CALIFORNIA STATE UNIVERSITY EAST BAY

COMMITTEE ON ACADEMIC PLANNING AND REVIEW
ANNUAL PROGRAM REPORT

College	CoS
Department	Math Computer Science
Program Unit	Computer Science
Reporting for Academic Year	2015-2016
Department Chair	Matt Johnson (Reported by Kevin Brown)
Date Submitted	6/20/2016

1. SELF-STUDY (about 1 page)

A. Five-year Review Planning Goals

The last Computer Science five year review identified the following goals for the Master's in Computer Science program:

Curriculum:
i) Revamp the currently confusing breadth category requirements in which students must take 2 courses each from courses identified as either Development/Theory/ or Systems/Architecture.
ii) Address issues arising from allowing graduate students to take 3000 or 4000 level undergraduate courses towards their Master's degree electives.
iii) Improve preparedness of students, insuring that prerequisite courses adequately prepare students for later coursework.

Students:
i) Grow program in order to help address need for qualified tech workers in California.
ii) Improve student experience and graduation rate. Increased course offerings and more formal advising should result in better retention.

Faculty:
i) Recruit new faculty to reduce reliance on lecturers and to provide opportunities to offer classes and research support in areas of current Computer Science areas of development.
ii) Address workload of Graduate Coordinator, who is responsible for evaluating 1500-2000 applications per year, in addition to advising enrolled graduate students.

Resources:
i) Facilities for Department faculty offices, teaching labs, research labs, including co-locating

office space to provide opportunities for faculty to work together more easily.
ii) Support from campus Information Technology for teaching and research needs.
iii) Need for funds for readers, TAs, and travel to academic conferences.
iv) Need for library resources, specifically to support graduate courses.

B. Five-year Review Planning Goals Progress

Please note: it has now been five years since the Department's last five year review (2010-2011). Due to semester conversion however, CAPR has instituted a modified five year review plan, moving the next department review to 2016-2017.

Three major developments have occurred in the last year which impacted several of the planning goals. They are the division of the Mathematics and Computer Science department into two separate departments, the move of the department office and faculty offices of the Department of Computer Science to the new Student and Faculty Support (SF) building, and the design, review, and approval of the new semester-based graduate and undergraduate programs as part of campus semester conversion.

Separation of department:
The faculty of the department of Mathematics and Computer Science had discussed the possibility of separation into two departments for many years. Combined departments were the result of Computer Science growing out of Mathematics departments, with some faculty teaching in both disciplines. Computer Science has been a distinct field of endeavor for several decades now, and only 2-3 faculty members of the combined department taught in both areas, limiting the need to house the two programs in one department. The great size of the department (nearly 30 faculty), and the large differences in the programs made it difficult to manage, and difficult for standard departmental policies to be developed. As a result, a request was made to separate the programs into two departments. That request was approved in July, 2015.

Move to SF building:
With the completion of the new Student and Faculty Support building, the Department of Computer Science department office and most faculty offices were moved from offices spread over all four floors of both Science buildings to the fifth floor of the new building.

New semester-based programs:
As part of the campus-wide semester conversion, the Department of Computer Science developed a new Master's program addressing shortcomings of the current program as well as plans for assessment. This program was reviewed and approved by the college and university curriculum committees.

Now, the progress towards the goals identified in section A will be addressed.

Curriculum:
i) Revamp the currently confusing breadth category requirements in which students must take 2 courses each from courses identified as either Development/Theory/ or

Systems/Architecture.

The department faculty felt that the division of courses into two categories was less than clear in some cases. As part of semester conversion, the breadth categories are eliminated. Instead, the number of core courses was increased from 2½ (10 quarter units) to 5 (15 semester units), providing a more comprehensive shared learning experience for the students, and more flexibility in the remaining electives. The new requirements are much a simpler to understand, which should reduce confusion and need for advising. A larger number of required courses will also simplify scheduling and lead to a more predictable annual schedule allowing students to plan their schedules far in advance.

ii) Address issues arising from allowing graduate students to take 3000 or 4000 level undergraduate courses towards their Master's degree electives.

While the CSU and WASC allow a portion of Master's degree requirements to be fulfilled using undergraduate courses, there was substantial difficulty in ensuring that students did not use the same course to address both a requirement in their undergrad and grad programs. This policy required substantial, careful evaluation of undergraduate transcripts, and extensive advising, and caused a great deal of confusion for the students. As a result, the department chose not to allow undergraduate courses to be used towards Master's degree requirements. This new policy will reduce the time needed to evaluate transcripts during admission, and greatly simplify one area of advising for students.

iii) Improve preparedness of students, insuring that prerequisite courses adequately prepare students for later coursework.

As part of the semester conversion, the department has clarified the requirements for admission to the Master's program, and added several admission prerequisites to ensure that students are adequately prepared to succeed in the program. Scores earned on the GRE test have also been made a mandatory component of the admission packet, and minimum scores on the test have been established.

