**ABET**

**Q2S Conversion Report**

**for the**

**Bachelor of Science Degree in**

****

**at**

**California State University, Los Angeles**

**Los Angeles, CA**

**Submitted January 10, 2016**

**(Updated November 1, 2016)**

Table of Contents

**INTRODUCTION. 3**

[**CRITERION 1. STUDENTS**](#Criterion1) **3**

A. Q2S Transitional Advising

A.1 Infrastructure to Support Q2S Advising 3

A.2 College and Departmental Q2S Advising Process 5

A.3 ESSC Undergraduate Advisement 6

A.4 Department Undergraduate Advisement 6

B. Graduation Requirements 7

**CRITERION 2. PROGRAM EDUCATIONAL OBJECTIVES 11**

**CRITERION 3. STUDENT OUTCOMES 12**

**CRITERION 4. CONTINUOUS IMPROVEMENT 13**

**CRITERION 5. CURRICULUM 14**

A. Program Curriculum 14

A.1. Course requirements 14

A.2. Course prerequisite(s) requirements 22

A.3. Capstone courses 22

A.4. Course coverage of required areas 23

A.5. Attainment of Student Learning Outcomes 25

**CRITERION 6. FACULTY 26**

**CRITERION 7. FACILITIES 27**

**Appendix- Sample Individualized Advisement Plan 28**

# INTRODUCTION

**­­**California State University, Los Angeles is now in the process of converting from a Quarter system of three quarters per academic year (10 weeks of educational instruction per term), to a semester system of two semesters per academic year (15 weeks of educational instruction per term) beginning Fall semester 2016. The Quarter-to-Semester (Q2S) conversion involves significant curriculum change, which makes transitioning advising critical for students who are admitted under Quarter System but need to graduate in Semester System. This documentation describes how the new Computer Science undergraduate program in Semester system still adheres to current ABET criteria. Specifically this document addresses ABET Criterion 1 (Student) and Criterion 5 (Curriculum), as both are highly related to Q2S conversion process.

# CRITERION 1. STUDENTS

Quarter-to-Semester (Q2S) conversion will certainly impact students in multiple ways. To ensure a smooth Q2S transition experience for our students, the University, College, and departments have made significant efforts to develop effective advising tools, implement Transitioning Advisement process, and streamline the graduate application and graduation check process. As a result, students will 1) have a good understanding of the curriculum change and receive help to plan for their academic schedule accordingly; 2) develop an Individualized Advisement Plan (IAP) to lay out a roadmap to graduation that takes into the curricular change into account; 3) have a smooth and effective graduation check process to ensure they meet graduation requirements in the degree program.

## Q2S Transitioning Advising

**A.1 Infrastructure to Support Q2S Advising**

To oversee the Q2S conversion process, Office of Semester Conversion was established by the University. This campus unit works with Colleges and University Curricular and Advisement committees to provides support and guidelines for both curriculum conversion and transitioning advisement. Office of Semester Conversion also implemented multiple tools such as Conversion Guide and Degree Planner to facilitate curriculum mapping, scheduling and advising in Q2S process. In particular, the Degree Planner is an added feature on GET (Golden Eagle Territory) to enable the development of individualized advisement plan (IAP). IAP maps out all the academic requirements term by term (ongoing Quarters or future Semesters) and serves as an individualized road map for a timely graduation. Training materials (handouts, video tutorials, and workshops) have been produced by Semester Conversion Office for faculty/staff advisors and students regarding IAP development. Bountiful information and supporting tools are provided on Q2S website to assist both advisors and students in Q2S transitioning process (<http://www.calstatela.edu/semesterconversion>).

