Computer Science Program Review

El Camino College – Fall 2013

Carl Broderick

Greg Fry

Massoud Ghyam

Greg Scott

Ralph Taylor

Table of Contents

I. Overview of the Program

a) Provide a brief narrative description of the current program, including the program's mission statement and the students it serves.

Mission Statement: The Computer Science Department offers quality, comprehensive educational programs and services to ensure the educational success of students from our diverse community.

The Computer Science Department at El Camino College provides an outstanding learning environment in which students can develop the skills and knowledge needed for today's technology-oriented jobs. We offer quality, comprehensive computer science courses to ensure the educational success of students from our diverse community, with an emphasis on preparing students to transfer to STEM-related majors at four year colleges and universities.

We have provided instruction for from 144-169 students each semester since fall 2008. The core course of the CS Department is CSCI 1 (Problem Solving and Program Design in C++), our introductory programming course. This has accounted for from 50-60% of our students each semester.

Our next tier of courses includes CSCI 2 (Intro to Data Structures, CSCI 3 (Java Programming), CSCI 30 (Advanced C++ Programming), and CSCI 40 (Intro to UNIX Operating Systems). Our first hybrid course, CSCI 12 (Programming Using PHP, Javascript and XHTML), was introduced successfully in Spring 2013.

Our overall departmental success rate has increased from a low of 49.1% in Fall 2008 to a high of 71.5% in Spring 2011, and has consistently exceeded the 60% level over the past two years. We are pleased that our retention rates have risen from a low of 63.3% in Fall 2008 to a high of 80% in Spring 2012, and has surpassed 70% over the past three years.

b) Describe the degrees and/or certificates offered by the program.

The department offers the A.S. Degree in Computer Science. We recently got approved to offer the AS-T Model Transfer Curriculum Degree. With the addition of CS 16 (Assembly Language Programming) to the Fall 2014 schedule we should soon be producing AS-T graduates.

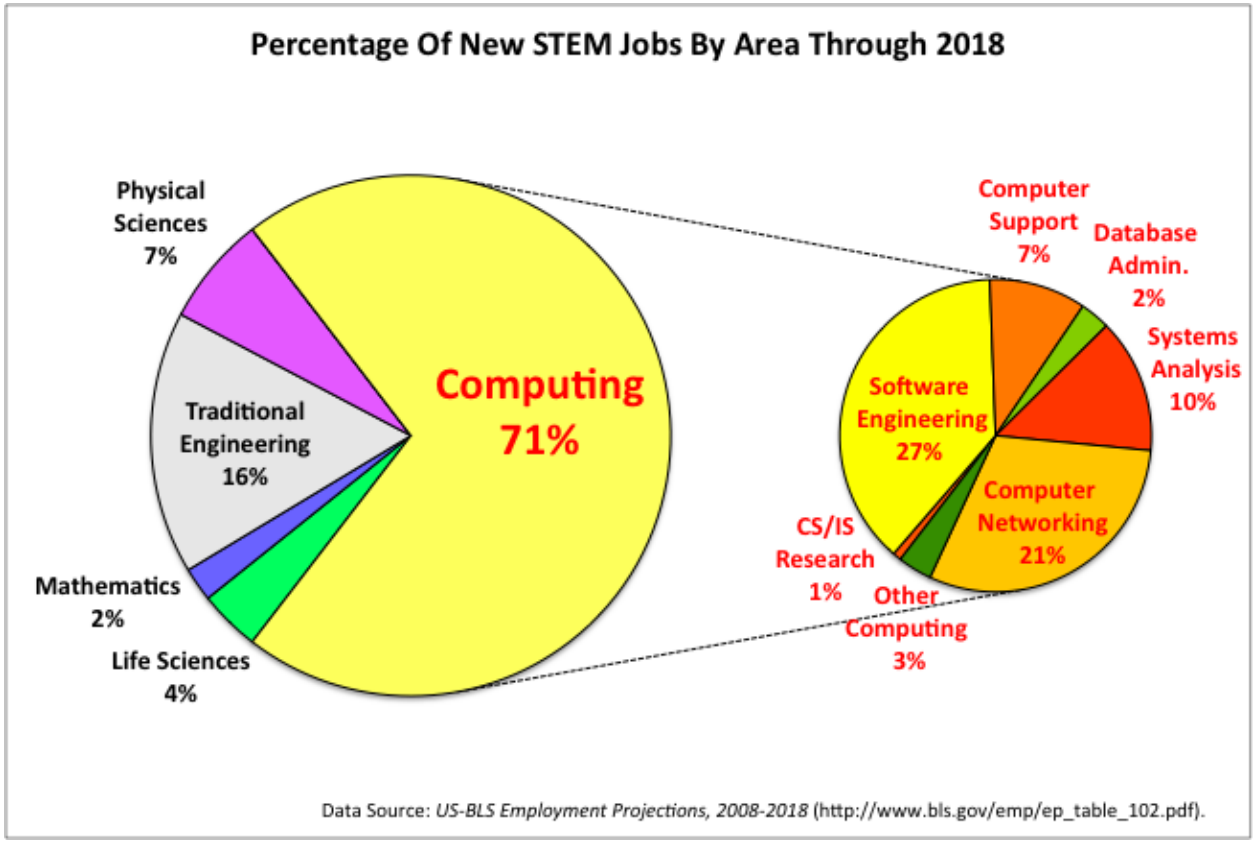
c) Explain how the program fulfills the college's mission and aligns with the strategic initiatives.

Computer Science is a vital field. In the U.S. News and World Report list of the top 100 jobs in 2013, no less than 5 of the top 13 fields are CS related, with Computer Systems Analyst placing 4th, Database Administrator in 6th, Software Developer in 7th, Web Developer in 9th, and

Computer Programmer in 13th. Students who take courses in the Computer Science Department are forming the core of the skills they will need to be successful in all of these fields.

The Bureau of Labor Statistics projects that employment of software developers is projected to grow 30% from 2010 to 2020, much faster than the average for all occupations. The main reason for the rapid growth is a large increase in the demand for computer software. The BLS website states: "Software developers usually have a bachelor's degree, typically in computer science, software engineering, or a related field. A degree in mathematics is also acceptable. Computer science degree programs are the most common, because they tend to cover a broad range of topics. Students should focus on classes related to building software in order to better prepare themselves for work in the occupation." They further state that "Although writing code is not their first priority, developers must have a strong background in computer programming. They usually gain this experience in school."

The graphic below (from Calvin College and based on BLS data) demonstrates how much Computer Science dominates the STEM disciplines.



There was a huge downturn in the Tech industry in the middle of the last decade, caused in large part by the bursting of the DotCom bubble, but the computer industry has recovered, prospered and grown. The ECC Computer Science Department has not been allowed to keep up. In the early part of the last decade the CS Department routinely taught 25 or more sections each semester, but we dropped to 7 sections by Spring 2008. We have hovered between 6 and 8 sections since then, even when demand has returned. This is barely enough to sustain a thriving environment for student learning. For several years, due to budget constraints, we haven't been able to offer the variety of courses that would more solidly prepare students for transfer. Nearly 60% of our offerings have been one course: CSCI-1. The dearth of more advanced offerings

frustrated many students by delaying their ability to transfer and denying them the opportunity to get a more well-rounded Computer Science background.

Our goal in the this program review is to make a case that the Computer Science Department at El Camino College be allowed to grow, that more sections be offered, that new faculty be hired, that support such as MESA workshops be offered, and that technology be routinely updated.

Programming Contest Success

Computer Science Students from El Camino College have participated in Progfest, a city-wide programming contest held at Cal State Los Angeles in February for the past 5 years (2009-2013). This contest has a Junior College and High School division. El Camino College students have placed 1st in the JC division in 4 of the 5 years, placing 2nd only once in 2010.

