Computer Sci (CSC 4102/1 credits) Spring 2015				
		Adv. Math Computer Scientists (CSC 7013/3 credits) Fall 2014				
		Systems Programming (CSC 2560/3 credits) Fall 2014				
		Game Programming (CSC 3650/3 credits) Fall 2014				

Faculty Member (name)	PT or FT[1]	Classes Taught (Course No./Credit Hrs.) Term and Year[2]	Program Activity Distribution[3]			% of Time Devoted to the Program[5]
			Teaching	Research or Scholarship	Other[4]	
Frits Lander	FT	Intro to Programming (CSC 1000/3 credits) Spring 2015	60%	5%	35%	20%
		Computer Applications (CSC 1100/3 credits) Spring 2015				
		Business Programming (CSC 2700/3 credits) Spring 2015				
		System Design & Implementation (CSC 4700/3 credits) Spring 2015				
		Intro to Programming (CSC 1000/3 credits) Fall 2014				
		Computer Applications Section 1 (CSC 1100/3 credits) Fall 2014				
		Computer Applications Section 2 (CSC 1100) Fall 2014				
		Systems Analysis and Design (CSC 3710/3 credits) Fall 2014				
N. Mahadev	FT	Computer Science I Section 1 (CSC 1500/3 credits) Spring 2015	70%	20%	10%	60%
		Computer Science I Section 2 (CSC 1500) Spring 2015				
		Computer Science I Section 3 (CSC 1500) Spring 2015				
		Software Engineering (CSC 4400/3 credits) Spring 2015				
		Sabbatical leave in Fall 2014				
Ricky Sethi	FT	Computer Science II Section 1 (CSC 1550/3 credits) Spring 2015	60%	25%	15%	60%
		Computer Science II Section 2 (CSC 1550) Spring 2015				
		Computer Science II Section 3 (CSC 1550) Spring 2015				
		Local Area Networks (CSC 3450/3 credits) Spring 2015				
		Computer Science I Section 1 (CSC 1500/3 credits) Fall 2014				
		Computer Science I Section 2 (CSC 1500) Fall 2014				
		Computer Science I Section 3 (CSC 1500) Fall 2014				
		Database Systems (CSC 2400/3 credits) Fall 2014				
Stephen Taylor	FT	Theory of Computation Section 52 (CSC7050/3 credits) Spring 2015	75%	15%	10%	75%
		Digital Electronics Section 1 (CSC 1650/4 credits) Spring 2015				
		Digital Electronics Section 2 (CSC 1650) Spring 2015				
		Mobile Application Development (CSC3560/3 credits) Spring 2015				
		Unpaid leave in Fall 2014				

1. FT = Full Time Faculty or PT = Part Time Faculty, at the institution
2. For the academic year for which the Self-Study is being prepared.
3. Program activity distribution should be in percent of effort in the program and should total 100%.
4. Indicate sabbatical leave, etc., under "Other."
5. Out of the total time employed at the institution.

The faculty contract provides for a teaching load of 12 credit-hours per semester, in addition to student advising and administrative duties. Course releases are available for several specific duties and some ad-hoc ones. For example, the Graduate Program chair gets one course release per year in return for scheduling courses and advising graduate students.

Undergraduate advisees are assigned to faculty by the department chair. Each faculty member has about twenty advisees. The faculty contract specifies that advising effectiveness is a criterion for faculty evaluation for promotion and tenure, along with teaching and research. The academic calendar includes a three-week period in each semester for academic advising during which students are expected to meet with their advisors to plan their schedules for the following semester. During this period all faculty have extended office hours, and post sign-up sheets in the lobby outside the department offices to facilitate visibility and provide convenient student access. To motivate students to seek a meeting, registration passwords for online registration change each semester, and the new passwords are distributed to advisors before the advising period. Although students may register on paper in person or may obtain their password directly from the registrar once registration begins, the system does work.

The department reserves three hours of dedicated time each week to discuss administrative issues, student awards, and the academic programs. A strong recent emphasis has been how to embed assessment of the courses into the traditional assessment of students.

C. Faculty Size

We currently have sufficient faculty to offer courses often enough for students to complete the program in a timely manner. In recent years we have not had a situation when a CS or CIS student's graduation was delayed because of problems in course offerings. Normally we have more than one faculty member prepared to teach a required course. In the past, we have been successful in recruiting new faculty members with terminal degrees.

In June 2012, one faculty member retired. During the 2011-12 AY we searched for a new faculty member with CIS credentials and offers were made to two successful candidates. However, the University could not match the salary expectations of the selected finalists. For the coming academic year, we made a request to resume the search that began in 2011.

Looking forward, five faculty members will be applying for sabbatical over the next 4 years making the faculty replacement urgent.

D. Professional Development

The university policy is to schedule one scholarship day in each week for each faculty member, during which there are no classroom responsibilities.

Every faculty member in the university is entitled to an annual professional development payment, which may go toward tuition, software, hardware, travel, or other professional development expenses. The amount is fixed annually; usually it is between $600 and $800. For 2012-2013 the amount was $732.

There is a university-wide procedure, whereby faculty can request the release of one course or a three-day schedule for research purposes.

The department has a small travel budget. Most faculty members attend at least one professional meeting in the year. Academic Affairs provides additional funds such as special project grants for faculty research and travel.

E. Authority and Responsibility of Faculty

Our department has a system of course "ownership", under which one faculty member is primarily responsible for each course, although others may teach it. The content of the course would not change without the oversight of its "owner". In addition, the department discusses the goals for most of the courses in the program, and from time to time we adjust the goals for individual courses to better meet the program objectives. For example, in the last two years, we have been experimenting with teaching CS I using the Python language instead of or in addition to Java. This is a change in content, but it is not a change in goal or direction for the course. The experiment was begun by the two professors who teach CS I. The rest of the department was interested and informed, but the teachers did not ask for approval.

We endeavor to evaluate many of our courses at least biennially. The responsible faculty member will prepare a report and present it at a department meeting. The faculty member reports on how well the course seems to be meeting its goals; the rest of the department shares responsibility for setting the goals, which include measurable student outcomes. Obviously the input of the person actually teaching the course has the biggest weight in these decisions.

New courses are usually created by individuals; the idea is worked out, presented to the department; often tested with a trial offering. Before a course can be placed in the university catalog, it must go through the (university-wide) faculty governance system. After it is approved by the department, a formal proposal for the new course must be presented to the university curriculum committee for approval, and finally endorsed by the All University Committee.

CRITERION 7. FACILITIES

A. Offices, Classrooms and Laboratories

Since our last evaluation, the third floor of the building has been completely remodeled, and the department and faculty offices have been moved there. The new space is bright and airy, and several faculty members' new offices will be bigger. The department secretary shares airspace with the Math department secretary in an open but windowless office/corridor, which gives both students and faculty easy access to her.

The new area is shiny and welcoming. An area with a few chairs is available for student loitering, which may prove convenient when the faculty arrive in their offices; luckily we also retain a vibrant area on the first floor which almost always holds students waiting for class, eating snacks, or even sometimes reading from the bookshelves and technical magazine rack.

The department uses three computer laboratories as classroom space, and an additional classroom has some computers for use in networking labs. Two additional classroom laboratories are primarily used for computer hardware and electronics laboratories, although some lectures take place in them. Almost all of our classes are held in Edgerly Hall, where the department offices and faculty offices are located.

Our building is the oldest on campus, built as a grade-school, and taken over as a campus school by the normal school that was the beginnings of Fitchburg State. It is not as grand as the college buildings built soon after it, but it has good bones and has been well-maintained. The classrooms are well-sized for the 20-25 student classes we strive to teach, and the acoustics are excellent. All of our classrooms and laboratories now have built-in video projectors. In two of the labs, the student computers are thin clients, with the file storage and some of the computing taking place remotely; in labs where hardware and networking are taught we have more traditional machines.

Classrooms and labs are available outside class hours. Computing facilities are nearly always available. Tools like oscilloscopes are locked in cabinets, for which several faculty members and the department office have keys, so they are usually available only during the day.

B. Computing Resources

Many instructional technology services in individual classrooms have improved over the past few years. All appropriate classrooms are now permanently mediated, containing a common setup consisting of a computer, DVD/VCR, speakers, projector and laptop connection port. There are 22 SMART board equipped classrooms across campus. The wired network infrastructure was completely replaced in 2007 and a new wireless network was installed in 2010.

Starting in 2006, the laptop initiative has extended to all students, who are required to have a laptop which meets specific requirements (http://www.fitchburgstate.edu/offices/technology/stucomp). Students receive free labor and warranty support for their computers and free loaner laptops if systems are not immediately

repaired. In 2009, the IT website was redesigned to improve ease of use and expand self-service capabilities. The IT Department contracted with a new vendor in 2009 to provide basic call center services 24/7/365 and include the ability to remotely control a problematic computer.

The walk-up helpdesk is open 6 days per week for onsite support. All full-time faculty members have University-owned laptops that are refreshed on a three year cycle. Desktop systems, used by some staff and in labs, are replaced on a four-year cycle. There are 25 computer labs on campus that students use for general computing or for specific software requirements. Additionally, since 2008, all entering students are required to complete computer literacy requirements as defined by each major (http://www.fitchburgstate.edu/academics/university-catalog/).

C. Guidance

Primary source of information for students regarding the use of various hardware and software resources is the classroom experience. Instructors provide hands-on training when the resources are introduced to the students for the first time. These include the following:

- Email: The University provides an email account to all the students, faculty and staff, which is the main source of communications for all the university-related work.
- Blackboard: This web tool, which is widely used across many colleges, is introduced to the students in classrooms. The instructors can post syllabus, handouts, assignments, tests, students grades etc. to the blackboard and registered students can access them, submit assignments, view grades, view instructor comments on graded assignments, tests etc.
- Hardware: In hardware courses such as Intro to Electronics, Digital Electronics, Computer Org, Microprocessors, Data Communications and Local Area Networks students are introduced to many hardware tools, components and software tools such as multimeter, oscilloscope, simulation software, etc.
- Software: Various programming languages and the associated IDEs, Databases, web servers, telnet and other software that are used in various courses are introduced in the classroom.
- J Drive: The University provides all the faculty and students with a network drive to store individual data and files. Students can access the J Drive from any of the campus computer labs as well as from dormitories. Faculty can access them online from anywhere.
- I Drive: Some departments may request a shared network drive for the strict use of their own faculty members. The computer science department has one. The faculty can access them online or from campus.
- Web Site: The University provides online account for interested students to host their own individual web pages. Students were introduced to this tool in web development courses.
- Screen capture video: the University provides software for faculty to produce screen-capture video either during a classroom experience or elsewhere (Camtasia Relay) and maintains a web site so students may access the content (lectures.fitchburgstate.edu/relay).

Another source of information for other computer resources is the advising period. Every student is assigned a faculty member from their own major as an advisor and during the advising period, students meet with the advisor regarding the study plan for the next semester and beyond. A major tool introduced at this time is the banner system (also known as Web4). During academic advising, faculty members show their advisees how to complete various registration-related tasks using Web4 such as:

- Viewing the schedule of course offerings for the following semester
- Registering for classes online.
- Obtaining a degree evaluation which shows the courses a student needs to complete the program.
- Viewing an unofficial transcript.
- Checking final grades.
- Viewing class schedule.

Finally, the Information Technology's web site provides a lot of useful information for both the students and faculty members regarding various resources available for their computing needs. These include information on:

- The list of university owned campus-wide site licenses.
- Free downloads of anti-virus software and Microsoft Office and Windows OS.
- Software available for free download specifically to Computer Science faculty and students.
- Helpdesk for maintenance and troubleshooting of computer resources.
- Servicing of faculty/student laptops.
- Printing facilities for the faculty and students.
- Virtual desktop systems (thin clients) available across campus.
- Various computer labs and hours they are open. Some of these are exclusively for the use of Computer Science faculty and students, as required by the license agreements.
- OneCard system for purchase of discounted software etc.
- Computer refresh schedule for the faculty (more on it in the next section).
- Policies and procedures governing the use of college-owned hardware and software.

D. Maintenance and Upgrading of Facilities

Maintenance and upgrading of the technology-related facilities is the responsibility of the Information Technology division. Here are some procedures used in this regard.

- Helpdesk is a call center used by students and faculty alike to report any issues with their accounts, and the electronic equipment provided in the classrooms, labs, library and other locations, and with various web tools, phone systems etc.
- All computer desktops and laptops provided by the university are refreshed periodically to ensure the most efficient and reliable use of these tools. In line with manufacturer's warrantees, Fitchburg State University replaces desktop computers every four years and laptop computers every three years. This refresh schedule is based on the fiscal year and individual systems will be refreshed after lab systems are upgraded.

- Maintain all the upgrades of web tools and classroom software such as the Blackboard, Banner system, software used in various labs etc. These upgrades usually take place between semesters to refresh the systems for the new academic year.
- Data backup on regular basis of all Blackboard accounts, the Banner system, and the faculty/staff email accounts, and the network drives etc. The university maintains all the Blackboard course accounts for up to four years. Some student records are removed when the student graduates.

E. Library Services

This section is divided into two parts: The Library Services and Library Materials.

The Library Services review provides a narrative description of the various services and provides usage statistics. Services include: the library web site, circulation of materials, interlibrary loans/document delivery, reserves, reference, and library instruction.

The Library Materials review provides a summary of the holdings and use of the books/monographs, journals and databases from 2008 to 2012. The books are arranged by call number and the reports include holdings, circulation and acquisitions as well as an assessment control check. The journal's section provides the number of titles provide by the library's database and an assessment control check. The database section lists the core and supplemental Computer Science titles that are provided and Appendix A provides a peer analysis/comparison of databases provided by 10 ABET certified institutional libraries.

Finally, there is a comparison of the previous 2007 ABET library report with the present ABET report.

The complete review is provided in Appendix F.

F. Overall Comments on Facilities

The university upgrades the systems in the open computer laboratories on a four-year schedule. Generally we have the equipment we need to teach our courses. The library is very responsive when we have needs which exceed their current collection.

CRITERION 8. INSTITUTIONAL SUPPORT

A. Leadership

The department chair is in charge of course scheduling, budget requests, and a myriad of administrative detail. Over the past nine years, the department chairperson has been Professor Frits Lander, and beginning in Fall 2013, the chair will be filled by Dr. Brady Chen. The Computer Science program is a central part of our departmental mission.

The primary support points for the CS program are course scheduling and staffing. Secondary needs include recruiting and advising. Professor Frits Lander has been very active in new student forums and new student advising. The university pays the department chair a small summer stipend, and during the summer there are several orientation meetings for new students, at which there is always someone present from our department.

B. Program Budget and Financial Support

The annual budget is submitted each spring. Our department budget includes funds for support of courses, for supplies, etc. It does not include salaries for faculty and staff, which are part of another process.

We do not use graders or teaching assistants in our department. In years past, the department hired lab monitors to oversee our open computer labs, but this budget line has been moved to the Information Technology Office, along with responsibility for maintaining and upgrading the labs themselves.

The technologies we principally rely upon to deliver instruction are 1) digital projectors in the classrooms; 2) computer facilities in the classrooms and laboratories, including specialized computer systems for teaching network administration and microprocessor development; 3) electronic tools and parts. Of these, items (1) and (2) are now supported and maintained by the IT department. This is slightly clumsy for our specialized systems. On the one hand, IT would like to standardize all computers on campus, which would clearly eliminate their support for our special needs. A part-time technical person to attend to our special needs is desirable.

Electronic tools like oscilloscopes and student breadboard systems are still part of the department budget; these are used by courses in the Computer Science program. Maintenance is performed by the faculty using the equipment.

C. Staffing

In 2010 our long-time department secretary Mary Bradford retired, and the office went through a rotation of temporary personnel before hiring our current administrative assistant, Ann Larsen. Ann has proven to be an excellent addition to the department with both a willingness to take on new roles and expand her responsibilities, and with strength in carrying through on her existing duties.

The university has a series of workshops on support software and the academic/budget database systems for administrative personnel. There are good benefits, but salaries seem low.

D. Faculty Hiring and Retention

The recruitment and hiring of new faculty is a shared process which includes faculty members, department chair, and the office of Academic Affairs. The process is defined in the MSCA contract and supported by the Human Resources Hiring Handbook. Human Resources also provides training in the hiring process and ensures that the search has been conducted properly. See (http://www.fitchburgstate.edu/uploads/files/AcademicAffairs/FitchburgStateFacultyHiringPolicySept2010.pdf). The process begins with requests for new positions from academic department chairs to the Vice President for Academic Affairs. The Vice President reviews these requests and makes recommendations to the President. If the position is approved, a search committee of faculty/librarians complete hiring training, review all application materials, contact candidates, schedule interviews, conduct phone/Skype and on-campus interviews, and request and communicate administrative details for the hiring process (parking passes, meal vouchers, and travel expense reimbursement) to the candidate. At the completion of a search process, the Search Committee submits its final three candidates unranked to the department chairperson or library director, who then submits the committee's recommendation and a separate chair or director recommendation to the Vice President for Academic Affairs. The VPAA reviews the recommendations and makes a separate one to the President, who then makes the final decision and sends the letter of appointment.

During the past five years, we have had one retirement. In response to suggestions after the last accreditation visit, we sought a candidate to strengthen our CIS program. We did not find many candidates with CIS degrees, but there were some with extensive information systems experience. Our search committee report led to offers, but the candidates declined the university's offers because of the salary. We expect to search again in 2013-2014.

Of our current six faculty members in the department, the two most recently hired will begin their eleventh year in the fall of 2013, and the other four have longer terms of service, ranging up to Professor Frits Lander's 31 years. Clearly we don't have a big problem with turnover. We are fortunate that our department has been able to work together very well.

E. Support of Faculty Professional Development

Currently, the institution supports faculty scholarship, research, and creative activity in several ways. Most notably, faculty members have a designated weekly research day. Support for professional development is contractually mandated. Tenured and tenure-track faculty and librarians having served six years at the University are eligible for sabbatical leave. Sabbatical leave is governed by Article XV in the current MSCA collective bargaining agreement. Article XIV calls for professional development funding using a formula for computation enumerated in the agreement. In each academic year, funds are disbursed as a one-time Continuing Scholarship stipend, for which faculty and librarians may apply by articulating the manner in which the funds will be spent. These funds are intended to suit the needs of individual faculty members and are not restricted or limited to certain activities. Many faculty members use these funds to attend conferences, for

subscriptions, membership in professional organizations, and to build personal libraries of materials, which thereby enrich their own personal teaching. Funds are available to all faculty and librarians. Approximately 90% of the faculty members apply for these funds. Any leftover monies are then redistributed among the original applicants.

The Vice President for Academic Affairs also provides additional professional development support. Each department receives a budget for Travel Funds, which is distributed according to departmental policies. The Vice President for Academic Affairs also has established special grants to support faculty research/creative activity, assessment projects, academic programming to enhance diverse/global perspective, and faculty travel with priority given to tenure track faculty presenting at peer-reviewed conferences. The participation of the faculty in national and international conferences as presenters is encouraged. For a small state university the involvement of the faculty in such professional activities is impressive and well documented in the Fitchburg State Today and other publications.

Other professional development support has been internally created. Faculty and librarians find professional development opportunities through the University's Center for Teaching & Learning (CTL). The CTL was launched in 2006 with the purpose of supporting faculty development through peer-facilitated workshops, presentations, mentoring, institutional monetary grants, and training. One or two full-time faculty receive workload reductions to serve as director(s) for the CTL, and it maintains an Advisory Board comprised of full-time faculty to aid in the development of programs, review applications for the CTL Innovation Grants, and coordinate a mentoring program for new faculty. Faculty professional development is also provided through workshops presented by the Distance Education Coordinator. Approximately 40 face to face sessions are offered annually with audio and video versions of the presentations available as well. GCE hosts a biennial faculty meeting which includes professional development sessions. The last gathering included such workshop topics such as innovative practices in online learning, developing effective rubrics and the challenges of grading. The Harrod Lecture series provides an additional forum for faculty to present their research to the University community. Faculty may also receive support for grant applications through the University's Grant Center, as well as the Office of Academic Affairs when appropriate. In addition, the Grant Center oversees the Ruth Butler Award, (http://www.fitchburgstate.edu/offices/grants-research/grant-center/ruth-butler-grants/), created to support professional or artistic development, or activities that are appropriate to furthering the goals of Fitchburg State University by full-time members of the faculty, librarians, administrators, and students of Fitchburg State.

Faculty performance is also rewarded through two yearly awards offered by the University. One, the Mara Award, is given to a professor for Excellence in Teaching; the second is the Faculty Award for Research and Scholarship. In addition, in 2010 in conjunction with the second annual graduate commencement ceremonies, an award was established to recognize excellence and contributions for graduate programs. In all cases, funds which support professional development directly benefit teaching, since they provide access to the most current and well-researched course material.

Additional scholarship, research and creative activities are currently supported through several initiatives including the Regional Economic Development Institute (REDi). The REDi's Advisory Board, comprised of North Central Massachusetts leaders from

businesses, municipalities, community organizations, media, and university department chairs, supports REDi research and outreach regionally. The REDi provides a research office and conference room (space) for faculty, with student assistance as interns, to perform on-campus and off-campus research in areas of economic development for North Central Massachusetts. Two distinct studies were completed in the first year of operation and a regional economic development summit co-sponsored by Fitchburg State University with the Massachusetts Lieutenant-Governor and Mount Wachusett Community College. Four more studies are underway, inclusive of one focused on regional broadband advancement and coordinated with the Massachusetts Governor's Office, with numerous public forums held and several studies published during REDi's second year.

PROGRAM CRITERIA

The Computer Science program satisfies all the applicable program criteria as set out in Criteria for Accrediting Computing Programs, 2013 – 2014.

In the area of Student Outcomes, both the general criteria (a) through (i) and the program criteria (j), (k) are enabled as demonstrated in the table under Criterion 3 Section D (Enabled Student Characteristics).

In the area of curriculum, the criteria are satisfied as demonstrated in Table 5-1. Table 5-1 shows the breakdown of program credit-hours based on the general topic areas of math and sciences (38), computing fundamentals (17), advanced computing (31), general education (24) and others (10).

Each course outline listed in the "Course Syllabi" section contains a table showing how each course objective aligns with student outcomes which in turn are mapped to program educational objectives. This mapping is given in a table shown earlier under Criterion 3 section A which is entitled "student outcomes".

As may be seen in Appendix B, 50% of the faculty members have PhD in areas of Computer Science.

APPENDICES

Appendix A – Course Syllabi

For those courses which may be applied toward the Computer Science major we indicate in the list of topics for the course the number of lecture hours devoted to each subtopic in the 2001 Computing Curricula Body of Knowledge list. The topics and core hours for each are only slightly changed in the 2008 Interim Curricular report, but we have not revised the labeling to match the 2008 labels.

The following table summarizes how required courses relate to core areas.

Course - CC 2001Body of Knowledge	CS Major Required Courses											
CC 2001 Body of Knowledge vs. courses supporting the major (core topics and required courses in bold)	CSC 1500 ST	CSC 1550 NM	CSC 1600 KA	CSC 1650 KA	CSC 1900 NK	CSC 2560 NK	CSC 2600 FA	CSC 3100 BC	CSC 3200 ST	CSC 3600 FA	CSC 3700 NK	CSC 4400 NM
HC: Human-Computer Interaction core hours	x	x										x
HC1: foundations of HCI		x										x
IM: Information Management core hours		x										x
IM1: information systems and models												
IM2: database systems												x
IS: Intelligent Systems core hours		x							x			
IS1: fundamental issues in intelligent sys												
NC: Net-Centric Computing core hours		x									x	x
NC2: communications & networking												
NC3: network security											x	
PF2: algorithms & problem solving	x					x					x	
SE: Software Engineering core hours	x					x						x
SE1: software design						x						x
SE3: software tools & environments												x
SP: Social & Professional Issues core hours	x		x	x		x	x					x
SP2: social context of computing	x					x						
SP3: methods & tools of analysis			x									x
SP4: prof. & ethical responsibilities	x											x
SP6: intellectual property	x											x
SP7: privacy & civil liberties												
AR: Architecture & Organization core hours		x	x	x			x		x	x		
AR1: digital logic & digital systems				x			x			x		
AR5: interfacing & communication			x	x			x			x		
DS: Discrete Structures core hours				x	x	x			x		x	
DS2: basic logic				x	x	x						
OS: Operating Systems core hours						x		x				
PF: Programming Fundamentals core hours	x	x				x			x		x	
PF1: fundamental programming constructs	x	x				x						
PL: Programming Languages core hours	x	x				x			x			

CSC 1500	Computer Science 1	3 cr., required	37.5 hours	N. Mahadev

Catalog Description: This course introduces Computer Science by using a high-level programming language. Students will be taught to design programs and implement them using object oriented programming techniques. This course provides a solid background for further studies in Computer Science by preparing students to enroll in the more specialized high-level software courses.

Prerequisite**:** Passing the Math placement exam or Math 0200.

Textbook: Java Programming From Problem Analysis to Program Design, 5th Edition by D.S. Malik, Thomson Course Technology 2012

Supplemental Material:

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: History and evolution of computers and programming languages.
- CO2: The process of designing, coding, compiling, debugging and executing programs.
- CO3: Basic structure, syntax and semantics of a programming language.
- CO4: Basic programming involving data type declarations, arithmetic expressions, methods and input/output.
- CO5: Using control structures such as if-else, switch, while and for loop.
- CO6: Designing and coding graphical user interfaces with event handling.
- CO7: Creating and using overloaded methods and constructors.
- CO8: Object oriented programming concepts such as encapsulation, inheritance and polymorphism.
- CO9: Working with arrays.

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1									
CO2					Y				
CO3					Y				
CO4		Y			Y		Y		
CO5		Y			Y		Y		
CO6		Y			Y		Y		
CO7		Y			Y		Y		
CO8		Y			Y		Y		
CO9		Y			Y		Y		

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO9
CO1									
CO2		Y							
CO3		Y							
CO4		Y							
CO5		Y							
CO6		Y							
CO7		Y							
CO8		Y							
CO9		Y							

Major Topics Covered in the Course**:**

1. Applications of computers and internet (SP2-.5). Security and ethical issues in the use of computers SP5-.5, SP4-.5). Evolution of computers and programming languages (SP1-1). (2.5 hours)
2. The process of designing, coding, compiling, executing and debugging computer programs (PF1-2, PF2-.5). (2.5 hours)
3. Structure, syntax and semantics of a program. Programming style conventions (PF1). (2.5 hours)
4. Data type declarations, arithmetic operations and basic input/output (PF3-2, PF1-3). (5 hours)
5. Control structures (PF1-4, PF2-1). (5 hours)
6. Classes, objects and constructor overloading (PL6-3, PL4-2). (5 hours)
7. Creating and using methods. Scope of variables (PF1-2, SE2-3). (5 hours)
8. Graphical user interface programming and event handling (HC2-1, PF5-2, SE2-2). (5 hours)
9. Introduction to arrays (PF3). (2.5 hours)

Estimate CSAB Category Content:

Category	**Core**	**Adv.**
Algorithms	0.5	
Data Structures	0.5	
Software Design	0.5	
Programming Language Concepts	1.5	
Computer Organization & Architecture		

CSC 1550	Computer Science II	3 cr., required	37.5 hours	S. Taylor

Current Catalog description: This course builds on the concepts covered in Computer Science I. Topics covered include inheritance, polymorphism, recursion, advanced GUI programming, exception handling, and input/output handling. Students use an integrated development environment to create, compile, run and debug programs.
Prerequisite**:** CSC 1500.

Textbook: D.S. Malik and Robert P. Burton, Java Programming: Guided Learning with Early Objects, Course Technology 2009

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

1. Data Structures in Java: Arrays and Strings. Simple Sorting and Searching Algorithms. Multidimensional Arrays.
2. HTML and Applet Basics. Event Driven Programming. The Applet Life Cycle. Interactive Applets.
3. How to use int(), start(), stop(), destroy(), paint(), repaint() methods within applets.
4. Graphics Basics. How to use drawString() method. How to use setFont() and setColor() methods. How to set an applet's background color. How to create graphic objects.
5. The concept of Inheritance. How to extend classes. What is a derived class.
6. Public and Private variables and methods. Constructors with and without arguments. How to access Superclass methods that have Constructors.
7. Advanced Inheritance concepts. Abstract Classes and Dynamic Method Binding.
8. Abstract Windows Toolkit.
9. Using Layout Managers and the Event Model.
10. Exception Handling in Java. The concept of Exception. How to throw and catch and exception?
11. Input/Output and File Techniques. File organization and streams. How to use streams. How to write a file and to read data from a file. Writing and reading formatted file data.