In a development that the Department applauds, the University will be enforcing course prerequisites as part of course enrollment. The Department has been checking course prerequisites by hand in the past, which is not practicable for large numbers of sections. As a result, some students successfully enrolled in courses for which they had not completed the course prerequisite often leading to less than successful results.

Lastly, and perhaps most importantly, all required courses will now be assessed to determine if students successfully learned the required course content. Now that the semester program has been approved, the Department will begin work on assessment instruments and rubrics for each required course. Assessment results will allow the Department to address any failing in the courses to ensure that students are prepared for later coursework.

In regards to the current quarter-based program, we successfully administered post-assessment examinations for four courses this year (most notably for the Capstone Experience). We are compiling results and modifying our process as needed.

Students:
i) Grow program in order to help address need for qualified tech workers in California.

Studies have shown for many years that there is a lack of qualified applicants for tech jobs in California and across the nation. The Department of Computer Science hopes to help address this shortage by growing our program and generating more qualified graduates. At the Master's degree level, there is limited demand from domestic students as most positions require a Bachelor's degree and some industry experience. As a result, the large majority of our student population is made up of students from other countries, predominantly India. They are attracted to the university due to our proximity to Silicon Valley. Our programs already possess the highest rate of degree-related job placement at CSUEB (from AACE data). Even so, the economic upturn has increased job opportunities for our graduates even more. The Bureau of Labor statistics projects another 22% increase in jobs over the next 6 years. Consequently, we are seeing a marked increase in the number of applications. Students are getting hired even more quickly and finding internships easily. Employers are contacting our department on a weekly basis.

Due to the fact that the students are mostly from abroad, however, the Department lacks the ability to directly affect the number of students who apply. The Office of International Admissions (IAO) makes periodic recruiting tours through India to visit institutions with strong ties to CSUEB and the Bay Area. These recruiting tours directly drive the number of applications that the program receives. In the last year, the number of students in the program dropped by 13%. Given discussions with IAO, we believe a large part of this drop was due to slow processing of applications. Both IAO and the Department of Computer Science moved into the SF building this year which caused significant delays. In response to the lower enrollment, the Department has been meeting with IAO to streamline the admissions process for the students in hopes of increasing the yield of applications to enrollees. In addition, the Department has asked IAO to increase its presence in China and any other potential markets, in attempts to both grow and diversify the program. Finally, the Department has been working with the College of Science Dean's office to manage staffing so that applications can be processed more quickly.

ii) Improve student experience and graduation rate. Increased course offerings and more formal advising should result in better retention.

The major curricular issues facing our students are the ability to plan for course offerings in advance, the ability to enroll in classes as needed, and availability of advising in particular in regards to choice of breadth courses and electives. All of these concerns should be met by the new semester-based curriculum. The larger set of required courses will lead to more predictability in scheduling, and more sections of the courses will be offered leading to ease of enrollment. The removal of the breadth requirements and the restriction to graduate level electives will make the program easier to understand and more flexible as students may take any graduate level elective offered in a quarter rather than having to choose from a particular set associated with a breadth category.

The Department of Computer Science will begin using the online advising software to advise students in the Master's program beginning in Fall quarter 2016. This software will replace the document folders that have been used in the past. The change will allow advising to be done more efficiently and to provide all necessary information to the students so that they are kept informed of any decisions regarding substitutions, change of status, etc. ITS is configuring the advising software for use in the Master's program in Summer 2016.

Faculty:
i) Recruit new faculty to reduce reliance on lecturers and to provide opportunities to offer classes and research support in areas of current Computer Science areas of development.

Faculty recruitment is one of the Department's main concerns. Even to maintain the program at its current size, new faculty will need to be hired as there are four faculty in the FERP program at this time. In order to grow the program and address new areas, even more faculty will be required. Two years ago, a faculty search ended in failure with only a handful of applicants. This despite extensive outreach efforts. Last year, the Department conducted a search for 2 positions. The Department was fortunate to fill one of the tenure track positions, and welcomes Dr. Varick Erickson, joining us in Fall 2016. The Department was unable to fill the second position though, and as has been typical, the first few applicants who have been offered a position have taken jobs elsewhere. The applicants have regularly pointed to insufficient compensation as the reason for turning down our offer, and it is typical that even other CSU campuses are able to provide more competitive offers. The Department will again be conducting a search for two positions next year, a roll-over of last year's position, and a new one. The Department continues to seek diverse applicants with interests in emerging fields of current interest, and again is making significant efforts to provide outreach to under-represented groups.

ii) Address workload of Graduate Coordinator, who is responsible for evaluating 1500-2000 applications per year, in addition to advising enrolled graduate students.