In fall 2013 a group of El Camino Students entered the Southern California Regional trials for the ACM International Collegiate Programming Contest, held at Riverside College. This contest is primarily for University level students, but junior college contestants are encouraged to participate for the experience. El Camino's team finished 25th out of 75, finishing ahead of teams from local Cal State schools such as CSULB, CSULA, and CSUF. The following link will take you to the results page for the contest. http://www.socalcontest.org/history/2012/results-2012.shtml

Members of the CS faculty get involved with the students in these extracurricular activities by 1) recruiting students via announcements in class, 2) organizing meetings of interested students to disseminate information, 3) organizing practice sessions to prepare students for the contests, and 4) raising funds to defer costs for students to participate.

d) Discuss the status of recommendations from your previous program review.

Status of Previous Recommendations from 2008

Recommendation 1: Develop new courses, or revitalize existing curriculum to cover newer technologies that are of interest to employers and our students**. Status:** CSCI 12, and online hybrid class was created. Budget constraints have limited the introduction of other new courses.

Recommendation 2: Obtain training for existing staff on newer hardware and software platforms that will form the basis of our newer courses. **Status:** No funding and no inspiration due to course offerings being scaled back to being barely on life-support.

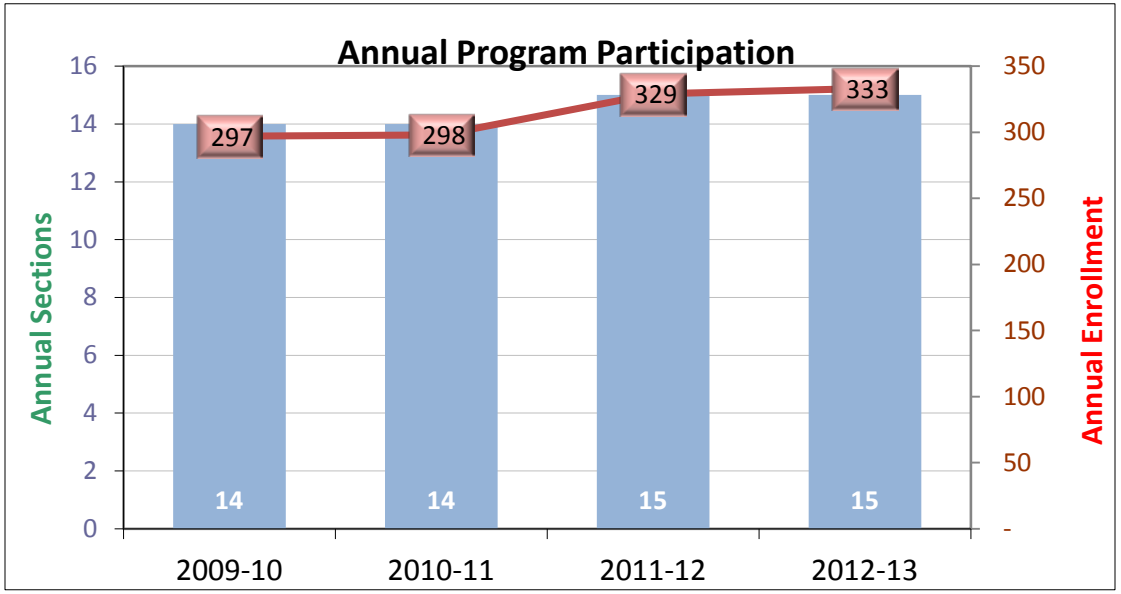
Recommendation 3: Continue to adhere to the three year cycle of upgrading the resources within our computer labs. **Status:** We got a new computer lab when we moved into the MBA building in January 2013, however we lost half of our lab space since we dropped from two labs to one lab.

Recommendation 4: Work to articulate El Camino Colleges Computer Science classes with the CSU, UC and other private universities. **Status:** All active CS courses are transferable and articulate.

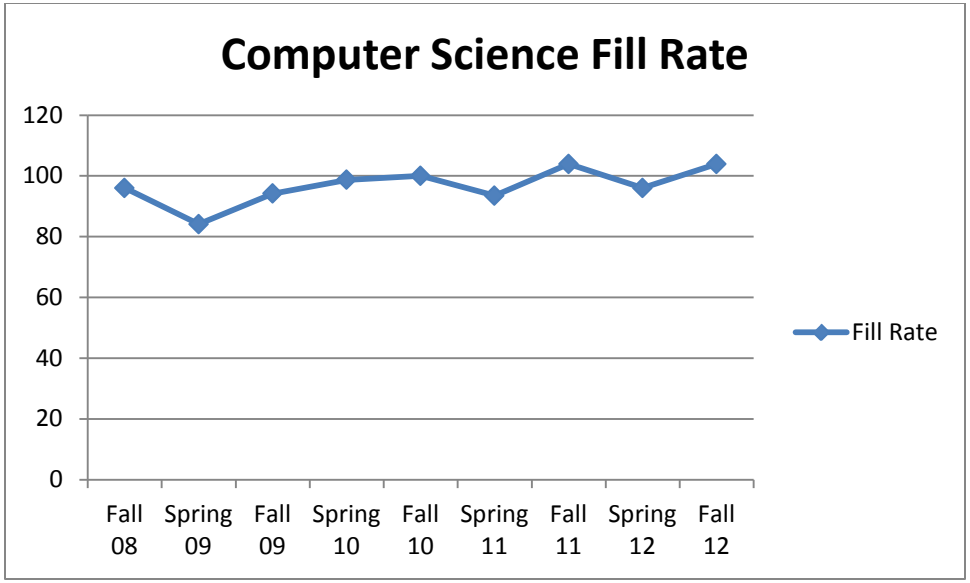
II. Analysis of Institutional Research and Planning Data

Enrollment Rates

The enrollment in Computer Science has held rather steady over the past five year**s.** This is due mostly to the budget constraints of recent years.



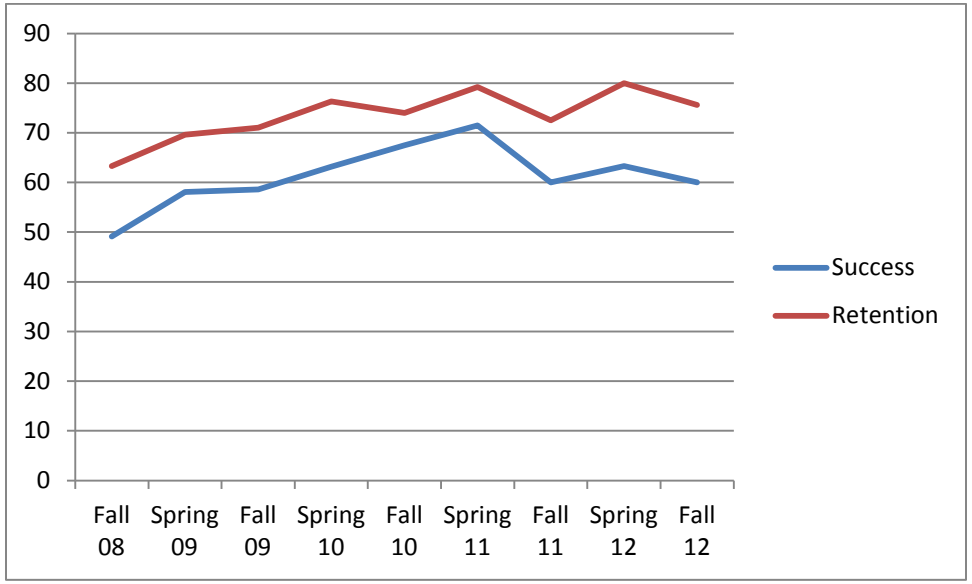
The demand for our courses exceeded our ability to provide them as illustrated by the near 100% fill rate each semester. Additionally, the classes and their wait lists fill up faster than most other courses indicating a very high demand for these core STEM courses.



Program Success and Retention Rates

We are pleased that our success and retention rates have risen from their lows in fall 2008. Since the low success rate of 49.1% in fall 2008 the rate has never been below 58% and has been over 60% every semester since spring 2010. A similar situation holds for the retention rates. From a low of 63.3% in fall 2008, the rates have been over 70% every semester since.

Computer Science Success and Retention Rates: 2008-2012



Data by course and semester can be found in Appendix A

Gender and Ethnicity of the Student Population

The data indicate that males outnumber females in the CSCI program by 7 to 1. This is similar the situation with many STEM fields.