At college level, Q2S Advising Coordinator was appointed to work closely with the associate Dean, department Chairs, and faculty/staff advisors to oversee the implementation of Q2S transitioning advisement process. In the College of Engineering, Computer Science and Technology, Advising Taskforce was formed in 2014 to proactively discuss students’ needs in Q2S conversion process and develop college level guidelines and strategies to implement Q2S transitioning advising. The members of Advising Taskforce include department chairs and principal faculty advisors, staff advisors from ECST Student Success Center, and the college Q2S Advising Coordinator. During the two-year Q2S conversion period, the Advising Taskforce has been instrumental to establish a collaborative advising model and an effective implementation process of IAP in the format of *IAP Clinic* where a group of students receive guidance from both faculty and staff advisors to develop IAP and have IAP reviewed and approved in a streamlined fashion. Table 1.1 lists the current members in ECST Advising Taskforce.

|  |  |  |
| --- | --- | --- |
| **Department** | **Faculty Advisor** | **Staff Advisors in ESSC** |
| Computer Science | Elaine Kang (Q2S Advising Coordinator)  Raj Pamula (Chair) | Frances Hidalgo  (Director)  Evelyn Crosby  T. Fox  Candi Marsh  Rodolfo Ramirez |
| Civil Engineering | Rupa Purasinghe  Mark Tufenkjian (Chair) |
| Electrical Engineering | Arash Jamehbozorg  Fred Daneshgaran (Chair) |
| Mechanical Engineering | Darrell Guillaume (Chair)  David Raymond |
| Technology | Keith Mew (Chair)  Stephanie Nelson  Mauricio Castillo |

Table 1.1 ECST Advising Taskforce Members (2015-2016)

At the department level, the Chair and designated faculty advisor (Dr. Elaine Kang) worked closely with ECST Student Success Center and Q2S Advising Coordinator to organize major-specific IAP clinics, to reach out to students and prepare them for IAP clinics, and develop essential tools for Q2S transitioning advising. The following is the list of advising tools that help students to understand the curriculum change in ME and guide them to develop IAP:

1. *Major Curriculum Conversion Map*: this is a critical tool to show the mapping between quarter and semester curriculum. Using the Conversion Map, students can easily see the equivalent semester course for the one in quarter system and vice-verse.
2. *ECST GE Conversion Map*: this map allows students to understand the different GE requirements in Quarter and Semester systems;
3. *Major Roads Maps (in quarter/semester)*: the road maps were developed by major advisors and serve as crucial reference for students to develop IAP, their individualized roadmap towards graduation;
4. *Projected Course Schedule in Semester*: this allows students to layout their IAP based on the actual timeline of course offering;
5. *Step-by-step Guide to Develop IAP*: developed collaboratively by the major advisors and Student Success Center, this Guide provides easy-to-follow steps for students to use Degree Planner in GET to develop their IAP.

The above Q2S transitioning advising tools as well as the IAP clinic information can be accessed on College Q2S website (<http://www.calstatela.edu/ecst/ecst-semester-conversion>).

## A.2. College and Departmental Q2S Advising

Effective planning to ensure timely completion of university and degree program requirements is of particularly importance throughout the transition process from quarter to semester system and thereafter.  The infrastructure described in section A.1 enables timely and informative support to all students who began their academic career under quarters and will complete it under semesters. This subsection will present the detailed Q2S advising process for Computer Science students.

In general, the Q2S advising process consists of the following steps:

1. The Semester Conversion Office works with Registrar Office to identify potential students who will be affected by Q2S transition based on their degree progress;
2. The department receives the student information from the Semester Conversion Office and inform the students by email (and telephone reminders) to register and attend one of the IAP clinics;
3. The department arranges multiple IAP clinics to conduct Q2S transiting advising in group settings. In Fall 2015, Computer Science offered four IAP clinics at different dates/time, so students can select one based on their availability. By Jan 19 2016, 43.72% CS students have got their IAPs completed, reviewed and approved by the advisors; we expect the remainder of the students to complete their planning in winter and spring quarters. An example of a completed IAP is shown in the Appendix.