The Graduate Coordinator evaluates applications for admission, and advises current students. The Coordinator receives 2 courses assigned time, which is insufficient in order to complete these tasks. The main concern is the number of applications, which have been in the range to 1500-2000 per year. Almost all applications are from abroad, and require significant evaluation in order to determine if the applicant has completed the 19 prerequisite courses that the Department has determined are necessary for successful preparation for entrance to the Master's program. A large percentage of the applicants are not prepared for the program and must be denied. The yield of enrolled students to applicants ranged from 3% to 8% over the last three quarters. An enormous amount of work was necessary on the part of International Admissions, the staff of the Department of Computer Science, and Graduate Coordinator for very little payoff in terms of enrolled students.

As described above, the Department has met with the Office of International Admissions (IAO) to better streamline the admissions process in hopes of increasing the yield. Also, a minimum requirement on GRE scores has been implemented, eliminating the need to evaluate some of the least competitive applications. The Department is also in discussions to modify the document deadlines and quarters in which applications are accepted in order to manage the workload.

Resources:
i) Facilities for Department faculty offices, teaching labs, research labs, including co-locating office space to provide opportunities for faculty to work together more easily.

The Department was pleased that space was made available in the new Student and Faculty Support (SF) building for the department office and faculty offices. Faculty had been spread over all four floors of both Science buildings previously. Proximity to colleagues and the department office has already led to increased communication between department faculty, more frequent department committee meetings, and increased productivity. Unfortunately, space was not made available for all department faculty. In particular, faculty participating in the FERP program, and all lecturers are still housed in the Science buildings. In addition, there is not enough space in the new building even to house the faculty that would result from successful searches that have been approved for next year. Ideally, it would be beneficial to house the entire department in one place with enough made available for desired growth.

The College of Science has provided some limited additional teaching lab space, but this is still a significant concern as current space is insufficient to meet department needs. As our outside reviewer mentioned, if we choose to seek accreditation, our relative lack of teaching and experimental lab space would be a major concern to the accrediting board. The Department is in discussions with the Dean of the College of Science to address these issues.

ii) Support from campus Information Technology Services for teaching and research needs.

The centralization of Information Technology Services (ITS) on campus left the Department of Computer Science with no dedicated support for its teaching and research support needs. The centralized model was unwieldy, slow, and has not served the Department well, leading to impact to the students as software and hardware testbeds and learning environments have become outdated or unusable. Our outside reviewer stated that even the support that was provided before centralization was insufficient. The Department is in discussions with the College of Science to provide dedicated lab course support as is the norm for other lab-based disciplines.

iii) Need for funds for readers, TAs, and travel to academic conferences.

The need for additional resources to fund readers, Teaching Assistants, and travel to academic conferences is little changed. Our outside reviewer specified lack of funding for continuing development and other department needs was a significant issue. The lack of funding is especially an important factor as we attempt to hire new faculty who are especially in need of grading support, and are expected to publish and present at conferences.

iv) Need for library resources, specifically to support graduate courses.

Library offerings have been uneven, with important database subscriptions cancelled and re-established. Access to a wide range of journals and conference proceedings is especially important as we plan to increase the rigor of our graduate courses under the semester system.

C. Program Changes and Needs

Program Changes:

i) Assessment
At the time of our last five year review, we were just starting to identify program learning outcomes and student learning outcomes. In the past three years, we have finalized our assessment process, and have conducted post-assessment examinations for targeted course. The exams are deployed through Blackboard and automatically scored. In the last two years, we have also closed the loop on assessment by making changes based on our assessment data.

Perhaps more importantly, going forward, we have incorporated assessment concepts in the new semester-based curriculum that has been reviewed and approved by the college and university curriculum committees. Specifically, the Department has increased the number of required courses in the program, each of which will be regularly assessed. This will allow data to be compared easily from course offering to offering and across student cohorts, providing opportunities to improve the quality of the program. The Department will be working to devise assessment instruments and rubrics for the semester-based system in the coming year.

ii) New courses
Since our last five year review, three new hybrid courses have been added to the curriculum: Security in Mobile, Wireless, Grid and Pervasive Computing (CS 6526), Security Management (CS 6527) and Cloud Computing (CS 6593). Additionally, the new course Statistical Learning and Data Analysis (CS 6831) was added to the Computer Science curriculum. These are courses that provide students the opportunity to learn about more current technologies.

iii) Oversubscribed courses
In our previous report, we mentioned our problem with oversubscribed courses. Students with early registration appointment times would sign up for the maximum allowed by the system (4 enrollments and 4 waiting lists). They would then drop the courses they did not want on the last day of the Add/Drop period. This meant that students with later registration appointments, including all incoming students, would have to get on waiting lists. The incoming students would be unable to register for courses that we had told them they were required to take, and for visa reasons had to sign up for courses for which they were unprepared. It also meant that sometimes extra sections created to meet supposed demand would turn out to be unnecessary after all the drops. Early attempts to mitigate the problem by appealing to students to stop enrolling in extra courses had no impact on the problem.