Mapping of course objectives to CS student outcomes:

	CSSO1	CSSO2	CSSO3	CSSO4	CSSO5	CSSO6	CSSO7	CSSO8	CSSO9	CSSO10	CSSO11
CO1		Y			Y			Y			
CO2		Y			Y						
CO3		Y			Y			Y			
CO4		Y			Y			Y			
CO5	Y	Y			Y			Y			
CO6	Y	Y			Y			Y			

CO7		Y			Y			Y			
CO8		Y			Y			Y			
CO9		Y			Y			Y			
CO10		Y			Y			Y			
CO11		Y			Y			Y			

Mapping of course objectives to CIS student outcomes:

	CISSO 1	CISSO 2	CISS O3	CISSO 4	CISSO 5	CISSO 6	CISSO 7	CISS O8	CISS O9
CO1		Y				Y			
CO2		Y				Y			
CO3		Y				Y			
CO4		Y				Y			
CO5		Y				Y			
CO6		Y				Y			
CO7		Y				Y			
CO8		Y				Y			
CO9		Y				Y			
CO10		Y				Y			
CO11		Y				Y			

Major Topics Covered in the Course:

1) Representing Advanced Data Structures in Java. ***CS:*** **PF3(3.0) IS2(1.0) IS2(3.0)**
CIS: **LU54(4) LU56(3)**

- Arrays and Introduction to Sorting and Searching Algorithms
- Linked Lists Representations in Java

2) Applets ***CS:*** **PF5(1.0) GV1(1.0) NC4(3.0) NC1(1.0)**

3) Graphics ***CS:*** **GV2(3.0) GV1(1.0) HC1(1.0)**

4) Inheritance ***CS:*** **IM3(2.0) PL6(3.0)**
CIS: **LU19(3) LU44(4) LU49(3)**

A. Introduction to Inheritance. Superclasses and Subclasses
B. Advanced Inheritance concepts. Abstract Classes and Dynamic Binding

5) Understanding the Abstract Windows Toolkit ***CS:*** **GV1(1.0) PF5(3.0)**
CIS: **LU69(3)**

6) Exception Handling in Java ***CS:*** **PF1(3.0)**
CIS: **LU31(1)**

7) Input/Output and File Techniques ***CS:*** **PF1(1.0) NC1(1.0)**
CIS: **LU63(1) LU43(3) LU53(3) LU57(3)LU58(2)**

8) Multithreading and Animation ***CS:*** **AR7(1.0)**

CSC 1600	Introduction to Electronics	4 cr., required	67.5 hours	K. Austin

Catalog description: This course provides an introduction to the electrical signals, circuits and solid-state devices fundamental to understanding the operational characteristics of digital computers. Methods for the theoretical analysis of electrical circuits containing resistive and capacitive elements are covered extensively. Electrical circuit analysis provides a firm scientific foundation for discussion of practical circuits relevant to computer science. Topics covered in the class include: electronic properties of materials, electrical signals, DC circuit analysis, transient circuit analysis, diodes, transistor switching circuits, integrated circuits, operational amplifier applications, digital-to-analog conversion and analog-to-digital conversion. In the laboratory, students build circuits and make electrical measurements using modern laboratory instruments to validate and reinforce the theory presented in class.

Prerequisite: MATH1300

Textbook: Floyd, Thomas L. Electronics fundamentals: circuits, devices and applications. Pearson Prentice Hall, 2010

Supplemental Materials: Electronic prototyping, test and measurement equipment, lab handouts, problem-solving handouts, notes and lectures are posted on Blackboard for student review.

Goals: The purpose of this course is to develop students' understanding of the electrical principles that underlie computing machinery. Upon completion of the course, a student should be able to do the following:

- CO1: demonstrate knowledge of fundamental electrical quantities;
- CO2: analyze DC circuits containing multiple resistors and voltage sources to solve for all branch voltages and currents;
- CO3: construct and troubleshoot simple electronic circuits containing resistors, capacitors, transistors and operational amplifiers using a solderless breadboard, digital multimeter and oscilloscope;
- CO4: compute standard metrics for various waveforms;
- CO5: perform transient and steady state analysis of circuits containing resistors and capacitors;
- CO6: use operational amplifiers as comparators and amplifiers;
- CO7: analyze circuits containing transistors and diodes;

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1	Y								
CO2	Y								
CO3	Y							Y	Y
CO4	Y								
CO5	Y								
CO6	Y								
CO7	Y								

Topic Outline:	**Lecture hours**	**BoK hours**
Why study electronics? Understanding electronic principles is essential as computers become smaller, faster and battery-powered Knowledge of electronics is highly valued in any environment where software and hardware are being co-developed The approach to problem-solving using deconstruction, analysis and iterative reconstruction used in this course is applicable to software design **(SP3)**	1	SP1: 0.1 SP3: 0.1
Electrical properties of materials: electrons, conductors, insulators, semiconductors	1.5	
Electrical concepts: SI units, voltage, current, resistance, conductance, energy, power Power sources, batteries, switches, pull-up, pull-down circuits	2	
Electrical signals and measurements DC electrical measurement, sine waves: period, frequency, amplitude, spectra Pulse trains: frequency, amplitude, rise time, fall time, pulse width, duty cycle, modulation Using the oscilloscope	1.5	
DC circuits Ohm's law, Kirchoff's laws, deconstruction, simplification, iterative reconstruction **(SP3)**, circuit simulation digital-to-analog conversion **(AR5, SP3)**, Thevenin's theorem, superposition theorem	8	AR5: 0.5 SP3: 1
Operational amplifiers Linear: inverting and non-inverting amplifiers, adders **(SP1)** Non-linear: comparators, analog-to-digital conversion **(AR5)**	4	AR5: 0.5 SP1: 0.1
RC circuits Capacitance, storage: time constant, charge and discharge **(SP1)** Transient analysis of integrators and differentiators, steady-state analysis, pulse-width modulation (PWM) **(AR5)**, digital-to-analog conversion revisited	4	AR5: 0.5 SP1: 0.1 SP3: 0.3
Semiconductor diodes P-N junction, diode V-I curve, light-emitting diodes, circuits containing diodes	3	
Transistors Bipolar transistors: characteristic curves, current gain, load lines, saturation, cutoff, switching circuits, controlling LEDs and motors through PWM	6	
TOTALS*	31	AR5: 1.5 SP1: 0.3 SP3: 0.5

A total of approximately 7 hours of classroom time is dedicated to exams and quizzes

CSC 1650	Digital Electronics	4 cr.	82.5 hours	S. Taylor

Catalog description**:** This course provides a study of the fundamental circuit building blocks that are used in the development of digital computers. The theory and practical application of both asynchronous and synchronous electronic logic circuits are covered. Topics included are: binary representations, data transfer methods, error detection and correction, logic gates, logic families, programmable logic devices, Boolean algebraic simplification, Karnaugh maps, combinational logic circuits, adders, comparators, encoders, decoders, multiplexers, demultiplexers, sequential logic circuits, latches, flip-flops, counters, shift registers and memory. Extensive laboratory work supplements the topics studied.

Prerequisites**:** CSC1600;

Textbook**:** Digital Fundamentals, Tenth Edition by Thomas L. Floyd. Prentice Hall 2008. ISBN-13: 978-0-13-235923-8

Course Objectives**:**
Upon successful completion of the course, a student will have:

- CO1: demonstrated how information is represented in a digital computer;
- CO2: demonstrated how to analyze and simplify Boolean expressions;
- CO3: analyzed digital logic circuits using truth tables, timing diagrams and state diagrams.
- CO3: developed combinational logic implementations using truth tables and K-maps;
- CO4: developed sequential logic implementations using transition tables, K-maps and state diagrams;
- CO5: demonstrated how to program a CPLD;
- CO6: planned a digital logic design and selected appropriate devices.

Mapping of course objectives to CS student outcomes:

	CSSO1	**CSSO2**	**CSSO3**	**CSSO4**	**CSSO5**	**CSSO6**	**CSSO7**	**CSSO8**
CO1	Y							
CO2		Y	Y					
CO3		Y	Y	Y				
CO4		Y	Y	Y				
CO5		Y	Y	Y	Y			
CO6		Y	Y	Y				

Topic Outline: **(**Lecture hours) BoK hours

1. Overview of digital device electronics
    a. Examples of digital systems using gates, adders, comparators, decoders, encoders, multiplexers, demultiplexers, counters, shift registers (1) AR1: 1
2. Digital representations
    a. Voltages represent logic states: high and low
    b. Input switches: pull-up, pull-down, DPST

c. Outputs: high, low, open-collector, high impedance
d. Number systems: decimal, binary, hexadecimal
e. ASCII code, Gray code, error detection and correction codes
f. Serial and parallel data transfer (2) AR2: 1.5 AR5: 0.5

3. Fundamental gates, truth tables and timing diagrams (2) DS2: 2
4. Electrical characteristics of digital logic devices
   a. Guaranteed logic levels and logic device compatibility
   b. Logic families: power, voltage, fan-out
   c. Propagation delay (1.5) AR1: 1.3 SP1: 0.1SP5: 0.1
5. Rules and laws of Boolean algebra
   a. Fundamental rules
   b. Distributive, Commutative and Associative laws
   c. DeMorgan's theorem
   d. Logic simplification
   e. Standard forms of expressions(3) DS2: 2.8 SP1: 0.2
6. Combinational logic circuits
   a. Analysis
   b. Implementation (3) DS2: 3
7. Logic simplification using K-maps
   a. Truth tables → K-maps and vice-versa
   b. Valid and invalid groupings (3) DS2: 2.8 SP9: 0.2
8. Functions of combinational logic
   a. Adders and comparators
   b. Decoders and encoders
   c. Multiplexers and demultiplexers (3) AR1: 3
9. Programmable logic and VHDL (0.5) SP1: 0.1
10. Latches
    a. Gate implementations of S-R and D latches
    b. Gating, edge-triggering
    c. Timing diagrams
    d. Memory devices (3) AR1: 3
11. J-K Flip-flops
    a. Clocking
    b. Synchronous and asynchronous inputs
    c. Timing diagrams
    d. Transition tables (3) AR1: 3
12. Sequential circuit analysis and design
    a. Standard counters
    b. State diagrams
    c. FSM design methods (3) AR1: 3
13. Shift registers
    a. Parallel in/out
    b. Serial in/out
    c. SPI interfaces (3)AR1: 3

CSC 1900	Discrete Mathematics	3 cr., required	37.5 hours	N. Kurtonina

Catalog Description: This course introduces discrete mathematics as applied to computer science. Topics covered include Boolean logic, elementary set theory, functions, relations, enumeration, proof techniques, number systems and trees.

Prerequisite**:**: Math1250 or Math 1300 or equivalent knowledge.

Textbook: Discrete Mathematics and Its Applications, 4th Edition by Kenneth H. Rosen, CB/McGraw-Hill 1998

Supplemental Material: Handouts and lectures notes are posted on Blackboard for student review.

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Fundamentals of propositional logic and Boolean expressions analysis
- CO2: Proof Techniques.
- CO3: Boolean Logic Techniques in Digital Electronics
- CO4: Fundamentals of Set Theory
- CO5: Basic Number Theory
- CO6: Basic Counting Principles
- CO7: Basic Graph Theory
- CO8: Elementary Finite Automata Theory

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1	Y	Y	Y						
CO2	Y	Y	Y						
CO3	Y	Y	Y						
CO4	Y	Y							
CO5	Y	Y							
CO6	Y	Y							
CO7	Y	Y							
CO8	Y	Y							

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO9
CO1	Y								
CO2	Y								
CO3	Y								
CO4	Y								
CO5	Y								
CO6	Y								
CO7	Y								

Major Topics Covered in the Course**:**

1. Fundamentals of Propositional Logic
    - Truth Tables
    - Equivalent Boolean Expressions
    - Justification of reasoning patterns
2. Proof Techniques
    - Proof by Contradiction
    - Mathematical Induction
    - Predicate Logic and Syllogisms
3. Propositional Logic and Boolean Algebra. Circuits and Gates.
4. Basics of Set Theory.
    - Set-theoretic Relations and Set-theoretic operations
    - Equivalence of Set-Theoretic Expressions
    - Logic and Set Theory
    - Property of Binary Relations
    - Relations and Functions
5. Basics of Number Theory
    - Euclid Algorithms
    - Prime numbers and Factorization
    - Modular Arithmetic and Cryptography
    - Summation Principles and Mathematical Induction
6. Counting Principles , Permutations, Elements of Probability Theory.
7. Graphs and Trees and Their Representations
8. Introductions to Models of Computation and Complexity Analysis

CSC 2560	Systems Programming	3 cr., required	37.5 hours	N. Kurtonina

Catalog Description: This course provides the student with the fundamentals of structured programming at the systems level. Students obtain a thorough knowledge of the C language, pointers, linked lists, trees and comfort in using the UNIX operating system.

Prerequisite**:**: CSC 1550 and CSC 1900 or MATH 1900.

Textbook: C Primer Plus. Stephen Prata. Pearson Technology Group 2005. ISBN: 0672326965

Supplemental Material: Handouts and lectures notes are posted on Blackboard for student review.

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Fundamentals of Structured Programming with C
- CO2: Dynamic Arrays, Linked Lists, Trees
- CO3: Dynamic memory management and Pointers Arithmetic
- CO4: Structures and Unions
- CO5: File Manipulations and IO Methods
- CO6: Problem Analysis and Design in Systems Programming
- CO7: Fundamentals of UNIX programming

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1		Y			Y				
CO2		Y			Y				
CO3		Y			Y				
CO4		Y			Y				
CO5					Y				
CO6					Y		Y		Y
CO7					Y				

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO9
CO1		Y							
CO2		Y							
CO3		Y							
CO4		Y							
CO5		Y							
CO6		Y				Y			
CO7		Y							

Major Topics Covered in the Course**:**

1. Brief history of C and of Structured Programming Paradigm.
2. Fundamentals of modular software design.
3. Anatomy of a C Program. Function prototypes and function definitions. Structures and Unions.
4. Introducing pointers. Dynamic variables and memory management. Using malloc() and calloc() functions.
5. Passing parameters by value and by reference.
6. Dynamic arrays and pointer arithmetic.
7. Representing Advanced Dynamic Data Structures in C
    - Linked Lists
    - Stacks and Queues
    - Trees

8. How to code functions for searching, sorting linked list, deleting and inserting nodes.
9. Programming with input and output files in C.
10. UNIX Shell Programming
11. Process generations in UNIX, managing inter-process Communications
12. Essential principles of Systems Programming.

Estimate CSAB Category Content:

Category	**Core**	**Adv.**
Algorithms	0.5	
Data Structures	0.5	
Software Design	0.5	
Programming Language Concepts	1.5	
Computer Organization & Architecture		

CSC 2600	Computer Organization	4 cr., required	82.5 hours	S. Taylor

Catalog Description: The purpose of this course is to provide a thorough discussion of the fundamentals of computer organization and architecture, and to relate these to contemporary computer design issues. After a brief review of the basic digital components used, the steps that a designer would go through in the design of an elementary system are covered followed by a discussion of the organization and design of the central processing unit (CPU) and various control system implementations. The input/output and memory subsystems are included as is a brief discussion of multiprocessing systems, pipelining, and virtual memory. Students registering for this course must also register for the accompanying Lab course.

Prerequisite**:** CSC 1650 Digital Electronics.

Textbook: Null and Lobur, Computer Organization and Architecture, Jones and Bartlett Computer Science 2010

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

CO1: Data representations / Digital logic design;
CO2: Register transfer language (RTL);
CO3: Hardwired controller design and implementation;
CO4: Microprogrammed controller design;
CO5: Instruction set architecture, processing, assembly and pipelining;
CO6: Memory hierarchy, cache techniques and virtual memory;
CO7: I/O methods, interrupts, raid techniques, data compression;
CO8: RISC versus CISC machines.

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSS09	CSS010	CSS011
CO1	Y	Y	Y						Y		
CO2		Y			Y						
CO3	Y	Y	Y								
CO4		Y	Y						Y		
CO5				Y	Y				Y		
CO6				Y							
CO7	Y	Y		Y							
CO8											

Topics:

1.Historical development SP1(.5)
   a. Four generations
   b. Computer level hierarchy
   c. Von Neumann model
2.Data representation AR1(1.5)

Binary, complement systems, codes

3.Digital logic review AR1(4.5)
   a. Boolean algebra
   b. Gates
   c. Combinational and sequential circuits
   d. Circuit design

4.Computer organization and design AR2(3), AR6(1)
   a. CPU Architecture, registers
   b. Buses, clock
   c. I/O, memory and interrupts

5.Instruction set architecture AR3(3),AR6(1)
   a. Fetch-decode cycle
   b. Formats, addressing modes
   c. Assembly

6.Control unit AR3(6), AR6(5)
   a. Register transfer language
   b. Hardwired design
   c. Microcoded design

7.Memory AR4(6)
   a. RAM, ROM, SRAM, DRAM
   b. Cache
   c. Mapping techniques
   d. Replacement algorithms
   e. Hit ratio
   f. Factors leading to poor performance
   g. Virtual Memory
   h. Paging
   i. Memory management

8.Input/Output AR5(3)
   a. Architectures and control
   b. Physical I/O devices and interfaces
   c. RAID techniques
   d. Compression techniques

9.RISC vs. CISC AR7(3)

CSC 3050	Web Programming with Java	3 cr., elective	37.5 hours	N. Mahadev

Current Catalog Description: There are three aspects to Web Programming: Front-end GUI programming, back-end Web or database server programming and the network programming to connect the two. In this course, students learn to use Java classes for creating applets, for server programming and for interfacing with a database. The course also reviews advanced concepts of object-oriented programming.

Prerequisite: CSC 2560.

Textbook: No textbook.

Supplemental Material: Web Programming Notes

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Types of Web Applications.
- CO2: The role of Hypertext Markup Language and platform independence.
- CO3: Use of Cascading Style Sheets in HTML.
- CO4: JavaScript and form validation.
- CO5: Three-tier architecture and MVC architecture.
- CO6: Coding with Java Server Pages.
- CO7: Database design and the role of JDBC.
- CO8: Creating JavaBeans for business logic.
- CO9: ER diagrams and entity class diagrams

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1									
CO2					Y				Y
CO3					Y				Y
CO4					Y				Y
CO5							Y		Y
CO6					Y				Y
CO7					Y				Y
CO8					Y				Y
CO9							Y		Y

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO9
CO1									
CO2									Y
CO3									Y
CO4									Y
CO5									Y
CO6									Y
CO7				Y					Y
CO8									Y
CO9				Y					Y

Major Topics Covered in the Course:

1. Overview of Java Programming Language.
2. Overview of web programming tools: HTML, JavaScript, Applets, Servlets and Java Server pages.
3. Client-side scripting with JavaScript.
4. Tomcat Web server configuration.
5. Overview of Java Server Pages and form processing with JSP.
6. Scripting with JSP elements.
7. Creating and using Java Beans with JSP.
8. Database programming using SQL, JDBC and JSP.
9. Creating JSP tag libraries.

CSC 3100	Operating Systems	3 cr., required	37.5 hours	Brady Chen

Catalog description**:** This course examines the internal structure and operation of operating systems with an emphasis on their design criteria and approaches. Topics covered include: process management, scheduling, deadlock, memory management, virtual memory, protection and security and distributed systems. A working knowledge of a higher-level system programming language and computer data structures is assumed.

Prerequisites**:** CSC2560 Systems Programming and CSC2600 Computer Organization.

Textbook**:** Modern Operating Systems Concepts Essentials, by Silberschatz, Galvin, Gagne. John Wiley & Sons 2011. ISBN: 978-0-470-88920-6

Course Objectives: Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Learn basic principles and components of an Operating System.
- CO2. Learn and be familiar with the notion of processes.
- CO3. Understand CPU scheduling, deadlock detection and deadlock avoidance.
- CO4. Understand how memory management is handled in an operating system.
- CO5. Understand how file systems are implemented.
- CO6. Learn and be familiar with the operating system security issues.

Mapping of course objectives to CIS student outcomes:

	CSSO1	**CSSO2**	**CSSO3**	**CSSO4**	**CSSO5**	**CSSO6**	**CSSO7**	**CSSO8**	CSSO9
CO1						Y			
CO2						Y			
CO3						Y			
CO4						Y			
CO5						Y			
CO6						Y			

Major Topics Covered in the Course**:**

1. Operating System concepts **OS1(2.5) OS2(2.5)**
   computer hardware review
   operating system concepts and structure
2. Processes and threads **OS3(6) OS4(4)**
   processes and threads
   interprocess communication (IPC) and classic IPC problems
   CPU scheduling
3. deadlocks **OS3(4)**
   introduction to deadlock
   deadlock detection and recovery
   deadlock avoidance

deadlock prevention

4. memory management **OS5(6)**
- basic memory management
- virtual memory
- page replacement algorithm

5. input/output **OS6(4)**
- i/o hardware
- i/o software

6. file systems **OS8(4)**
- files
- directories

7. operating system security **OS7(4)**

CSC 3200	Programming Languages	3 cr.; CS req; CIS elec	37.5 hours	S. Taylor

Catalog Description: This course studies the hierarchy of programming languages starting with Assembly Language. It covers general principles of languages within imperative, object oriented and functional paradigms as well as logic programming. Students have an opportunity to learn the basic concepts and constructs of various programming frameworks and practice software design skills in languages like Ada, Lisp and/or Prolog.

Prerequisite**:** CSC 2560

Textbook: *Concepts of Programming Languages,* Robert Sebesta. Addison Wesley 2009

Course Objectives:
After finishing this course, students will be able to:

C. Recognize different styles and programming techniques promoted by different programming languages and programming paradigms
D. Identify the strengths and weaknesses of each programming paradigm.
E. Describe the implementation structures supporting the various paradigms.
F. Design and code programs in Scheme using the functional programming paradigm.
G. Design and code programs in Prolog using the logic programming paradigm.

Mapping of course objectives to CS student outcomes:

	CSSO1	CSSO2	CSSO3	CSS04	CSS05	CSS06	CSSO7	CSS08	CSS09
CO1	Y				Y		Y		Y
CO2		Y				Y			Y
CO3		Y		Y		Y			
CO4		Y			Y		Y		
CO5		Y			Y		Y		

Mapping of course objectives to CIS student outcomes:

	CISSO1	CISSO2	CISSO3	CISS04	CISS05	CiSS06	CiSSO7	CISS08	CISS09
CO1	Y	Y		Y		Y	Y		Y
CO2		Y				Y			Y
CO3		Y		Y		Y			
CO4		Y							
CO5		Y		Y					

Major Topics Covered in the Course**:**

- Brief history of Programming Languages (2 hours) **PL1(2.0)**
- General Hierarchy of Programming Languages. The concepts of High and Low Level Programming Language

- Fundamental issues in language design (2 hours)
- Design goals
- Data structure models
- Control structure models
- Virtual Machines
- Introduction to language translation (2 hours) **PL3(2.0)**
- Comparison of interpreters and compilers
- Language translation phases (lexical analysis, parsing, code generation, optimization)
- Machine-dependent and machine-independent aspects of translation
- Programming Paradigms
- Programming in Assembly (6 hours) **PL2(1.0) PL5(3.0) AR2(1.0) AR6(1.0)**
- High-Level Programming
- Procedural paradigm (4 hours) **PF3(1.0) PL4(3.0)**
- Object-oriented paradigm (6 hours) **PL6(6.0)**
  Object creation and initialization
  Inheritance and dynamic dispatch
  Comparing different Object Oriented Languages
- Functional paradigm (6 hours) **PF3(1.0) PF4(2.0) PL4(3.0)**
  Overview and motivation
  Recursion over lists, natural numbers, trees, and other recursively-defined data
  Pragmatics (debugging by divide and conquer; persistency of data structures)
- Logic paradigm (8 hours) **DS3(1.0) PF4(1.0) PL5(1.0) AL2(1.0) IS2(2.0) IS3(2.0)**
  Overview and motivation
  Recursion in Goal Oriented Programming
  Logic Programming and Artificial Intelligence

CSC 3300	Assembly Language Programming	3 cr., elective	37.5 hours	K. Austin

Catalog description: Assembly language provides the means for programming a computer at the most basic machine level. In this course, we explore the fundamental operations of a modern computer system using software tools. Topics examined include numerical and character representations, microprocessor register usage, machine instructions, addressing modes, input/output processing, parameter passing, interrupt processing and simple data structure realizations on the Intel 80x86 processor.

Prerequisites: CSC1550

Textbook: Hyde, Randall, The Art of Assembly Language Programming. William Pollock 2010. (PDF made available through Blackboard)

Supplemental materials: Various topic handouts and practice problems, in-class notes and screen-capture video of lectures are posted on Blackboard for student review.

Course objectives: The purpose of this course is to develop students' understanding of digital computer organization through assembly language programming. Having successfully completed this course, a student will have:

- CO1: demonstrated how information is represented and operated upon in a digital computer;
- CO2: demonstrated knowledge of how information is processed at the machine level within a CPU;
- CO3: analyzed assembly language code sequences to perform arithmetic, calculate addresses, determine processor flags, predict conditional branches;
- CO4: written assembly language programs to perform console I/O;
- CO5: written assembly language programs that draw directly to screen memory;
- CO6: written assembly language programs that demonstrate bit masking techniques;
- CO7: written assembly language programs that work with arrays using indexed addressing;

	CSSO 1	**CSSO 2**	**CSSO 3**	**CSSO 4**	**CSSO 5**	**CSSO 6**	**CSSO 7**	**CSSO 8**	**CSSO 9**
CO1	X	X							X
CO2		X							X
CO3		X			X				X
CO4		X		X	X	X	X		
CO5		X		X	X		X		
CO6		X			X		X		
CO7		X			X		X		

Topic Outline:	Lecture hours	BoK hours
1. Digital representations a. Bits, bytes, nibbles, words b. Binary, hexadecimal, BCD, ASCII and two's complement representations	4	AL3: 1 AR2: 2 PF2: 1
2. Architecture a. Digital computer components i. CPU ii. Busses iii. RAM and ROM iv. I/O devices b. CPU i. Registers ii. ALU iii. Addressing iv. Instruction fetch/decode c. DOS memory map	4	AR1: 2 AR2: 1
3. Programmer's Register Model a. General purpose, segment, stack pointer, program counter, index b. Low byte / High byte	3	AR1: 1 AR3: 2
4. Assembly Language Fundamentals a. Program structure, entry and exit b. Variables and constants c. MOV, ADD, SUB, NEG, AND, OR, NOT, XOR d. DOS and BIOS function calls	5	DS2: 1 OS2: 0.5 SE2: 0.5 PF1: 2.0 PF3: 0.5 PL4: 0.5
5. The Stack and the Processor Flags a. Subroutines and software interrupts b. Stack operations / context saving c. Instruction results and processor flags d. Unconditional branching e. Conditional branching	5	AR3: 1 DS2: 1 PF1: 2 PF3: 1
6. Addressing modes and data structures a. Immediate, direct and indirect b. Based and indexed addressing c. Arrays	5	AL3: 1 AR3: 1 PF3: 2
7. I/O devices a. Keyboard b. Screen c. File I/O	5	AL3: 1 GV1: 2
31 hours* - AL3: 3, AR1: 3, AR2: 3, AR3: 4, DS2: 2, GV1: 2, OS2: 0.5, PF1: 4, PF2: 1, PF3: 3.5, PL4: 0.5, SE2: 0.5		

A total of approximately 7 hours of classroom time is dedicated to exams and quizzes

CSC 3350	Small-scale Embedded Systems Development	3 cr., elective	37.5 hours	K. Austin

Catalog description: Small-scale embedded systems are everywhere. Students will learn about how to develop functional devices using microcontrollers and become familiar with microcontroller architecture and the various function-specific hardware modules such as timer/counters, analog-to-digital converters, pulse-width modulators, liquid-crystal displays and keypads. Hands-on projects will require the writing of device drivers and the development of a useful embedded system.

Prerequisites: CSC1650 and CSC2560

Textbook: Hyde, Randall, The Art of Assembly Language Programming. William Pollock 2010. (PDF made available through Blackboard)

Supplemental materials: Various topic handouts and practice problems, in-class notes and screen-capture video of lectures are posted on Blackboard for student review.