The percentage of Latinos in the program has ranged from 24.8% to 33.1% from 2009 to 2012, which is short of the 44.7% Latino population in the overall ECC student population. The percentage of African-Americans has ranged from 3.8% to 7.1% from 2009 to 2012, which isn't close to the overall student population proportion of 17.0%.

We believe that MESA could do a great job of improving these numbers. The MESA Director, Arturo Hernandez, agrees that we should try to get a MESA workshop for our introductory course, CSCI 1. Later we could attach workshops to key courses like CSCI 2 and CSCI 3.

Fall						ECC Student Population	District Boundary Population
		Term					
		2009	2010	2011	2012	Fall 2012	2010 Census
Term Headcount		141	153	160	159	*23,409*	*556,400*
Gender	F	19.9%	12.4%	12.5%	17.6%	*52.5%*	*51.0%*
	M	80.1%	87.6%	87.5%	81.8%	*47.5%*	*49.0%*
Ethnicity	African-American	7.1%	3.9%	3.8%	5.7%	*17.0%*	*15.1%*
	Amer. Ind. or Alask. Native	0.0%	0.0%	0.0%	0.0%	*0.2%*	*0.2%*
	Asian	38.3%	45.1%	33.8%	43.4%	*16.1%*	*13.6%*
	Latino	24.8%	24.2%	33.1%	26.4%	*44.7%*	*34.5%*
	Pacific Islander	0.0%	1.3%	0.0%	0.6%	*0.5%*	*0.5%*
	White	17.0%	17.0%	15.6%	16.4%	*15.6%*	*32.8%*
	Two or More	0.0%	0.0%	5.6%	3.8%	*3.8%*	*2.9%*
	Unknown or Decline	12.8%	8.5%	8.1%	3.8%	*2.0%*	*0.4%*

Survey Analysis

We administered a survey to 82 of our students in the week before final exams in spring 2013**.** Detailed results can be found in Appendix B.

It is gratifying to see that all our students want to get a Bachelor's Degree, and more than half want a higher degree, with 54.9% indicating that they want to pursue a Masters or a Doctorate. The vast majority want to do this at a UC or CSU campus, and a whopping 70.7% want to major in Computer Science or Computer Engineering.

Around 56% of our students sought the help of our one computer science tutor at least once a week. We hope that we can increase the hours for our tutor and also work with MESA to get workshops for CSCI 1, 2 and 3.

The survey indicates that there is great demand for several of our more neglected courses, neglected because we haven't been able to offer them very often in recent years. This includes CSCI-40, Unix Programming, and CSCI-15P (now CSCI-16), Assembly Language Programming. There is also a plea for us to reactivate dormant courses relating to graphics programming and web programming. Finally, there is a desire for us to offer game programming, programming for mobile devices, and recently popular languages like Python.

III. Curriculum

The Computer Science Department currently offers a very small number of courses compared to the offerings ten years ago. Most of the courses in the inactivated list are still viable in a robust STEM curriculum; we have simply not been allowed to reactivate them due to lack of funding and attention from administration. Other courses could be created that could be equally attractive the highly motivated Computer Science student. We want the department to grow. We need to increase the offerings of the basic courses first: CSCI 1, 2, and 3. Then we need to give the students other courses to aspire to. All CSCI courses are transferable and all CSCI students take many other STEM courses in other departments. These are hungry students – we can only help them achieve their goal of transferring or finding a great programming job if we offer the variety of coursework that will make them attractive to employers and college admissions officers.

Current Computer Science Courses

CSCI 1 – Intro to Problem Solving and Programming Using C++
CSCI 2 – Intro to Data Structures in C++
CSCI 3 – Intro to Java
CSCI 12 – Prog Using PHP, JS, XHTML
CSCI 16 – Assembly Language Programming (reactivated for CS-Transfer Degree)
CSCI 30 – Advanced Programming in C++
CSCI 40 – Intro to UNIX Programming

Course Review Schedule

Course	Last Course Review	Next Course Review
CSCI 1	**Fall 2012**	**Spring 2018**
CSCI 2	**Fall 2012**	**Fall 2017**
CSCI 3	**Spring 2009**	**Spring 2014**
CSCI 12	**Fall 2010**	**Spring 2015**
CSCI 16	**Spring 2013**	**Fall 2018**
CSCI 30	**Spring 2009**	**Spring 2016**
CSCI 40	**Fall 2010**	**Fall 2016**

Inactivated Computer Science Courses:

CSCI 4 – Intro To C#
CSCI 10 – Intro to FORTRAN
CSCI 23 – Advanced Java
CSCI 25 – Programming in C
CSCI 36 - .NETs Windows Forms for GUI CS Programming
CSCI 55 – Three Dimensional Computer Graphics Programming
CSCI 60 – Programming with ASP.NET and C# in Web Based CS Application

IV. Assessments and Student Learning Outcomes (SLOs)

The computer science department has assessed 100% of the course level SLOs. We are at the Proficiency stage and we are working towards the Sustainable Continuous Quality Improvement stage.

Program Level SLOs

Upon completion of their course of study in the Computer Science Department:

PLO #1 Students, when given a specification for a program or program segment, will be able to design, code, compile, test and document a solution;

PLO #2 Students, when given a code segment, will be able to trace the execution and give the output;

PLO #3 Students, when given a code segment with errors, will be able to identify and correct the problems;

PLO #4 Students will be able to explain concepts specific to a particular language.

Course Level SLOs

Each SLO matches up with the corresponding PLO with the same number.

CSCI 1 Problem Solving and Program Design Using C++:

SLO #1 Students will write correct and detailed algorithms. (Properly analyze a problem using top down design, and write an algorithm that can be translated into computer code).

SLO #2 Students will write C++ code that uses correct syntax. (When declaring data types, writing algebraic and logical expressions, naming variables, etc.)

SLO #3. Students will write C++ code that correctly uses control structures (and nested control structures) [Including conditionals (like "if"), loops (like "while" and "for") and user defined functions (both void and value returning)]

SLO #4. Students will write C++ code that correctly uses basic data structures (Including strings, arrays, and structs)

CSCI 2 Introduction to Data Structures:

SLO #1 Students will design, code, compile, test and document a programming solution to a problem involving the basic data structures: lists, stacks, queues, trees, and related abstract data types.

SLO #2 Students, when given a C++ code segment, will be able to trace the execution, give the output, and analyze the efficiency of the basic data structures and techniques involved.

SLO #3. Students, when given a C++ code segment with errors, will be able to identify and correct the problems.

SLO #4. Students will be able to explain the C++ concepts related to pointers, classes, recursion, searching, sorting, templates and dynamic memory allocation.

CSCI 3 Computer Programming with Java:

SLO #1 Students, when given a specification for a program or program segment, will be able to design, code, compile, test and document a solution.

SLO #2 When given a code segment involving control structures, iteration structures and method calls, students will be able to trace the execution and give the output.

SLO #3 When given a code segment involving data objects, graphical interface objects, and processing objects, students will be able to describe what the users sees and the events that take place as the user interacts with the interface.

SLO #4 Students understand and be able to develop both desktop and web applications involving graphical user interfaces, animations, sound manipulations, File I/O, database, applets, and threads.

CSCI 12 Internet Applications using PHP, JavaScript, and XHTML:

SLO #1 Interactive Web Pages Design and implement an interactive web page.

SLO #2 Side-Scripts Develop a client-side script to create a drop down menu for a web page.

SLO #3 Web Pages Date and Time Design and implement a program or function to process data collected from a web form.

SLO #4 Processing Web Data Design and implement a server-side program or function to a database and interact (insert, delete, update records) with it.

CSCI 16 Assembly Language Programming for the IBM PC and Compatibles

SLO #1 Students will design, code, compile, test and document programming solutions to problems by developing PC assembly language code that makes direct use of processor instructions, interrupts, registers, the stack, as well as existing macro and procedure libraries.

SLO #2 Students, when given a code segment will be able to trace the execution, providing the real-time content of registers during operations, the dynamic content of the stack during procedure calls and returns, and tracing the conditional execution of code generally, and within looping structures specifically.