The department addressed this by limiting early registration for courses starting in the Fall quarter of 2014. After seven quarters, we have to say this has been the most beneficial change to the program since the start of the assessment process. New students have been able to enroll in appropriate courses, existing students have been able to find courses to satisfy requirements, and enrollment has reflected actual demand.

iv) Academic dishonesty

Another issue with broad impact has been academic dishonesty. The department chair receives copies of all academic dishonesty reports, and had to process a large stack of reports each quarter. In meetings with the students involved, he also found that students were dismissive of the measures taken at the university level. Anecdotally, some faculty observed repeated incidents with the same students despite previous reports on their file. Inevitably, some faculty had ceased to use the academic dishonesty process since they felt that the reports had no impact and were thus a waste of time. Students not involved in academic dishonesty were frustrated as well, feeling that they had to compete with students with an unfair advantage. Some did come forward to mention their frustrations and request the department to do something, though as might be expected, they did so anonymously. In a rather dramatic incident, a research group from another university investigating the online solicitation of illicit aid for coursework provided an instructor with proof that a student in that instructor's course had offered to pay for someone to do their assignments. It turned out that work was shared with the rest of the class.

To handle this issue, the Chair proposed, and the Department accepted, levying department-level sanctions to a sufficient level to at least diminish the frequency. The first incident of reported academic dishonesty deprives the student of any opportunity or benefit requiring a department signature, whether a grader or TA position, a scholarship, or the opportunity to do an internship through the university's curricular practical training program. The second incident of reported academic dishonesty, even in the same academic term, results in declassification of the student. The effect of this is to remove them from the program. As with the early registration limits, this policy was started in the Fall quarter of 2014. While the punishments may seem harsh, in the seven quarters of enforcement, the number of academic dishonesty reports has plummeted. During that period, the teaching faculty has been essentially the same, as have the courses offered. The drop in reports may be partly due to a significant number of students leaving after the Fall quarter of 2014, either after being academically disqualified or to transfer to other (often unaccredited) regional universities. Departing students were found to be only those whose grades were below the standard for acceptable academic standing. This policy serendipitously has also allowed us to address the 4th ILO of the university, specifically "understanding the implications of values and ethics for leadership, teamwork and collaboration".

Faculty:

Name	Time Base
Brown, Kevin	1.0
Christianson, Leann	1.0
Ertaul, Levent	1.0
Grewe, Lynne	1.0
Johnson, Matt	1.0
Jurca, Dan	0.44 (FERP)
Reiter, Eddie	0.5 (FERP)
Roohparvar, Farzan	1.0
Simon, Steve	0.44 (FERP)
Yang, David	1.0
Yu, Ytha	0.44 (FERP)
Zhong, Fay	1.0

TOTAL FTEF	**9.82**

Resources and Needs:

i) Lab space and lab tech support

The Computer Science Department was dramatically impacted by ITS centralization several years back. Up until two years ago, the Department had only one small computing lab with less than a dozen machines -- despite the number of students in the majors – and only one computer classroom. Two years ago we were finally able to obtain at least primary usage to a second newly renovated computer classroom, and access to a second small computer lab in VBT. The Department is still severely underequipped. Students often try to make do with their own laptops and general purpose space (like the Cave of the Science building), but this often leads to difficulties from incompatibilities among their laptops, or lack of required software. Many courses in the curriculum require dedicated servers that are isolated from the campus networks, as students write programs to interact or query these servers. Getting these configurations set up is difficult or impossible when ITS centrally manages all systems on campus. Classroom space, retiring faculty, equipment and software shortages, and lack of dedicated ITS support are all issues that severely impact the program and its future growth.

2. SUMMARY OF ASSESSMENT (about 1 page)

A. Program Student Learning Outcomes

Students graduating with an M.S. in Computer Science from CSU East Bay will be able to:

1. apply advanced computer science theory to computational problems
2. demonstrate advanced understanding of the mechanisms, components and architecture of current computing systems
3. apply emerging technologies and advanced algorithmic design
4. critique, plan and produce complex software applications
5. research and analyze current computer science literature

B. Program Student Learning Outcome(s) Assessed

Following our assessment plan, we are closing the loop on PLO #3 this year. The department does collect assessment data for all targeted courses each year, however, so as to track trajectories for scores on all PLOs.

Post-assessment quizzes were administered for three courses (addressing PLO #3):

CS 6260 (Computational Complexity – Core Requirement) Developing PLO 1 and PLO 3
CS 6560 (Operating Systems Design – Core requirement) Developing PLO 2 and PLO 3
CS 6901 (Graduate Capstone) Mastering PLO 1, PLO 2, and PLO 3

C. Summary of Assessment Process

The Department created SLOs and PLOs for the Master in Computer Science program in the academic year 2012-2013. The Department made the decision to use Blackboard as a means to provide students with an assessment exam that addresses the SLOs of each course. The SLOs for each course have been mapped to the program PLOs and the ILOs of the university. The assessment exams were developed for the required courses in the program, as well as a representative set of elective courses. The assessment instruments were then made available to the department faculty via a BlackBoard repository. Instructors teaching courses which were to be assessed in a given year deployed the tests and reported the results back to the Graduate Coordinator.