Course objectives: The purpose of this course is to develop students' understanding of structured software design for embedded applications. Students work in pairs to develop device drivers in the C programming language for microcontroller-based applications. Having successfully completed this course, a student will have:

- CO1: demonstrated understanding of the top-down problem analysis and structured device driver design process;
- CO2: measured interrupt overhead and demonstrated an understanding of interrupt processing;
- CO3: analyzed and programmed several hardware-based timer configurations;
- CO4: demonstrated an understanding of how to improve code readability using C language structs to create variables that represent bit-fields within register;
- CO5: written a device driver to operate an LCD display;
- CO6: written a device driver to scan a keypad matrix;
- CO7: written a device driver to implement an SPI interface to an EEPROM;

	CSSO 1	**CSSO 2**	**CSSO 3**	**CSSO 4**	**CSSO 5**	**CSSO 6**	**CSSO 7**	**CSSO 8**	**CSSO 9**
CO1	X	X							X
CO2		X							X
CO3		X			X				X
CO4		X		X	X	X	X		
CO5		X		X	X		X		
CO6		X			X		X		
CO7		X			X		X		

Topic Outline:	Lecture hours	BoK hours
1. Embedded Software Basics a. PICC int data types b. Bit-field structs c. Mapping variables to memory d. Transferring code to the CPU e. Embedded app structure	3	PF/FundamentalConstructs (1.5) PF/DataStructures (1) SE/SpecializedSystems (0.3) SE/SoftwareReliability (0.2)
2. Microcontrollers a. Architecture i.CPU, Busses, RAM and ROM ii.Instruction pipeline iii.I/O ports, embedded devices iv.Harvard vs. Princeton v.RISC vs. CISC vi.Clock b. PIC assembly language overview	3	AR/ComputerArchitectureAndOrganization (1) AR/FunctionalOrganization (0.5) AR/MemoryArchitecture (0.5) AR/InterfacingAndI/OStrategies (0.5) PL/BasicLanguageTranslation (0.5)
3. Interrupt processing a. Service routines, overhead b. Device identification	3	AR/InterfacingAndI/OStrategies (3)
4. Hardware timers a. Tone generation, Interrupt pacing b. Switch de-bouncing c. Pulse-width modulation	3	AR/InterfacingAndI/OStrategies (1.5) AR/Devices (1.5)
5. External Device Interfaces a. Parallel b. Asynchronous/Synchronous serial	3	AR/InterfacingAndI/OStrategies (3)
6. Structured Software Design a. Requirements definition b. Top-down design/Bottom-up development c. Testing / debugging methods	3	SE/RequirementsSpecifications (1) SE/SoftwareDesign (2)
7. Device Driver Projects a. Loudspeaker b. LCD display c. Keypad matrix d. Serial EEPROM e. A/D, D/A conversion	15	AR/DigitalLogicandDataRepresentation (1) SE/RequirementsSpecifications (1) SE/SoftwareDesign (1) SE/ToolsAndEnvironments (1) AR/InterfacingAndI/OStrategies (10) AR/Devices (1)
AR/InterfacingAndI/OStrategies (18), AR/Devices (2.5), SE/ToolsAndEnvironments (1), SE/SoftwareDesign (3), SE/RequirementsSpecifications (2), PL/BasicLanguageTranslation (0.5), AR/ComputerArchitectureAndOrganization (1), AR/MemoryArchitecture (0.5), PF/FundamentalConstructs (1.5), PF/DataStructures (1), SE/SpecializedSystems (0.3), SE/SoftwareReliability (0.2), AR/DigitalLogicandDataRepresentation (1), AR/FunctionalOrganization (0.5)		

A total of approximately 4 hours of classroom time is dedicated to exams and quizzes

CSC 3400	Data Communications & Networking	3 cr., elective	37.5 hours	Brady Chen

Catalog Description**:** This course investigates the means by which data is exchanged by two digital devices. Topics include the history of data communications, the public switching network (PSTN), standards bodies (OSI, IEEE, etc), serial synchronous/ asynchronous data flow, channel characteristics (bandwidth, noise, capacity, physical implementations), modulation techniques (modems and standards), circuit and packet switching (Asynchronous Transfer Mode (ATM)), multiplexers, Integrated Service Digital Network (ISDN), Digital Subscriber Lines (DSL), etc. An introduction to Wide Area Networks (WAN) is included.

Prerequisites: CSC 2560 and CSC 2600.

Textbook**:** Data Communications and Computer Networks, A Business User's Approach, 7th Edition by Curt M. White. Course Technology 2012. ISBN 978-1-133-62646-6

Course Objectives**:**

Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Data representation, conversion and signaling techniques;
- CO2: Media types and tradeoffs-;
- CO3: Modems types and operational principals;
- CO4: Asynchronous and synchronous communications;
- CO5: Multiplexing techniques;
- CO6: Error causes, detection and control techniques;
- CO7: Local area networks, access techniques, protocols and components;
- CO8: Wide are networks, routing, and switching types;
- CO9: The Internet and protocols used.

Mapping of course objectives to CIS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSS09
CO1							Y		
CO2							Y		
CO3							Y		
CO4							Y		
CO5							Y		
CO6							Y		
CO7							Y		
CO8							Y		
CO9							Y		

Major Topics Covered in the Course**:**

1. Overview of networks and their protocols

    a. Configurations, LAN and WAN
    b. OSI
    c. TCP/IP
2. Fundamentals of Data and Signals
    a. Analog vs. digital signals
    b. Attenuation and noise
    c. Fourier analysis of waveforms; frequency spectra
    d. Signal transmission techniques
3. Data Codes
    a. EBCDIC, ASCII, Unicode
4. Conducted and Wireless Media
    a. Twisted pair, coaxial, fiber
    b. Cellular, infrared, bluetooth, satellite
    c. Selection criteria
5. Getting Connected
    a. Modems
    b. Cable, ISDN, and DSL
    c. Interface Protocols
    d. Asynchronous vs. synchronous connections
    e. Half-duplex, full-duplex, and simplex
6. Multiplexing
    a. Frequency division
    b. Time division
    c. Wavelength division
    d. Discrete multitone
    e. Code division
7. Error Detection and Control
    a. Noise types --White, Impulse, crosstalk, echo, jitter, delay distortion
    b. Prevention
    c. Parity and CRC
    d. Acknowledgements and correction
8. Local Area Network Basics
    a. Topologies
    b. Medium access control
    c. IEEE standards
9. LAN Internetworking
    a. Hubs, bridges, switches
    b. Routers
    c. Congestion control
10. The Internet
    a. Protocols - IP, TCP, ICMP, UDP, ARP, DHCP, NAT, VPN, Email, FTP, Telnet, VOIP
    b. IPV6

CSC 3450	Local Area Networks	3 cr., elective	37.5 hours	Brady Chen

Catalog Description**:** This course examines local area network (LAN) technology and architecture both through general concepts and practical hands-on experience. All networking fundamentals are presented based on the modular approach of the ISO standards. Topics covered include file servers, configurations and protocols, installation and management of server hardware and software, system monitoring, maintenance and troubleshooting. Due to its importance, the TCP/IP protocol will be stressed.

Prerequisites: CSC 2560

Textbook**:**
Networking Basics, 2nd Edition by Ciampa. Course Technology 2003. ISBN 0-619-05583-9
Hands-On MS Windows Server 2003, by Palmer. Course Technology 2003. ISDN 978-0-619-18608-1

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Networking topologies and access methods;
- CO2: Network architectures and standards;
- CO3: Network protocols, hardware and software;
- CO4: Network planning;
- CO5: Server installation and configuration;
- CO6: Network management, monitoring and troubleshooting.

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISS09
CO1							Y	Y	
CO2							Y	Y	
CO3							Y	Y	
CO4							Y	Y	Y
CO5							Y	Y	Y
CO6							Y	Y	Y

Major Topics Covered in the Course**:**

1. Introduction to Networking LU62(2), LU63(2), LU2(2), LU43(1), Network communications LU32(2), LU37(3), LU38(3), Types of networks, Data transmission and media, Network interface cards
2. Network Design LU62(2), LU63(2), LU32(2), LU37(3) Topologies LU38(3), Channel access methods
3. Network Architectures LU32(2), LU35(2), LU37(3),LU38(3) IEEE standards: Ethernet, token ring, wireless
4. Protocols LU32(2), LU35(2), LU37(3), LU38(3)

OSI reference model, Real world implementations and TCP/IP

5. Window 2003 Server
   Different Editions and features, TCP/IP configuration and address resolution (ARP)
6. Server Installation LU63(2), LU34(3). LU38(3)
   Hardware requirements and compatibility, Disk partitioning and file systems, Licensing modes, Protocol(s) selection, Domains and workgroups, Unattended installations, Automated system recovery and troubleshooting
7. Server Configuration LU34(3), LU38(3)
   Configuration of hardware, OS configuration: performance, environment, startup/recovery, protocols, additional components, Windows registry
8. Active Directory and Account Management LU34(3), LU38(3)
   Schema, global catalogue, namespace, Forests, trees and domains, Organizational units and sites, User account creation, configuration and management, Security group management: local, domain local, global, universal
9. Resource Access LU34(3)
   Objects and object security, Attributes, permissions, auditing, Ownership, Sharing and permissions, Troubleshooting a security conflict, Distributed file system, Standalone and domain based, Disk quotas
10. Network Printing LU63(2), LU34(3), LU38(3)
    Printer and port specifications, Sharing and permissions, Configuration and Management, Troubleshooting
11. Data Storage LU63(2), LU34(3), LU38(3)
    Basic and Dynamic disks, Disk management, Fault tolerance: RAID, backup
12. Services LU34(3), LU38(3)
    Dynamic Host Configuration protocol (DHCP), Domain Naming service (DNS), Internet Information server (IIS), Telnet
13. Remote Access LU62(2), LU32(2), LU33(2), LU35(2)
    Protocols LU37(3), LU38(3), Security, Virtual private network, Terminal services
14. Server Security LU34(3)
    Group policies, Encryption
15. Server and Network Monitoring LU34(3)
    Server services, Task manager, System monitor, Logs and alerts, Network monitor

CSC 3500	Object Oriented Programming	3 cr., elective	37.5 hours	N. Kurtonina

Catalog Description: This course covers the fundamental principles of Object Oriented Paradigm. It studies the concepts of procedural and data abstraction, classes, inheritance, polymorphism, virtual functions, templates, exception handling and file manipulations. Students learn the main tools of a popular language supporting the object-oriented framework and various techniques of object-oriented design..

Prerequisite**:**: CSC 2560

Textbook : Absolute C++ by Walter Savitch, Addison Wesley 2005. ISBN 0-321-33023-4

Supplemental Material: Handouts and lectures notes are posted on Blackboard for student review.

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Main techniques, and paradigms in Object Oriented Programming
- CO2: Classes and objects. Inheritance and Interface.
- CO3: Implementations of dynamic and static objects, copy constructors, virtual destructors, multiple inheritance, virtual inheritance, public, private and protected access control templates
- CO4: Computational Problem Analysis in Object Oriented Paradigm. Problem solutions in various Object Oriented Programming Languages.
- CO5: Principles of Object Oriented Design

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1		Y			Y		Y		
CO2		Y			Y				
CO3					Y				
CO4		Y			Y		Y	Y	
CO5		Y			Y		Y	Y	

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO9
CO1		Y							
CO2		Y							
CO3		Y							
CO4		Y							
CO5		Y							
CO6		Y				Y			
CO7		Y							

Major Topics Covered in the Course**:**

1. An overview of Object Oriented Programming Paradigm..
2. Designing structures and classes in C++. The concept of an object in a programing context. Encapsulation and Information Hiding.
3. Class designs in C++ Java and other programming languages. Dynamic and Static Objects. Inheritance and Interface.
4. C++ Programming Tools. Comparing C++ tp other Object Oriented Languages.
5. Object Oriented context of Arrays, Vectors and Strings in C++
6. Object Oriented Programming and Inheritance
    - Inheritance in C++ and Java
    - Understanding Virtual Functions
    - Public, Private and Protected Inheritance
    - Virtual Inheritance
    - Multiple Inheritance in C++ and its application
7. Polymorphism and Virtual functions
    - Dynamic Binding
    - Abstract classes and virtual functions
    - Virtual Destructors
8. Templates
    - Function Templates
    - Class Templates
    - Templates and Inheritance
9. Namespaces
10. Exception Handling Techniques
11. Patterns and UML

CSC 3600	Microprocessors	4 cr., required	67.5 hours	K. Austin

Current Catalog Description: This course provides a detailed study of the microprocessor and its applications. Emphasis is placed on a current microprocessor, its hardware and software and its associated family of integrated circuits. Students design a microprocessor system, configuring the random access memory, the read-only memory, and peripheral devices using peripheral interface adapters. Students reinforce theory with extensive laboratory work. Students registering for this course must also register for the accompanying lab course.

Prerequisite**:** CSC 2600.

Textbook: The Intel Microprocessors: Architecture, Programming and Interfacing, 7th Edition, Barry B. Brey, Pearson/ Prentice Hall 2002

Supplemental materials: ISA bus interface hardware, lab handouts, problem-solving handouts, in-class notes and screen-capture video of lectures are posted on Blackboard for student review.

Course Goals: The purpose of this course is to develop students' understanding of a microprocessor-based computer system to include both hardware and software in order that they will better understand the tradeoffs involved in programming. Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Machine architecture and real mode operation
- CO2: Assembly language for Intel based computers
- CO3: CPU hardware specifications
- CO4: Interfacing memory
- CO5: Input/ Output interfacing
- CO6: Using the hardware interrupt structure
- CO7: Direct memory access in a PC

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSS09
CO1		Y		Y					Y
CO2		Y		Y	Y	Y	Y	Y	Y
CO3	Y	Y		Y					Y
CO4	Y	Y	Y	Y	Y			Y	Y
CO5	Y	Y	Y	Y	Y		Y	Y	Y
CO6	Y		Y	Y	Y		Y	Y	Y
CO7				Y	Y				Y

Topics**:**

1. History of the Microprocessor SP1(.5)
    a. CPU's, busses, memory and I/O maps, clock speed, caches
    b. Number systems and data formats
2. 80x86 architecture AR3(1)
    a. Registers, real mode addressing
3. Assembly language AR3(7), AR2(3)
    a. Addressing modes
    b. Memory models
    c. Machine language
    d. Assembly language
    e. MASM directives
    f. Stack operations
4. 8088 hardware specifications AR1(1)
    a. Pin out
    b. Buffering
    c. Detailed timing: bus cycle and instruction execution
    d. Wait states
5. Memory interfacing AR4(3)
    a. ISA bus interface
    b. Memory types
    c. SIMMs and DIMMs
    d. Decoding circuits
6. Basic IO port interfacing AR5(3)
    a. Input buffer, output latch
    b. Decoding circuits
7. Basic Interrupt handling AR5(6)
    a. Purpose
    b. Vector table
    c. Vector types
    d. Real mode operation
8. Programmable peripheral interfaces AR5(6)
    a. Parallel I/O: 8255 in 3 modes
    b. Serial I/O: 16550 UART
    c. 8259 Interrupt controller
9. DMA processing AR4(2)
    a. Handout on a disk to memory DMA file load: hardware and software

CSC 3700	Algorithms and Data Structures	3 cr., required	37.5 hours	N. Kurtonina

Catalog Description: This course provides an introduction to modern study of computer algorithms and complex data structures. It explores the logic of algorithmic analysis and design as well as practical utility and implementation techniques. The basic ideas from Complexity Theory will help students understand the concept of efficiency of algorithms and its role in the design of algorithms. The analysis of data structures is focused on the concepts of linked list, queues, stacks and trees. Various programming techniques such as divide and conquer; dynamic programming and backtracking are presented.

Prerequisite**:**: CSC 2560

Textbook: Introduction to Algorithms. T.H.Cormen, C.E.Leiserson R.I.Rivest. MIT Press 2009.

Supplemental Material: Handouts and lectures notes are posted on Blackboard for student review.

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Sorting and Search Algorithms
- CO2: Graph Theory Algorithms
- CO3: Cryptography Algorithms
- CO4: Dynamic Data Structures and related algorithms
- CO5: Fundamentals of Computational Complexity Theory
- CO6: Algorithmic Design Paradigms

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1	Y	Y			Y				Y
CO2	Y	Y			Y				Y
CO3	Y	Y			Y				Y
CO4	Y	Y			Y				Y
CO5	Y	Y			Y				Y
CO6	Y	Y			Y				Y
CO7	Y	Y			Y				Y

Major Topics Covered in the Course**:**

1. Introduction to Complexity Analysis of Algorithms
2. Fundamental Searching and Sorting Algorithms: analysis and principles of implementation. Complexity of searching and sorting algorithms. Implementations in various programming paradigms.
3. Data Structures and Related Algorithms. Linked Lists, Stacks and Queues, Binary Search Trees, Red-Black Trees Spanning Trees, Hash Tables.
4. Advanced Design and Analysis Techniques
    - Dynamic Programming
    - Greedy Algorithms
    - Backtracking Techniques
5. Graph Algorithms
6. Cryptographic Algorithms
7. Computability and NP-Completeness

CSC 3710	Systems Analysis and Designs	3 cr., elective	37.5 hours	Brady Chen

Catalog Description: As an overview of the system development life cycle, this course introduces the student to the fundamental concepts and techniques of systems analysis. Classical and structured methods of systems documentation are explored as well as information gathering and reporting activities.

Prerequisites**:** CSC 3450.

Textbook: Systems Analysis and Design, 5th Edition Alan Dennis, Barbara Haley Wixom, Roberta M. Roth. John Wiley & Sons 2012. ISBN: 978-1-118-05762-9

Course Objectives: Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Planning, analysis, design, and implementation phases of the systems development life cycle and systems development methodologies
- CO2: Project identification, feasibility analysis.
- CO3: Identifying project size, creating and managing the workplan, staffing the project
- CO4: Requirement determination, requirement analysis techniques, requirement-gathering techniques
- CO5: Use case
- CO6: Data flow diagram

Mapping of course objectives to CIS student outcomes:

	CISSO1	CISSO2	CISSO3	CISSO4	CISSO5	CISSO6	CISSO7	CISSO8	CISSO9
CO1			Y		Y	Y		Y	Y
CO2		Y			Y			Y	Y
CO3		Y	Y		Y			Y	Y
CO4					Y	Y		Y	Y
CO5					Y	Y		Y	Y
CO6		Y			Y	Y		Y	Y

Major Topics Covered in the Course**:**

1. The systems development life cycle (LU13.16(2), 9(3), 12(2), 16(2), 17(2), 25(2), 28(3), 29(1), 30(1), 62(1), 56(2), 60(2), 61(1), 72(3), 79(2), 114(2), 115(1), 116(4), 105(3), 106(4), 110(2), 121(1), 122(1))
    a. Planning phase (LU26.3(3))
    b. Analysis phase
    c. Design, and phase (LU7(2), 32(1), 33(1), 34(1))
    d. Implementation phase
2. Systems development methodologies (LU29(1), 30(1), 48(2), 110920)
    a. Structure design
    b. Rapid application development (LU76(2))
    c. Agile development

3. Project identification, feasibility analysis (LU18(2), 29(1), 77(4), 78(3), 80(2))
    a. Economic feasibility
    b. Technical feasibility
    c. Organizational feasibility
4. Project management (LU17(2), 18(20, 73(3), 75(4), 79(2), 80(20, 94(3), 100(2), 104(1), 11291), 107(4), 108(3), 126(3), 99(1))
    a. Identifying project size (LU52(3))
    b. Creating and managing the workplan (LU52(3))
    c. Staffing the project
5. Requirement determination (LU8(20, 18(20, 74(3), 78(3), 79(2), 80(2), 96(3))
    a. Requirement analysis techniques
    b. Requirement-gathering techniques
6. Use case (LU46(2), 79(2), 80(2))
7. Data flow diagram (LU24(2), 46(2), 47(2), 54(2), 79(2), 80(2))

CSC 4200	Compiler Construction	3 cr, elective	37.5 hours	S. Taylor

Catalog description: The course studies the techniques involved in the analysis of source language code and the generation of efficient object code. The focus is more on the front end of the compiler (analyzer) than it is on the back end (generator). By working in small project teams, students modify an existing compiler. Topics covered in the class include language definition; lexical, syntactic, and semantic analysis; and code generation.

Prerequisite: CSC 2600.

Textbook: Kenneth C. Louden, "Compiler Construction Principles and Practice", PWS 1997

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- Regular Expressions.
- Backus-Naur Form.
- Recursive descent parsing.
- Bottom-up LR and LALR parse algorithms.
- flex and yacc programs for describing computer languages.

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1	Y	Y			Y		Y		Y
CO2	Y				Y		Y		Y
CO3		Y			Y		Y		Y
CO4	Y	Y			Y				
CO5		Y			Y		Y		

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO 9
CO1	Y					Y			Y
CO2	Y					Y			Y
CO3						Y			Y
CO4	Y					Y			
CO5						Y			

Course objectives are assessed by:

	Tests/quizzes	Problems sets	Programming projects
CO1	Y	Y	Y
CO2	Y	Y	Y
CO3			Y
CO4	Y	Y	Y
CO5			Y

Major Topics Covered in the Course**:**

1. Bootstrapping and Porting
2. Lexical Scanning; regular expressions and regular languages; the flex language.
3. Context-Free Grammars and parse trees
4. Recursive Descent parsing
5. Bottom-up parsing. The yacc language.
6. Attribute Grammars
7. Implementations of Dynamic memory allocation
8. P-code and Assembly code. Compiling to C.

CSC 4400	Software Engineering	3 cr., required	37.5 hours	N. Mahadev

Catalog Description: This course examines main features of software life cycle. It covers the main issues in design creation, principles of programs verification, system testing and evaluation criteria. This course provides students with an opportunity to obtain practical experience in software design using CASE tools. Students work in project teams and apply principles of software design, verification, testing and coding toward the solutions of assigned problems.

Prerequisite**:** CSC 2560.

Textbook: Object Oriented Software Engineering, 2nd Edition by T.C. Lethbridge and R. Laganiere, McGraw Hill 2005

Supplemental Material:

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Professional and ethical responsibilities of a software engineer.
- CO2: Software Engineering processes and CASE tools.
- CO3: Requirements Analysis and documentation.
- CO4: Architecture design and documentation.
- CO5: Database design and documentation.
- CO6: User interface design and documentation.
- CO7: Object oriented analysis, design and documentation.
- CO8: Planning, documenting and implementing verification, validation and testing.
- CO9: Working in groups to complete a software project.

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1									
CO2							Y		
CO3							Y	Y	
CO4							Y	Y	
CO5							Y	Y	
CO6							Y	Y	
CO7							Y	Y	
CO8							Y	Y	
CO9					Y			Y	Y

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO9
CO1									
CO2									
CO3					Y	Y			
CO4					Y	Y			
CO5				Y	Y	Y			
CO6					Y	Y			
CO7		Y			Y				
CO8									
CO9				Y				Y	Y

Major Topics Covered in the Course**:**

1. Software engineering overview and professional and ethical responsibilities (SP3-.5, SP4-1, SP5-2, SP6-.5). (4 hours)
2. Overview of systems engineering process (.5 hours)
3. Overview of software processes and tools (SE3-1, SE4-2). (3 hours)
4. Project management overview (SE8). (3 hours)
5. Requirements engineering and the role of use cases (SE5). (2 hours)
6. Requirements documentation presentations (SE5). (2 hours)
7. Architecture models (NC4). (1 hours)
8. Data processing models and documentation (IM2-.5, IM3-4.5). (5 hours)
9. User interface design and documentation (HC1-1.5, HC2-2.5). (4 hour)
10. Object oriented design and documentation (SE1). (5 hours)
11. Planning verification, validation and testing (SE6). (3 hours)
12. Software evolution and maintenance processes (SE7). (3 hours)

Estimate CSAB Category Content:

Category	**Core**	**Adv.**
Algorithms		
Data Structures		
Software Design		3
Programming Language Concepts		
Computer Organization & Architecture		

CSC 4550	Database Programming	3 cr., elective	37.5 hours	N. Mahadev

Catalog Description: Many businesses, small and large, utilize user-interfaces to access databases to provide security as well as multiple user access to the databases. This course will introduce the student to software components designed for database programming in a higher-level language. Each student works on a project of applied nature that involves documenting the requirements of a database, creating the database, and designing the user-interface to access the database. Working knowledge of the host language is required.

Prerequisite**:** CSC 2400 and permission from the instructor.

Textbook: No textbook. Only course notes and online material.

Supplemental Material:

Course Objectives**:** Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Relational databases; entities, fields and records; relationships and keys.
- CO2: The process of designing a database.
- CO3: Components of a structured query language.
- CO4: Database security and data control.
- CO5: Three-tier architecture and MVC architecture.
- CO6: Analysis, design and implementation of an application using database programming.

Mapping of course objectives to CS student outcomes:

	CSS01	CSS02	CSS03	CSS04	CSS05	CSS06	CSS07	CSS08	CSSO9
CO1							Y		Y
CO2							Y		Y
CO3									Y
CO4							Y		Y
CO5							Y		Y
CO6					Y		Y		Y

Mapping of course objectives to CIS student outcomes:

	CISS01	CISS02	CISS03	CISS04	CISS05	CISS06	CISS07	CISS08	CISSO9
CO1									
CO2				Y					
CO3									
CO4				Y					
CO5									
CO6		Y		Y					Y

Major Topics Covered in the Course**:**

1. Relational database management systems.
2. High level language (SQL)
3. Transaction management and control commands
4. Data security and data control commands
5. Data entry and retrieve commands
6. Database design.
7. MVC and 3-tier architectures.
8. Using JDBC and ODBC API in database programming.
9. Design, analysis and implementation of applications using database programming.

CSC 4700	Systems Design and Implementation	3 cr., elective	37.5 hours	Brady Chen

Current Catalog Description: As an overview of the system development life cycle, this course introduces the student to the fundamental concepts and techniques of systems analysis. Classical and structured methods of systems documentation are explored as well as information gathering and reporting activities.

Prerequisites**:** CSC 3710

Textbook: Systems Analysis and Design, 5th Edition Alan Dennis, Barbara Haley Wixom, Roberta M. Roth. John Wiley & Sons 2012. ISBN: 978-1-118-05762-9

Course Objectives: Upon successful completion of this course, a student will have demonstrated knowledge of:

- CO1: Data flow diagram
- CO2: The entity relationship diagram
- CO3: System acquisition, architecture design, user interface design
- CO4: Logical and physical process models, designing programs, structure chart, and program specification,
- CO5: Data storage formats, logical and physical data models
- CO6: Managing programming process, testing, developing documentation,
- CO7: Transition to the new system, the migration plan, post-implementation activities

Mapping of course objectives to CIS student outcomes:

	CISSO1	CISSO2	CISSO3	CISSO4	CISSO5	CISSO6	CISSO7	CISSO8	CISSO9
CO1		Y	Y		Y	Y		Y	Y
CO2				Y	Y	Y		Y	Y
CO3					Y	Y	Y	Y	Y
CO4		Y	Y	Y	Y	Y	Y	Y	Y
CO5				Y	Y	Y	Y	Y	Y
CO6		Y			Y	Y	Y	Y	Y
CO7					Y	Y		Y	Y

Major Topics Covered in the Course**:**

(LUs that cover all topics: LU28(2), 75(4), 79(2), 80(2), 116(2), 106(4), 107(4), 110(2))

8. Data flow diagram (LU24(3), 46(3), 72(4))
9. The entity relationship diagram (LU47(3), 88(2))
10. System acquisition (LU29(3), 125(3))
11. Architecture design (LU62(2), 32(2), 34(3))
    a. Architectural components (LU62(2), 32(2), 33(2), 34(3), 36(1), 38(2))
    b. Hardware and software specification (LU7(3), 32(2))
12. User interface design
    a. User interface design process
    b. Navigation design

    c. Input/output design
13. Program design (LU50(2), 117(40, 105(3))
    a. Logical and physical process models (LU72(4))
    b. Designing programs
    c. Structure chart
    d. Program specification
14. Data storage design (LU12(1), 47(3), 53(2), 58(3), 59(1), 81(3), 88(2), 90(2), 91(2), 92(2))
    a. Data storage formats
    b. Logical and physical data models (LU72(4), 73(4), 95(3))
15. Implementation (LU52(30, 60(3), 94(3), 98(3), 117(4), 105(3))
    a. Managing programming process (LU60(3), 61(2))
    b. Testing (LU60(3), 81(3), 117(4), 103(3))
    c. Developing documentation (LU94(3))
16. Transition to the new system
    a. The migration plan (LU97(3))
    b. Post-implementation activities (LU78(2), 127(2), 109(2))

Appendix B – Faculty Vitae

Kevin B. Austin

Education

- Ph.D. Biomedical Engineering, Worcester Polytechnic Institute, Worcester, MA (1987)
- M.S. Engineering Science, Rensselaer Polytechnic Institute, Troy, NY (1985)
- B.S. Electrical Engineering, Polytechnic Institute of New York, Brooklyn, NY (1980)

Academic experience

- Graduate Program Chair, Computer Science, Fitchburg State University, Fitchburg, MA (2012-present, part-time)
- Professor of Computer Science, Fitchburg State University, Fitchburg, MA (2012-present, full-time)
- Associate Professor of Computer Science, Fitchburg State College, Fitchburg, MA (2006-2011, full-time)
- Assistant Professor of Computer Science, Fitchburg State College, Fitchburg, MA (2000-2006, full-time)
- Regular Fellow, Department of Pharmacology, University of Colorado Health Sciences Center, Denver, CO (1993-98, full-time)
- Research Associate, McGill University, Montreal, Quebec, Canada (1987-93, full-time)
- Adjunct Professor of Electrical Engineering, Worcester Polytechnic Institute, Worcester, MA (1987, part-time)
- Instructor, Electrical Engineering, Worcester Polytechnic Institute, Worcester, MA (1985-86, part-time)

Non-academic experience

- Software Engineer, ComputerBoards, Inc. Middleboro, MA (1999-2000, full-time)
- President, Eclectic Engineering Studio, Denver, CO (1998-99, full-time)
- President, Eclectic Engineering Studio, Denver, CO (1997-98, part-time)

Certifications or professional registrations - NONE

Current membership in professional organizations

- Association for Computing Machinery (SIGCSE)
- Institute for Electrical and Electronic Engineers (EMBS, Education & Computer)
- Society for Neuroscience