SLO #3 Students, when given PC assembly language code with errors, will be able to identify what those errors are and will be able to modify the PC assembly language code to eliminate those errors.

SLO #4 Students will be able to explain the concepts of PC assembly language registers, interrupts, data segment organization, addressing modes, internal data representation, decision structures, macros and procedures.

CSCI 30 Advanced Programming in C++:

SLO #1 Students will design, code, compile, test and document programming solutions to problems requiring the development of C++ classes (by inheritance, by composition; templates), requiring C++ operator overloading, requiring effective use of the Standard Template Library, requiring effective use of pointers and dynamic memory allocation.

SLO #2 Students, when given a code segment involving use of a class, will be able to trace the construction of class objects, trace the destruction of class objects, verify whether memory leaks have occurred, trace object assignment operations, verify when copy constructors are invoked and when overloading of copy constructors is required

SLO #3 Students, when given C++ code with errors, will be able to identify what those errors are and will be able to modify the C++ code to eliminate those errors.

SLO #4 Students will be able to explain the concept of C++ class templates and how they relate to the concept of generics, the concept of virtual functions and polymorphism, the concept of multiple inheritance and virtual base classes, the concept of container types and the circumstances where specific containers should or should not be used.

CSCI 40 Introduction to UNIX and LINUX Operating Systems:

SLO #1. Given a specification for a set of operating system tasks, students will create, edit, move, display, copy and delete files and subdirectories.

SLO #2. *Students* use shell programming to create file processing applications and control user interaction.

SLO #3 Students create, schedule, filter, monitor, format, sort and redirect and delete input / output of programs and processes.

SLO #4 Students perform basic administration functions in system installation and maintenance, network services, user

Timeline for the four-year cycle for course and program level SLO assessments

Computer Science – 4 Year Cycle Plan

COURSES IN PROGRAM:

CSci 1 Problem Solving and Program Design using C++
CSci 2 Introduction to Data Structures
CSci 3 Computer Programming in Java
CSci 12 Internet Applications using PHP, JavaScript, and XHTML
CSci 16 Assembly Language Programming for the IBM PC and Compatibles
CSci 30 Advanced Programming in C++
CSci 40 Introduction to UNIX Operating Systems

SLO Timeline:

During the past 4 years all Computer Science department SLO courses have been assessed for all offered sections. Computer Science department has completed 100% of assessments. Based on the assessments and student survey results CS department started a tutoring program. Based on the SLO assessments and their alignments with PLO are also assessed indirectly.

Next 4 years timeline:

Academic Year 2014		Academic Year 2015		Academic Year 2016		Academic Year 2017	
SLO/PLO #1		SLO/PLO #2		SLO/PLO #3		SLO/PLO #4	
S2014	F2014	S2015	F2015	S2016	F2016	S2017	F2017
						Year of Program Review	
CSCI 12 CSCI 30 CSCI 40	CSCI 1 CSCI 2 CSCI 3 CSCI 16	CSCI 12 CSCI 30 CSCI 40	CSCI 1 CSCI 2 CSCI 3 CSCI 16	CSCI 12 CSCI 30 CSCI 40	CSCI 1 CSCI 2 CSCI 3 CSCI 16	CSCI 12 CSCI 30 CSCI 40	CSCI 1 CSCI 2 CSCI 3 CSCI 16

Assessment results and recommended/implemented changes resulting from course and program level SLO assessment

During this program review cycle, all offered courses were assessed, and the assessments were conducted in spring 2009, fall 2010, spring 2011, Fall 2011, Spring 2012, Fall 2012, and Spring 2013. Overall, the results met expectations.

Program SLOs and manner of assessment

The CS program level SLO's have not yet been assessed. Now that the department has a clearly outlined assessment plan, the program can be accurately assessed once more courses are evaluated. The CS program level SLO's are scheduled for assessment in spring 2016 and will be based on raw data collected during academic years 2012 through 2016. With more data collected, the program level SLO's will better measure the success rates, average grades, success/retention rates (with and without grade distribution), and demographics.

Program's level of SLO/assessment implementation: Awareness; Development; Proficiency; or Sustainable Continuous Quality Improvement

The CS programs are at the proficiency level defined in its SLO implementation:

- SLOs and assessments are in place for all of its courses, as well as at the program level.
- Results of assessments have been used to guide the department in improving its courses and program.
- There is department-wide dialogue about the results of assessment.
- Decision-making is purposefully directed towards improving student learning based on the results of assessment.

Students are aware of these student learning outcomes because they are on course syllabi.

Alignment grid for courses, program and institution.

Mathematical Sciences Institutional (ILO), Program (PLO), and Course (SLO) Alignment												
Program: Computer Science			**Number of Courses: 6**		**Date Updated 1.26.13**	**Submitted by** Junko Forbes Ext. 7217						
Institutional SLOs	I. Content Knowledge	II. Critical, Creative, and Analytical Thinking		III. Communication and Comprehension	IV. Professional and Personal Growth	V. Community and Collaboration	VI. Information and Technology Literacy					
Program Rating	4	4		3	2	2	3					
Program Level SLOS							**ILOs to PLOs Alignment** (Rate 1-4)					
							I	II	III	IV	V	VI
PLO #1 Using Specifications: Upon completion of their course of study in the Computer Science Department, students, when given a specification for a program or program segment, will be able to design, code, compile, test and document a solution.							4	4	3	2	2	3
PLO #2 Tracing Execution Upon completion of their course of study in the Computer Science Department, students, when given a code segment, will be able to trace the execution and give the output.							4	4	2	2	2	3
PLO #3 Identifying and Correcting Problems Upon completion of their course of study in the Computer Science Department, students, when given a code segment with errors, will be able to identify and correct the problems.							4	4	2	2	2	3
PLO #4 Concepts of Computer Language Upon completion of their course of study in the Computer Science Department, students will be able to explain concepts specific to a particular language.							2	2	4	2	2	3

Course Level SLOs	Course to Program SLO Alignment Mark with an X				ILOs to Course SLOs Alignment (Rate 1-4)					
	P1	P2	P3	P4	I	II	III	IV	V	VI
CSCI 1 Problem Solving and Program Design Using C++:SLO #1 Writing Algorithms Student will write correct and detailed algorithms. (Properly analyze a problem using top down design, and write an algorithm that can be translated into computer code)	X			X	4	4	3	2	2	3
CSCI 1 Problem Solving and Program Design Using C++:SLO #2 Using Correct Syntax Student will write C++ code that uses correct syntax. (When declaring data types, writing algebraic and logical expressions, naming variables, etc.)	X		X	X	4	4	3	2	2	3
CSCI 1 Problem Solving and Program Design Using C++:SLO #3. Input and Output Information Student will write C++ code that correctly input and output information. (Moving data to and from the screen and to and from text files. Also inputting predefined functions, etc, from included libraries.)	X	X		X	4	4	3	2	2	3
CSCI 1 Problem Solving and Program Design Using C++:SLO #4. Nested Control Structures Student will write C++ code that correctly uses control structures (and nested control structures) [Including conditionals (like "if"), loops (like "while" and "for") and user defined functions (both void and value returning)]	X	X		X	4	4	3	2	2	3
CSCI 1 Problem Solving and Program Design Using C++:SLO #5. Basic Data Structures Student will write C++ code that correctly uses basic data structures (Including strings, arrays, and structs)	X	X			4	4	3	2	2	3
CSCI 1 Problem Solving and Program Design Using C++:SLO #6. Debugging and Running Programs Student will use software environment to debug and run C++ programs and to test them using various inputs.		X			4	4	3	2	2	3