The Department has been using this assessment mechanism for three years now and can evaluate its advantages and disadvantages. Unfortunately, evaluating the results of the assessment exams as they stand is challenging, as each assessment contains questions addressing multiple PLOs. Due to a BlackBoard limitation, the results for individual PLOs cannot be automatically aggregated and compared across multiple courses, and instead must be tabulated by hand. To solve this problem, for the semester-based program, the Department agreed to develop assessment instruments that address only one PLO at a time. This will allow assessment to be automated, providing the opportunity to assess more courses, and assess those courses more frequently. The Department will be developing the new assessment instruments for the semester-based program beginning next year.

An additional challenge in the current system is assessing PLOs for both the Master's in Computer Science program and the Master's in Computer Network program. Since the programs share the great majority of the courses, but have different PLOs, it has been necessary to provide separate mappings of course SLOs to the PLOs of the two different programs, or to include additional questions on the assessment instruments to address the different PLOs. Fortunately, this difficulty will be eliminated under the semester-based program as the Master's in Computer Science and the Master's in Computer Networks have been combined into a single program with common PLOs.

In evaluating the assessment scores for PLO #3, we find very positive results in three of the four courses assessed. CS 6260, CS 6560, and CS 6901 are required courses for all students, except a small number of students who elect to complete a thesis rather than taking the comprehensive exams (CS 6901.) As a result, these assessment scores reflect the performance of the program as a whole in achieving PLO #3.

In regards to closing the loop and using the results of the assessment process to improve student learning for PLO #3, it would appear that the CS 6260 and CS 6560 need only fine tuning, while CS 6901 instructors could attempt to address consistency. In Fall 2015, CS 6901 students successfully demonstrated their proficiency regarding PLO #3, while Spring 2016 students were much less successful.

CS 6260:
This required course in Computational Complexity addresses material which is well-defined and theoretical in nature. It requires advanced understanding of algorithmic design, which is why it particularly addressed PLO #3. There is little room for addressing different or additional material, so fine-tuning would be aimed towards providing additional learning opportunities for students who were unsuccessfully served by the current class format. These opportunities might include high impact educational practices such as collaborative projects or swapped classrooms.

CS 6560:
This required course in Operating Systems Design addresses material which has a well-defined core of material but is constantly being enhanced by new research and advances in the industry. It clearly maps to both parts of PLO #3, in applying emerging technologies, and requiring advanced knowledge of algorithmic design. While perhaps two thirds of the course material is necessary for all students in the program, the remaining concepts may be presented through a number of avenues. Study of existing commercial operating systems, review of seminal research papers, and creation of software artifacts may all provide opportunities for learning the desired concepts. All of these avenues are included in the current incarnation of the course, with some students finding one or another more useful in providing a grasp of the necessary ideas. The instructor regularly selects or removes research papers for review as they prove more or less successful in delivering the needed material. Programming assignments are also modified or enhanced as required. This sort of fine tuning may provide for even better student success.

CS 6901:
This course comprises the capstone examinations, in which students take tests in 6 different areas of Computer Science. The material is well-defined and requires mastery of both theoretical and applied concepts. All material has been developed in earlier courses in the program, but here must be mastered outside of the structure of the earlier courses and shown to be current and comprehensive. To help prepare students for this evaluation, instructors typically run study groups throughout the quarter in which students solve problems from older exams, writing proofs, and designing software solutions to problems. As can be seen from the Fall 2015 results, students can successfully achieve the PLO goal via this mechanism. The poorer performance in Spring 2016 may be due to the composition of the study groups, the student need for different learning avenues, or another unknown factor. Since study groups are usually formed by the students themselves, it may be that less proficient students joined the same groups, and then did not benefit from the more experience of the more proficient students. This could be addressed by the instructor by assigning the students to groups, or by using BlackBoard functionality to allow groups to easily share their work. Wikis or blogs could also help support students who are less comfortable in the study group environment, and could learn more effectively using online tools. The Department will suggest these modifications to instructors in the coming year to enhance the learning opportunities for students, and attempt to provide consistently excellent student performance on PLO #3.

D. Summary of Assessment Results

2015-2016 Assessment Results	**1**	**2**	**3**	**4**	**5**
CS 6260 Computational Complexity	**7.8**		**8.7**		
CS 6560 Operating Systems Design		**8.9**	**8.9**		
CS 6901 Graduate Capstone (Fall 2015)	**6.7**	**6.8**	**8.2**		
CS 6901 Graduate Capstone (Spring 2016)	**4.6**	**8.1**	**6.4**		

<u>3. STATISTICAL DATA</u> (about 1 page)

<u>Student Demographics:</u> Gender breakouts of demographic data for Fall 2015 were not provided by Institutional Research.