Honors and awards

- Principal Investigator, "Acquiring a Bat's Perspective on Biosonar Echoes", National Science Foundation, $576,463 (2008-12)
- Co-Principal Investigator, “Exploring the Ecology of Flash Communication in Photinus Fireflies Through Collaborative Undergraduate Research with Computer-Simulated Signaling Behavior”, National Science Foundation, $663,228 (2003-2006)
- Principal Investigator, “Social Impact of Information Technology”, Commonwealth Information Technology Initiative (2001)
- NSERC International Fellowship in Science and Engineering. Department of Psychology, McGill University, Montreal, Quebec, Canada (1991-93)
- NSF-NATO Postdoctoral Fellowship in Science and Engineering. Department of Psychiatry, Douglas Hospital Research Center, McGill University, Montreal, Quebec, Canada (1989-90)
- Research Institute of the Royal Victoria Hospital Postdoctoral Fellow. Department of

Physiology, McGill University, Montreal, Quebec, Canada (1988-89)
- Eta Kappa Nu. International Electrical Engineering honor society. Department of Electrical Engineering, Worcester Polytechnic Institute, Worcester, MA (1985)
- Greater Hartford Clinical Engineering Internship. University of Connecticut Health Sciences Center, Farmington, CT (1980-82)
- U.S. patent number 4250554. "System for Ordered Measurement and Computation" (1981)

Service activities (within and outside of the institution)
- Graduate Program Chair, Computer Science, Fitchburg State University, Fitchburg, MA (2012-present)
- Animal Care and Use Committee, Fitchburg State University, Fitchburg, MA (2011-2012)
- Youth Basketball Coach, Worcester JCC
- Youth Baseball Coach, Jesse Burkett Little League, Worcester, MA
- Beta tester for Rock Prodigy (guitar instruction application for iOS)

Most important publications and presentations from the past five years
- Austin, K.B., P.R. Moosman, Jr, H.H. Thomas, "Eavesdropping on echolocation: Recording the bat's auditory experience." Proceedings of the 33rd Annual Conference of the IEEE / Engineering in Medicine and Biology Society, pp.7682-7686 (2011)
- Moosman, P. R., Jr., K. B. Austin, H. H. Thomas. "Using in situ recordings to study echolocation: a comparison of bat-based versus traditional ground-based devices." Presented at the 40th North American Symposium on Bat Research, Denver, Colorado (October, 2010)
- Moosman, P. R., Jr., K. B. Austin, H. H. Thomas, B. J. Crepeau, R. K. Farnsworth, B. A. Huff, P. J. Lustig, M. F. Tatro, and K. Veasna. "Gaining a bat's perspective on echolocation: recording ultrasound calls and their resulting echoes in situ from flying bats." Presented at the 39th North American Symposium on Bat Research, Portland, Oregon (October, 2009)
- Austin, K.B., P. Lustig, K. Veasna, H. Thomas, B. Crepeau, M. Tatro, P. Moosman, B. Huff, R. Farnsworth. Capturing the Auditory Experience of Behaving Bats: A Preliminary Study. International Conference on Computing in Engineering and Information, pp. 123-6 (April, 2009)

Most recent professional development activities
- Ongoing analysis of bat echolocation signals collected under NSF-funded research program 2009-2011 with Dr. Howard Thomas (Biology Department)
- Attended and presented at 33rd Annual Conference of the IEEE / Engineering in Medicine and Biology Society (August, 2011)
- Traveled to SIGCSE 2010 with four student researchers. Attended workshops on Cloud Computing and Computer Science Outreach activities (February, 2010).
- Development of a minimalist 2-digit binary to hexadecimal display decoder using a single microcontroller and a dual 7-segment display
- Step-by-step assembly language exercises leading to the development of a simple interactive animation (video game)
- Building a simple computer architecture from scratch: simulation using Multisim

1. Xuzhou Chen
2. Education:

- MCS, Computer Science, North Carolina State University, Dec. 1998
- Ph.D., Applied Mathematics, North Carolina State University, Dec. 1995
- M.S., Numerical Analysis, Shanghai Teachers University, July 1987
- B.S., Mathematics, Shanghai Teachers University, July 1984

3. Academic experience

Fitchburg State University	Full Professor (9/2013 –)	Department Chair	2013 –	full time
Fitchburg State University	Associate Professor	Evening Program Chair (2008 – 2010)	2008 – 2013	full time
Fitchburg State University	Assistant Professor	Evening Program Chair (2003 – 2008)	2002 – 2008	full time
East China University of Science & Technology	Assistant Professor		1987 – 1991	full time

4. Non-academic experience

Nortel Networks	Member of Scientific Staff	R & D	1997 – 2002	full time
Fujitsu Network Communications	Software Engineer	R & D	5/1996 – 12/1996	full time

5. Certifications or professional registrations

6. Current membership in professional organizations

ACM, CCSC, ILAS

7. Honors and awards
   - FSU Ruth Butler Grant to attend the 18th International Linear Algebra Society conference in Providence, RI, June 3-7, 2013.
   - FSU Special Research and Travel grants for research in Robotics Programming, 2012.
   - Fitchburg State University Center for Teaching and Learning CTL Innovation Grant: Enhancing Undergraduate Research and Learning of Advanced Algorithms using Robotics Programming, December 2010. (Joint with Dr. Mahadev)

8. Service activities (within and outside of the institution)

9. Briefly list the most important publications and presentations from the past five years – title, co-authors if any, where published and/or presented, date of publication or

presentation

1. Xuzhou Chen, Xinghua Shi, and Yimin Wei. The stationary iterations revisited. Numerical Algebra, Control and Optimization, Vol. 3, No. 2, 2013, pp. 261-270.
2. Yan Luo, Xuzhou Chen, and Jie Wang. A Virtual Network Embedding Algorithm for Providing Stronger Connectivity in the Residual Networks. Accepted for publication in Journal of Networks.
3. Xuzhou Chen and N.V.R. Mahadev. Enhancing the Undergraduate Teaching and Research Using Robotic Programming, Journal of Computing Sciences in Colleges, Vol. 28, No. 2, 2012, pp. 57-64.
4. Xuzhou Chen and N.V.R. Mahadev. A PC Based Robot for Learning Computer Vision and Advanced Programming. ACM SIGCSE, 2012, Raleigh, North Carolina, USA, pp. 664-665.
5. Xuzhou Chen, Yan Luo, and Jie Wang. Virtual Network Embedding with Border Matching. The Proceedings of the Fourth International Conference on Communication Systems and Networks (COMSNETS), Bangalore, India, January 3-7, 2012.
6. Xuzhou Chen and Jun Ji. Computing the Moore-Penrose inverse of a matrix through symmetric rank-one updates. American Journal of Computational Mathematics, Vol. 1, No. 3, 2011, pp. 146-150.
7. Xuzhou Chen and Jun Ji. The minimum-norm least squares solution of a linear system and symmetric rank-one updates. Electron. J. Linear Algebra, 22:480–489, 2011.
8. Xuzhou Chen and Robert E. Hartwig. The Conditions for the Convergence of Power Scaled Matrices and Applications. American Journal of Computational Mathematics, Vol. 1, No. 2, 2011, pp. 63-71.

10. Briefly list the most recent professional development activities

- 18th International Linear Algebra Society conference in Providence, RI, June 3-7, 2013. Topic: The stationary iterations revisited.
- 21st Annual RM CCSC annual conference, October 12-13, 2012. Topic: Enhancing the Undergraduate Teaching and Research Using Robotic Programming.
- Invited talk in the computer science department at UMass Boston, April 1, 2010. Topics: A network embedding algorithm with border approach
- International Workshop of Scientific Computing and its Applications, June 28-30, 2009, Fudan University, Shanghai, China. Invited Talk: Discussion on Dimensionality Reduction and Data Representation.
- International Conference on Applied Analysis and Scientific Computation, June 25-28, 2009, Shanghai Normal University, Shanghai, China. Talk in Mini Symposium: How the changes of inlinks impact the PageRank
- Lead a team of Fitchburg State College students to Zhejiang GongShang University for study aboard and exchange program, Hangzhou, China. May 17 - June 8, 2009.
- Served as a member of Advisory Board for the Computer Information Systems in Mount Wachusett Community College.

1. Frits Lander

2. Education – degree, discipline, institution, year

Artium Degree	Science and Mathematics	Rødovre State School (Denmark),	1962
Academy Engineer Degree	Mechanical Engineering	Engineering Academy of Denmark	1968
Postgraduate Courses	Computer Science	Technical University of Denmark	1971-78

3. Academic experience – institution, rank, title, when (ex. 1990-1995), full time or part time

Fitchburg State University	Instructor		1982-86	Full
	Assistant Professor		1986-93	Full
	Associate Professor		1993-2004	Full
	Associate Professor	Chair, Computer Science	2004-2013	Full

4. Non-academic experience – entity, title, brief description of position, when, full time or part time

Danish State Railroads	Chief Engineer	Logistic simulation, scheduling, facilities planning, heuristic modeling. Teaching FORTRAN, lecturing on operations research.	1974-82	Full
Brüel & Kjær	Production Manager Assistant	Simulations for complex decision making. Programming in GPSS and FORTRAN.	1970-74	Full
Royal Danish Air Force	2nd Lieutenant, Tactical Air Command	Teaching & analyzing weapons delivery. Computer systems for combat operations.	1968-70	Full
Royal Danish Air Force	Lieutenant	Teaching & analyzing weapons delivery. Computer systems for combat operations.	1970-72 & 1976-80	Part
AVCO Systems Division	Consultant	Simulation of conical dipole antennas.	1983	Part
Digital Equipment Corporation	Consultant	Measurement of the electrical performance of ceramic substrate test circuits.	1984	Part
Digital Equipment Corporation	Consultant	Consulting the design and construction of an EMI measurement site.	1988	Part

North East Systems Associates	Consultant	Software engineering of Scope54, a semi-automated package designed to fully support and enhance the performance of the HP 54120 digitizing oscilloscope.	1989-95	Part

5. Current membership in professional organizations
   - Danish Society of Chemical, Civil, Electrical, and Mechanical Engineers.
   - European Federation Of National Engineering Associations. Entry in the European Register of Higher Technical Professions.
   - Association for Computing Machinery.

6. Honors and awards

   - Departmental Academic Performance Award, September 1983.
   - Granted a Distinguished Service Award, June 1985.
   - Granted Tenure, September 1987.
   - Merit Bonus, December 2002.
   - Faculty Member of the Year 2003.

7. Service activities (within and outside of the institution)

   - AUC Curriculum Committee, from September 1986-88. The new Liberal Arts and Science program was developed in this period.
   - AUC Curriculum Committee, 1995-6.
   - Committee on Promotion, 1998-2001.
   - AUC Curriculum Committee, from September 2002-3.
   - Served on the AUC Curriculum Committee, 2003-4.
   - Serving as Department Chairman, 2004-13.

8. Briefly list the most recent professional development activities

   - ABET Best Assessment Processes Symposium, April 25-26 April 2008, Atlanta, GA
   - ABET Symposium, April 14-16, 2011, Indianapolis, IN
   - ABET Symposium, April 19-21, 2012, St. Louis, MO

1. Natasha Kurtonina

2. Education

- Postdoctoral Studies and Research (1997-1999)
- Institute for Research in Cognitive Science

Department of Computer Science
University of Pennsylvania

- Ph. D. in Computer Science (1995)

(cross listed with Mathematical Logic and Computational Linguistics)
University of Amsterdam, the Netherland

- Ph. D. (1989) in Artificial Intelligence

Department of Applied Mathematics and Department of Philosophy, Moscow State University

3. Academic experience

Fitchburg State University, 2000 – currently (full time)
Massachusetts
Associate Professor

University of Pennsylvania, PA 1997-2000 (full time)
Institute for Research in
Cognitive Science
Research Scholar

Ghent University 1996-1997 (full time)
Belgium
Department of Logic
Research Scholar

Institute Logic, Language and Computation, 1993-1995 (full time)
University of Amsterdam
Ph.D. Research Associate

Leuven Catholic University, 1991-995 (full time)
Department of Logic
Visiting Assistant Professor

Moscow State University 1990-1991 (full time)
Department of Artificial Intelligence
Assistant Professor

4. Non-academic experience
5. Certifications or professional registrations
6. Current membership in professional organizations

- American Association for Computing Machinery

- American Association for Symbolic Logic
- American Association for Philosophical Logic
- European Association for Symbolic Logic
- European Association Association for Computational Linguistics

7. Honors and awards
- Fitchburg State University Gallucci Grant
- Fitchburg State University City Grant to develop graduate curriculum
- *University of Pennsylvania, USA* Research Award
- *Institute for Advanced Studies, Princeton, USA.* Research Grant
- *National Science Foundation of Belgium* Research Grant
- *National Science Foundation of The Netherlands* Research Grant
- *Dutch National Institute for Mathematics and Computer Science* Research Grant
- *Austrian National Science Foundation. Salzburg University. Research Grant.*

8. Service activities (within and outside of the institution)

FSU Graduate Council, (2006-2007), College Curriculum Committee (2004-2206), College International Council (2008,2011), host Women in Computer Science events
Volunteer to teach Computer Literacy to female inmates of Massachusetts
President of North American Chapter of Rachmaninoff Society

9. Briefly list the most important publications and presentations from the past five years – title, co-authors if any, where published and/or presented, date of publication or presentation

- New edition (2012) of the book: Frames and Labels. A modal Analysis of Categorial Inference. *Institute for Logic, Language and Computation. University of Amsterdam* ISBN 90-7479528-5
- Relational Semantics for the Lambek-Grishin Calculus (with Michael Moortgat). In "The Mathematics of Language". Springer Verlag. 2010
- Completeness Results for the Lambek-Grshin Calculus (with Michael Moortgat). In "Proceedings of the Mathematics of Language Coference". Los Angeles, UCLA, 2012
- Partial Proof Trees, Hybrid Logic and Quantifiers Scope (with Aravind Joshi and Seth Kulick) In Logical Aspects of Computational Linguistics, Selected Papers, Springer Verlag, 2011

10. Briefly list the most recent professional development activities

Developing a course on Game Programming
Working on presenting results on complexity of propositional proofs in various international conferences

1. Nadimpalli Mahadev

2. *Education* :

Ph. D.	Combinatorics & Optimization	University of Waterloo, Canada	1984
M. Math.	Combinatorics & Optimization	University of Waterloo, Canada	1980
B. Stat (Hons).	Mathematics & Statistics	Indian Statistical Institute, Kolkata	1978

3. Academic experience :

Fitchburg State	Professor beginning 9/2002	Department Chair Graduate Program Chair	2000-2004 2000-2005	Full-time
Northeastern University	Associate Professor		1992-2000	Full-time
	Assistant Professor		1986-1992	Full-time
University of Winnipeg, Canada	Assistant Professor		1984 - 1986	Full-time
Ecole Polytechnique Federale de Lausanne	Visiting Professor		Several times since 1987	Part-time

4. Non-academic experience :

5. Certifications or professional registrations

6. *Current membership in professional organizations :*

ACM SIGCSE

7. Honors and awards:

Special Projects Grants from Fitchburg State University for work in “Use of Robots in Advanced Computer Science Courses” and “Speech Recognition Algorithms”.

8. *Service activities (within and outside of the institution):*
   Served on college-wide committees such as Information Technology Advisory Board, Promotions Committee etc.

   Devised the assessment plan for the two undergraduate programs in the department.

   Served as a volunteer chief judge in Spelling Bee, Geography Bee etc., organized by North South Foundation.

9. *Briefly list the most important publications and presentations from the past five years*

   - Xuzhou Chen and N.V.R. Mahadev. Enhancing the Undergraduate Teaching and Research Using Robotic Programming, Journal of Computing Sciences in Colleges, Vol. 28, No. 2, 2012, pp. 57-64.
   - Xuzhou Chen and N.V.R. Mahadev. A PC Based Robot for Learning Computer Vision and Advanced Programming. ACM SIGCSE, 2012, Raleigh, North Carolina, USA, pp. 664-665.

10. Recent professional development activities:

- Built PC Robots.
- Wrote NSF proposals for using PC Robots in classroom for teaching advanced algorithms.
- Developing a course in Speech Recognition.
- Poster presentation at ACM SIGCSE conference in 2012.

1. Ricky J. Sethi

2. Education

Institution	Major	Degree	Year
University of California, Berkeley	Neurobiology/Physics	B.A.	1996
University of Southern California	Physics/Business (I.S.)	M.S.	2001
University of California, Riverside	Computer Science	Ph.D.	2009

3. Academic experience

2014-present	Assistant Professor, Fitchburg State University
2014	Postdoctoral Associate, UMass Amherst/UMass Medical School
2010-2013	NSF Computing Innovation Fellow, UCLA/USC Information Sciences Institute
2009-2010	Postdoctoral Scholar, University of California, Riverside
2007-2009	Research Associate, University of California, Riverside
2006	Researcher, Physics Department, United States Coast Guard Academy, Department of Homeland Security (DHS)

4. Non-academic experience

2013-present	Director of Research, The Madsci Network
2001-2002	Network Operations Manager, Iviewit Technologies

5. *Certifications or professional registrations*

- Fellow, North American Academy of Arts and Sciences

6. Current membership in professional organizations

- Member, YSP/Madsci Financial Board
- Member, American Institute of Physics
- Member, IEEE

7. *Honors and awards*

• CTL Innovation Award, Fitchburg State University	2015	
• NSF Computing Innovation Fellowship	2010 – 2012	$285,371
• Faculty Technology Grant, DeVry University	2004	$5,000

8. Service activities (within and outside of the institution)

- NSF Panel Service, Panelist, NSF Cyberlearning DIP Review Panel, 2012 – 2013
- NSF Panel Service, Panelist, NSF Cyberlearning CAP Review Panel, 2013

- Co-Chair, Scientific Workflows for Machine Learning Applications (SWMLA) in conjunction with International Conference on Machine Learning (ICML), 2013
- Local Organizing Chair and Registration Chair, International Conference on In- telligent User Interfaces (IUI), 2013
- Demos Chair, International Conference on Collaboration Technologies and Sys- tems (CTS), 2013 – 2014
- Center for Teaching and Learning, Fitchburg State Univeristy, 2014 - Present
- Editorial Board Member, International Journal of Computer Vision & Signal Pro- cessing, 2011 - Present
- Associate Editor-in-Chief for the Journal of Postdoctoral Research, 2012 - Present

9. Most important publications and presentations from the past five years

- Balaji Polepalli Ramesh, Ricky J. Sethi, and Hong Yu, "Figure-Associated Text Summarization and Evaluation". PLOS ONE (2014).
- Ricky J. Sethi, Hyunjoon Jo, and Yolanda Gil, "Structured Analysis of the Atomic Pair Actions Dataset using Workflows". Smart Approaches for Human Action Recognition, Pattern Recognition Letters (PRL) (2013).
- Ricky J. Sethi, "Towards Defining Groups and Crowds in Video Using the Atomic Group Actions Dataset". IEEE International Conference on Image Processing (ICIP) (2015).
- Ricky J. Sethi, "A Perceptually-Inspired Stochastic Framework for Video Search and Analysis". Perception Inspired Video Processing held in conjunction with ACM International Conference on Multimedia (ACM MM) (2014).
- Ricky J. Sethi, Yolanda Gil, Hyunjoon Jo, and Andrew Philpot, "Large-Scale Multimedia Content Analysis Using Scientific Workflows". ACM International Conference on Multimedia (ACM MM) (2013).
- Ricky J. Sethi, Hyunjoon Jo, and Amit K. Roy-Chowdhury, "A Generalized Data-Driven Hamiltonian Monte Carlo for Hierarchical Activity Search". IEEE International Conference on Image Processing (ICIP) (2013).
- Ricky J. Sethi and Lynn Bry, "The Madsci Network: Direct Communication of Science from Scientist to Layperson". 21st International Conference on Computers in Education (ICCE) (2013).

10. Most recent professional development activities

- Program Development: Data Analytics and Data Science
- Course Development: Introduction to Data Science
- Research: Scientific Workflows, Image Analysis, and Visual Stylometry in the Digital Analysis of Art
- Research: Group Activity Detection
- Research: Virtual Organizations and Collaborative Argumentation

1. Stephen Taylor

2. *Education* :

Ph. D.	Computer Science	Worcester Polytechnic Institute	1995
M. S.	Computer Science	Worcester Polytechnic Institute	1990
A. B.	Mathematics	Harvard College	1968

3. Academic experience :

Fitchburg State	Professor beginning 9/13	Several-time Graduate Program Chair	2002 - present	Full-time
Holy Cross	Visiting Assistant Professor		19992002	Full-time
United Arab Emirates University	Lecturer		1997-1998	Full-time
Worcester Polytechnic Institute	Instructor, Visiting Assistant Professor		1993-1997	Full-time after 1994

4. *Non-academic experience :*

Data General	Principal Systems Programmer	Developed and maintained office productivity software	1985-1993	Full-time
Interactive Sciences	Senior Systems Programmer	Developed and maintained operating system and compilers.	1969-1985	Full-time

5. *Certifications or professional registrations*

6. *Current membership in professional organizations :* ACM, Arabic Linguistics Society, American Machine Translation Association, IEEE Computer Society

7. *Honors and awards:*

Fulbright Fellowship, Damascus, Syria, September 2009 – June 2010;
FSC Ruth Butler Grant of work-study student, Spring 2008
FSC Gallucci Grant to attend *Computability in Europe* conference in Siena, Italy, June 2007

8. *Service activities (within and outside of the institution):*

FSU Graduate Council, 2008-2012
Technical Support for Follow the Women Middle East bicycle ride for peace, May 2008.
Chair of Department Accreditation Effort 2005-2008
FSC All-College Curriculum Committee, 2004-2008
Host for High School Programming Contests at Fitchburg State, 2005-
Association for Machine Translation Webmaster, 2004-2009

9. *Briefly list the most important publications and presentations from the past five years*

"A multi-tier annotation regimen for Arabic ammiya
", International Arabic Linguistics Symposium, June 2013
Chapter in IALS series from Johns Benjamins, to appear.
Shamy Dictionary, an Android phone app for an Arabic Dialect Dictionary, released 2012
Soft Arabic Keys an Android phone app for Arabic Text Input, released 2012
Keynote address, *Making Plans*, Clark University Graduate Student Multidisciplinary Conference, April 2011
"A Morphological Analyzer for Damascene Colloquial Arabic Words," 4th International Arabic Linguistics Symposium, Alexandria, Egypt, December 2010

10. Recent professional development activities:

International Arabic Linguistics Symposium submission, Spring 2013
Language Resources and Evaluation Conference, Istanbul, May 2012
Arabic Linguistics Symposium, New York City, March 2012

Appendix C – Equipment

Laboratory facilities are available for student use in conjunction with various courses. Some courses require special equipment and/or software. Here, equipment and software are listed by room. Every classroom on campus is technology-mediated with an instructor podium and projector. Our department has membership in the Microsoft Developer Network Academic Alliance which allows us to deliver many development tools to them for free via a DreamSpark e-commerce site (http://msdnaa.fitchburgstate.edu).

E201 and 202 labs (used for CS 1, CS 2, Systems Programming, Programming Languages, Object Oriented Programming, Algorithms and Data Structures)
24 thin client computers running Windows
Microsoft Visual Studio
Java Netbeans
Eclipse
XAMPP for MySQL and TomCat
Python
DrRacket
Perl
Visual Prolog
Microsoft Project
Microsoft Visio
Dia diagramming software
Greenfoot
Textedit

E207 lab (used for Microprocessors, Computer Organization and Data Communications)
8 Lab Stations w/ISA bus dual-boot PCs (DOS / Windows XP)
Digital Sampling Oscilloscopes
EEPROM programmers
Wire-wrapped microprogrammed controllers
Lab Benches
ISA bus breakout system (PCμLab)
Prototyping systems
Microsoft Assembler
LVSim-Dcomm Digital Communications Simulation Software
EEPROM programmer software
MARIE computer simulation software

Other (virtual)
Linux server used in various programming classes
Oracle database server

E203 lab (used for Introduction to Electronics and Digital Electronics)

8 Lab Stations w/thin client computers running Microsoft Windows
Multisim circuit simulation software
Digital Sampling Oscilloscopes
Circuit prototyping systems w/ power supplies, signal generators, digital inputs/outputs
Digital Multimeters
PICC compiler
MPlab IDE
PICkit device programmers

E102 (used for Local Area Networks)
18 Computers used for LAN class
Microsoft Windows 2003 Server and XP Client
Wireshark Protocol Analyzer

E205 Project Lab
3 – Matlab programming/data analysis workstations
3 – USB data acquisition subsystems
1 GHz digital sampling oscilloscope
Network analyzer
Logic Analyzer
Sherline CNC Milling machine
Sherline CNC Lathe
Drill press
Band saw
Printed circuit board (PCB) fabrication materials
Ultiboard PCB layout software

Appendix D – Institutional Summary

1. The Institution

Fitchburg State University
160 Pearl Street
Fitchburg, MA 01420

The President of the university is Dr. Robert Antonucci and the Vice President for Academic Affairs is Dr. Robin Bowen. This report is submitted by Associate Vice President for Academic Affairs, Dr. Paul Weizer. FSU is accredited by NEASC. The most recent accreditation evaluation visit was in 2012.

2. Type of Control

Fitchburg State University is one of six comprehensive State Universities in the Massachusetts higher education system; and like its sister institutions, the University's governance procedures and organizational structure are largely mandated by either state law or collective bargaining agreements.

The official governing body is the Board of Higher Education (BHE) which is staffed by the state's Department of Higher Education (DHE), led by the Commissioner of Higher Education. The BHE consists of the Secretary of Education, ex officio, or his designee, thirteen voting members appointed by the governor to reflect regional geographic representation, and three members chosen to represent public institutions of higher education.

According to the Department of Higher Education's website, the Board has four fundamental responsibilities:

1. Define the mission of and coordinate the Commonwealth's system of higher education. The Department, in conjunction with the universities' Boards of Trustees, holds the system accountable for achieving its goals and establishing a comprehensive system to measure quality by defining educational achievement and success with the use of standards and measurements.

2. Approve the awarding of degrees and define and authorize new functions new programs or consolidate, discontinue, or transfer existing functions, educational activities, and programs.

3. Analyze present and future goals, needs, and requirements of public higher education and establish goals to achieve a well-coordinated quality system of public higher education in the Commonwealth.

4. Develop a rational and equitable statewide tuition plan for state universities and the community colleges.

In addition, the Board, through the BHE, establishes policy for state colleges and universities and outlines performance measures to use as comparisons with peer institutions in other states. As the employer of record for all those represented by the four collective bargaining agreements, the BHE is responsible for collective bargaining negotiations, contract, and grievance administration (see http://www.mass.edu/aboutus/summary.asp). Contracts and collective bargaining agreements are available on the Office of Human Resources website. (See http://www.fitchburgstate.edu/humanres/collectivebargaining.cfm)

Massachusetts law delineates the roles of the Board of Trustees and the chief operating officers of the state universities. At the local level, Fitchburg State is overseen by eleven voting members of a Board of Trustees. Nine are appointed by the governor and serve terms up to five years; these appointments are renewable one time. A student trustee, elected by the student body, serves a one year term, and the Alumni Association elects an alumni trustee who serves a five-year term. No trustee may serve more than two consecutive terms. The Board elects its chairperson from among its members.

The duties and responsibilities of the Board of Trustees are defined in the General Laws of the Commonwealth of Massachusetts Chapter 15a, section 22. This law states that the Fitchburg State University trustees are charged with the fiduciary management of the institution, including determination of fees, establishment of personnel management policy, staff services, and the general business of the institution. They elect the University President with the approval of the Board of Higher Education, adopt an annual plan of financial operation, award degrees in approved fields, and develop the university's mission statement consistent with the mission of the Commonwealth's system of public higher education. The trustees conduct an annual evaluation of the president based on goals they have jointly established with him. This review is forwarded to the BHE which is the hiring authority for all chief executive officers in the state university system.

The Board of Trustees, operating under by-laws revised in October 2000, meets at least four times annually; the chairperson, the University president, and five trustees may, through petition, also call special meetings. The meetings and minutes are public, and executive sessions are limited to purposes specified in the by-laws. There are five standing Trustee committees: Academic Affairs, Student Life, Administration and Finance, Personnel, and the Executive Committee. The Board of Trustees also performs a self-evaluation annually.

Massachusetts General Law Chapter 15a, section 9 clearly defines the authority and responsibilities of the University President who has the responsibility for establishing the structure of his administration. In June 2003, Dr. Robert Antonucci was appointed President and reduced the number of Vice Presidents from five to two. The roles and responsibilities of each administrator are defined and kept on file, along with the procedures for their selection, in the office of the Assistant Vice President for Human Resources and Payroll Services. Currently, the President is advised by an Executive Committee (EC) composed of the Vice President for Academic Affairs, Vice President for Finance and Administration, Executive Assistant to the President for External Affairs, Associate Vice President for Academic Affairs, the Dean of Graduate and Continuing Education, the Chief Operating Officer, the Chief Information Officer and Assistant Vice President for Human Resources and Payroll Services. In addition to his meetings with the EC, the President meets frequently with the Vice President for Academic Affairs and the Vice President for Finance and Administration. The EC also meets regularly; and each of these administrators holds

regular staff meetings. The Vice President for Academic Affairs meets weekly with the academic affairs administrators.

