

EL CAMINO COLLEGE
MATHEMATICAL SCIENCES
Computer Science Advisory Board Meeting

May 22, 2015

Present: David Akins, Carl Broderick, Greg Fry, Kevin Judge (John Deere), Marta Maaza, Matthew Mata, Brad Rumery (Sampra Energy), Solomon Russell, Greg Scott, Jacquelyn Sims, Satish Singhal, Ralph Taylor

1. Welcome:
 - A. This is the first advisory board meeting for the Computer Science CTE program.
 - B. Being designated as CTE allows the program to compete and request certain funds based on grants.
 - C. The Computer Science department applied for a grant through CTEA requesting upgraded laptops for CS faculty. The grant awards have not been determined yet. For the next CTEA grant, we will look into creating a dual classroom and computer lab.
2. Introduction of committee members
3. Data on expansion of Computer Science department, new faculty appointment, current lab space and need for additional lab space (data attached). Perkins CTE funding requirements.
 - A. The goal is for growth and expansion. The CS Department is seeking out industry advice to help make students better employees and ensure they meet industry standards.
4. Department Updates
 - A. There are currently three full-time instructors in the CS department.
 - B. The Mathematical Sciences Division has three departments including Computer Science, Pre-Engineering and Mathematics.
 - C. There are five full-time mathematics instructors and five adjunct instructors who also teach CS courses.
 - D. There are seven active CS classes currently being offered. The department has more than doubled the CS class offerings within the last three years. One of the main issues with such growth is capacity and resources. All of the courses offered have a lecture and lab setup, which allows the department to stand out and be more competitive (not all schools have labs as part of their classes). However, that setup causes a capacity shortage.
 - E. MBA 113 has 22 workstations and is the only designated computer lab for CS. We are advocating for more resources to establish another computer lab.

- F. In the recent past, the Math Division had an AS degree for CS. However, with the AS-T (Transfer) degree we were mandated to change the curriculum and change a lot of the current content which would jeopardize ECC's articulation for transfers. The CS department inactivated the AS degree a year ago and decided to continue with a certificate program.
5. Report on current courses and future trajectories of these courses:
- A. CSCI 1 Programming and Problem Solving with C++, CS 12 Web Development: This is the entry-level CS class. Trigonometry is a prerequisite. The largest program in CS 1 is 400 - 500 lines of code. At the end of the class, students must create a program like payroll processing where they read a data file, search and sort. The goal of this class is to ensure students understand structures and records and can process data files.
 - B. CSCI 2 – Data Structure: The first part of CS 2 is devoted to programming. All the items learned in CS 1 are used in CS 2, except the data is hidden. The second part of the course goes into operative programming. Pointer, linked lists and binary trees are also covered in CS 2. The final project is essentially a component of a search engine. The future plan for CS 2 is to add more components like graph data and hash tables.
 - C. CSCI 3 Java: A component for Java is database applications and introduction to Android programming. The final project is a game using sound and user interaction.
 - D. CSCI 40 UNIX: Greg Scott started this course as a Community Education class and built the program using grant funds. The problem with the course is support because many people do not know UNIX. The class has six weeks of shell programming and six weeks of introduction to the command line. There is also the introduction to networking and system administration. There are no prerequisites. UNIX provides good structure. This year, the CS and ITS departments discussed the old computer hardware used for UNIX. ITS is now working on creating a virtual server. Twenty-five external hard drives were purchased so each student can install UNIX on their workstations.
 - E. CSCI 16 Assembly Language: Essentially, assembly language helps students understand what is happening in their computers. Students learn about the ways things are stored into a computer, the stack frame, how information goes back and forth, etc.
 - F. CSCI 30 Advanced C++: This course is the advanced programming in C++. The topics include classes, virtual function, abstract base class and inheritance.

- G. Discrete Math (Math 210): This class starts with logic, sets, number theory, algorithm analysis, accounting, graph theory, relations and functions. Throughout the class, there are methods of proof. It is offered once a semester, however, with the growth of CS, more sections may be offered at a later time. There is not a uniform description of discrete math class in transfer institutions. ECC's Math 210 matches up with UCLA and CSULB. It's very important that our courses articulate. The state has a new requirement of the CID system. This course is part of a degree and we are being forced to conform to CID. If they want us to comply with CID, we need to add another unit to the course to ensure we don't lose its articulation.
- H. The Math Division has tutoring programs for all classes.
6. Report on future courses, timelines and contents (Python, and others)
- A. It was reported that Dr. Ghyam introduced a course on Python at another school and plans to use this to create a similar course at ECC. It's an additional option for other disciplines on campus. This course may be offered in fall or spring 2016.
 - B. Dr. Mata is an expert on MATLAB which may be a future possibility.
 - C. There is a new course that the College Board is pushing at the high school level called Computer Science Principles. The course is language independent, but there are ideas of ethical concepts, programming, security, and Internet. It uses a visual program. LAUSD is starting to pilot this course so its something to look out for in fall 2016.
7. Report by Computer Science faculty on conferences attended
- A. The importance of an Advisory Board was emphasized at the Chancellors Conference on CTE for Faculty conference attended by Satish Singhal. It is important to have an advisory board for CTE programs.
8. Internships for Computer Science students
- A. What can we do to better prepare students coming into the industry?
 - B. It would provide a lot of benefits if students learn how to work on hardware.
 - C. It would also be beneficial to teach students how to write programs.
 - D. Practical examples of security would also prepare students that are entering the industry. Place an emphasis on stack management as hackers usually come in through the stack. It would also be beneficial if instructors taught students the ways in which hackers come in.
 - E. Computer safety should also be discussed each semester.
 - F. John Deere, along with many other companies, use MATLAB so an understanding of UNIX and MATLAB is very important to the industry.