M.S. Computer Science		**Fall 2009**	**Fall 2010**	**Fall 2011**	**Fall 2012**	**Fall 2013**	**Fall 2014**	**Fall 2015**
Female	**Black, non-Hispanic**							
	American Indian or Alaska Native		1					
	Asian	24	14	7	5	7	4	5

	Pacific Islander							
	Hispanic	1	1	1	1			
	White	2	6	5	2	2	2	2
	Multiple ethnicity		2					
	Race/ethnicity unknown	8	12	6	3	1	1	1
	Nonresident aliens	51	59	56	43	59	108	87
Male	**Black, non-Hispanic**	3	2	1	2	3	3	1
	American Indian or Alaska Native							
	Asian	6	9	11	5	4	4	10
	Pacific Islander							
	Hispanic		2	1	1	1		
	White	7	12	10	10	3	2	7
	Multiple ethnicity					1	1	1
	Race/ethnicity unknown	10	5	4	3	5	3	3
	Nonresident aliens	71	59	46	30	66	94	66
Total	**Black, non-Hispanic**	3	2	1	2	3	3	1
	American Indian or Alaska Native		1					
	Asian	30	23	18	10	11	9	15
	Pacific Islander							
	Hispanic	1	3	2	2	1		
	White	9	18	15	12	5	3	9
	Multiple ethnicity		2			1	1	1
	Race/ethnicity unknown	18	17	10	6	6	4	4
	Nonresident aliens	122	118	102	73	125	202	153

Annual Data:

A. Student Headcount:

		Fall Quarter				
Headcount Enrollment	**2010**	**2011**	**2012**	**2013**	**2014**	**2015**
Computer Science						
1. Undergraduate	260	290	315	326	355	424
2. Postbaccalaureate	8	4	0	1	1	0
3. Graduate	184	148	105	152	222	183
4. Total Number of Majors	452	442	420	479	578	607
Computer Network						
1. Undergraduate	0	0	0	0	0	0
2. Postbaccalaureate	0	0	0	0	0	0
3. Graduate	35	25	26	53	57	37
4. Total Number of Majors	35	25	26	53	57	37

B. Degrees Awarded:

		College Years				
Degrees Awarded	**09-10**	**10-11**	**11-12**	**12-13**	**13-14**	**14-15**

Computer Science						
1. Undergraduate	41	54	51	63	74	72
2. Graduate	62	102	89	57	31	59
3. Total Number of Majors	103	156	140	120	105	131
Computer Network						
1. Undergraduate	0	0	0	0	0	0
2. Graduate	5	12	16	12	18	17
3. Total Number of Majors	5	12	16	12	18	17

Over the past three quarters, a total of 59 students received their MS degrees. This is a significant increase from the 31degrees awarded last year and similar to the 57 awarded the year before. It also reflects the typical pace of our students. Most take 2 courses per quarter, and many are required to take prerequisite courses that do not count towards their degree. It is common for students to take 2 or more years to complete the degree requirements, so the rise in degrees awarded appears to match the rise in the number of students in the program two years ago.

Accepted Applications for MS Computer Science (Data is from Graduate Coordinator records):

Fall 2015:
There were 996 applicants, of whom 130 were accepted into the program. Of these, 30 students enrolled in the program. This is an acceptance rate of 13% and a yield of 3%.
Winter 2016:
There were 229 applicants, of whom 71 were accepted into the program. Of these, 15 students enrolled in the program. This is an acceptance rate of 31% and a yield of 6.5%.
Spring 2016:
There were 306 applicants, of whom 109 were accepted into the program. Of these, 20 students enrolled in the program. This is an acceptance rate of 35% and a yield of 6.5%.

As can be seen, even when admission rates were greatly increased in Winter and Spring quarters, yield of applicants to enrollees was very low. We hope that, by streamlining the admissions process, we can increase the yield, enrolling more students with less administrative overhead.

C. Faculty Information:

Please note that the university does not calculate separate data for the Math and Computer Science programs.
Please see above (Program Needs) for information on Computer Science and Computer Network faculty.