The structure of academic governance at Fitchburg State University, as in the other state universities, is governed by the contractual agreement between the BHE and the Massachusetts State College Association (MSCA). The duties and responsibilities of department chairpersons are defined by contract; they are elected by department faculty in accordance with procedures established by Article VI of the contract and may serve up to three consecutive three year terms. In addition to meeting with department faculty, the chairs meet at least monthly with the Academic Vice President and, at Fitchburg State, they meet monthly on their own. These separate meetings were intended to facilitate communication with the Academic Vice President and focus on developing agenda items for upcoming Vice President/Chairs meetings.

A separate contractual agreement between the BHE and the MSCA lists the responsibilities of the University and the faculty teaching in Graduate and Continuing Education. GCE programs are administered by graduate program chairs and evening undergraduate program managers; these are positions created by the University and filled by the Graduate and Continuing Education Dean, in consultation with department chairs, who also meets with her appointees several times a year.

At the departmental level, governance begins with curriculum committees which meet to consider new course offerings, program revisions and new program development. Approved proposals, along with those made by faculty, students, or administrators and those relating to changes in academic or student life policies are then submitted to the All University Committee (AUC). The AUC is composed of eight faculty members elected by their peers, three administrators appointed by the President, and three students selected by the Student Government Association (SGA). The AUC has three standing committees -- Curriculum, Academic Policies, and Student Affairs--as well as ad hoc committees to deal with special issues. The Curriculum and Academic Policies committees are composed of 16 faculty members appointed by the MSCA, three administrators appointed by the President, and three students selected by the SGA, while the Student Affairs committee is comprised of nine students, five administrators and five faculty members. The standing committees recommend approval or disapproval to the AUC which then makes a recommendation of approval or disapproval to the President who has the final authority on each change.

There is a separate, contractually defined governance structure for graduate policies and curriculum. Departments with a graduate program also have a graduate curriculum committee which may recommend changes to graduate curriculum or policies. If both the graduate program chair and the department chair approve the changes, they are reviewed by the Graduate Council. The Graduate Council then makes a recommendation of approval or disapproval to the President who has the final authority on each change. The Graduate Council is composed of five faculty members, appointed by the MSCA, three administrators appointed by the President, and one graduate student elected by the other Graduate Council members.

At the beginning of each academic school year the President hosts two opening addresses, one for classified personal and administrators and the other for faculty, librarians, and

administrators. These addresses typically review major past events and forecast those expected in the new academic year.

Every other year, the graduate program coordinators, undergraduate program managers, and graduate and continuing education faculty meet at the beginning of the academic year. These meetings include a series of professional development workshops, departmental meetings, and updates from the Graduate Dean, Vice President for Academic Affairs, and President. Additionally, the Dean and Associate Dean of Graduate and Continuing Education meet with graduate chairs and undergraduate program managers throughout the year as needed. Once each semester, there is a combined meeting for all department chairs, graduate chairs, and undergraduate program managers with the entire academic affairs team for the purpose of discussing items of common interest.

3. Educational Unit

As of July1, 2013, Dr. Brady Chen will be assuming the chair of the department. The chairs report to Vice President of Academic Affairs Dr. Robin Bowen, who reports to President Robert Antonucci.

4. Academic Support Units

The chair of the Mathematics Department is Dr. Mary Ann Barbato.
The chair of the Geo/Physical Science Department is Dr. Bruce Duncan.

5. Non-academic Support Units

The head of the library is Robert Foley.
The Chief Information Officer is Steven Swartz.
The Tutoring Center is run by Tom Rousseau.
Career services is headed by Director Erin Kelleher.
Counseling is headed by Robert Hynes.
Disability Services is run by Julie Maki.

6. Credit Unit

It is assumed that one semester or quarter credit normally represents one class hour or three laboratory hours per week. One academic year normally represents at least 28 weeks of classes, exclusive of final examinations. If other standards are used for this program, the differences should be indicated.

7. Tables

Complete the following tables for the program undergoing evaluation.

Table D-1. Program Enrollment and Degree Data

Computer Science

Academic Year		Enrollment Year					Total Undergrad	Total Grad	Degrees Awarded			
		1st	2nd	3rd	4th	5th			Associates	Bachelors	Masters	Doctorates
2013	FT								0	13	35	0
	PT								0			0
2012	FT	14	16	14	19		63		0	12	24	0
	PT	0	4	5	4		13		0			0
2011	FT	13	20	20	10		63		0	4	28	0
	PT	1	4	3	6		14		0			0
2010	FT	20	23	9	12		64		0	5	21	0
	PT	2	2	3	5		12		0			0
2009	FT	18	11	6	7		42		0	8	53	0
	PT	1	3	5	4		13		0			0
2008	FT	15	9	7	4		35		0			0
	PT	1	2	4	4		11		0			0

FT--full time
PT--part time

Table D-2. Personnel

Computer Science

Year[1]: 2012-13

	HEAD COUNT		FTE
	FT	PT	
Administrative[2]	0.375	0	0.375
Faculty (tenure-track)[3]	5.625	0	5.625
Other Faculty (excluding student Assistants)	0	0	0
Student Teaching Assistants[4]	0	0	0
Technicians/Specialists	0	0	0
Office/Clerical Employees	1	0	1
Others[5]	0	0	0

[1] Data on this table should be for the fall term immediately preceding the visit. Updated tables for the fall term when the ABET team is visiting are to be prepared and presented to the team when they arrive.

[2] Persons holding joint administrative/faculty positions or other combined assignments should be allocated to each category according to the fraction of the appointment assigned to that category.

[3] For faculty members, 1 FTE equals what your institution defines as a full-time load.

[4] For student teaching assistants, 1 FTE equals 20 hours per week of work (or service). For undergraduate and graduate students, 1 FTE equals 15 semester credit-hours (or 24 quarter credit-hours) per term of institutional course work, meaning all courses — science, humanities and social sciences, etc.

[5] Specify any other category considered appropriate, or leave blank.

Appendix E – Four-Year Plan of Study

C
S
C

SUGGESTED FOUR-YEAR PLAN OF STUDY

COMPUTER SCIENCE

2013-2014

Assumes placement exams cleared

*Also includes suggested plan for Mathematics Minor.

FRESHMAN YEAR

Fall Semester			Spring Semester		
ENGL 1100	Writing I	(3)	ENGL 1200	Writing II	(3)
CSC 1500	Computer Science I	(3)	CSC 1550	Computer Science II	(3)
MATH 1300	Precalculus (if needed)	(3)	CSC 1900	Discrete Math (SMT)	(3)
PHYS 2300	General Physics I (SMT)	(4)	PHYS 2400	General Physics II (SMT)	(4)
History	LA&S elective (CTW)	(3)	EXSS 1000	LA&S elective (SMT)	(3)
	Total Credits	(16)		Total Credits	(16)

SOPHOMORE YEAR

Fall Semester			Spring Semester		
CSC 1600	Intro. to Electronics	(4)	CSC 1650	Digital Electronics	(4)
CSC 2560	Systems Programming	(3)	CSC 3700	Algorithms & Data Struct.	(3)
MATH 2300	Calculus I (Option B)	(4)	MATH 2400	Calculus II (Option B)	(4)
Literature	LA&S elective (ARTS)	(3)	SPCH 1000	Speech (ARTS)	(3)
	LA&S elective (CTW)	(3)	Behavior	LA&S elective (CTW)	(3)
	Total Credits	(17)		Total Credits	(17)

JUNIOR YEAR

Fall Semester			Spring Semester		
CSC 2600	Computer Organization	(4)	CSC 3600	Microprocessors	(4)
CSC 3200	Programming Languages	(3)	CSC 3100	Operating Systems	(3)
MATH 1800	Business Stat (Option B)	(3)	MATH 2600	Linear Alg. (Option B)	(3)
CSC 3XXX	CSC Elective	(3)	CSC 3XXX	CSC Elective	(3)
*MATH 3300	Calculus III	(3)	*MATH 4200	Probability & Stat I	(3)
	Total Credits	(16)		Total Credits	(16)

* Suggested courses for Math minors; all others can substitute free electives.

SENIOR YEAR

Fall Semester			Spring Semester		
CSC 3XXX	CSC Elective	(3)	CSC 4400	Software Engineering	(3)
CSC 3XXX	CSC Elective	(3)	MATH 3XXX	Math Elective	(3)
Art/Music	LA&S elective	(3)	CSC 3XXX	CSC Elective	(3)
	Free Elective	(3)		Free Elective	(3)
	Total Credits	(12)		Total Credits	(12)

SUGGESTED CSC/MATH ELECTIVES

Fall Semester		Spring Semester	
CSC 3050	Web Programming with Java	CSC 3300	Assembly Language
CSC 3400	Data Comm & Networking	CSC 3450	Local Area Networks
CSC 3500	Object Oriented Programming	CSC 4550	Database Programming
CSC 4940	Internship	CSC 4940	Internship
		MATH 3000	Geometry

Appendix F – Review of Library Services and Library Materials

Services Provided by the Library Staff to the Computer Science Department Students and Faculty

Standards for Service

The library staff strives to meet the Association for College and Research Libraries (ACRL) standards for services.

> "The library should establish, promote, maintain, and evaluate a range of quality services that support the institution's mission and goals. The Library should provide competent and prompt assistance for its users. Hours of access to the library should be reasonable and convenient for its users. Reference and other special assistance should be available at times when the institution's primary users most need them."

The Library staff of the Amelia V. Gallucci-Cirio Library is assigned to four departments and they provide a full range of services to the College community. The professional library staff consists of 6.37 fte librarians, the library is open 90-95 hours per week, and week-end and holiday coverage is provided. The librarians are assigned as liaisons to each academic department, they all assist at the reference desk, they all participate in the library instruction program, and they all assist in updating and editing the various online subject guides.

The Library is housed in a 6 story multi-use building. It occupies one-half of the 1st floor, one-half of the 2nd floor, part of the 3rd floor and all of the 4th floor. The building is shared with the Campus Center, commuter's cafeteria, and numerous academic student support services on the 3rd floor. A building reconstruction was begun in 2011

The Library mission statement is the following:

> *Our mission is to enhance the learning experience for our students by providing appropriate resources and technologies to stimulate their intellectual growth, initiate a successful career path, and develop leadership skills. Working in collaboration with faculty we support the needs of the curriculum. We strive to teach students information literacy skills needed for lifelong critical thinking and learning.*

The library's professional and support staff is the primary resource and consist of 6.3 librarians, 1 administrator, 8.6 support staff, and 3.00 (fte)) student assistants.

Since 2007, the library staff has completed their 2004-09 strategic plan, the 2012 NEASC self-study was completed during the spring 2012 semester, and has been implementing the 2010-14 strategic plan. The staff has been able to provide enhanced additional services. A facilities

consultant was hired to review and recommend suggestions for the Library building use and future space usage/needs.

The Library Catalog and Website

The library utilizes the VOYAGER library management systems and the Library's WEB site includes the online catalog, electronic reserves, a journal locator service, online forms, and information about the library, etc. More importantly it provides access to over 143 electronic databases. These include reference services; index/abstracting services, journal full-text services, etc. There is access to approximately 45,000 unique journal titles. Off-campus access to the database is also provided.

Circulation Services

Students are allowed to borrow items for four weeks with one renewal and faculty are allowed to borrow materials for six months with one renewal. Document delivery services for the faculty began in the fall of 2012 and the library staff will deliver physical items to faculty offices as well as provided electronic/scanned documents to the individual faculty members e-mail address.

Reserves

The Access Services Department provides for physical reserves and electronic reserves. Various types of resources and materials are eligible: books, reports, photocopies, DVDs (Audio and ROM), power point presentations, equipment. Annual usage statistics can be found in the library's annual report.

CIRCULATION Statistics

	2006/07	2007-08	2008-09	2009-10	2010-11	2011-12
Circulation	14,386	12,110	12,007	13,531	11,131	9,623
Reserves	19,194	11,709	12,923	12,068	12,418	6,276

Reference

The Reference staff consists of two librarians however the five Librarians participate in providing traditional reference services. The reference desk is staffed for 76 hours per week during the academic year and 52 hours during the summer semesters. This includes coverage until 9PM. Weekend coverage is provided by a part -time professional and the desk is staffed from noon to 6PM on Saturdays and 1-9PM on Sundays. Each Librarian works a night and they all conduct Instruction classes. There is a toll free number for reference questions and there is an online reference form for questions for the Distributed Learning students.

REFERENCE Statistics

	2006/07	2007-08	2008-09	2009-10	2010-11	2011-12
Questions	5,825	4,632	6,417	6,005	6,752	5,708
Hours of service						
Summer	52	52	40	45	45	45
Academic Year	76	76	74	74	74	67

Distance Education /Distributed Learning Support

Not needed for CS programs

Interlibrary Loans

Fitchburg State University, students, staff, and faculty have access to millions of titles via Interlibrary Loan services. We also have interconsortia borrowing agreements with the Academic & Research Collaborative libraries of Worcester County. Students and faculty are eligible to borrow materials. Student borrowing became available in 2003. In addition students, faculty and staff can take advantage of WILL (Walk In Interlibrary Loan). This agreement encompasses Community Colleges, State Colleges and Public Universities in Massachusetts. Unfortunately, UMASS Medical-Worcester does not participate in the walk in loan system. The interlibrary loan service implemented the web-based ILLiad document delivery service in 2012 and this provides for electronic self-populating of item requests and web-based delivery of scanned copies.

INTERLIBRARY LOAN Statistics

	2006/07	**2007-08**	**2008-09**	**2009-10**	**2010-11**	**2011-12**
FSU Borrowing	1,624	1,400	709	819	794	1,044
Lending	5,118	4,771	3,122	3,236	3,340	2,922

Library Instruction

The Library Instruction services offered by the Librarians are available to all classes, the graduate and undergraduate programs, and to on and off campus courses. The librarians work with the faculty to design handouts, exercises, projects, which will enable students to become effective library researchers in their discipline. A library Instruction classroom is located on the 3rd floor of the Library and contains 30 student workstations, one instruction workstation, an overhead projector, and an electronic white board. In order to provide flexibility and offer instruction classes in other classroom buildings and off-site, additional equipment includes a portable lap-top computer, a portable screen and a portable projector.

INSTRUCTION **Statistics**

	2006/07	**2007-08**	**2008-09**	**2009-10**	**2010-11**	**2011-12**
Classes	133	82	92	116	109	167
Students	2,583	2,687	1,929	2,522	2,293	3,298
Graduate classes	10	10	9	N/A	10	18

There were no instruction classes for the Computer science programs

Conclusion

The library staff offers a wide variety of quality services. Library hours and reference desk hours are sufficient, numerous activities such as online forms, and toll free numbers are available; on campus and off campus access to databases is available; there are sufficient workstations; e-reserves are readily available; interlibrary loans are available to all users; all user have borrowing privileges at other Mass. Public Higher Ed. Libraries (MPHE) and the academic libraries of Worcester County.

Reference, circulation, and instructional services are designed to enable users to take full advantage of the resources. We should be having CS Instruction classes.

Interlibrary loans are available to all faculty, staff and students.

Library hours are consistent with the demand for services and they meet the average of the other MPHE Libraries.

Library services equal to the on campus users are offered to the off-campus users. They include but are not limited to: Library Instruction, Reference Assistance, Circulation, Interlibrary Loans, and Document Delivery. All off-campus users have access to the Library web sites and there are dedicated pages for these users.

The library director prepares an annual report and it includes statistical information, financial information and a narrative summary of each department's activities. In addition the ACRL Points of Comparison are completed as well as an internal spread sheet that includes statistics from 1987 and they include: materials expenditures, funding, services, collection size-annual, and collection size-summary.

Library *Materials* provided to the Computer Science faculty and students

PROGRAM DESCRIPTION

The Computer Science offers a Bachelor of Science in Computer Science and Computer Information Systems. The department also offers minors in Computer Science and joint Graphics minor with the Communications Media and Industrial Technology departments. The department also offers a Master's degree program in Computer Science. The graduate program specializes in hardware, software, networking and state-or-the-art knowledge. There is a thesis option for the graduate students.

COLLECTION DESCRIPTION

General Guidelines for Collection Depth

1 Minimal A level that consists mostly of basic works

2 Basic Information

2a A level that introduces and defines the subject and that indicates the varieties of information available elsewhere

2b Basic Instructional Support A level that introduces course work and research for undergraduate courses, including a wide range of basic monographs and reference tools pertaining to the subject and targeted to undergraduate students

3 Study or Instructional Support

3a Basic Study A level that supports undergraduate courses

3b Intermediate Study A level that supports upper division undergraduate courses

3c Advanced Instructional Support A level that supports course work and research for graduate and undergraduate courses, including a wide range of basic monographs and reference tools pertaining to the subject

4 Research A level that supports independent research and preparation of doctoral dissertations

5 Comprehensive Inclusion of all significant works for a defined topic

The library staff strives to have our library collections support course work and research for graduate and undergraduate courses, with the inclusion of a wide range of basic monographs and reference tools pertaining to the subject. This meets the goal of Level 3c, **Advanced Instructional Support** A level that supports course work and research for graduate and undergraduate courses, including a wide range of basic monographs and reference tools pertaining to the subject

The following review includes information regarding the monograph collections, serials, electronic databases, and primary sources. Assessment of the collections is based upon indicators outlined in the 1998 *WLN* (Washington Library Network) *Collection Assessment Manual.*

Library Materials Collections

As of July 1, 2012 the material resources of the Gallucci-Cirio Library included 205,759 cataloged books; 1,029 electronic books; 95,999 microfilm or microfiche and 16,750 bound journal volumes. The Library also provides access to 49,007 unique journals; 143 electronic databases, including subject based indexes and abstracts, reference titles, and multi-subject "aggregator" services such as ProQuest Central, EbscoHOST and Gale/INFOTRAC. There are approximately 98 standing orders. In addition there are 2,920 audio visual materials.

There is an established Collection Development policy and it is based on a liaison model. The plan is based on having each Librarian assigned to a department(s) and they in turn are responsible for contacting the departments, selection, weeding, etc. Each discipline (major, minors and master's programs) has a written conspectus analysis based on the WLN *Collection Assessment Manual*, 4th and 5th editions. The resulting profile includes goals and objectives for monographic purchasing. The cooperation and coordination effort by the faculty and library staff has built and maintains library resources which are more than adequate. The staff uses various resources to select titles. These include *CHOICE* online, subject bibliographies, bibliographic essays, "Best Book" lists, and the approval plan from Yankee Book Peddler.

In 2011 the librarians began using a formula based allocation system based on the departmental degrees and student FTEs. Prior to this, a flat dollar amount per department was allocated. Journals and database are selected based on bibliographic essays, Katz's *Magazines for Libraries* and FSU usage. Usage statistics are kept and they are reported in the Library's annual reports

Monographs/Books-Holdings

Subject Area-Holdings	Classification Area	2005	2008	2009	2010	2011	2012
Computer Science/EDP	QA75-76.95	2,978	N/A	N/A	1,553	1,565	1,519
Mathematics	QA	3,026	N/A	N/A	2,000	1,973	1,751
Machine Theory	QA150-272	604	N/A	N/A	480	475	484
Analysis	QA300-433	467	N/A	N/A	385	384	388
Physics	QC	368	N/A	N/A	1,204	1,218	1,196
Business Apps.	HD30.2	91	N/A	N/A	74	54	54
Computer Industry	HD 9696	142	N/A	N/A	95	95	94
Office Automation	HF5548	375	N/A	N/A	3	3	3
Educational Uses	LB1028	285	N/A	N/A	41	41	43
Operations Research, Information Systems, Systems Analysis	T58	103	N/A	N/A	7	7	4
Data Processing	TA1630-1660	33	N/A	N/A	38	38	25
Telecommunications	TK1501-6720	848	N/A	N/A	771	776	707
Electronics & Computer Engineering	TK7807-7895	704	N/A	N/A	650	652	594
Misc.	TA156 & ZA	4	N/A	N/A			
Print Total		10,028			7,301	7,281	6,862
Electronic total (Approximate)					150	150	150

General Collection

The books are located on the 4th floor of the library; there are approximately 238 shelves of books; each range is seven shelves high and each shelf is being used. Each shelf is 80% full. The books include paperbacks and hard covers. Extensive weeding has been done to replace the dated materials.

Reference Collection
Due to the nature of this discipline there is very little need for in-depth subject and literature analysis. The basic ready reference questions (who, when, what, etc.) are answered by the online reference services *Credo* and *Gale Virtual Reference.* Mathematics has approximately 10-15 reference titles.

E-books collections
The library began a subscription ($5,500/yr.) to *Safari Books Online* in 2007 and funds are annually budgeted to provide access to 165 books covering Web Development, Information Technology & Software Development and Web Design. There are approximately 150 titles available for use (the balance is for Communications/Photography). Usage has been minimal and they have revised their access model so that only one person can use the entire database at a time. The library staff and faculty are presently rethinking this subscription. A trial subscription to *Books 24 X 7* as a possible replacement was undertaken during the spring 2013 semester however it was not favorably received.

In 2013, the library began selecting and purchasing individual e-books from EBSCO and usage will be monitored on an annual basis.

In the spring of 2013 Fitchburg State University was authorized to provide a Bachelor's degree in Game Design in the Communications Department. Additional institutional funds have been assigned in a three year spending plan to establish an opening day collection of books which will be mostly in electronic format. Subjects include programming, game design, graphics, etc.

Comments
- As of July 1, 2012 Computer Science volumes comprised 3% of the collections (6,862/205,759).
- From 2010 to 2012, the staff undertook an extensive analysis of the book collection and based upon the following criteria: copyright date and subject and withdrew approximately 3,000 volumes.
- E-books via *Safari Books online* began in 2007. This database provides titles for Communication Media and Computer Science. The *Safari Books Online* e-books are being searched and there is quite a bit information retrieved but it is concentrated in 30-40 books. A meeting with faculty is recommended to select more appropriate book titles can be made available.

Acquisitions

Subject Area-Acquisitions	Classification Area	2008	2009	2010	2011	2012	2013
Computer Science/EDP	QA75-76.95			28	27	53	
Mathematics	QA (non-CS)			39	18	53	
Machine Theory	QA150-272			19	8	15	
Analysis	QA300-433			9	1	9	
Physics	QC			33	17	33	
Business Apps.	HD30.2			-	-	-	
Computer Industry	HD 9696			-	-	-	
Office Automation	HF5548			-	-	-	
Educational Uses	LB1028			1	1	3	

Subject Area-Acquisitions	Classification Area	2008	2009	2010	2011	2012	2013
Operations Research, Information Systems, Systems Analysis	T58			-	-	-	
Data Processing	TA1630-1660			1	-	1	
Telecommunications	TK1501-6720			-	-	14	
Electronics & Computer Engineering	TK7807-7895			13	4	5	
Misc.	TA156 & ZA						
Totals		N/A	N/A	143	76	186	

Comments:
The 1998 *WLN* (Washington Library Network) *Collection Assessment Manual* recommends a reasonable guideline to consider for assigning an acquisition indicator. It is a purchasing rate of at least 20-25% or more of the appropriate universe of titles published annually in a division. In this case it would be to purchase on an annual basis at least 20-25% of the new Computer Science titles.

1a-2a =less than 5%
2b =5-10%
3a =10-15%
3b =20%
3c =25%
4 =30%

A comparison of the library's holdings was conducted with a random sample of 80 out of 389 Computer Science book titles published between 2008 and 2013 reviewed by ***Choice*** magazine and its results indicated the library had purchased 22 or 27% of the essential, highly recommended and recommended titles for undergraduate and graduate students. ***Choice*** is a publishing unit of the Association of College & Research Libraries, a division of the American Library Association and provides reviews to approximately 3,500 titles appropriate for academic libraries.

The library is purchasing 27% of the recommended titles and the liaison is ordering the electronic version. It is hoped this will improve usage.

Currency of the Collection
This includes the age of the collection and is shown at the mean or mode level. The 1992 *Manual* recommends generally 10% or more of the collection should have a copyright date within the last ten years.

Subject Area-Currency	Classification Area	Mean 2011-12	Median 2011-12	Mode 2011-12
Computer Science/EDP	QA75-76.95	1997	1999	2000
Mathematics	QA (non-CS)	1984	1989	1990
Machine Theory	QA150-272	1983	1990	1990
Analysis	QA300-433	1986	1991	1990
Physics	QC	1982	1984	2000
Business Apps.	HD30.2	1998	1998	1998
Computer Industry	HD 9696	1992	1993	1996

Subject Area-Currency	Classification Area	Mean 2011-12	Median 2011-12	Mode 2011-12
Office Automation	HF5548	2002	2000	
Educational Uses	LB1028	1993	1992	1986
Operations Research, Information Systems, Systems Analysis	T58	1950	1951	
Data Processing	TA1630-1660	1995	1992	1991
Telecommunications	TK1501-6720	1993	1998	
Electronics & Computer Engineering	TK7807-7895	1983	1982	1980
Miscellaneous	TA156			
Miscellaneous	ZA	2004	2007	2009

Circulation and Usage

Subject Area-Circulation	Classification Area	2008	2009	2010	2011	2012	2013*
Computer Science/EDP	QA75-76.95	101	65	85	72	70	54
Mathematics	QA (non-CS)	58	76	70	50	73	47
Machine Theory	QA150-272	15	9	20	48	10	6
Analysis	QA300-433	4	9	7	7	14	8
Physics	QC	65	74	57	64	20	31
Business Apps.	HD30.2	4	5	3	4	6	4
Computer Industry	HD 9696	1	4	3	2	19	2
Office Automation	HF5548	3	1	0	3	0	6
Educational Uses	LB1028	4	17	18	18	24	15
Operations Research, Information Systems, Systems Analysis	T58	1	0	0	0	0	0
Data Processing	TA1630-1660	0	1	4	0	0	2
Telecommunications	TK1501-6720	17	12	11	13	3	18
Electronics & Computer Engineering	TK7807-7895	12	7	9	2	7	3
Misc.	TA156 & ZA	0	5	4	1	1	2
Total print circulation for Computer Sciences		**285**	**285**	**291**	**284**	**247**	**198**
Total Library Circulation		12,110	12,007	13,531	11,131	9,623	
Percentage of CS with total circulation		2.35%	2.37%	2.15%	2.55%	2.57%	
Total-book usage							
***Safari Books online*-Sections viewed**		756	390	3,024	2,356	1,706	1,079
***Safari Books online*-Total e-books viewed**		N/A	N/A	N/A	N/A	N/A	44
* July-May 2013 only							

E-book usage

There were 44 “hits” from the 150 e-book titles from *Safari Books Online* for the broad subject areas Desktop and Web Applications and Information Technology & Software Development in 2012

SERIALS HOLDINGS-Access to journals

Lists of journals in printed and electronic format dealing with History can be located on the Gallucci-Cirio Library web and use the Journal Locator search box. The Gallucci-Cirio Library currently has access to approximately 1,305 journal titles classified as History (all categories). See below

Definitions being used:
1 & 2a = some general periodicals and some major indexes
2b = 30% or more of the titles indexed in *Magazines for Libraries* and access to indexes
3a = 50% of the titles indexed in *Magazines for Libraries* and access to indexes
3b = 75% of the titles indexed in *Magazines for Libraries* and access to indexes and access to non-bibliographic databases
3c = 80%-100% of the titles indexed in *Magazines for Libraries* and access to the major indexing and abstracting services in the field

In assigning a ranking to the quality of the serials collection the *WLN Collection Assessment Manual*, 1995, recommended 75% comparison rating to support a major and 90% to support the Masters Level. The 1997 edition deleted all references to journal holdings. However it did continue to recommend access to the major indexing and abstracting services in the field.

Journals

The following is a subject analysis of the journals presently provided to the faculty and students.

Computer Science - 1,321 titles

- Computer Science – 679 titles
- Electrical & Computer Engineering
    - Electrical Engineering – 295 titles
    - Information Technology - 98 titles
    - Telecommunications – 249 titles

Mathematics - 624 titles

- Applied Mathematics – 70 titles
- Mathematics
    - Algebra – 44 titles
    - Calculus – 20 titles
    - Elementary Mathematics & Arithmetic – 8 titles
    - Geometry – 18 titles
    - Mathematical Statistics – 79 titles
    - Mathematical Theory – 183 titles
    - Mathematics – General – 170 titles
    - Mathematics Teaching & Research – 32 titles

Business & Economics - 5,176 titles

Analysis

Magazine's for Libraries, 2010 edition, is a major source for reviewing and recommendation of a wide variety of journals for the general and academic audience. Published by ProQuest it was begun in 1969 by Prof. Katz from the SUNY Albany School of Library Science. It provides journal recommendations, they are arranged by subjects and FSU uses it for acquisition's decisions. We also use it as a list checker for assessment. We compare our journal holdings with these recommendations. It also recommends

access to the major indexing and abstracting services in the field. The following is a comparison of the library holdings:

In comparing the recommendations Katz's *Magazines for Libraries* the Library provides the following:

- Computer, Control and Systems Engineering: 6 titles out of 9 title recommendations or 69%
- Computer Systems and Forecasting: 9 out of 12 or 33%
- Computer and Information Technology: 27 out of 33 or 82%
- Mathematics: 21 out of 31 or 68%
- Business: 350 out of 418 or 84%

Comments:
The library provides access to 1,321 journal titles in Computer Science, 624 journal titles in Mathematics and 5,176 journal titles in Business and Economics.