CSCI 2 Introduction to Data Structures: SLO #1 Programming Solutions Students will design, code, compile, test and document a programming solution to a specified problem requiring basic data structures	X	X			4	4	3	2	2	3
CSCI 2 Introduction to Data Structures: SLO #2 Output of Program Segments Students will trace the execution and give the output of a given program or program segment pertaining to data structures		X			4	4	3	2	2	3
CSCI 2 Introduction to Data Structures: SLO #3. Correcting Errors Students will identify and correct the errors in a given program or program segment pertaining to data structures.		X		4	4	3	2	2	3	
CSCI 2 Introduction to Data Structures: SLO #4.Basic Data Structures Students will implement and explain the concepts underlying the basic data structures: lists, stacks, queues, trees, and related abstract data types.				X	4	4	3	2	2	3
CSCI 2 Introduction to Data Structures: SLO #5. Data Structure Techniques Students will explain and implement basic data structure techniques: pointers, classes, recursion, searching, sorting, templates and dynamic memory allocation.	X			X	4	4	3	2	2	3
CSCI 2 Introduction to Data Structures: SLO #6. Efficiency of Basic Data Students will analyze the efficiency of the basic data structures and techniques.	X	X			4	4	3	2	2	3
CSCI 3 Computer Programming with Java: SLO #1 Designing, Coding, Compiling and Testing Students, when given a specification for a program or program segment, will be able to design, code, compile, test and document a solution.	X				4	4	3	2	2	3
CSCI 3 Computer Programming with Java: SLO #2 Tracing Execution When given a code segment involving control structures, iteration structures and method calls, will be able to trace the execution and give the output.		X			4	4	3	2	2	3
CSCI 3 Computer Programming with Java: SLO #3 Describing What Users See When given a code segment involving data objects, graphical interface objects, and processing objects, will be able to describe what the users sees and the events that take place as the user interacts with the interface.		X			4	4	3	2	2	3

CSCI 3 Computer Programming with Java: SLO #4 Using File Objects and Interfaces Understand and be able to correctly use the File input/output objects and interfaces.			X	X	4	4	3	2	2	3
CSCI 3 Computer Programming with Java: SLO #5 Java Applications Understand and be able to develop Java applications involving graphical animations and sound manipulations.			X	X	4	4	3	2	2	3
CSCI 3 Computer Programming with Java: SLO #6 Web Applications Understand and be able to develop web applications involving applets with threads.			X	X	4	4	3	2	2	3
CSCI 12 Internet Applications using PHP, JavaScript, and XHTML: SLO #1 Interactive Web Pages Design and implement an interactive web page.	X			X	4	4	3	2	2	3
CSCI 12 Internet Applications using PHP, JavaScript, and XHTML: SLO #2 Side-Scripts Develop a client-side script to create a drop down menu for a web page.	X				4	4	3	2	2	3
CSCI 12 Internet Applications using PHP, JavaScript, and XHTML: SLO #3 Web Pages Date and Time Write a PHP program segment to display the current date and time on a web page.	X			X	4	4	3	2	2	3
CSCI 12 Internet Applications using PHP, JavaScript, and XHTML: SLO #4 Processing Web Data Design and implement a program to process data gathered from a web form.	X				4	4	3	2	2	3
CSCI 12 Internet Applications using PHP, JavaScript, and XHTML: SLO #5 Cookies Design and implement a program to define and manage cookies.	X				4	4	3	2	2	3
CSCI 12 Internet Applications using PHP, JavaScript, and XHTML: SLO #6 Server Side Program Design and implement a server side program to connect to a database and interact with it.	X			X	4	4	3	2	2	3
CSCI 30 Advanced Programming in C++: SLO #1 Not User Member Initialization Students should be able to		X	X		4	4	3	2	2	3

follow the consequences of not using the member initialization list in class constructors.										
CSCI 30 Advanced Programming in C++: SLO #2 Data Abstraction Develop C++ classes that show an understanding of data abstraction.	X		X		4	4	3	2	2	3
CSCI 30 Advanced Programming in C++: SLO #3 Virtual Functions Develop C++ classes that use friend functions, and that contain virtual functions.	X		X	X	4	4	3	2	2	3
CSCI 30 Advanced Programming in C++: SLO #4 Inheritance and/or Compositions Develop new C++ classes based on existing classes by inheritance, by composition, or by a combination of both inheritance and composition.	X		X		4	4	3	2	2	3
CSCI 30 Advanced Programming in C++: SLO #5 Virtual Base Classes Develop C++ classes derived from multiple existing classes, and where applicable/necessary implement virtual base classes.	X		X	X	4	4	3	2	2	3
CSCI 30 Advanced Programming in C++: SLO #6 Standard Template Library Understand and effectively use the Standard Template Library.	X			X	4	4	3	2	2	3
CSCI 30 Advanced Programming in C++: SLO #7 Functions and Classes Develop C++ template functions and template classes.	X			X	4	4	3	2	2	3
CSCI 40 Introduction to UNIX and LINUX Operating Systems: SLO #1. Shell Script Solutions Given a specification for a set of operating system tasks, the student will design, code, test and document a shell script solution.	X				4	4	3	2	2	3
CSCI 40 Introduction to UNIX and LINUX Operating Systems: SLO #2. Creating Networks Understand and be able to create TCP/IP network addressing schemes for subnets, gateways, DHCP and DNS servers.				X	4	4	3	2	2	3
CSCI 40 Introduction to UNIX and LINUX Operating Systems SLO #3 Shell Program Segment When given a shell program segment, will be able to trace the execution and give the output.	X				4	4	3	2	1	3
CSCI 40 Introduction to UNIX and LINUX Operating Systems SLO #4 Identifying and Correcting Problems When given a shell program with errors, will be able to identify and correct the problems.		X			4	4	3	2	1	3

CSCI 40 Introduction to UNIX and LINUX Operating Systems: SLO #5. Process Monitoring and Scheduling Understand and be able to use process monitoring and scheduling			X		4	4	3	2	1	3
CSCI 40 Introduction to UNIX and LINUX Operating Systems: SLO #6. Installation Understand and be able to administer operating system installation, user installation, hardware and software installation, system maintenance and system services.			X		4	4	3	2	1	3

V. Facilities, Equipment and Technology

A: Facilities and Equipment

Computer Lab

In the previous math building, we had two computer labs dedicated to computer science courses. In our current building, we now have only one computer lab dedicated to computer science courses. This is already causing scheduling problems, which will only get worse as we continue to grow. In order to cope with this problem, it is recommended that we purchase a sufficient number of laptops to effectively add two labs, without any construction costs. The laptops could be signed out to students at the beginning of a lab session, and returned at the end. Given this, any classroom could effectively be "converted" to a computer lab. Since the number of students in any given lab session has been capped at 22, we would need to purchase at least 44 laptops, and make sure that any relevant software site licenses would apply to them. Also, we would need to purchase at least 2 wireless printers for the rooms. More details on these items follow.

B. Technology

Computers

Student Computers

Dedicated Lab: Our dedicated lab (MBA 113) will need upgrading to current technology levels periodically in order provide the platform to train our students using the industry standard tools. Since Visual Studio has updated to major revisions every two years (2006, 2008, 2010, 2012) and has just moved to the 2013 version, we need to have 24 desktops(22 in

servcice+2 spares) equipped to the handle the latest technology requirements for development using Visual Studio. The computers in our current lab are capable of supporting the 2013 version, but only have 2010 installed. Cost: approx. $1000 ea. x 24 =$24,000

Mobile Labs: We need to have two "mobile" computer labs that move in and out of general classrooms, requiring 48 laptops (44 in servcice+4 spares) equipped to the handle the latest technology requirements for development using Visual Studio. Cost: approx. $1200 ea x 48 = $57,600

Instructor Computers

It is recommended that instructor computers be replaced at least every three years (and possibly earlier) in order to keep up with the technology changes related to the courses we teach. It is also recommended that faculty be given an additional monitor for their office so that the problem of small laptop screens with the associated eye strain would be alleviated. We currently have 6 instructor positions (5 current + 1 opening to be filled by fall 2014) needing this support. Cost: approx. $2000 ea x 6 = $12,000

Printers Two wireless, portable printers for the mobile labs. Cost approx $300 ea. x 2 = $600

Final Analysis:

After interviewing IT decision makers at this school (John Wagstaff) and other schools, we feel that a 3-4 year plan is all that is feasible right now. The industry trend is to move computer software and computing power to the Cloud computing model. So the items outlined above are expected to last for up to 4 years. After 4 years, we would need to re-evaluate our position relative to the industry trends.

So, our bottom line will be approximately $94,200 ($24,000 + $57,600 + $12,000 + $600 = $94,200).