	Fall Quarter					
	2010	**2011**	**2012**	**2013**	**2014**	**2015**
C. Faculty						
Tenured/Track Headcount	Computer Science and Mathematics Combined					
1. Full-Time	25	25	23	21	22	19
2. Part-Time	4	2	1	1	2	3
3a. Total Tenure Track	29	27	24	22	24	22

3b. % Tenure Track	80.6%	62.8%	58.5%	52.4%	57.1%	48.9%
Lecturer Headcount	Computer Science and Mathematics Combined					
4. Full-Time	1	1	1	2	2	4
5. Part-Time	6	15	16	18	16	19
6a. Total Non-Tenure Track	7	16	17	20	18	23
6b. % Non-Tenure Track	19.4%	37.2%	41.5%	47.6%	42.9%	51.1%
7. Grand Total All Faculty	36	43	41	42	42	45
Instructional FTE Faculty (FTEF)	Computer Science and Mathematics Combined					
8. Tenured/Track FTEF	22.4	19.4	16.5	17.4	17.0	21.0
9. Lecturer FTEF	11.1	18.1	19.0	19.3	18.4	13.7
10. Total Instructional FTEF	33.5	37.4	35.4	36.7	35.4	34.7
Lecturer Teaching	Computer Science and Mathematics Combined					
11a. FTES Taught by Tenure/Track	439.1	307.1	288.1	314.9	356.4	260.3
11b. % of FTES Taught by Tenure/Track	58.7%	38.7%	36.0%	36.2%	39.4%	28.3%
12a. FTES Taught by Lecturer	308.5	487.1	513.2	553.9	547.3	660.0
12b. % of FTES Taught by Lecturer	41.3%	61.3%	64.0%	63.8%	60.6%	71.7%
13. Total FTES taught	747.7	794.2	801.3	868.7	903.7	920.4
14. Total SCU taught	11215.0	11913.0	12019.0	13031.0	13566.0	13806.0

D. Student Faculty Ratios:

	Fall Quarter					
	2010	**2011**	**2012**	**2013**	**2014**	**2015**

D. Student Faculty Ratios	Computer Science					
1. Tenured/Track	16.8	14.7	17.1	19.4	20.9	23.1
2. Lecturer	26.4	23.6	27.5	30.2	29.2	25.6
3. SFR By Level (All Faculty)	17.5	15.5	18.5	21.5	23.0	24.0
4. Lower Division	24.6	22.5	20.8	24.9	28.9	26.2
5. Upper Division	17.0	17.5	20.2	21.4	23.8	23.2
6. Graduate	15.9	10.1	14.5	19.8	19.9	23.6

E. Sections:

	Fall Quarter					
	2010	**2011**	**2012**	**2013**	**2014**	**2015**

E. Sections	Computer Science					
1. Number of Sections Offered	39.7	47.8	37.0	45.8	49.0	47.0
2. SCU taught	3016.0	2962.0	3054.0	3938.0	4556.0	3872.0
3. Average Section Size	21.1	17.8	20.9	22.5	24.0	22.9
4. Average Section Size for LD	33.5	26.4	29.5	27.0	27.0	25.9
5. Average Section Size for UD	20.2	18.8	21.4	22.9	25.5	22.5
6. Average Section Size for GD	18.7	12.5	15.5	19.5	21.1	21.3

7. LD Section taught by Tenured/Track	4	5	5	5	2	2
8. UD Section taught by Tenured/Track	18	21	19	21	15	14
9. GD Section taught by Tenured/Track	16	18	10	12	16	12
10. LD Section taught by Lecturer	0	2	1	3	7	8
11. UD Section taught by Lecturer	1	0	3	5	5	5
12. GD Section taught by Lecturer	2	3	3	3	4	6

CALIFORNIA STATE UNIVERSITY EAST BAY

College of Science (CSCI)
North Science 135
25800 Carlos Bee Boulevard, Hayward CA 94542

2015-2016 CSCI EETF Assessment Year End Report, June, 2016

Program Name(s)	EETF Faculty Rep	Department Chair
M.S. Computer Science	Matt Johnson	Matt Johnson

[NOTE: Items A, B, C, and D are identical to your Page 2 on your Annual Report for CAPR. Please simply cut and paste from there. Item E is unique to the CSCI EETF.]

A. Program Student Learning Outcomes

Students graduating with an M.S. in Computer Science from CSU East Bay will be able to:

1. apply advanced computer science theory to computational problems
2. demonstrate advanced understanding of the mechanisms, components and architecture of current computing systems
3. apply emerging technologies and advanced algorithmic design
4. critique, plan and produce complex software applications
5. research and analyze current computer science literature

B. Program Student Learning Outcome(s) Assessed

Following our assessment plan, we are closing the loop on PLO #3 this year. The department does collect assessment data for all targeted courses each year, however, so as to track trajectories for scores on all PLOs.

Post-assessment quizzes were administered for three courses (addressing PLO #3):

CS 6260 (Computational Complexity – Core Requirement) Developing PLO 1 and PLO 3
CS 6560 (Operating Systems Design – Core requirement) Developing PLO 2 and PLO 3
CS 6901 (Graduate Capstone) Mastering PLO 1, PLO 2, and PLO 3

C. Summary of Assessment Process

The Department created SLOs and PLOs for the Master in Computer Science program in the academic year 2012-2013. The Department made the decision to use Blackboard as a means to provide students with an assessment exam that addresses the SLOs of each course. The SLOs for each course have been mapped to the program PLOs and the ILOs of the university. The assessment exams were developed for the required courses in the program, as well as a representative set of elective courses. The assessment instruments were then made available to the department faculty via a BlackBoard repository. Instructors teaching courses which were to be assessed in a given year deployed the tests and reported the results back to the Graduate Coordinator.