There are seven individual library subscriptions; all other journals are in the databases.

Usage: There is minimal retrieval of full-text documents from the databases. There is a sizable amount of sessions and searching taking place but the retrieval of documents is minimal. See Core Databases chart for usage.

Databases

Electronic databases provide online access to a variety of materials. They include but are not limited to: e-books, full-text journals, indexing/abstracting titles with or without some full-text journals, technical reports, standards, etc. Some database a very general in nature and some are specific to Computer Science.

The databases recommended by *Magazines for Libraries* are: *Computer and Control Abstracts (INSPEC), Computer Literature Index, Computing Reviews,* and *Internet & Personal Computing Abstracts*

These are indexing/abstracting services and they allow and provide for online searching of journals.

The library subscribes to *ACM Digital Library* and *Computers and Applied Science Complete* and they provide access to *Computing Reviews* and *Computer Literature Index.*

Core Databases provided by the library

Title	2013 Expen	2010 Sessions	2010 Searches	2010 Docts	2011 Sessions	2011 Search	2011 Docts	2012 Sessions	2012 Searches	2012 Docts
ACM Digital Library (370 subs)	$4,041	450	338	417	275	155	136	131	114	92
Computers & Applied Sciences Complete	$4,994	1,084	4,960	243	1,827	8,481	255	2,287	9,731	269
ProQuest Computer Science Journals		2,111	6,745	171	1,985	6,511	228	1,796	5,693	129
Safari Tech Books	$5,555	2,113	440	3,203	230	249	2,356	220	121	1,706

Supplemental Databases provided by the Library

ABI/INFORM-Dateline
ABI/INFORM-Global
ABI/INFORM-Trade & Industry
Academic One file
Academic Search Premier
Business Source Premier
CREDO Reference
DOAJ
Expanded Academic ASAP
Gale Science in Context
General One File
Highwire Press
Library, Information Science & Technology Abstracts
MathSciNet
ProQuest Central
ProQuest Science Journals
SpringerLink
WorldCat
WorldCat.org

Peer Comparison-Appendix A

A comparison of database holdings from 10 ABET certified institutions' libraries from New England with less than 10,000 students and not offering Doctorates can be seen in the Appendix. The institutions included:

Daniel Webster
Norwich Un.
Roger Williams Un.
Salem State Un.
UMass-Dartmouth
Un. of New Haven
Un. of Southern Maine
Vermont Technical College
Wentworth Institute
Westfield State Un

The list includes approximately 112 unique database titles. Fitchburg State subscribes to approximately 24 of the titles (the vast majority of the databases were titles unique to one library only). There were five databases that were owned by at least 3 or more libraries and Fitchburg did not subscribe to them. They were *Ebrary-Academic Search Complete*, this is the complete e-book collection from EBSCO and we are purchasing individual titles instead of the entire package*; IEEE Xplore* (indexing/full-text journals), 3 libraries, this service is not appropriate for Fitchburg State; *Science Direct* (full-text journals), 6 libraries, funds have been requested for a subscription to begin in 2013; *ENGnetBASE*, (e-books) 3 libraries, we are not sure if this is appropriate for Fitchburg State; and *ProQuest Telecommunications (full-text journals)*, we are not sure if this is appropriate for Fitchburg State.

Comments

- FSU's core database titles are favorably compared to the ABET libraries
- There is minimal retrieval of full-text documents from the databases. There is a sizable amount of sessions and searching taking place but the retrieval of documents is minimal. This could be due to improper searching techniques or the documents may not be appropriate.
- The library instruction/information literacy program is available to the department but no classes have been scheduled

Expenditures

Computer Science-Materials Expenditures Summary										
Year	Journals (Individual subscriptions)	Databases	Standing Orders	Books	e-books	Video	Audio	Total	Discipline based FTE	Spending per FTE
2008	$1,575	$19,848						$21,423	140.75	$152.21
2009	$1,870	$14,346	$304	$2,873				$19,393	118.95	$163.03
2010	$1,939	$8,131	$462	$3,945				$14,477	127.17	$113.84
2011	$2,049	$12,893	$631	$1,829		$311		$17,713	145.28	$121.92
2012	$1,488	$13,495	$482	$5,058				$20,523	144	$142.52
2013*	$2,126	$9,035	$508	$4,716	$5,555			$21,940	114	$192.46
*estimate										

Conclusions

Monograph/Books Comparison

2007 Report

- The monographic level is not sufficient to sustain the Bachelor's CS program, the library owns approximately 8,500 volumes (including Mathematics) and the count should be 12,000+. While there have been decent rate of growth to the collection, the purchasing rate is low. It was recommended in 2006 to investigate the purchase of e-books. This has been done and a subscription to *Safari Books* was begun in April 2007. This will add approximately 150 e-books to the collection.
- Even though book circulation has declined in all subject areas, additional titles are needed to reach a reasonable goal of 12,000 volumes. We need to purchase additional titles for computer applications in business. They need to be added on a consistent and high level when older titles are deleted they need to be replaced.

2013 Report

- Extensive weeding has been done (approximately 3,000volumes) to remove dated materials and remove books that no longer deal with the course offerings.
- Print book usage is very low averaging 278 volumes per year or 2.4% of the book circulation
- The *Safari Books Online* e-books are being searched and there is quite a bit information retrieved but it is concentrated in 30-40 books. A meeting with faculty is recommended to select more appropriate book titles to be made available.
- The library is purchasing 27% of the recommended titles and the liaison is ordering the electronic version. It is hoped this will improve usage.
- In the spring of 2013 Fitchburg State University was authorized to provide a Bachelor's degree in Game Design in the Communications Department. Additional institutional funds have been assigned in a three year spending plan to establish an opening day collection of books which will be mostly in electronic format. Subjects include programming, game design, graphics, etc.

Journal Comparison

2007 Report

- The library provided access to 410 Computer journals and 3,000 titles dealing with Business.
- The library subscribed to *The ACM Digital library package, ACM Proceedings, the ACM Portal* and *Science Direct* which provided 27 full-text computer science journals.
- Results from the 2007 report:
    - After the staff conducted a peer analysis of databases, four meetings were held with a representative of the Computer Science department and it was decided to add *Computer Source* and *Computer Science Index.* These were later merged to create *Computers and Applied Science Complete* and this subscription began in 2008. This added 500 full-text journals
    - A subscription to *ProQuest Computer Science Journals* began in 2007. This added 350 full-text journals

2013 Report

- The library provides access to 1,321 journal titles in Computer Science, 624 journal titles in Mathematics and 5,176 journal titles in Business and Economics.
- The library continues to subscribe to *Computers and Applied Science Complete* and *ProQuest Computer Science Journals*. The ProQuest Central was added during the spring of 2013.
- Funds have been requested to purchase a subscription to *Science Direct-Physical Sciences* (full-text journals) for July 2014. This will enhance the journal collections for Chemistry, Computer Science, Earth/Geophysical Sciences, Mathematics, and Physics.

Database Comparison

2007 Report

- The library provided access to the following: *ACM Digital Library, BusManagement, and Business Source Premier, Lexis-Nexis Academic Universe,* and Dialog. *Computer Science Index* was cancelled and replaced with in *Computers and Applied Sciences Complete* in Dec. 2008. *Computer Source*, was cancelled and replaced with in *Computers and Applied Sciences Complete* in Dec. 2008
- Discussions were held to see if it was necessary to provide reports, standards, papers, and e-books. I t was decided they weren't necessary.

2013 Report

- The two business databases were cancelled, *ProQuest Central* has been added and this provides more appropriate business sources (journals and reports).
- *Lexis-Nexis* was not cancelled; it was just removed from the Computer Science database options as not being appropriate.
- *Safari Books Online* was added but it is getting minimal use,

6/10/13

Appendix A (of Review of Library Materials)
ABET Peer Database Comparison

Computer Science Databases-Comparison with ABET Institutions	**Database Type**	**Totals**	**FSU Holdings**
A Dictionary of Computing	Reference	1	
ABI/INFORM-Dateline	Reference	1	X
ABI/INFORM-Global	Reference	2	X
ABI/INFORM-Trade & Industry	Reference	1	X
Abstracts in New Technologies and Engineering		1	
Academic One file	Aggregator	2	X
Academic Search Complete	Aggregator	2	
Academic Search Premier	Aggregator	3	X
Academic Video Online		1	
Access Science	Reference	1	
ACM Digital Library	Standards	6	X
American National Standards Institute (ANSI)	Standards	1	
Annual Reviews	Full-text journals	1	
Applied Science & Technology Full Text	Aggregator	2	
Applied Science & Technology Index	Aggregator	1	
BABEL: A Glossary of Computer Oriented Abbreviations and Acronyms	Reference	1	
Business Source Premier	Aggregator	1	X
Caltech Computer Science Technical Reports	Tech. Reports	1	
Cambridge Un Press e-books	e-books	1	
CogPrints Electric Archive	Tech Reports	1	
Collection of Computer Science Bibliographies	Tech. Reports	1	
Computer Database	Aggregator	2	
Computer Science and Artificial Intelligence Lab (CSAIL)	Tech. Reports	1	
Computer science Bibliographies	Reference	1	
Computer Science Index	Aggregator	1	
Computers and Applied Sciences Complete	Aggregator	1	X
Computing Research Repository (CoRR)	Tech. Reports	1	
CRCnetBASE	e-books	1	
CREDO Reference	Reference	0	X
Current Research information System (CRIS)		1	
CYBER Bullying Searchable Info. Center		1	
Dictionary of Algorithms and Data Structures	Reference	1	

Computer Science Databases-Comparison with ABET Institutions	Database Type	Totals	FSU Holdings
Digital Commons Network	Reference	1	
Dissertations & Thesis	Aggregator	1	
DOAJ	Aggregator	2	X
Ebook Library	e-books	1	
ebrary	e-books	2	
ebrary-Academic Complete/Collection	e-books	3	
EBSCO e-book Collection	e-books	1	
EBSCO Science Sources	e-books	1	
EECS Technical Reports	Tech. Reports	1	
Encyclopedia of Computer Science	Reference	1	
Encyclopedia of Computer Science and Technology	Reference	1	
Encyclopedia of Data Warehousing & Mining	Reference	1	
Encyclopedia of Mobile Computing and Commerce	Reference	1	
Encyclopedia of Portal Technologies and Applications	Reference	1	
Encyclopedia of Technology and Innovation Management	Reference	1	
Engineering Village	Aggregator	2	
ENGnetBASE	e-books	3	
E-Print Network		1	
Expanded Academic ASAP	Aggregator	1	X
Films on Demand	Films	2	
Find White Papers Research Library	Tech. Reports	1	
FOLDOC: Free On-line Dictionary of Computing	Reference	1	
Gale Science in Context	Aggregator	1	X
General One File	Aggregator	1	X
General Science Abstracts	Aggregator	1	
Google Scholar		2	
Highwire Press	Aggregator	2	X
Homeland Security Digital Library		1	
HP Labs: Technical Reports	Tech. Reports	1	
IBM Research: Technical Paper Search	Tech. Reports	1	
IEEE Standards	Standards	1	
IEEE Xplore/IEEE Compute Society Digital Lib		3	
Information Science & Technology Collections		1	
Ingenta Connect		1	
INSPEC	Aggregator	1	
INSPEC Archive	Aggregator	1	
International Organization for Standardization (ISO)	Standards	1	

Computer Science Databases-Comparison with ABET Institutions	Database Type	Totals	FSU Holdings
Internet Corporation for Assigned Names and Numbers (ICANN)	Standards	1	
Journal Citation Report	Aggregator	1	
Library, Information Science & Technology Abstracts	Aggregator	1	X
Mathematical Programming Glossary	Reference	1	
MathSciNet	Aggregator	3	X
National Information Standards Organization (NISO)	Standards	2	
National Institute of Standards and Technology (NIST)	Standards	1	
National Technical Information Service	Tech. Reports	2	
Nature	journal	1	
NetLingo	Reference	1	
Network Protocol Suite Directory and Index	Standards	1	
NSSN: A National Resource for Global Standards	Standards	1	
Open Science Directory		1	
Oxford Journals Online	Aggregator	1	
Oxford Scholarship Online		1	
ProQuest Central	Aggregator	3	X
ProQuest Computer Science Articles/Collection	Aggregator	2	
ProQuest Computing Journals	Aggregator	5	X
ProQuest Science Journals	Aggregator	3	X
ProQuest Telecommunications	Aggregator	3	
Safari Books Online	e-books	3	X
Science Citation Index, Expanded	Aggregator	1	
Science Direct	Full-text journals	6	
Science Reference Center	Reference	1	
Science.gov		2	
ScienceResearch. Com		1	
Scirus		2	X
Scitopia		1	
SpringerLink	Aggregator	1	X
TechRepublic: White Papers	Tech. Reports	1	
Techstreet		1	
Tech-White-Papers	Tech. Reports	1	
Web of Knowledge	Aggregator	1	
Web of Science	Aggregator	2	
Webopedia: Online Computer Dictionary for Computer and Internet Terms and Definitions	Reference	1	
Whatis	Reference	1	

Computer Science Databases-Comparison with ABET Institutions	Database Type	Totals	FSU Holdings
Wiley Encyclopedia of Electrical and Electronics Engineering	Reference	1	
Wiley Interscience	Full-text journals	1	
Wiley Online Library	Full-text journals	2	
World Wide Web Consortium (W3C)	Standards	1	
WorldCat	Reference	1	X
WorldCat.org	Reference	1	X

Appendix G – Interim Report (Submitted on July 2009)

INTRODUCTION

This is a report by the Computer Science Department at Fitchburg State College which responds to the weaknesses and concerns identified in ABET's final statement following its on-site evaluation of our computer science (CS) and computer information systems (CIS) programs. The site visit took place during the 2007-2008 academic year and the final statement was issued in August of 2008. The actions and explanations provided below are the product of many hours of consideration and deliberation by our faculty and administration. We believe that we have addressed the weaknesses and concerns identified in the final statement, and have accordingly improved the educational experience of our students.

The ABET final report identified weaknesses in the Objectives and Assessments category (Standards I-3 and I-4.) A description of our efforts to correct these weaknesses will constitute the first section of the report. We believe these weaknesses have been eliminated and replaced by a robust assessment program which provides to the Department multiple external and internal sources of information about what our students know and can do. We are particularly proud of a new, on-line system which aligns assessment measures based directly on student performance with our stated program outcomes. This system is implemented using the TK20 on-line assessment management system provided by the Office of Academic Affairs at Fitchburg State College. The Computer Science Department Chair, Prof. Frits Lander, and the College's recently hired Director of Assessment, Dr. Stephen Wall-Smith, made a presentation about our positive experiences with this system at the April 2009 ABET Best Assessment Practices Symposium in Indianapolis, a year after attending the same conference in Atlanta along with faculty member Dr. Xuzhou Chen.

The ABET final statement identified concerns in Faculty, Curriculum, and Program Delivery categories. Concerns were expressed relating to Standards IV-7, IV-15, IV-16, and IV-17. These will be addressed in the second section of the report which follows.

We are grateful for the extensive review our program has received from the ABET committee and appreciate the opportunity to improve our program so that it meets requirements of each Standard. We are certain that our responses to the review provided in the ABET final statement have already resulted in an improved educational experience for our students. We believe that you will find our efforts to correct the identified concerns and weaknesses and meet the ABET Standards to be satisfactory in every way.

SECTION I: WEAKNESSES

The following weaknesses were identified in ABET's final statement:

- *Standard I-3 ("Most of the data gathered for assessment purposes are primarily the opinions of the persons surveyed and does not give conclusive evidence of the attainment of objectives.")*
- *Standard I-4 ("There is limited evaluation of the extent to which the program objectives are being met.")*

Specifically, the final statement requested that the Computer Science interim report "include the following to address the weaknesses:

1. Identify the data gathered for assessment purposes. In particular, indicate how the data collected is more than just personal opinions.
2. Identify the methods being used to evaluate whether program objectives are being met, and also identify the extent to which the program objectives are currently being met."

The assessment plan under which the Computer Science (CS) and Computer Information Systems (CIS) programs presently operate now includes six formal data sources. Three complementary and objective data sources provide an external perspective on our programs and the students in them. Three internal sources tell us how our students are doing with respect to our program outcomes and how they rate what they've been through as they finish. Only one data source involves self-reports of student learning. Data and methods detailed in the remainder of this section.

The external data sources mentioned in the preceding paragraph are: intern evaluations; a revivified Industrial Advisory Board, which meets annually; and disaggregated results from the bi-annual Alumni Survey, last administered in the Fall of 2008.

Internal data sources include course-embedded assessments tied to program outcomes; course surveys, which ask students to reflect on the extent of their learning around stated course objectives (and which are also aligned with departmental program outcomes); and results from the Graduating Student Survey, an extensive exit survey offered to all students who apply to receive degrees from the College.

All of these sources indicate that our students meet the stated objectives of the CS program and that our programs are successful in preparing graduates for the workplace and for graduate study. All of them also highlight areas where we can improve curriculum and strengthen the assessment process itself, and we have already implemented changes in curriculum based on information derived from our data. The data we have gathered and processed since receiving ABET's final statement, as well as the ways in which we have used the data to improve instruction and departmental processes, are discussed in detail below.

Intern evaluations. Though undergraduate internships are relatively difficult to arrange, we have placed a renewed emphasis on internships since the ABET visit. Prof. Lander is now the departmental point of contact for student interns. Results from "Employer's Evaluation of

Intern Student" forms received by the Department are presented below in Table 1. Results indicate the FSC students are successful in workplace environments.

Table 1
Employer's Evaluation of Student Interns
Computer Science Program
Fall 2005-Fall 2007

	5	4	3	2	1	Mean
Relations with others						
5 = Exceptionally well accepted						
4 = Works well with others						
3 = Gets along satisfactorily						
2 = Has some difficulty working with others						
1 = Works poorly with others	2					5.0
Attitude and application to work						
5 = Outstanding in enthusiasm						
4 = Very industrious						
3 = Average diligence and interest						
2 = Somewhat indifferent						
1 = Definitely not interested	1	1				4.5
Judgment						
5 = Exceptionally mature						
4 = Above average						
3 = Usually makes right decisions						
2 = Often uses poor judgment						
1 = Consistently poor		2				4.0
Dependability						
5 = Completely dependable						
4 = Above average						
3 = Usually dependable						
2 = Sometimes neglectful						
1 = Unreliable	2					5.0
Ability to learn						
5 = Learns very quickly						
4 = Learns readily						
3 = Average in learning						
2 = Rather slow to learn						
1 = Very slow to learn	2					5.0
Quality of work						
5 = Excellent						
4 = Very good						
3 = Average						
2 = Below average						
1 = Very poor	1	1				4.5

Table 1 – continued	5	4	3	2	1	Mean
Attendance 5 = Regular 4 = Irregular	2					5.0
Punctuality 5 =-Regular 4 = Irregular	2					5.0
Overall performance 5 = Outstanding 4 = Very good 3 = Average 2 = Marginal 1 = Unsatisfactory		2				4.0

Industrial Advisory Board. A joint CS/CIS Industrial Advisory Board was convened on April 24, 2009, with 16 members present. Attendees included the College President, Dr. Robert Antonucci, as well as the Vice President and Associate Vice President for Academic Affairs, the Director of Assessment, Computer Science Department faculty, and six well-placed representatives from area employers. Most were graduates of either the undergraduate or graduate CS or CIS programs at FSC, and were invited for their frankness, knowledge, and ability to discern and articulate the present and future needs of industry for FSC's CS and CIS graduates. The minutes of the IAB meeting are included in Appendix A of this report.

IAB members indicated that all CS and CIS graduates they hire – not only those from FSC – tend to need additional experience with algorithm design skills, UNIX programming, and management skills, especially project management. They noted that having made contributions to extensive, real-world projects makes graduates much more competitive in the job market. They also emphasized generally desirable workplace skills for new graduates, including communication skills, flexibility, work ethic, integrity, and problem-solving skills and experience.

As a first step toward addressing recommendations from the IAB, several courses (CSC 2400 Database Systems, CSC 3710 Systems Analysis Methods, CSC 4400 Software Engineering, and CSC 4700 Systems Design and Implementation) have added or reemphasized project management. Workplace skills are modeled in all classes, and explicitly addressed in orientation to internships.

Alumni Survey Results. In the Fall of 2008, a survey of Fitchburg State College graduates was advertised by the College's Alumni and Development Office and posted on the web at the College's Select Survey™ site. Graduates were solicited with notices in the emailed Alumni Affairs newsletter and in the hard copy alumni magazine, *Contact*. Six undergraduate CS/CIS majors responded to the survey. (The survey asked only the department from which respondents received degrees and did not distinguish among programs. All respondents are therefore referred to as "Computer Science" majors in this section.) Four received their degrees in 2003 or earlier. Two had graduated within the previous five years. Although the number of Computer Science respondents is small and may not be representative, the available data

suggests that graduates of CS or CIS programs are at least as likely to be satisfied, employed, and successful in work and post-graduate education as other students from the College.

By design, the survey was short. Most responses were close-ended, that is, respondents had to select their answers from a drop-down menu or set of radio buttons. About two thirds of the items dealt with issues pertinent to alumni affairs, e.g., satisfaction with *Contact* (very high) and interest in a variety of proposed activities. The rest of the items dealt broadly with questions typically asked about alumni by regional and program accreditors: How satisfied are students with their educational experiences, how employable are they when they graduate, how successful are they in their careers, and how likely are they to go on for graduate education? Answers to these questions, based on this survey, are the summarized below.

Alumni satisfaction. Item 16 on the survey asked "At this point, how satisfied are you with the education you received at Fitchburg State College?" Respondents were offered a drop-down Likert scale that ranged from "Very Dissatisfied" (value = 1) through "Very Satisfied" (value = 5). The mean for Computer Science alumni was 4.50 for recent graduates and 4.00 for earlier graduates. These numbers compare favorably to satisfaction among all alumni respondents.

Alumni employability. Item 9 on the survey asked, "How soon after graduation were you able to gain employment in your field of study?" Typically, about 60% of graduates either had jobs on graduation or found them within six months. Among Computer Science graduates, 59% of those graduating before 2004 had offers upon graduation or within six months, and 100% of more recent graduates obtained employment immediately or within six months.

Career Success: Salary. Item 12 posed the income question broadly: "If you are currently working for pay, how would you assess your present compensation?" Respondents were offered a drop-down five-point Likert scale, and were also allowed to indicate that they were not working for pay or were retired. Choices for those currently working for pay were "Much better than average for my field," "Somewhat better than average for my field," "Typical for the field in which I work," "Somewhat below average for my field," and "Well below average for my field." Item 12 did not distinguish between full- or part-time employment, or employment in the major field rather than some other. College-wide, about 60% of respondents assess their income as being typical for their field or below average to some degree, while 40% believe they are paid better than is typical for their field. Computer Science respondents who graduated in 2003 or earlier follow exactly that distribution, while 100% of more recent graduates believe that their pay is typical.

Career Success: Responsibility. Item 10 asks, "Which of the following best describes your present level of responsibility?" Bulleted explanations for some of the choices follow:

- "Entry-level professional" includes beginning artists and entrepreneurs as well as technicians, nurses, teachers, managers, etc., in the early stages of their careers.
- "Advanced professional" includes well-established artists and entrepreneurs as well as clinical specialists, teacher leaders, etc.
- "Management-level" includes artists and entrepreneurs who develop large-scale operational concepts and supervise others in the execution of work, as well building principals in education, head nurses, etc.

Other options include “Experienced professional, artist, or entrepreneur;” “Senior executive (e.g., President, CEO;”” Retired;” and “Other.”

Both recent and more experienced Computer Science graduates listed various levels of responsibility, from entry level to advanced professional (40% of pre-2004 graduates, and 50% of more recent graduates). One fifth of earlier graduates and one quarter of recent graduates identified their level as “management.”

Graduate Education. College wide, 44% of undergraduate respondents had never been to graduate school, while 56% had at least some graduate education. Of Computer Science graduates, 57% had no graduate school while 43% had at least some.

Course-embedded assessment. The arrival on campus of a full-time Director of Assessment has allowed, among other things, College-wide deployment of an e-portfolio and assessment-management system called Tk20. This system permits results of performance-based, rubric-rated classroom assignments to be reported around the eleven learning outcomes of the CS program and eight learning outcomes of the CIS program. Rubrics utilized are three-point scales with “3” indicating exceptional proficiency, “2” indicating the basic proficiency expected of an entry-level professional, and “1” indicating deficient performance. As a faculty, we are still identifying and/or creating assessment opportunities in our classes and we have not completely met our goal of at least one significant assessment per outcome in each division (lower division = 1000- and 2000-level courses, upper division = 3000- and 4000-level courses). Still, we have made significant progress in this area. Results tabulated below indicate that we are directly assessing half or more of our outcomes, and that students are generally performing at or above expectations for young professionals.

Note that student counts are (at least potentially) duplicated. That is, the same student may be counted twice for assessments of two standards in one class, or multiple times for several standards across several courses. Computer Science is relatively small program, with 46 majors as of Fall 2008 and 8 BS degrees awarded in academic year 2008-2009.

Courses which contributed one or more assignments to Table 2 in 2008-2009 include CSC 1550 (Computer Science II), CSC 2600 (Computer Organization), CSC 3200 (Programming Languages), CSC 3400 (Data Communications and Networking), CSC 3450 (Local Area Networks), CSC 3600 (Microprocessors), CSC 3700 (Algorithms and Data Structures), and CSC 4400 (Software Engineering).

Table 2
Course Embedded Assessment Summary by Departmental Outcomes
Computer Science Program
Fall 2008-Spring 2009

	Fall 2007 – Spring 2008				Fall 2008 – Spring 2009			
Outcome	**N All Students**	**N All Classes**	**N All Assignments**	**Mean All Students**	**N All Students**	**N All Classes**	**N All Assignments**	**Mean All Students**
CSSO-1 (Mathematics)	7	1	1	1.83				
CSSO-2 (Logic)	84	6	6	1.98	72	5	8	2.16
CSSO-3 (Circuits)	6	1	1	2.50	7	2	2	2.43
CSSO-4 (Interfaces)	6	1	1	2.50	34	3	3	2.06
CSSO-5 (Programming)	81	3	4	1.93	125	8	22	2.19
CSSO-6 (Operating systems)	11	2	2	2.45	8	1	1	2.56
CSSO-7 (Writing/speaking)	22	3	3	2.57	48	5	13	2.37
CSSO-8 (Software development)	8	1	1	2.75	67	3	5	2.29
CSSO-9 (Teamwork)	14	2	2	2.64	11	2	7	2.53
CSSO-10 (Ethics)	18	1	1	2.33	6	1	1	2.84
CSSO-11 (Advanced topics)	8	1	1	2.43	6	1	2	2.42

3-point scale: "Does not meet standards" = 1; "Meets standards" = 2; "Exceeds standards" = 3

While no classroom-based assessments provided data for CSSO-1 in 2008-2009, it should be noted that this particular standard is largely met by satisfactory completion of pre- and co-requisite courses in mathematics and physics. All program completers met or exceeded the mathematics and science requirements of the Computer Science program.

In the span of a single academic year, we increased the number of student assessments from 265 to 384 (44.9%); increased number of courses with standards-based assessments from 6 to 8 (33.3%); and increased the number of assignments assessed from 23 to 64 (178.3%). The mean for all standards assessed in 2008-2009 was at least 2.00.

Course surveys. In past years, course surveys were a key source of information for instructors about the extent to which they succeeded in getting course content across to students. A typical course survey contains a detailed table of course content objectives. Students are asked to evaluate, first, the extent to which they believe the content was taught, and then the extent of their sense of mastery. The scale ranges from 1 (was not taught or was not learned) to 5 (was covered completely or learned completely). Historically, there is a one-half- to one-point difference between the taught and learned ratings, with the latter lower than the former. This

indicates that students recognize there is more they might learn concerning the topics covered during a semester. Occasionally, students have a very different perception of how extensively a topic was treated than the instructor intended. When that occurs, it provides a basis for changing instruction.

With the advent of Tk20, it became possible not only to administer course surveys online, but to align each item on each survey with one or more program outcomes. Reporting survey results can be almost instantaneous.

We have also learned by hard experience that students are jaded to online surveys and their participation has to be monitored in order for it to be certain. This capability exists in Tk20, which lists participants and non-participants separately from their responses, so that responses are all anonymous. However, we did not utilize this capability during the two semesters of online survey administration, and the response rate and depth of available data suffered for it. The return rate for all courses was low (mean = 23.1%, range = 0.00% - 83.3%). In every area where data exists, however, students are generally confident that they have mastered the topics covered in their course work.