9. CTE review for Computer Science program (due fall 2015)
 - A. CS faculty members are currently working on the CTE review. It will be complete in fall 2015.
10. Student Activities
 - A. Our students are active and involved in various competitions. The Robotics Club is very active.
 - B. We may want to consider a monetary award for Computer Science academic awardees.
 - C. Computer Club is headed by Professor Greg Scott and meets regularly.
11. The Computer Science Advisory Board is a commitment of a couple meetings per year. The next advisory board meeting will be set for fall 2015.

Computer Science Department Information

Computer Science Faculty

Full-Time Instructors

1. Massoud Ghyam
2. Solomon Russell
3. Satish Singhal

Math/Computer Science Instructors

1. Carl Broderick
2. Greg Fry
3. Matthew Mata
4. Greg Scott
5. Ralph Taylor

Non Full-Time Math/Computer Science Instructors

1. David Akins
2. Edwin Ambrosio
3. Joseph Hyman
4. Juan Leon
5. Esmaail Nikjeh

Computer Science Courses Offered (*all courses are lab and lecture*)

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

Computer Lab Usage/Need

Each section \approx three hours of lab per week

20 sections for fall 2015 semester

Total lab time usage \approx 60 hours per week (roughly 12 hours per day)

One devoted lab (MBA 113) with 22 workstations

Lab	MW	TTh	Sat	Total Hours
MBA 113	20	8.5	5	33.5
MBA 120	8.5	6	0	14.5
MBA 220	3	3	3	9
				57

Approximately 57 hours/week of lab usage

Spring and Fall Comparisons

Computer Science Classes Offered	Spring 2014	Spring 2015
CS 1	7	9
CS 2	2	3
CS 3	2	4
CS 12	1	0
CS 30	1	2
CS 40	1	1
Total	14	19
36% Increase from Spring 2014 to Spring 2015		

Computer Science Classes Offered	Fall 2014	Fall 2015
CS 1	9	9
CS 12	0	1
CS 16	1	2
CS 2	4	4
CS 3	3	3
CS 30	0	1
Total	17	20
18% Increase from Fall 2014 to Fall 2015		

Degrees and Certificates Awarded 2009-2014

Degrees Awarded					
Major	2009-10	2010-11	2011-12	2012-13	2013-14
Computer Science	1	5	2	7	9
Mathematics	9	31	38	66	49
Pre-Engineering	11	12	22	29	29
Total	21	48	62	102	29

Certificates Awarded					
Major	2009-10	2010-11	2011-12	2012-13	2013-14
Computer Science	2	0	0	2	2
Total	2	0	0	2	2

Course Descriptions

Computer Science 1 - Problem Solving and Program Design Using C++

This course is an introduction to problem solving and program design using structured, top-down algorithmic development techniques applied to the solution of numeric and nonnumeric problems. Software engineering topics such as analysis, design, implementation, testing, documentation, and maintenance of software are discussed. Laboratory work will be done using the C++ computer language.

Computer Science 2 - Introduction to Data Structures

In this course, the C++ computer language is used to demonstrate methods of representing and manipulating data. The student will learn the object oriented problem solving skills necessary to read, write, and correct complex computer programs, and to make important design decisions. Topics include lists, stacks, queues, trees, searching, sorting, modeling and algorithm analysis.

Computer Science 3 - Computer Programming in Java

This course includes a detailed coverage of the Java programming language including Java data types, operators and expressions, control structures, iterations, functions, arrays, classes and inheritance, files, graphical user interface (GUI) applications with event handling, and applets for world wide Web applications.

Computer Science 12 - Programming for Internet Applications using PHP, JavaScript, and XHTML

In this introductory Internet programming course, students learn the fundamentals of Internet programming with JavaScript and Hypertext Preprocessor (PHP), a widely used, open source, general-purpose server-side programming language. Students design and write applications that extend web servers. These applications use backend databases to process data submitted through web forms and provide access to dynamically generated web pages with the retrieved data from the database.

Computer Science 16 - Assembly Language Programming for the IBM PC and Compatibles

This course includes detailed coverage of assembly language programming, based on the IBM compatible personal computer. Topics include hexadecimal arithmetic, two's complement

arithmetic, memory organization, addressing modes, procedure calls, the stackframe, macros, calling assembly language procedures from Pascal or C/C++, recursion, BIOS and DOS interrupts, the floating point unit and instructions, and the debugger.

Computer Science 30 - Advanced Programming in C++

This course presents an advanced coverage of the C++ programming language. Topics presented include templates, the Standard Template Library, data abstraction, operator overloading, inheritance, friend functions, virtual functions, multiple inheritance, and virtual base classes. An emphasis will be placed on object-oriented programming.

Computer Science 40 - Introduction to UNIX and LINUX Operating Systems

This course covers UNIX and LINUX operating system concepts and include basic commands, file structures, editors, file management utilities, shell programming, process control, and remote messaging as well as network and system administration.