The Department has been using this assessment mechanism for three years now and can evaluate its advantages and disadvantages. Unfortunately, evaluating the results of the assessment exams as they stand is challenging, as each assessment contains questions addressing multiple PLOs. Due to a BlackBoard limitation, the results for individual PLOs cannot be automatically aggregated and compared across multiple courses, and instead must be tabulated by hand. To solve this problem, for the semester-based program, the Department agreed to develop assessment instruments that address only one PLO at a time. This will allow assessment to be automated, providing the opportunity to assess more courses, and assess those courses more frequently. The Department will be developing the new assessment instruments for the semester-based program beginning next year.

An additional challenge in the current system is assessing PLOs for both the Master's in Computer Science program and the Master's in Computer Network program. Since the programs share the great majority of the courses, but have different PLOs, it has been necessary to provide separate mappings of course SLOs to the PLOs of the two different programs, or to include additional questions on the assessment instruments to address the different PLOs. Fortunately, this difficulty will be eliminated under the semester-based program as the Master's in Computer Science and the Master's in Computer Networks have been combined into a single program with common PLOs.

In evaluating the assessment scores for PLO #3, we find very positive results in three of the four courses assessed. CS 6260, CS 6560, and CS 6901 are required courses for all students, except a small number of students who elect to complete a thesis rather than taking the comprehensive exams (CS 6901.) As a result, these assessment scores reflect the performance of the program as a whole in achieving PLO #3.

In regards to closing the loop and using the results of the assessment process to improve student learning for PLO #3, it would appear that the CS 6260 and CS 6560 need only fine tuning, while CS 6901 instructors could attempt to address consistency. In Fall 2015, CS 6901 students successfully demonstrated their proficiency regarding PLO #3, while Spring 2016 students were much less successful.

CS 6260:

This required course in Computational Complexity addresses material which is well-defined and theoretical in nature. It requires advanced understanding of algorithmic design, which is why it particularly addressed PLO #3. There is little room for addressing different or additional material, so fine-tuning would be aimed towards providing additional learning opportunities for students who were unsuccessfully served by the current class format. These opportunities might include high impact educational practices such as collaborative projects or swapped classrooms.

CS 6560:
This required course in Operating Systems Design addresses material which has a well-defined core of material but is constantly being enhanced by new research and advances in the industry. It clearly maps to both parts of PLO #3, in applying emerging technologies, and requiring advanced knowledge of algorithmic design. While perhaps two thirds of the course material is necessary for all students in the program, the remaining concepts may be presented through a number of avenues. Study of existing commercial operating systems, review of seminal research papers, and creation of software artifacts may all provide opportunities for learning the desired concepts. All of these avenues are included in the current incarnation of the course, with some students finding one or another more useful in providing a grasp of the necessary ideas. The instructor regularly selects or removes research papers for review as they prove more or less successful in delivering the needed material. Programming assignments are also modified or enhanced as required. This sort of fine tuning may provide for even better student success.

CS 6901:
This course comprises the capstone examinations, in which students take tests in 6 different areas of Computer Science. The material is well-defined and requires mastery of both theoretical and applied concepts. All material has been developed in earlier courses in the program, but here must be mastered outside of the structure of the earlier courses and shown to be current and comprehensive. To help prepare students for this evaluation, instructors typically run study groups throughout the quarter in which students solve problems from older exams, writing proofs, and designing software solutions to problems. As can be seen from the Fall 2015 results, students can successfully achieve the PLO goal via this mechanism. The poorer performance in Spring 2016 may be due to the composition of the study groups, the student need for different learning avenues, or another unknown factor. Since study groups are usually formed by the students themselves, it may be that less proficient students joined the same groups, and then did not benefit from the more experience of the more proficient students. This could be addressed by the instructor by assigning the students to groups, or by using BlackBoard functionality to allow groups to easily share their work. Wikis or blogs could also help support students who are less comfortable in the study group environment, and could learn more effectively using online tools. The Department will suggest these modifications to instructors in the coming year to enhance the learning opportunities for students, and attempt to provide consistently excellent student

performance on PLO #3.

D. Summary of Assessment Results

2015-2016 Assessment Results	1	2	3	4	5
CS 6260 Computational Complexity	**7.8**		**8.7**		
CS 6560 Operating Systems Design		**8.9**	**8.9**		
CS 6901 Graduate Capstone (Fall 2015)	**6.7**	**6.8**	**8.2**		
CS 6901 Graduate Capstone (Spring 2016)	**4.6**	**8.1**	**6.4**		

E. Suggestions and Recommendations for the CSCI EETF in the Future

NONE