Table 3
Course Evaluation Summary by Departmental Outcomes
Computer Science Department
Fall 2008 – Spring 2009

	Fall 2008				Spring 2009			
Outcome	**N All Students**	**N All Classes**	**N All Assignments**	**Mean All Students**	**N All Students**	**N All Classes**	**N All Assignments**	**Mean All Students**
CSSO-1 (Mathematics)	18	1	1	4.50	4	2	2	3.50
CSSO-2 (Logic)	21	2	2	4.52	9	2	2	4.25
CSSO-3 (Circuits)	18	1	1	4.55	5	1	1	4.32
CSSO-4 (Interfaces)	9	1	1	4.67	11	3	3	4.41
CSSO-5 (Programming)	91	3	3	3.65	12	4	4	4.43
CSSO-6 (Operating systems)					8	2	2	4.71
CSSO-7 (Writing/speaking)								
CSSO-8 (Software development)					3	2	2	4.50
CSSO-9 (Teamwork)								
CSSO-10 (Ethics)	1	1	1	4.00	1	1	1	5.00
CSSO-11 (Advanced topics)								

5-point scale from 1 = "Totally Disagree That I Learned The Material" to 5 = "Totally Agree That I Learned the Material."

Graduating Student Survey. This is a new instrument administered by the Office of Assessment in cooperation with the College Registrar. Departmental results are reported annually for students. Students are identified by major department, hence there is no distinction between CS and CIS program completers.

The Graduating Student Survey is an extremely detailed exit survey offered to every degree applicant at Fitchburg State College. Slightly more than 20% have so far availed themselves of the opportunity, though not every student responds to every question. The survey encompasses more than 125 items covering general education, satisfaction with student services, demographic characteristics, and experiences in the major field. Approximately 130 undergraduates completed all or part of the Graduating Student Survey. Five undergraduate Computer Science majors were among them. Information about experiences in the major and overall satisfaction with FSC are included in the tables below.

Some of the means are flattering and some are lower than we might like, but the small number of respondents makes it difficult to draw conclusions in the absence of trend data. We will continue to monitor results from this survey annually, and encourage graduating students to complete it.

Table 4
Undergraduate Program Completers' Perceptions of Major Coursework
FSC Graduating Student Survey
Summer 2008-Spring 2009

	FSC Undergraduates			Computer Science		
	N	Mean	SD	N	Mean	SD
Availability of classes.	135	3.21	1.07	4	3.25	1.26
Frequency of course offerings.	135	2.97	1.06	4	3.00	1.41
Size of classes.*	135	3.86	0.99	4	4.50	0.58
Overall quality of instruction.	135	3.70	0.92	4	3.50	1.29
Overall quality of texts and other instructional materials.	134	3.27	0.94	4	3.75	1.26
Availability of faculty outside of class time.	135	3.75	0.99	4	3.75	1.50
Timeliness and relevance of course content.	134	3.62	0.88	4	4.00	1.15
Helpfulness of non-teaching staff in your major department.	126	3.64	1.13	3	4.00	1.73
Quality of facilities for learning (e.g., classrooms, labs, studios, gyms).	132	3.48	0.99	4	4.50	1.00
Quality of extra-curricular experiences (e.g., practica, internships)	110	3.52	1.11	2	4.50	0.71
Match between career goals and course-of-study requirements.	132	3.45	1.00	4	2.50	1.91
Overall level of challenge.	133	3.62	0.93	4	3.50	1.29

Table 5
Undergraduate Program Completers' Summary of Satisfaction
FSC Graduating Student Survey
Summer 2008-Spring 2009

	N	Mean	SD	N	Mean	SD
Overall satisfaction with my experience at FSC.	61	3.85	0.87	4	3.67	1.41
Likelihood that I will recommend the College to others.	62	3.87	1.01	4	3.00	1.73

In the short time since our on-site visit in Fall 2007, the assessment plan for the Computer Science program has been expanded significantly to incorporate additional sources of internal and external data. The next steps for the Department are to expand the data collection, insure that all objectives are being assessed, and analyze the data obtained to make appropriate changes to the program.

We believe that the information presented above reflects a strong commitment to assessment which specifically addresses our program objectives and which is based on much more than personal opinion.

SECTION II: CONCERNS

In addition to the two weaknesses identified in ABET's final statement, concerns related to four standards were identified. They are addressed in this section.

- *Standard IV-7. The course display materials examined at the time of the visit showed limited development of problem analysis and program design within the introductory core courses. (CS program)*

We acknowledge that the course display materials presented at the site visit were inconsistent. We did not fully appreciate how these materials would be used in the ABET evaluation process. As mentioned above (in response to Standard I-4), we are now using an online course management system to establish consistency in our course displays in the future.

Problem analysis and program design has always been a consistent and important aspect of our program. Although the approach may vary with instructor, the goal of introducing problem analysis and program design is considered important. Through discussion and reexamination of our activities it has become clear that the time we had originally ascribed to problem analysis and design in the introductory courses was underestimated. Table 6, below describes how problem analysis and design are integrated into each course at the introductory level in the computer science curriculum.

Problem analysis methods are also presented in detail in the introductory hardware classes (CSC 1600 and CSC 1650). Although these are not programming classes, per se, they do require students to deconstruct larger problems into simple, easily solvable, units. The solutions to the simpler problems are combined to form a complete solution. This process of problem breakdown and iterative reconstruction models and reinforces the problem-solving processes that students implement when developing computer programs.

Table 6
Problem Analysis and Design Activities

Courses	Problem Analysis and Design Activity
Computer Science I (CSC 1500)	All stages of problem analysis and program design are explained through code examples worked out in the textbook and in class. Each assignment requires students to analyze a given problem, design and plan an implementation, write code and perform tests to verify the code. As students learn syntax and the fundamental programming constructs in this course, the assignments necessarily result in similar designs: input data -> process input -> generate output.
Computer Science II (CSC 1550)	In addition to the analysis and design activities described above, these assignments also require students to develop software requirements, interpret class diagrams and implement event-driven programs. In these assignments, students work with objects in a more sophisticated way that requires them to demonstrate understanding of polymorphism, inheritance and other aspects of object-oriented programming. Data processing implementations often require

Table 6 – continued	
Courses	**Problem Analysis and Design Activity**
	students to implement multiple sequential steps and use a variety of data structures.
Systems Programming (CSC 2560)	Assignments require students to break down problems into realizable sub-problems using structured decomposition. Top-down and bottom-up design are introduced and practiced. Students must implement various fundamental data structures and functions to initialize and manipulate them using a procedural language. Some assignments demonstrate how the choice of data structure may simplify the design of a solution to a programming problem. Problem solving using recursion and the design of recursive functions are also introduced.

- *Standard IV-15 ("Although at present all students in the program appear to have an adequate written communications experience, the opportunity for all students to acquire written communication skills may be limited in the future.")*
- *Standard IV-16 ("Although at present all students in the program appear to have an adequate oral communications experience, the opportunity for all students to acquire oral communication skills may be limited in the future.")*

All students in the CS and CIS programs (along with all students in the College) are required to take Writing I (ENGL1100) and Writing II (ENGL1200). All students must also complete a Speaking and Listening requirement which is defined by each program. The Computer Science Department has chosen to require that all students take Introduction to Speech Communication (SPCH1000). This requirement was not listed in the College Catalog, but is included in the Web4 degree audit system (please see Appendix B). General education distribution requirements also require coursework in liberal arts and social sciences, which invariably require students to demonstrate written and oral communication skills.

In hardware lab courses (CSC 1600, CSC 1650, CSC 2600 and CSC 3600) appropriate and precise oral communication is of primary importance. Students have always been regularly required to appropriately describe lab issues as they arise and explain their intermediate and final lab results to the instructor. This experience, teaches students how to express themselves appropriately in a technical domain that requires precision of language. To successfully complete lab assignments students must explain technical issues, formulate technical questions and provide appropriate responses to the instructor. In addition, students taking CSC1600 and CSC1650 are periodically asked to present the solutions to various homework and in-class problems at the whiteboard. In the process, students practice orally presenting a problem and describing a solution.

Furthermore, although it may not have been apparent from our course displays, writing and oral presentation have always been a consistent part of our program's capstone experiences (CSC 3710 and CSC 4700). In these courses, students are presented with a real-world problem that they must analyze and develop a solution for. Both proposal and implementation phases require students to write and orally present their findings. CSC 1500, CSC 1550, CSC 1900 and

CSC 2560 (marked with the dagger symbol† in Table 7, below) also have written and/or oral communication requirements, and are courses taken by both CS and CIS majors).

Table 7
Oral and Written Communication Skills in Required CS Courses

Course	Communication		Description
	Type	***Style***	
Computer Science I (CSC1550)†	Written	Formal	**Students write short paragraphs demonstrating their understanding of ethical principles and how they apply to the field of computer science.***
Computer Science II (CSC1550) †	Written	Formal	Students keep a development diary that documents the difficulties encountered during programming assignments.
Introduction to Electronics (CSC1600)	Oral	Informal	Students regularly report on laboratory progress and difficulties. Students are orally quizzed to provide feedback on their understanding of the laboratory exercise.
Digital Electronics (CSC1650)	Oral	Informal	Students regularly report on laboratory progress and difficulties. Students are orally quizzed to provide feedback on their understanding of the laboratory exercise.
Discrete Mathematics (CSC1900) †	Oral	Formal	Students present a proof associated with a discrete math topic that was not presented in class.
Systems Programming (CSC2560) †	Oral	Formal	Students make a presentation on an aspect of the UNIX operating system.
	Written	Formal	**Students write a short paper on open source software and intellectual property rights.***
Computer Organization (CSC2600)	Written	Formal	A report describing the final project is required.
	Oral	Informal	Students regularly report on laboratory progress and difficulties.
Microprocessors (CSC3600)	Written	Formal	A report describing the final project is required.
	Oral	Informal	Students regularly report on laboratory progress and difficulties.
Algorithms and Data Structures (CSC3700)	Oral	Formal	Students present an algorithm that is not presented in class.
	Written	Formal	**Students write a paper describing an aspect of the ACM ethical guidelines with respect to a newsworthy current event.***
Software Engineering (CSC4400)	Oral	Formal	Students make four presentations in this class: use case scenarios, requirements analysis, database design and user interface design.

	Written*	Formal	**Students are required to write an essay describing realistic situations that conflict with ACM ethical guidelines*** and submit a requirements analysis document for their project.
Systems Analysis Methods (CSC3710)	Oral	Formal	Students are required to give five presentations in this class to demonstrate progress in their systems analysis projects.
	Written	Formal	Each of the five presentations must be accompanied by written reports. Student's required written products include: mission statements, project descriptions, system requests, feasibility studies, work plans, risk assessments, requirements definitions, interview questionnaires, use case analyses and progress reports.
Systems Design and Implementation (CSC4700)	Oral	Formal	Students are required to give five presentations in this class to demonstrate progress in their systems design and implementation projects.
	Written	Formal	Each of the five presentations must be accompanied by written reports. Student's required written products include: database design documents, progress reports, user interface design documents and testing plans.

**** indicates assignment that relates to coverage of ethical and social issues***
† indicates courses that are required of both CS and CIS majors

- *Standard IV-17 ("The limited coverage of social and ethical implications of computing may impact the ability of students to gain an understanding of the broad range of issues in this area.")*

Our course displays did not adequately demonstrate the extent to which the global, economic, social and ethical implications of computing are actually implemented in the CIS curriculum. For example, the required textbook for Introduction to Computer Information Systems (CSC1400) has an entire chapter devoted to the social and ethical aspects of computing. This chapter is required reading and three hours of lecture are devoted to the material. In CSC 1500 and CSC2560 students are required to complete assignments specific to the social and ethical implications of computing (see Table 6).

Table 7, above, indicates that students must engage ethical issues in computing at multiple points in both the CS and CIS degree programs. Dr. Mahadev, who has developed at least one major assignment recorded as a performance assessment tracked in the Tk20 system, has been appointed as ethics coordinator. In this role, he will assist other faculty in developing such assessments throughout the curriculum and to monitor the breadth of global, economic, and social topics treated.

It should also be noted that all students at Fitchburg State College have distribution requirements which direct that at students take at least two courses which have been designated as fulfilling the global diversity requirement. One of these must be non-Western.

Faculty members also are developing ways to get students to give students global experiences within the Department curriculum. Dr. Taylor and Dr. Chen recently taught two college level courses (CSC 1010, Computer Science Basics, and CSC 1001, Technology and Business in China) in Hangzhou, China as a summer study abroad program. Eight FSC students, including one major in the Department, took part in this study-abroad program. Students lived on-campus at Zhejiang Gongshang University outside Shanghai for three weeks, devoting mornings to their courses and afternoons to tutoring Chinese students in English and visiting cultural sites in Shanghai and Beijing.

CONCLUSION

We believe that this interim report captures and reports strong evidence that the weaknesses in FSC's CS program identified in the final statement last year have been thoroughly corrected. We also believe we have shown that identified concerns have been addressed. We therefore look forward to full accreditation of the program and to a long and beneficial partnership with ABET.

APPENDIX A (of Interim Report, 2009)

Minutes of Industrial Advisory Board Meeting April 24, 2009.

Present: R. Antonucci (for welcome), F. Archambeault, K. Austin, K. Berry, M. Fiorentino, B. Gauvin, A. Gelinas, N. Kurtonina, F. Lander, N. Mahadev, D. McKenna, D. Shadbegian, S. Taylor, S. Wagner, S. Wall-Smith, R. Wisenski.

The meeting began at 9:00, breakfast was served and introductions were made.

Dr. Antonucci arrived at 10:30 as planned and explained the importance of professional accreditation for individual departments and college-wide assessment. He went on to thank the group for gathering to discuss how the College may best serve the community and provided a general description of how the college will continue to grow in this economically challenging time.

Discussions were the concerned with the following:

There was a discussion about the assessment process. Frits and Stephen Wall-Smith explained the assessment process. Assessment was described as system we can use to determine if our overall objectives are being met. Frits said the purpose of an Industrial Advisory Board (IAB) is to help us assess how well our program is meeting the needs of local employers and identify areas for program improvement. Frits mentioned that the IAB ought to have some governing rules that will be developed over time.

There were comments (Ken and Ron) that some of their best employees have come from our program. Ken described his hiring experiences. Good employees have both talent and a good work ethic. Most agreed that flexibility and attention to detail is important and often lacking. Employees with experience balancing many different life challenges often do well. Hires with great resumes sometimes do not survive because they do not attend to details or lack a good work ethic.

It was generally agreed that UNIX experience is marginal in our curriculum. Ron said that working knowledge of several platforms is a valuable asset. Employers want people with the flexibility to learn new things when needed. New hires must be able to admit ignorance and have the drive to learn new skills on the fly. Ken said that new hires that misrepresent themselves do not survive.

Ron went on to describe that in his experience, algorithm design skills are poor in new hires. He finds they have little flexibility when it comes to choosing an appropriate data structure to solve a particular problem. Many do not realize that the choice of data structure impacts algorithm development and vice versa. He went on to describe other weaknesses in new hires: project management skills, the ability to work within a group or organization and experience estimating how long a task will take.

This led to a discussion of how our curriculum can better prepare students to manage projects, work in groups and estimate the effort required to complete tasks. Our capstone courses attempt to address these areas. It is difficult to find good, fresh, real-world project

opportunities. Brady stressed that institutionalization is a problem. The WPI major project that every senior must complete to graduate was mentioned as a model for such real world experiences. Ken explained that it is a good idea for students to describe some of their school projects on their resumes and during interviews. This gives them the opportunity to describe their approach to problem solving and demonstrate their enthusiasm for the work. He also brought up the topic of group interaction. It is important to give students substantial group problem-solving experiences. Someone (Dave) suggested that a course in the "Foundations of Project Management" would be useful.

Mahadev responded by describing what he does in CS II and Software Engineering. Each CS II project has a requirements specification, a design description, coding standards, UML description and use cases. He went on to explain how all of the projects in Software Engineering require students to work in groups, use databases and understand client/server architectures. It was suggested that the student's project experiences (in the capstone and elsewhere) could be used to build a portfolio that would be useful during interviews. There was agreement that a portfolio presentation would be useful in demonstrating understanding.

Everyone agreed that communication skills are very important. It was mentioned that our small class size contributes to good communication. We also require students to take a speech class. Dave suggested that the students may benefit from more technical presentation opportunities in our required classes. Someone (Dale) explained that there is a need for employees that have the ability to facilitate communication between the technical and business sides of an organization. This was described as a "not so technical" job that is vital to maintaining the viability of a technical endeavor. People in these positions must have an understanding of technical project management. They must be able to get accurate estimates of both time and cost from technical staff and explain the project to executive level management. This led the discussion back to the importance of being able to estimate how long it will take to accomplish a particular task. Exercises that can be done at the curricular level to get students in the habit of estimating their effort may be easy to implement. Everyone agrees that meeting assignment deadlines is an important part of the educational experience.

The discussion returned to the interview project portfolio idea. Is a physical portfolio better than a web link? Someone mentioned that time spent discussing a physical portfolio can help an interviewer assess a prospective employees "pride of accomplishment". The idea of a web link spurred a discussion of web presence and social networking. Ron (and others) said that he searches the internet to find information on prospective hires from Facebook, LinkedIn, etc. He also looks to see if job applicants have contributed intelligently to projects/discussions on sites like sourceforge, codeproject, etc. Bill suggested that social networking/engineering and gaming are good areas for student project development.

Frits: One of our strengths, relative to the other state colleges, is our hardware emphasis. Are there any places that you see that strength paying off in your work?

- Networking has a big hardware aspect, and is vitally important.
- The new virtual machine operating systems have a hardware orientation, even though they are software.
- Driver work – there are lots of Windows CS systems, and they all need drivers for everything.

- Performance is important, and you have to understand the hardware to optimize to it.
- "The biggest thing I took out of the hardware courses was the integration of math, physics, engineering, and software."

Frits talked briefly about the 'assessment of our assessment process' by the accreditation visitors, and some of the strategies we're trying to use to respond: TK20, direct assessment by translating assignment goals and exam grades into student objectives and department mission.

Ken: Is web programming still and elective? Mahadev's class was defining for my career.
Ron: The web is everywhere!
Adam: Web services – we're evaluating and buying web services and software as a service. You've got to include the Web in all your planning; you have to understand the interfaces and the algorithms.
Dave: For CIS you have to understand the hardware and the software, but I spend my day thinking about how to bring my department into alignment with the business.
Ron: CIS students, even if they don't end up programming, need to understand the possibilities.
Adam: In a small company, you have to be able to do everything. Especially when you first start out.

Frits: We're planning to meet once a year. I'm going to send out a proposed set of bylaws. Send me an email if you have any ideas. Thank you all!

The meeting adjourned at noon.

Detail Requirements

@01053373 Admin Only
Jun 24, 2009 11:47 am

Information for Admissions Applicant

This is NOT an official evaluation.

Program Evaluation

Program :	Computer Science	**Catalog Term :**	Fall 2008
Campus :		**Evaluation Term :**	Spring 2009
College :	Fitchburg State College	**Expected Graduation Date :**	
Degree :	Bachelor of Science	**Request Number :**	1201
Level :	Undergraduate	**Results as of :**	Jun 24, 2009
Majors :	Computer Science	**Minors :**	
Departments :	Computer Science	**Concentrations :**	

		Met Credits		Courses	
		Required	Used	Required	Used
Total Required :	No	120.000	0.000	0	
Required Institutional :	No	45.000	0.000	0	
Maximum Institutional Non-Traditional :		12.000	0.000	0	
Maximum Transfer :		75.000	0.000	0	
Program GPA :	Yes	.00	.00		
Overall GPA :	No	2.00	.00		
Other Course Information					
Transfer :		75.000	0.000	0	

*** This is NOT an official evaluation. ***

Area : **Computer Science Required LAS (8 courses) - Not Met**
Description : Courses used in this area are still available for use in other areas below.

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Cre
No		(	MATH		1800									
No		)AND(	MATH		1900									
No		)OR(	CSC		1900									
No		)AND(	MATH		2300									
No		AND	MATH		2400									
No		AND	MATH		2600									
No		)AND(	PHYS		2300									
No		AND	PHYS		2400									
No		)AND(	SPCH		1000									
		)												

Total Credits and GPA (

*** unofficial evaluation ***

Area : **Computer Science Major (16 courses) - Not Met**

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Credits	G
No		(	CSC		1500										

No	AND	CSC	1550			
No	AND	CSC	1600			
No	AND	CSC	1650			
No	AND	CSC	2560			
No	AND	CSC	2600			
No	AND	CSC	3100			
No	AND	CSC	3200			
No	AND	CSC	3600			
No	AND	CSC	3700			
No	AND	CSC	4400			
No	)AND(	CSC	3000	5999	5	
	)					

Total Credits and GPA 0.000

*** unofficial evaluation ***

Area : **Global Diversity (6.000 credits AND 2 courses) - Not Met**

Description : This area will act as a "checkpoint" in order to verify that a Global Diversity course is taken from two of the next three clusters by checki GDS, GDC, and GDA. One of the two courses must also have the Non-Western attribute of GDSN, GDCN, or GDAN. These courses will be this area in order to satisfy requirements in other areas below.

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Cre
No	(		GDCN and a GDS, GDSN, GDA or GDAN											
No	)OR(		GDSN and a GDC, GDCN, GDA, or GDAN											
No	)OR(		GDAN and a GDS, GDSN, GDC, or GDCN											
	)													

Total Credits and GPA (

*** unofficial evaluation ***

Area : Science, Math & Technology (12.000 credits AND 4 courses) - Not Met

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Credits	G
No	(		MATH		1200	4999		1							
No	)AND(			LAB				1							
No	)AND(	H&F	3 Credit Health & Fitness Req												
No	)AND(			SMT			3.000								
	)														

Total Credits and GPA 0.000

*** unofficial evaluation ***

Area : Citizenship & The World (9.000 credits AND 3 courses) - Not Met

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Credits	G
No	(		SOC		1100										
No	)OR(		PSY		1100										
No	)OR(		PSY		2200										
No	)AND(		HIST		1000	4999		1							
No	)AND(			CTW				1							
	)														

Total Credits and GPA 0.000

*** unofficial evaluation ***

Area : **The Arts (15.000 credits AND 5 courses) - Not Met**

Description : AOM is the attribute for the Art or Music requirement.

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Cre
No	(		ENGL		1100									
No	AND		ENGL		1200									
No	)AND(			AOM			3.000							
No	)AND(			LIT				1						
No	)AND(			ART			3.000							
	)													

Total Credits and GPA (

*** unofficial evaluation ***

Area : **Advanced LA&S Options (12.000 credits) - Not Met**

Description : This area requires that you fulfill the requirements for one of the three following options. In order for your courses to appear in this area Registrar's Office which option you choose by filling out an "Advanced LA&S Option Election" form available in the Registrar's Office. ----- OPTION A requires 6 credits of a Foreign Language and 6 credits of 2000+ LAS Electives from ONE LAS DISCIPLINE. ---------- requires 12 credits in ONE LAS DISCIPLINE, 6 credits of which must be 2000+. ---------- ----------OPTION C requires 12 credit designed by you and your advisor tailored to meet your specific goals. 6 of the 12 credits must be at the 2000+ level. Option C curriculu the Dean of Student & Academic Life and submitted to the Registrar's Office before the completion of 60 credits.

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Cre
No	READ TEXT ABOVE FOR DETAILS													

Total Credits and GPA (

*** unofficial evaluation ***

Area : **Free Electives - Not Met**

Description : Any courses not used above will appear in this area. NOTE: Even if all areas above are "Met," you must have at least 120 college level cr graduation. Developmental courses such as MATH 0100 and ENGL 0100 do not count toward a degree and will not appear on your CAPP.

Met	Condition	Rule	Subject	Attribute	Low	High	Required Credits	Required Courses	Term	Subject	Course	Title	Attribute	Cre
No					100	9999E								

Total Credits and GPA (

*** unofficial evaluation ***

Back to Display Options

RELEASE: 7.4.0.1

Appendix H – Sample Transcripts of Recent Graduates

Below are six randomly selected transcripts of recent CS graduates.

U N O F F I C I A L

Record of: [redacted]
*** WARNING ***
--No Address--

Date Issued: 24-JUN-2013
Date of Birth: 25-JAN-1991
Student ID: [redacted]
Level: Undergraduate

Issued To: ml-ld

Undergraduate
Continuing

Course Level: Undergraduate
Student Type: Continuing

Current Program
Major : Computer Science
Minor : Mathematics

Degree Awarded Bachelor of Science 18-MAY-2013
Primary Degree
Major : Computer Science
Minor : Mathematics
Inst. Honors: Summa Cum Laude

SUBJ NO.	COURSE TITLE	CRED GRD	PTS R

TRANSFER CREDIT ACCEPTED BY THE INSTITUTION:

Fall 09 AP Exam Credit

CHEM 1300	General Chemistry I	4.00 TR	
CHEM 1990	Transfer Elective	3.00 TR	
MATH 2300	Calculus I	4.00 TR	

Ehrs: 11.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

INSTITUTION CREDIT:

Term: Fall 2009
Computer Science

CSC 1500	Computer Science I	3.00	4.0	12.00
CSC 1600	Intro to Electronics	4.00	4.0	16.00
ENGL 1100	Writ I:Race / Gender in Horror	3.00	4.0	12.00
MATH 2400	Calculus II	4.00	3.7	14.80

Term: Ehrs: 14.00 GPA-Hrs: 14.00 QPts: 54.80 GPA: 3.91
Dean's List
Good Standing

Term: Spring 2010
Computer Science

CSC 1550	Computer Science II	3.00	4.0	12.00
CSC 1650	Digital Electronics	4.00	4.0	16.00
ENGL 1200	Writing II:	3.00	4.0	12.00
EXSS 1000	Health and Fitness	3.00	4.0	12.00

********** CONTINUED ON NEXT COLUMN **********

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS R

Institution Information continued:

MATH 1900	Discrete Mathematics	3.00	4.0	12.00

Term: Ehrs: 16.00 GPA-Hrs: 16.00 QPts: 64.00 GPA: 4.00
Dean's List
Good Standing

Term: Fall 2010
Computer Science

CSC 2560	Systems Programming	3.00	4.0	12.00
CSC 2600	Computer Organization	4.00	3.5	14.00
MATH 2600	Linear Algebra	3.00	4.0	12.00
MATH 3300	Calculus III	3.00	4.0	12.00
SPCH 1000	Intro to Speech Communication	3.00	3.7	11.10

Term: Ehrs: 16.00 GPA-Hrs: 16.00 QPts: 61.10 GPA: 3.82
President's List
Good Standing

Term: Spring 2011
Computer Science

CSC 3100	Operating Systems	3.00	4.0	12.00
CSC 3450	Local Area Networks	3.00	3.3	9.90
CSC 3600	Microprocessors	4.00	4.0	16.00
MATH 1700	Applied Statistics	3.00	4.0	12.00
MATH 2500	Intro to Mathematical Thought	3.00	4.0	12.00

Term: Ehrs: 16.00 GPA-Hrs: 16.00 QPts: 61.90 GPA: 3.87
President's List
Good Standing

Term: Fall 2011
Computer Science

CSC 3200	Programming Languages	3.00	3.3	9.90
MATH 3000	Geometry I	3.00	4.0	12.00
MATH 4000	Real Variable Theories	3.00	3.0	9.00
MUSC 2000	Commonwealth of the Arts	3.00	4.0	12.00
PSY 1200	Human Growth and Development	3.00	4.0	12.00

Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 54.90 GPA: 3.66
Dean's List
Good Standing

********** CONTINUED ON PAGE 2 **********

Of: [redacted]

Date Issued: 24-JUN-2013
Date of Birth: 25-JAN-1991
Student ID: [redacted]
Level: Undergraduate

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:					
Term: Spring 2012					
Computer Science					
CSC 3300	Assembly Language	3.00	3.5	10.50	
CSC 3700	Algorithms Data Structure	3.00	4.0	12.00	
CSC 4400	Software Engineering	3.00	4.0	12.00	
ITEC 1700	Evolution of Industrial Tech	3.00	3.5	10.50	
SPAN 1000	Spanish for Beginners I	3.00	4.0	12.00	
Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts:		57.00	GPA:	3.80	
Dean's List					
Good Standing					
Term: Fall 2012					
Computer Science					
CSC 3050	Web Programming w/Java	3.00	4.0	12.00	
CSC 3400	Data Comm/Networking	3.00	3.7	11.10	
CSC 3500	Object-Oriented Programming	3.00	4.0	12.00	
PHYS 2300	General Physics I	4.00	4.0	16.00	
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts:		51.10	GPA:	3.93	
Dean's List					
Good Standing					
Term: Spring 2013					
Computer Science					
ENGL 2620	Classic Mythology	3.00	4.0	12.00	
MATH 3150	Elementary Number Theory	3.00	4.0	12.00	
PHYS 2400	General Physics II	4.00	3.7	14.80	
SOC 1100	Introduction to Sociology	3.00	4.0	12.00	
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts:		50.80	GPA:	3.91	
President's List					
Good Standing					