Students who began their academic programs under the requirements of the quarter system (i.e., prior to Fall 2016) may choose to complete their requirements with semester course equivalents or they may switch to the new semester program requirements in place. During IAP Clinic, advisors will help students to review their degree progress and help them to determine if they should stay with Quarter curriculum or switch to Semester. Since IAP clinic plays a critical role in Q2S transitioning advising, its flow is elaborated below:

* Advisor (IAP clinic facilitator) explains the function of IAP to help students in Q2S transition as well as the steps of IAP development.
* Students review their degree progress with advisors and determine the remaining requirements.
* With the guidance from advisors, students decide on the most efficient GE catalog year (quarter versus semester requirements).
* With the guidance from advisors, students select the most efficient major catalog year (quarter versus semester requirements for the major).
* Using Degree Planner, students select courses to take in future quarter/semester and generate an academic plan to meet all remaining graduation requirements.
* Both faculty and staff advisors are on site to provide timely help to students’ questions regarding both GE and major courses.
* Advisors help students to review their IAPs ensuring that the IAP meets the degree requirements under the selected catalog date.
* Competed IAPs are approved on spot.

The IAP becomes a dynamic document that can be updated by the students during any future advisement visits with an academic advisor. In addition to IAP clinics, the college and department continues to offer comprehensive advisement in one-to-one basis to answer students’ questions about Q2S transition, transfer credit evaluation, career consulting, graduation requirements, etc. In general, incoming freshman students receive mandatory advising provided by Student Success Center (ESSS) in their first two years; while transfer students and junior/senior students are advised by faculty advisors in the department. The following sections summarize the advisement service provided by ESSS and the department respectively.

## A.3. College (ESSC) Undergraduate Advisement

The College of ECST through ESSS provides an array of services to incoming freshman and second year students in all majors. Advisement is mandatory every quarter for the first two years. These services include: academic advisement through professional staff and peer advisors, the engagement of freshman in learning communities and academic excellence workshops.

* Students must make an appointment in advance to guarantee advisement time with a Staff Advisor or Peer Advisor. To make an appointment:
  + Visit the office located in the Engineering and Technology building, A-127  or
  + Call the office front desk at 323-343-4574, or
  + Visit <http://www.calstatela.edu/ecst/success/advising>

## A.4. Department Undergraduate Advisement

All students will receive academic advisement at the department level to help them make informed academic choices. Both CSNS (Computer Science Network Services, a department level management system) and GET (Golden Eagle Territory, a university record keeping system) are used during the advisement process. The advisor and student go over the student’s degree progress data available on CSNS and GET. The advisor documents the visit on CSNS and relevant changes are reflected in the IAP on GET.

Normally, student advising in the department is facilitated in a number of ways. It can be categorized into two categories: Open Advisement and Mandatory Advisement.

Open Advisement:

1. Advisor/staff office visitation: Office hours for the Advisor and the Department chair are posted in the Department Office every quarter. Students meet with the faculty advisor to evaluate class work to date, to discuss issues (if any) impacting their present load, to resolve any GPA issues, and to plan subsequent classes. Students also seek the advice of any faculty in evaluating career choices.
2. Email advisor: Students can seek to get clarifications from the advisor or staff concerning any particular queries. Email is often used by students who need immediate clarification.
3. CSNS Advisement Forum: Various topics exist that help clarify generic questions posed by the students. These are replied to either by other students or the advisor and are monitored by the Department staff and advisor.

Mandatory Advisement:

Every student has to discuss academic plans with the advisor at least once a year. Students individually meet with the advisor to plan upcoming schedule and formulate a road map to complete all the remaining requirements. The mandatory visits are scheduled as follows: (i) Entering First Time Freshmen or Transfer Students, (ii) Targeted advising at least once a year.