VI. Technology and Software

Software

Programming:

We need the latest version of Visual Studio which will soon be 2014. Cost for VS Professional with MSDN is approximately $4700 ($800 (1st copy) + 50 per machine site license = 50*78 = 3900). Note that the licensing fee would be yearly.

Specifically related to CSCI 1, CSCI 2, and CSCI 30, Visual Studio allows software development in the C++ programming language.

Specifically related to CSCI 16, Visual Studio allows software development in assembly language.

Related to courses we have offered in the past, and potentially may offer again in the future, Visual Studio allows software development in the C# programming language, software development implementing graphical user interfaces generally, and graphical user interfaces that specifically may be implemented and accessed through the internet.

It should be noted that for the spring 2014 term, there are features of the most recent version of C++ (C++ 11) that we are unable to investigate for our students (specifically anyone teaching CSCI 30, which is our advanced C++ course) because Visual Studio 2010, the current version in the computer lab, and also in the classrooms, does not support those features.

Unix/Linux Operating Systems

To provide a productive Unix/Linux environment (alternative operating systems) we would need a webhosting site to allow students to have both on and off campus access to their accounts. Our IT dept has vetoed allowing students to login to accounts running on campus hosts for security reasons. The cost for such a site would be $10-$15 per month. ($180 per year)

For on campus access, currently we have two Dell OptiPlex 260 desktop to run as Linux servers. These computers were purchased in 2003 and are in need of replacing. The current format of the class is to have the students login in to a Linux server from a Windows machine. This gives them user level interaction and rights only. They get to install and configure the other server as the administrator only for one lab period, passing the computer to the next student so that the process can be repeated until every student has had the opportunity to install and configure a server.

The ideal environment to learn Unix/Linux is to allow each student to have their own server assigned to them so they can gain more time and experience in the role of the administrator. This requires they boot up their own server, either from an existing hard-drive, an external hard-drive, or a flash drive installation (which is usually a minimal installation due to space). I believe the best option is the external hard-drive which provides sufficient storage for the server installation and the portability allowing the class to move to different rooms. It may be in MBA113 were there are desktops one term. The next term it might be in 120 (like this semester) where there are laptops.

What the class will need is

Component	Number	Price(approx.)	Extension
Dell Inspiron 3000 Wrksta./Monitor/KB/Mouse	2	600	1200
Seagate External HDrv /500GB USB3.0	24	60	1440
Storage Cabinet, rolling, lockable(for hard-drives)			100
		Total	2740

Software Training for Course Development

It is recommended that budgeting be made available to allow faculty to explore newer technologies in order to meet student demand for courses that allow software development for those newer technologies. An example would be the purchase of iPads, at a cost of $600 to $800 each (total $3,600 to $4,800), for faculty with the intent of being able to fully explore the development of apps for such technology, and finally the ability to offer a course in that software development to our students. It should be noted that it is one thing to have a concept of an application, to have mapped out the development of software related to that concept, and it is another thing entirely to actually provide a real world implementation of that concept. Without the hardware to be able to test the software, no one is able to test the validity or real usefulness of the software.

In summary, the total here will be approximately $22,180 ($4,700 * 3 + $180 * 3 + $ 2740 + $4,800 = $22,180), assuming a three year period.

VII. Staffing

a) Describe the program's current staffing, including faculty, administration, and classified staff.

Currently there is only one full-time Computer Science faculty, and four Math/CS hybrid faculty members, considered part-time. The last time the college hired a full-time Math/CS hybrid was 2002. The last time college hired a full-time computer science faculty was over thirty years ago. There have been several retirements since then, and demand for courses and the improving economy necessitate the hiring of more faculty members capable of teaching a wide spectrum of the current CS curriculum. Also, new faculty could help develop new courses to help us meet the demands of the next generation of computer science students, who need to be exposed to a wide variety of programming languages and ideas, both old and new.

Faculty	CS Load	
Massoud Ghyam	**100%**	
Greg Scott	**50%**	
Greg Fry	**50%**	
Ralph Taylor	**50%**	
Carl Broderick	**50%**	

Fall 2013 - Full-Time Faculty FTEF: 1.0 Percentage of Program: 30%
Part-Time FTEF: 2.3 Percentage of Program: 70%

b) Explain and justify the program's staffing needs in the immediate (1-2 years) and long-term (2-4+ years). Provide cost estimates and explain how the position/s will help the program better meet its goals.

The Computer Science Department is poised to grow if given the proper tools, one of the most important of which is a group of full-time faculty dedicated to the success of the department.

An increase of full-time faculty in the department will provide the impetus needed to modernize the curriculum at El Camino College, to make it a place where students will come from far away to prepare for transfer and career. The infusion of new blood will help to build courses built on new languages and new technology, such as a games programming curriculum.

Staffing Recommendation #1: The college should hire two new full-time computer science faculty in 2014.

Staffing Recommendation #2: The college should hire three new part-time computer science faculty in 2014, to build a pool of qualified instructors and allow the course and section offerings to grow without uncertainty about staffing (it is better for students to see who is teaching a CS class, rather than to see STAFF, so that they can make an informed decision about which classes to take).

VIII. Direction and Vision

We don't want to be alarmist when we say that the computer department at El Camino College is dying due to lack of funding. However, it's a shame that our funding has been so severely limited at the same time that we see so many bright, hard-working students, and during a period during which El Camino- trained CS students have done so well in programming contests.

Over ten years ago El Camino College offered 25 sections of computer programming. This semester we are offering we are offering just 7 sections, 5 of which are in a single language, C++.

The ECC computer science information page offers a stark picture:
http://www.elcamino.edu/academics/mathsciences/CompSciDeptInformation.asp

The first paragraph of the description of our computer science program lists these courses: Unix, Pascal, Fortran, C, C++, and assembly language. But 3 of these six courses have been permanently discontinued at ECC: Pascal, C and Fortran. Of the remaining 3, one has not been taught for many years: Assembly language. In fact only one of the 6 courses we "advertise" was offered this semester at ECC.

This is a sign of a dying computer science program. What is sad is that we are turning away students that want to be educated. For several years there has been enough demand to fill more sections if they had been opened. It is economics that is shutting us down.

Students are learning that they cannot count on ECC for an education in computer programming … I have had to tell some bright students in C++ (the first computer programming course) that they will be lucky if they can get into CSCI 2 without being turned away since only one or 2 sections being offered and those sections fill fast. Thirty-eight percent of our students polled in our two lowest CS courses, those who had managed to get into a class at all, indicated that they had been turned away previously due to full classes. (This is a strong indicator that many students in 2 or 3 attempts did not manage to get into any CS class). And these students are supposed to go on to complete a certificate at ECC? They sometimes have to wait a semester or a year for a course they need to be offered. More and more students beginning at ECC are finding difficulty completing their computer certificate here. The certificate requires: CS1, CS2 and three courses from among these twelve: Math 210, CS3, 4, 10, 15P, 23, 25, 30, 36, 40, 55 and 60, almost none of which are offered at ECC. (CS 3, 30 and 40 are offered…. And Math 210, because the math department is funded)

A primary goal at ECC is to get students to transfer to 4 year colleges. There likely is no group anywhere on campus that has a greater potential of doing this than students enrolling in our computer science courses. 98% of them plan to transfer to a 4 year college. They have had trigonometry (the prerequisite for CS1) so they have already made huge progress toward transferring before they enroll in a CS class. Seventy-five percent of them are majoring in computer science or computer engineering or gaming. We are slowing them down due to the inability to fund and offer the courses they need.

The once thriving CS department has had its sections hacked to to a mere 7 per semester, down from highs of over 25 per semester. That is 28% of what it was… in spite of the fact that we continue to turn out quality students and have great demand for more CS courses…and in spite of the fact that CS is one of the most-in-demand job opportunities anywhere.

The math department this semester is offering 210 sections of math, not 7. And though their cuts have been fierce, they pale beside the 72% cut made to our once-prosperous computer science department. A 72% cut in the math department would reduce their current 210 sections to 59 sections (a cut of 151 sections).