********** TRANSCRIPT TOTALS **********

	Earned Hrs	GPA Hrs	Points	GPA
TOTAL INSTITUTION	118.00	118.00	455.60	3.86
TOTAL TRANSFER	11.00	0.00	0.00	0.00
OVERALL	129.00	118.00	455.60	3.86

********** END OF TRANSCRIPT **********

Record of: [redacted]
*** WARNING ***
--No Address--

Date Issued: 24-JUN-2013
Date of Birth: 13-SEP-1960
Student ID: [redacted]
Level: Undergraduate

Issued To: ml-ld

Undergraduate
Continuing

Course Level: Undergraduate
Student Type: Continuing

Current Program
Major : Computer Science

Degree Awarded Bachelor of Science 18-MAY-2013
Primary Degree
Major : Computer Science
Inst. Honors: Cum Laude

SUBJ NO.	COURSE TITLE	CRED GRD	PTS R

TRANSFER CREDIT ACCEPTED BY THE INSTITUTION:

Benjamin Franklin Inst Boston

CSC 2990	Elective	3.00 TR	
ENGL 1100	Writing I	3.00 TR	
ITEC 2990	Elective	4.00 TR	
MATH 1300	Elementary Functions	4.00 TR	
POLS 1000	United States Government	3.00 TR	

Ehrs: 17.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

Benjamin Franklin Inst Boston

ITEC 2990	Elective	4.00 TR	
MATH 2300	Calculus I	3.00 TR	
PHYS 1300	General Physics I	4.00 TR	
SOC 2990	Elective	3.00 TR	

Ehrs: 14.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

Benjamin Franklin Inst Boston

MATH 2400	Calculus II	3.00 TR	

Ehrs: 3.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

Benjamin Franklin Inst Boston

ENGL 1200	Writing II	3.00 TR	
PHYS 1400	General Physics II	4.00 TR	

********** CONTINUED ON NEXT COLUMN **********

SUBJ NO.	COURSE TITLE	CRED GRD	PTS R

Transfer Information continued:
Ehrs: 7.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

Benjamin Franklin Inst Boston

ENGL 2990	Elective	3.00 TR	

Ehrs: 3.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

Benjamin Franklin Inst Boston

CSC 1650	Digital Electronics	3.00 TR	

Ehrs: 3.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

Benjamin Franklin Inst Boston

CSC 1600	Introduction To Electronics	3.00 TR	
ITEC 2992	Elective	3.00 TR	
ITEC 2993	Elective	3.00 TR	

Ehrs: 9.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

INSTITUTION CREDIT:

Term: Spring 1998
Computer Science

CSC 1500	Intro Pascal	3.00	3.5	10.50	

Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 10.50 GPA: 3.50
Good Standing

Term: Summer 2 1998
Computer Science

CSC 1550	Data Structures	3.00	3.0	9.00	

Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 9.00 GPA: 3.00
Good Standing

Term: Fall 1998
Computer Science

CSC 2560	Systems Programming	3.00	1.5	4.50	

********** CONTINUED ON PAGE 2 **********

of: [redacted]

Date Issued: 24-JUN-2013
Date of Birth: 13-SEP-1960
Student ID: [redacted]
Level: Undergraduate

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 4.50 GPA: 1.50						
Good Standing						
Term: Spring 1999						
Computer Science						
CSC	3600	Microprocessors	4.00	3.0	12.00	
Term: Ehrs: 4.00 GPA-Hrs: 4.00 QPts: 12.00 GPA: 3.00						
Good Standing						
Term: Summer 1 1999						
Computer Science						
CSC	2500	Assembly Language Programming	3.00	3.5	10.50	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 10.50 GPA: 3.50						
Good Standing						
Term: Fall 1999						
Computer Science						
CSC	2600	Computer Organization	4.00	3.0	12.00	
CSC	3450	Local Area Networks	3.00	3.5	10.50	
Term: Ehrs: 7.00 GPA-Hrs: 7.00 QPts: 22.50 GPA: 3.21						
Good Standing						
Term: Spring 2000						
Computer Science						
CSC	3500	Object-Oriented Programming	3.00	3.5	10.50	
CSC	4500	Database Design	3.00	W	0.00	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 10.50 GPA: 3.50						
Good Standing						
Term: Fall 2000						
Computer Science						
CSC	3100	Operating Systems	3.00	3.5	10.50	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 10.50 GPA: 3.50						
Good Standing						

********** CONTINUED ON NEXT COLUMN **********

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
Term: Spring 2001						
Computer Science						
CSC	3400	Data Communications/Networking	3.00	2.0	6.00	
CSC	3700	Algorithms And Data Structures	3.00	4.0	12.00	
Term: Ehrs: 6.00 GPA-Hrs: 6.00 QPts: 18.00 GPA: 3.00						
Good Standing						
Term: Spring 2003						
Computer Science						
CSC	3050	Web Programming w/Java	3.00	4.0	12.00	
CSC	3250	Adv Topics in Web Development	3.00	3.0	9.00	
Term: Ehrs: 6.00 GPA-Hrs: 6.00 QPts: 21.00 GPA: 3.50						
Good Standing						
Term: Summer I 2006						
Computer Science						
CSC	1900	Discrete Math	3.00	4.0	12.00	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 12.00 GPA: 4.00						
Term: Fall 2009						
Computer Science						
CSC	3200	Programming Languages	3.00	4.0	12.00	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 12.00 GPA: 4.00						
Good Standing						
Term: Spring 2010						
Computer Science						
HIST	1400	United States History I	3.00	3.0	9.00	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 9.00 GPA: 3.00						
Good Standing						

********** CONTINUED ON PAGE 3 **********

of: [redacted]

Date Issued: 24-JUN-2013
Date of Birth: 13-SEP-1960
Student ID: [redacted]
Level: Undergraduate

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:					
Term: Fall 2010					
Computer Science					
SPCH 1000	Intro to Speech Communication	3.00	3.3	9.90	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 9.90 GPA: 3.30					
Good Standing					
Term: Spring 2011					
Computer Science					
MATH 1800	Business Statistics	3.00	3.7	11.10	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 11.10 GPA: 3.70					
Good Standing					
Term: Summer I 2011					
Computer Science					
ENGL 2300	Literature and Disability	3.00	2.7	8.10	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 8.10 GPA: 2.70					
Term: Fall 2011					
Computer Science					
MATH 2600	Linear Algebra	3.00	2.5	7.50	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 7.50 GPA: 2.50					
Good Standing					
Term: Winter 2013					
Computer Science					
EXSS 1000	Health and Fitness	3.00	3.0	9.00	
Term: Ehrs: 3.00 GPA-Hrs: 3.00 QPts: 9.00 GPA: 3.00					
Term: Spring 2013					
Computer Science					
IDIS 1800	Global Issues	3.00	3.3	9.90	
********** CONTINUED ON NEXT COLUMN **********					

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:					
MUSC 2000	Commonwealth of the Arts-COTA	3.00	4.0	12.00	
SOC 1100	Introduction to Sociology	3.00	3.7	11.10	
Term: Ehrs: 9.00 GPA-Hrs: 9.00 QPts: 33.00 GPA: 3.67					
Good Standing					

********** TRANSCRIPT TOTALS **********

	Earned Hrs	GPA Hrs	Points	GPA
TOTAL INSTITUTION	74.00	74.00	240.60	3.25
TOTAL TRANSFER	56.00	0.00	0.00	0.00
OVERALL	130.00	74.00	240.60	3.25

********** END OF TRANSCRIPT **********

Record of: [redacted]

*** WARNING ***
--No Address--

Date Issued: 24-JUN-2013

Date of Birth: 16-FEB-1991

Student ID: [redacted]

Level: Undergraduate

Issued To: ml-ld

Undergraduate
Continuing

Course Level: Undergraduate
Student Type: Continuing

Current Program
Major : Computer Science

Degree Awarded Bachelor of Science 18-MAY-2013
Primary Degree
Major : Computer Science
Inst. Honors: Magna Cum Laude

SUBJ NO.	COURSE TITLE	CRED GRD	PTS R

TRANSFER CREDIT ACCEPTED BY THE INSTITUTION:

Fall 09 AP Exam Credit

MATH 2300	Calculus I	4.00 TR		

Ehrs: 4.00 GPA-Hrs: 0.00 QPts: 0.00 GPA: 0.00

INSTITUTION CREDIT:

Term: Fall 2009
Computer Science

CSC 1000	Intro to Programming	3.00	4.0	12.00
CSC 1500	Computer Science I	3.00	4.0	12.00
ENGL 1100	Writing I	3.00	4.0	12.00
HIST 1400	United States History I	3.00	4.0	12.00
MATH 1300	Precalculus	3.00	4.0	12.00

Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 60.00 GPA: 4.00
Dean's List
Good Standing

Term: Spring 2010
Computer Science

CSC 1550	Computer Science II	3.00	3.7	11.10
ENGL 1200	Writing II	3.00	4.0	12.00
IDIS 1800	Global Issues	3.00	3.7	11.10
PSY 2200	Human Growth and Development	3.00	2.5	7.50
SOC 1100	Introduction to Sociology	3.00	3.3	9.90

Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 51.60 GPA: 3.44
Dean's List
Good Standing

SUBJ NO.	COURSE TITLE	CRED GRD	PTS R

Institution Information continued:

Term: Fall 2010
Computer Science

CSC 2560	Systems Programming	3.00	4.0	12.00
CSC 3500	Object-Oriented Programming	3.00	3.7	11.10
EXSS 1000	Health and Fitness	3.00	3.7	11.10
MATH 2300	Calculus I	4.00	4.0	16.00

Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 50.20 GPA: 3.86
Dean's List
Good Standing

Term: Spring 2011
Computer Science

CSC 3300	Assembly Language	3.00	2.5	7.50
CSC 3700	Algorithms Data Structure	3.00	3.5	10.50
MATH 2400	Calculus II	4.00	2.5	10.00
MUSC 3700	19th Century Music	3.00	3.3	9.90
PHYS 2600	Calculus-Based Physics I	4.00	3.0	12.00

Term: Ehrs: 17.00 GPA-Hrs: 17.00 QPts: 49.90 GPA: 2.94
Good Standing

Term: Fall 2011
Computer Science

ART 2570	Northern Renaissance Art	3.00	3.5	10.50
ART 3300	History of Architecture	3.00	3.7	11.10
CSC 1600	Intro to Electronics	4.00	3.7	14.80
CSC 1900	Discrete Math	3.00	3.5	10.50
FREN 1000	French for Beginners I	3.00	W	0.00
MATH 2600	Linear Algebra	3.00	2.3	6.90

Term: Ehrs: 16.00 GPA-Hrs: 16.00 QPts: 53.80 GPA: 3.36
Dean's List
Good Standing

Term: Spring 2012
Computer Science

CSC 1650	Digital Electronics	4.00	3.3	13.20
CSC 2700	Business Programming	3.00	3.3	9.90
CSC 4400	Software Engineering	3.00	4.0	12.00

********** CONTINUED ON PAGE 2 **********

Record of: [redacted]

Date Issued: 24-JUN-2013
Date of Birth: 16-FEB-1991
Student ID: [redacted]
Level: Undergraduate

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:					
ENGL 3220	American Novel to 1950	3.00	3.7	11.10	
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 46.20 GPA: 3.55					
Dean's List					
Good Standing					
Term: Fall 2012					
Computer Science					
CSC 2600	Computer Organization	4.00	3.5	14.00	
CSC 3200	Programming Languages	3.00	3.7	11.10	
MATH 1700	Applied Statistics	3.00	4.0	12.00	
PHYS 2300	General Physics I	4.00	4.0	16.00	
Term: Ehrs: 14.00 GPA-Hrs: 14.00 QPts: 53.10 GPA: 3.79					
Dean's List					
Good Standing					
Term: Spring 2013					
Computer Science					
CSC 3100	Operating Systems	3.00	4.0	12.00	
CSC 3600	Microprocessors	4.00	3.5	14.00	
CSC 4950	Internship: Computer Science	6.00	4.0	24.00	
MATH 3150	Elementary Number Theory	3.00	2.7	8.10	
SPCH 1000	Intro to Speech Communication	3.00	3.3	9.90	
Term: Ehrs: 19.00 GPA-Hrs: 19.00 QPts: 68.00 GPA: 3.58					
Dean's List					
Good Standing					

********** TRANSCRIPT TOTALS **********

	Earned Hrs	GPA Hrs	Points	GPA
TOTAL INSTITUTION	122.00	122.00	432.80	3.55
TOTAL TRANSFER	4.00	0.00	0.00	0.00
OVERALL	126.00	122.00	432.80	3.55

********** END OF TRANSCRIPT **********

U N O F F I C I A L

Record of: [redacted]
*** WARNING ***
--No Address--

Date Issued: 24-JUN-2013
Date of Birth: 11-FEB-1991
Student ID: [redacted]
Level: Undergraduate

Issued To: ml-ld

Undergraduate
Continuing

Course Level: Undergraduate
Student Type: Continuing

Current Program
Major : Computer Science

Degree Awarded Bachelor of Science 18-MAY-2013
Primary Degree
Major : Computer Science
Inst. Honors: Cum Laude

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
INSTITUTION CREDIT:						
Term: Fall 2009						
Comp Info Systems						
CSC	1000	Intro to Programming	3.00	4.0	12.00	
CSC	1500	Computer Science I	3.00	3.3	9.90	
ENGL	1100	Writing I	3.00	3.5	10.50	
HIST	1400	United States History I	3.00	4.0	12.00	
MATH	1300	Precalculus	3.00	3.7	11.10	
Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 55.50 GPA: 3.70						
Dean's List						
Good Standing						
Term: Spring 2010						
Computer Science						
CSC	1550	Computer Science II	3.00	3.0	9.00	
ENGL	1200	Writing II	3.00	4.0	12.00	
IDIS	1800	Global Issues	3.00	3.7	11.10	
PSY	2200	Human Growth and Development	3.00	1.5	4.50	
SOC	1100	Introduction to Sociology	3.00	2.5	7.50	
Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 44.10 GPA: 2.94						
Good Standing						
Term: Fall 2010						
Computer Science						
CSC	2560	Systems Programming	3.00	3.5	10.50	
CSC	3500	Object-Oriented Programming	3.00	3.7	11.10	
EXSS	1000	Health and Fitness	3.00	3.7	11.10	

********** CONTINUED ON NEXT COLUMN **********

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
MATH	2300	Calculus I	4.00	3.5	14.00	
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 46.70 GPA: 3.59						
Dean's List						
Good Standing						
Term: Spring 2011						
Computer Science						
CSC	3300	Assembly Language	3.00	2.5	7.50	
CSC	3700	Algorithms Data Structure	3.00	3.5	10.50	
MATH	2400	Calculus II	4.00	2.5	10.00	
MUSC	3700	19th Century Music	3.00	3.3	9.90	
PHYS	2600	Calculus-Based Physics I	4.00	2.5	10.00	
Term: Ehrs: 17.00 GPA-Hrs: 17.00 QPts: 47.90 GPA: 2.82						
Good Standing						
Term: Fall 2011						
Computer Science						
ART	2570	Northern Renaissance Art	3.00	3.5	10.50	
ART	3300	History of Architecture	3.00	3.7	11.10	
CSC	1600	Intro to Electronics	4.00	3.7	14.80	
CSC	1900	Discrete Math	3.00	3.5	10.50	
CSC	4940	Internship: Computer Science	3.00	4.0	12.00	I
FREN	1000	French for Beginners I	3.00	W	0.00	
MATH	2600	Linear Algebra	3.00	2.3	6.90	
Term: Ehrs: 19.00 GPA-Hrs: 19.00 QPts: 65.80 GPA: 3.46						
Dean's List						
Good Standing						
Term: Spring 2012						
Computer Science						
CSC	1650	Digital Electronics	4.00	3.3	13.20	
CSC	2700	Business Programming	3.00	3.5	10.50	
CSC	4400	Software Engineering	3.00	4.0	12.00	
ENGL	3220	American Novel to 1950	3.00	3.5	10.50	
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 46.20 GPA: 3.55						
Dean's List						
Good Standing						

********** CONTINUED ON PAGE 2 **********

[redacted]

UNOFFICIAL

Date Issued: 24-JUN-2013
Date of Birth: 11-FEB-1991
Student ID [redacted]
Level: Undergraduate

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
Term:		Fall 2012				
Computer Science						
ISC	2600	Computer Organization	4.00	3.5	14.00	
ISC	3200	Programming Languages	3.00	3.7	11.10	
IATH	1700	Applied Statistics	3.00	4.0	12.00	
HYS	2300	General Physics I	4.00	3.0	12.00	
erm:	Ehrs: 14.00	GPA-Hrs: 14.00 QPts:	49.10	GPA:	3.51	
ean's List						
ood Standing						
Term:		Spring 2013				
Computer Science						
SC	3100	Operating Systems	3.00	4.0	12.00	
SC	3600	Microprocessors	4.00	3.0	12.00	
SC	4940	Internship: Computer Science	3.00	4.0	12.00	I
ATH	3150	Elementary Number Theory	3.00	2.7	8.10	
CH	1000	Intro to Speech Communication	3.00	4.0	12.00	
rm:	Ehrs: 16.00	GPA-Hrs: 16.00 QPts:	56.10	GPA:	3.51	
an's List						
od Standing						

********** TRANSCRIPT TOTALS **********

	Earned Hrs	GPA Hrs	Points	GPA
TAL INSTITUTION	122.00	122.00	411.40	3.37
TAL TRANSFER	0.00	0.00	0.00	0.00
ERALL	122.00	122.00	411.40	3.37

********** END OF TRANSCRIPT **********

Record of: [redacted]
*** WARNING ***
--No Address--

Date Issued: 24-JUN-2013
Date of Birth: 10-DEC-1989
Student ID: [redacted]
Level: Undergraduate

Issued To: ml-ld

Undergraduate
Continuing

Course Level: Undergraduate
Student Type: Continuing

Current Program
Major : Computer Science

Degree Awarded Bachelor of Science 18-MAY-2013
Primary Degree
Major : Computer Science

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS	R
INSTITUTION CREDIT:					
Term: Fall 2008					
Computer Science					
CSC 1500	Computer Science I	3.00	3.7	11.10	
ENGL 1100	Writing I	3.00	3.3	9.90	
HIST 1400	United States History I	3.00	3.0	9.00	
MATH 1300	Pre-Calculus	3.00	2.0	6.00	
PSY 1100	General Psychology	3.00	3.5	10.50	
Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 46.50 GPA: 3.10					
Good Standing					
Term: Spring 2009					
Computer Science					
CSC 1550	Computer Science II	3.00	3.3	9.90	
ENGL 1200	Writ II:Punk,Style & Rebellion	3.00	2.7	8.10	
EXSS 1000	Health and Fitness	3.00	0.0	0.00	E
MATH 1800	Business Statistics	3.00	0.0	0.00	E
Term: Ehrs: 6.00 GPA-Hrs: 6.00 QPts: 18.00 GPA: 3.00					
Good Standing					
Term: Fall 2009					
Computer Science					
CSC 1600	Intro to Electronics	4.00	1.5	6.00	
CSC 1900	Discrete Math	3.00	3.0	9.00	
CSC 2560	Systems Programming	3.00	3.5	10.50	
CSC 3200	Programming Languages	3.00	3.0	9.00	
MATH 2300	Calculus I	4.00	0.0	0.00	E
********** CONTINUED ON NEXT COLUMN **********					

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:					
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 34.50 GPA: 2.65					
Good Standing					
Term: Spring 2010					
Computer Science					
ART 1100	Art Appreciation	3.00	0.0	0.00	E
CSC 1650	Digital Electronics	4.00	2.5	10.00	
CSC 3700	Algorithms Data Structure	3.00	3.3	9.90	
EXSS 1000	Health & Fitness	3.00	2.7	8.10	I
Term: Ehrs: 10.00 GPA-Hrs: 10.00 QPts: 28.00 GPA: 2.80					
Good Standing					
Term: Fall 2010					
Computer Science					
CSC 2600	Computer Organization	4.00	2.7	10.80	
CSC 3500	Object-Oriented Programming	3.00	3.7	11.10	
SOC 1100	Introduction to Sociology	3.00	3.3	9.90	
SPCH 1000	Intro to Speech Communication	3.00	3.7	11.10	
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 42.90 GPA: 3.30					
Dean's List					
Good Standing					
Term: Spring 2011					
Computer Science					
CSC 3100	Operating Systems	3.00	0.0	0.00	E
CSC 3300	Assembly Language	3.00	4.0	12.00	
CSC 3450	Local Area Networks	3.00	2.5	7.50	
CSC 3600	Microprocessors	4.00	0.0	0.00	E
Term: Ehrs: 6.00 GPA-Hrs: 6.00 QPts: 19.50 GPA: 3.25					
Good Standing					
Term: Fall 2011					
Computer Science					
CSC 3400	Data Comm/Networking	3.00	1.7	5.10	
ENGL 2000	Am Lit I: Explore to Civil War	3.00	3.7	11.10	
MATH 1800	Business Statistics	3.00	3.5	10.50	I
********** CONTINUED ON PAGE 2 **********					

Record of: [redacted]

Date Issued: 24-JUN-2013
Date of Birth: 10-DEC-1989
Student ID: [redacted]
Level: Undergraduate

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
MATH	2300	Calculus I	4.00	3.0	12.00	I
SPAN	1000	Spanish for Beginners I	3.00	0.0	0.00	E
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 38.70 GPA: 2.98						
Good Standing						
Term: Spring 2012						
Computer Science						
CSC	3600	Microprocessors	4.00	4.0	16.00	I
ENGL	2100	Am Lit II:Civil War to Present	3.00	2.0	6.00	
MATH	2400	Calculus II	4.00	1.5	6.00	
PHYS	2300	General Physics I	4.00	3.0	12.00	
Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 40.00 GPA: 2.67						
Good Standing						
Term: Fall 2012						
Computer Science						
ART	1100	Art Appreciation	3.00	2.0	6.00	I
CSC	3050	Web Programming w/Java	3.00	2.0	6.00	
MATH	2600	Linear Algebra	3.00	W	0.00	
PHYS	2400	General Physics II	4.00	2.5	10.00	
SPAN	1000	Spanish for Beginners I	3.00	2.5	7.50	I
Term: Ehrs: 13.00 GPA-Hrs: 13.00 QPts: 29.50 GPA: 2.27						
Good Standing						
Term: Spring 2013						
Computer Science						
CSC	3100	Operating Systems	3.00	2.0	6.00	I
CSC	4400	Software Engineering	3.00	2.5	7.50	
MATH	2600	Linear Algebra	3.00	2.3	6.90	
MATH	3000	Geometry I	3.00	2.3	6.90	
MUSC	1500	World Music	3.00	3.3	9.90	
SPAN	1100	Spanish for Beginners II	3.00	1.5	4.50	
Term: Ehrs: 18.00 GPA-Hrs: 18.00 QPts: 41.70 GPA: 2.32						
Good Standing						

********** CONTINUED ON NEXT COLUMN **********

********** TRANSCRIPT TOTALS **********

	Earned Hrs	GPA Hrs	Points	GPA
TOTAL INSTITUTION	122.00	122.00	339.30	2.78
TOTAL TRANSFER	0.00	0.00	0.00	0.00
OVERALL	122.00	122.00	339.30	2.78

********** END OF TRANSCRIPT **********

U N O F F I C I A L

Record of: [redacted]

*** WARNING ***
--No Address--

Date Issued: 24-JUN-2013
Date of Birth: 26-APR-1991
Student ID: [redacted]
Level: Undergraduate

Issued To: ml-ld

Undergraduate
Continuing

Course Level: Undergraduate
Student Type: Continuing

Current Program
Major : Computer Science

Degree Awarded Bachelor of Science 18-MAY-2013
Primary Degree
Major : Computer Science

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
INSTITUTION CREDIT:						
Term: Fall 2009						
Computer Science						
CSC	1500	Computer Science I	3.00	3.5	10.50	
ENGL	1200	Writing II: Graphic Novel	3.00	3.3	9.90	
HIST	1100	World Civilizations II	3.00	1.0	3.00	
MATH	0200	Basic Mathematics II	3.00	3.3	9.90	
PSY	1100	General Psychology	3.00	2.3	6.90	
Term: Ehrs: 15.00 GPA-Hrs: 15.00 QPts: 40.20 GPA: 2.68						
Good Standing						
Term: Spring 2010						
Computer Science						
CSC	1550	Computer Science II	3.00	3.0	9.00	
ENGL	2500	World Literature II	3.00	3.0	9.00	
EXSS	1000	Health and Fitness	3.00	1.3	3.90	
MATH	1300	Precalculus	3.00	2.3	6.90	
Term: Ehrs: 12.00 GPA-Hrs: 12.00 QPts: 28.80 GPA: 2.40						
Good Standing						
Term: Fall 2010						
Computer Science						
CSC	1600	Intro to Electronics	4.00	2.5	10.00	
CSC	1900	Discrete Math	3.00	3.3	9.90	
MATH	1700	Applied Statistics	3.00	2.0	6.00	
MATH	2300	Calculus I	4.00	1.7	6.80	

********** CONTINUED ON NEXT COLUMN **********

SUBJ	NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:						
Term: Ehrs: 14.00 GPA-Hrs: 14.00 QPts: 32.70 GPA: 2.34						
Good Standing						
Term: Spring 2011						
Computer Science						
ART	1100	Art Appreciation	3.00	2.5	7.50	
CSC	1650	Digital Electronics	4.00	2.7	10.80	
CSC	2560	Systems Programming	3.00	3.0	9.00	
CSC	3300	Assembly Language	3.00	3.0	9.00	
SOC	1100	Introduction to Sociology	3.00	2.7	8.10	
Term: Ehrs: 16.00 GPA-Hrs: 16.00 QPts: 44.40 GPA: 2.78						
Good Standing						
Term: Fall 2011						
Computer Science						
CSC	2400	Database Systems	3.00	3.5	10.50	
CSC	2600	Computer Organization	4.00	2.7	10.80	
CSC	3500	Object-Oriented Programming	3.00	3.5	10.50	
MATH	2600	Linear Algebra	3.00	1.7	5.10	
SPCH	1000	Intro to Speech Communication	3.00	3.5	10.50	
Term: Ehrs: 16.00 GPA-Hrs: 16.00 QPts: 47.40 GPA: 2.96						
Good Standing						
Term: Spring 2012						
Computer Science						
CSC	3100	Operating Systems	3.00	3.5	10.50	
CSC	3700	Algorithms Data Structure	3.00	4.0	12.00	
CSC	4400	Software Engineering	3.00	4.0	12.00	
MATH	2400	Calculus II	4.00	1.5	6.00	
PHYS	2300	General Physics I	4.00	2.3	9.20	
Term: Ehrs: 17.00 GPA-Hrs: 17.00 QPts: 49.70 GPA: 2.92						
Good Standing						
Term: Fall 2012						
Computer Science						

********** CONTINUED ON PAGE 2 **********

Record of: [redacted]

Date Issued: 24-JUN-2013
Date of Birth: 26-APR-1991
Student ID: [redacted]
Level: Undergraduate

SUBJ NO.	COURSE TITLE	CRED	GRD	PTS	R
Institution Information continued:					
CSC 3050	Web Programming w/Java	3.00	4.0	12.00	
CSC 3200	Programming Languages	3.00	3.3	9.90	
CSC 3400	Data Comm/Networking	3.00	2.0	6.00	
CSC 4903	Ind St: Game Programming	3.00	4.0	12.00	
MATH 4300	Abstract Algebra	3.00	1.5	4.50	
PHYS 2400	General Physics II	4.00	2.5	10.00	
Term: Ehrs: 19.00 GPA-Hrs: 19.00 QPts:		54.40	GPA:	2.86	
Good Standing					
Term: Spring 2013					
Computer Science					
COMM 3305	Interactive Media I	3.00	4.0	12.00	
CSC 2700	Business Programming	3.00	3.3	9.90	
CSC 3600	Microprocessors	4.00	1.0	4.00	
GAME 2000	Elements of Game Design	3.00	3.5	10.50	
GAME 2200	Introduction to Game Art	3.00	3.5	10.50	
Term: Ehrs: 16.00 GPA-Hrs: 16.00 QPts:		46.90	GPA:	2.93	
Good Standing					

********** TRANSCRIPT TOTALS **********

	Earned Hrs	GPA Hrs	Points	GPA
TOTAL INSTITUTION	125.00	125.00	344.50	2.76
TOTAL TRANSFER	0.00	0.00	0.00	0.00
OVERALL	125.00	125.00	344.50	2.76

********** END OF TRANSCRIPT **********

Signature Attesting to Compliance

By signing below, I attest to the following:

That Computer Science (*Name of the program(s)*) has conducted an honest assessment of compliance and has provided a complete and accurate disclosure of timely information regarding compliance with ABET's *Criteria for Accrediting Computing Programs* to include the General Criteria and any applicable Program Criteria, and the ABET *Accreditation Policy and Procedure Manual.*

Paul Weiss

Dean's Name (As indicated on the RFE)

Paul W

Signature

6/25/13

Date