Our students' trust cannot be regained in a single semester. They will need to relearn that once they begin work on a CS certificate here at ECC, they will be able to complete it here without difficulties? We are losing students to other colleges that are willing to offer them what they need. Why shouldn't El Camino be a leader in community college computer science education?

<u>Changes in the Field</u>

1. Jobs are increasing faster in Computer Sciences. This makes a need for the program to grow.

2. The AS-T degree is a change
    a. One of the recent changes in computer science at the community college level is the AS-T degree. Community colleges which comply with its courses can award their students a computer science degree that is uniform from college to college.

3. The smart phone is a change. It is a computer in disguise. Our computer program department in the future must address apps for cell phones.

4. Money is a change. There is more money now for computer science courses. Our department has suffered severe budget cutbacks that have not allowed us offer the courses our students need in a timely manner

5. The commonness of Laptops is a change: We are very close to the point where every student will soon own their own lap-top or ipad-type device that they can bring with them to school.

Vision and Plan

1. Prepare students better for transferring. The plan: Offer students the Computer Science courses they need. (Perhaps no group of student is more on-the-verge-of-transferring than CS students.)
2. Institute the AST degree at ECC. We have reactivated Assembly language (CS15) as Computer Science 16 toward this goal. This will fulfill the computer architecture requirement. Once this is done we should be in conformity with the degree.
3. We plan to finish achieving this goal by offering CS 16 in the Fall of 2014.
4. Students that have taken the proper courses will be able to write Apps for the cell phone. We have already begun working on this, and plan to give instruction on apps as applicable.
5. Be able to be relied on by our students to offer the courses they need. Gradually increase the number of sections, as funds allow, until we are meeting the needs of our students.
6. Open another computer lab, allowing the program to strengthen and expand (and to make up for losing a lab when we moved to the new building. Supply the laptops to the students, or let them bring their own. In this way, any room can be a computer lab. We hope laptops for students to use will be funded.

IX. Prioritized Recommendations

Status of Previous Recommendations from 2008

Recommendation 1: Develop new courses, or revitalize existing curriculum to cover newer technologies that are of interest to employers and our students**. Status:** CSCI 12, and online hybrid class was created. Budget constraints have limited the introduction of other new courses.

Recommendation 2: Obtain training for existing staff on newer hardware and software platforms that will form the basis of our newer courses. **Status:** No funding and no inspiration due to course offerings being scaled back to being barely on life-support.

Recommendation 3: Continue to adhere to the three year cycle of upgrading the resources within our computer labs. **Status:** We got a new computer lab when we moved into the MBA building in January 2013, however we lost half of our lab space since we dropped from two labs to one lab.

Recommendation 4: Work to articulate El Camino Colleges Computer Science classes with the CSU, UC and other private universities. **Status:** All active CS courses are transferable and articulate.

Prioritized Recommendations for 2013 and Beyond

Recommendation 2013-A: (Increase Course Offerings) It is recommended that we increase our course offerings to keep up with demand. We should increase our offerings of CSCI-1 to 6 sections each semester. We should have 2 sections of CSCI-2 and CSCI-3 each semester. Sections of later courses should be offered more often, some courses should be reactivated, and the development of new courses focusing on new trends in programming be supported.

Recommendation 2013-B: (Faculty Hiring) It is recommended that a full-time Computer Science faculty and/or a hybrid Math/CS faculty be hired. Also, a search for part-time faculty capable of teaching the CS curriculum is desirable.

Recommendation 2013-C: (MESA Workshops and Tutoring) It is recommended that a MESA workshop and facilitator be created for CSCI-1 for the Spring 2014 semester. MESA workshops for CSCI-2 and CSCI-3 would also be desirable by Fall 2014. Fund more hours for the Computer Science tutor.

Recommendation 2013-D: (Technology)

1) We need the latest version of Visual Studio which will soon be 2014. Cost for VS Professional with MSDN is approximately $4700 ($800 (1st copy) + 50 per machine site license = 50*78 = 3900). Note that the licensing fee would be yearly.

2) To provide a productive Unix/Linux environment (alternative operating systems) we would need a webhosting site to allow students to have both on and off campus access to their accounts. Our IT dept has vetoed allowing students to login to accounts running on campus hosts for security reasons. The cost for such a site would be $10-$15 per month. ($180 per year). For on campus access, replacing equipment costs $2740.
3) It is recommended that budgeting be made available to allow faculty to explore newer technologies in order to meet student demand for courses that allow software development for those newer technologies. An example would be the purchase of iPads, at a cost of $600 to $800 each (total $3,600 to $4,800), for faculty with the intent of being able to fully explore the development of apps for such technology, and finally the ability to offer a course in that software development to our students.

In summary, the total here will be approximately $22,180 ($4,700 * 3 + $180 * 3 + $2740 + $4,800 = $22,180), assuming a three year period.

Recommendation 2013-E: (Curriculum) Establish an AS-T Degree for Computer Science. This has already been presented at the College Curriculum Committee. The current status of the degree is being reviewed at the Chancellor's office level.

Appendix A – Computer Science Success, Retention and Fill Rates By Course and Semester

Fall 2008 – Fill Rate: 96.0%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	4	22	9	2	7	10	5	29	84	39.3%	59.5%
CSCI-2	1	6	2	5	2	0	1	5	21	61.9%	71.4%
CSCI-3	1	7	7	3	0	4	0	3	24	70.8%	87.5%
CSCI-10	1	4	2	4	0	1	0	16	27	37.0%	40.7%
CSCI-60	1	6	2	2	0	0	0	3	13	76.9%	76.9%
Totals	**8**	**45**	**22**	**16**	**9**	**15**	**6**	**56**	**169**	**49.1%**	**63.3%**

Spring 2009 – Fill Rate: 84.1%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	4	18	15	8	5	8	1	26	81	50.6%	66.7%
CSCI-2	1	3	8	1	0	1	3	3	19	63.2%	68.4%
CSCI-4	1	8	4	0	0	0	0	3	15	80.0%	80.0%
CSCI-30	1	12	0	0	0	0	1	3	16	75.0%	75.0%
CSCI-40	1	4	1	4	0	3	0	5	17	52.9%	70.6%
Totals	**8**	**45**	**28**	**13**	**5**	**12**	**5**	**40**	**148**	**58.1%**	**69.6%**

Summer 2009 – Fill Rate: 95.5%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	2	8	7	9	0	0	1	17	42	57.1%	57.1%
Totals	**2**	**8**	**7**	**9**	**0**	**0**	**1**	**17**	**42**	**57.1%**	**57.1%**

Fall 2009 – Fill Rate: 94.2%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention

CSCI-1	4	17	16	12	5	5	6	23	84	53.6%	65.5%
CSCI-2	1	8	6	2	1	0	2	2	21	76.2%	81.0%
CSCI-3	1	12	3	3	0	2	1	5	26	69.2%	76.9%
CSCI-10	1	1	3	2	1	4	0	3	14	42.9%	78.6%
Totals	**7**	**38**	**28**	**19**	**7**	**11**	**9**	**33**	**145**	**58.6%**	**71.0%**

Spring 2010 – Fill Rate: 98.7%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	4	23	16	13	9	8	4	18	91	57.1%	75.8%
CSCI-2	1	8	6	1	1	1	0	7	24	62.5%	70.8%
CSCI-30	1	6	6	3	0	0	0	1	16	93.8%	93.8%
CSCI-40	1	3	7	4	0	1	1	5	21	66.7%	71.4%
Totals	**7**	**40**	**35**	**21**	**10**	**10**	**5**	**31**	**152**	**63.2%**	**76.3%**

Summer 2010 – Fill Rate: 97.7%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	2	7	6	7	3	6	1	13	43	46.5%	67.4%
Totals	**2**	**7**	**6**	**7**	**3**	**6**	**1**	**13**	**43**	**46.5%**	**67.4%**

Fall 2010 – Fill Rate: 100%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	5	38	20	14	4	5	8	22	111	64.9%	73.0%
CSCI-2	1	13	1	1	0	0	1	6	22	68.2%	68.2%
CSCI-3	1	6	9	2	0	1	0	3	21	81.0%	85.7%
Totals	**7**	**57**	**30**	**17**	**4**	**6**	**9**	**31**	**154**	**67.5%**	**74.0%**

Spring 2011 – Fill Rate: 93.5%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention

CSCI-1	4	36	20	11	2	7	3	7	86	77.9%	88.4%
CSCI-2	1	4	5	2	0	2	2	7	22	50.0%	59.1%
CSCI-3	1	9	0	0	0	0	1	3	13	69.2%	69.2%
CSCI-40	1	10	5	1	0	0	1	6	23	69.6%	69.6%
Totals	**7**	**59**	**30**	**14**	**2**	**9**	**7**	**23**	**144**	**71.5%**	**79.2%**

Summer 2011 – Fill Rate: 95.5%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	1	3	7	1	1	2	0	7	21	52.4%	66.7%
Totals	**1**	**3**	**7**	**1**	**1**	**2**	**0**	**7**	**21**	**52.4%**	**66.7%**

Fall 2011 – Fill Rate: 103.9%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	5	24	18	27	2	14	6	22	113	61.1%	75.2%
CSCI-2	1	4	2	1	2	0	3	13	25	28.0%	36.0%
CSCI-3	1	11	5	4	0	2	0	0	22	90.9%	100%
Totals	**7**	**39**	**25**	**32**	**4**	**16**	**9**	**35**	**160**	**60.0%**	**72.5%**

Spring 2012 – Fill Rate: 96.0%

Course	Sections	A	B	C	D	F	DR	W	TOTAL	Success	Retention
CSCI-1	5	25	25	15	6	16	2	17	106	61.3%	82.1%
CSCI-2	2	7	10	10	2	1	3	7	40	67.5%	75.0%
CSCI-40	1	11	2	2	0	3	0	5	23	65.2%	78.3%
Totals	**8**	**43**	**37**	**27**	**8**	**20**	**5**	**29**	**169**	**63.3%**	**80.0%**

Appendix B – Survey Results – Administered June 2013

82 students surveyed

15) Course Currently Enrolled In

CSCI-1	34
CSCI-2	29
CSCI-30	19
M-210	5
CSCI-12	1

2) Educational Goal

M.A.	35	42.7%
B.A.	34	41.5%
PhD.	11	13.4%
unsure	2	2.4%

3) Where do you want to transfer?

UC Irvine	18
CSU Long Beach	17
UCLA	14
UC Berkeley	9
USC	6
CSU Dom Hills	6
Cal Poly Pomona	4
Other	9
Undecided	4

4) Major?

Comp Science/Comp Engineering	58	70.7%
Other Engineering	13	15.9%
Mathematics	6	7.3%
Game Design	4	4.8%
Business	2	2.4%
Physics	1	1.2%
Undecided	1	1.2%

5) Future Job?

Programmer/Software Engineer	35	42.7%
Undecided	14	17.1%
Other Engineer	9	11.0%
Teacher	6	7.3%
Game Designer	6	7.3%
Business	5	6.1%
Computer Security	3	3.7%
Other	5	6.1%

6) How often do you use the CS Tutor?

Never	35	43.8%
less than once per week	18	22.5%
one to two times per week	25	31.3%
more than two times a week	2	2.5%

7) Other CS Courses taken at ECC?

CSCI-1	46
CSCI-2	19
CSCI-3	15
M-210	12
CSCI-40	5

8) Courses you want to take at ECC?

CSCI-3	47
CSCI-30	30
CSCI-2	27
M-210	26
CSCI-40	23
CSCI-15P	21
CSCI-12	17

9) Are you getting an AA Degree at ECC?

No	36
Yes	35
Maybe	7

If yes, then in what subject?

Computer Science	16
Mathematics	12
Physics	2
Engineering	2
Economics	2
CIS	1

10) Where you ever turned away from an ECC CS class because it was full?

No	51
Yes	31

If yes, then what was the course you couldn't get in?

CSCI-1	15
CSCI-3	14
CSCI-2	7
CSCI-30	2

11) Did you ever have to take CS courses elsewhere?

No	57
Yes	5

If yes, then where?

SMC	3
no response	2

15) What other CS courses would you like ECC to offer?

Graphics Programming	23
Advanced Java	20
C# Programming	14
Game Programming	13
.NET – CSCI-36	10
C Programming	9

ASP.NET – CSCI-60	8
Android/Mobile Apps	8
Python	5
HTML	5
Assembly Language	3
Fortran	3
CSS	3
Javascript	2
Advanced Discrete Math	2
Objective C	2
Data Structures in Java	1
Perl	1
Ruby	1
Networking	1

Appendix C

U.S. News and World Report – Top 100 Jobs of 2013 (the top 13)

1) Dentist
2) Registered Nurse
3) Pharmacist
4) **Computer Systems Analyst**
5) Physician
6) **Database Administrator**
7) **Software Developer**
8) Physical Therapist
9) **Web Developer**
10) Dental Hygienist
11) Occupational Therapist
12) Veterinarian
13) **Computer Programmer**

Career and Technical Education- Supplemental Questions

The Computer Science Program at El Camino College.

1. **How strong is the occupational demand for this program?**
   According to the ECC Computer Science Analyst Report, there are over 102,000 computer jobs in California. There are predicted to be about 1,600 new jobs this year which is a large part of all STEM-related new jobs.

2. **How has the demand changed in the last 5 years and what is the outlook for the next 5 years?** The growth in computer jobs in California has averaged about 1.6% per year for the last 5 years… but the job growth from year to year has been volatile. For instance, in 2011 the growth was a mere 1%, but in 2012 (the last year we have data for) it was 3%. The estimated job growth over the next 5 years is estimated at about 8%, which far exceeds the population growth. According to US-BLS employment projections, in the next 5 years (thru 2018) 71% of all new STEM jobs will be in computing.

3. **What is the El Camino College District's need for the program?**
   Jobs in Computer science are the fastest growing of any STEM field. We need to be a part of this. It is our mission to serve the community and to offer affordable education … education that leads to jobs. Our school has produced excellent students, winning 1st or 2nd place in the Progfest programming contest the last 5 years in a row, which builds school spirit and gives evidence for the excellence of our school.

4. **What is the state's need for the program?**
   Computer programming is everywhere…in our cell phones and computers, our cell phones, our automobiles. Computer programming has changed the face of what technology is and what it can do, it has changed the way we live our lives here in California and everywhere. It creates jobs, creating income for individuals and for the state. California has always been a leader in Technology. Much of that technology is computer-based.

5. **How does the program address needs that are not met by similar programs in the region?**
   We are doing a good job of meeting those needs that are met by other community colleges in the region (after all, we are winning contests). We are meeting needs that are not addressed at all by the four year colleges… we have people still in work who just need a course or two to improve their work skills, we educate people who do not yet qualify to enter a 4 year college. We educate students at reduced prices that the 4 year colleges cannot compete with.

6. **Are the students satisfied with their preparation for employment?**
   As instructors it is our experience our students, including those students who are currently employed and seek to prepare themselves to be better employed enjoy the courses we offer and are positive about the usefulness of what they learn. If, due to the huge cuts in our offerings they are not able to get into a course and complete their preparation then they, of course, are frustrated.

7. **Are the employers in the field satisfied with the level of preparation of our graduates?**
   El Camino College does not keep or collect this data. Only those students already in a job and seeking to better their knowledge go directly into a job after their education at El Camino College. 98% of our students intend to go on to earn a BA and put off getting a computer job until they have graduated from a 4-year college. We are not informed of their eventual employers.

8. **What are the completion, success, and employment rates of the students?**
   - Our completion rate has been over 70% every semester since the fall of 2008. Our latest data (Fall of 2012) shows a 76% retention rate.
   - Our success rate has been over 60% every semester since 2010, and for the latest semester we have data on (Fall 2012) it was just over 60%.
   - We do not have access to data on computer-science related employment, which generally occurs 4 to 8 years after leaving El Camino, after graduating from at a 4 year college or university with a BA, MA, or PhD.

9. **What is the role of the advisory committee?**
   The computer science committee acts as the advisory committee and makes the decisions that affect every course and the way they are run. We are expanding our advisory committee to include outside persons.

10. **There is no licensure exam for computer science students.**