1. Entering first term students: Admitted freshmen and transfer students attend university-sponsored advising sessions, which are attended by the Department chair and program advisors. Students are informed of the degree requirements, course pre-requisites, laboratory access, computer-related student clubs, and advising process. Students are given information about the program requirements that are posted on the online university catalog and the Department website. Students are also given information in the form of a Undergraduate Student Handbook, which is also posted on the Department website.
2. Freshmen level in CS1010: CS1010 is required to be taken by every student within the first terms as entering freshmen or a first year transfer students. CS1010 gives a comprehensive overview of higher education. Topics covered include: University rules and regulations; general education requirements; major requirements; evaluate transfer units; sample road maps; individualized quarterly planners; graduation checks. Instructional videos used in the CS1010 class are always accessible even after completion of the course.
3. Sophomore level in CS2011: CS2011 is a required course usually taken as the first required course at the sophomore level. Students are then advised to keep pace with the Math and physics requirements along with CS requirements. Students Individualized Roadmaps are checked and given clearances to proceed with their roadmap.
4. Junior level in CS3112: CS3112 is usually taken as the first required course at the junior level. Students roadmaps are checked to see if they are on pace to take the senor design next year. Students also make any adjustments to the planned electives.
5. Senior level in CS4961. CS4961 is the front end of the senior design sequence. Students do a graduation check with the advisor to ensure that they are on track to graduate by the end of the year.
6. **Graduation Requirements**

Since students admitted prior to Fall 2016 can select their catalog year (Quarter or Semester), their graduation requirements differ with their decisions. For students who switch to Semester, in order to graduate with a Bachelor of Science degree in Computer Science, they must complete 120 units and have grade point averages of at least 2.0 in all course work attempted as an undergraduate (including transfer credit), in all course work attempted at CSULA, and in all course work attempted in the major. Students are continuously evaluated from the time they apply for admission to the University until they graduate. These evaluations occur in every situation in which a student must attain a satisfactory ("passing") result to make progress toward graduation. Each evaluation plays a role in ensuring that the program educational objectives and student learning outcomes for the Computer Science program are met.

In Q2S transition process, the Academic Advisement Requirement Module (CAAR) on the graduation tracking system in the GET continues to play a critical role to monitor the student degree progress and to track if a student satisfies the graduation requirement. In Spring 2015, CAAR template reflecting graduation requirements Computer Science program was created and tested. Course mapping between quarter and semester systems are enabled based on the Curriculum Conversion table provided by the department. Using “What-if report” on GET system, students and advisors can easily check if graduation requirements are met under either quarter or semester curriculum.

The graduation check process stays the same for the most parts, however, the availability of IAP helps to make this process easier. Figure 1.1 shows the updated graduation check process which utilizes IAP to simplify academic planning for the remaining quarter/semester and uses “what-if” report to ensure the student meets the graduation requirements with the plan specified in IAP.

**Apply for graduation**

Check CAAR on GET

Fix Problems

Planning

Run “what-if” to ensure meeting the requirement

* **Evaluate progress towards degree**
* **Identify missing courses/requirement**

**e.g. Course substitution for equivalent transfer credit**

**IAP**

**Figure 1.1 An illustration of graduation check process on GET**

Except for transfer credits granted for equivalent coursework, no substitution for required courses is allowed without very serious and compelling reasons. Such substitutions are studied carefully and require the approval of the student's advisor and the department chair. All students are formally evaluated two-quarter or one semester before their anticipated graduation. The graduation check process ensures that students have satisfied all curricular requirements as described below:

* + - * 1. Students initiate this process by filling out an Application for Graduation (degree check) on a form available at the Cal State L.A. Graduation Office website, academic department/division/school, and Enrollment Services in Administration 146. (Filing periods are published in the Graduation information section of the Schedule of Classes.)



**Figure 1.2: Academic Requirements Report on GET**

* + - * 1. The application is to be filed two quarters prior to the end of the term of their expected graduation. Students are able to access their Academic Requirements Report at any time thorough their college career. The Academic Requirements Report (an audit report generated on GET) gives a visually clear picture of the requirements; completed requirements; transfer credits etc. as shown in Figure 1.2. Students thus have a general idea of the graduating quarter and discuss the program requirements with the advisor.
        2. Students submit their completed application to the Department for processing. The undergraduate faculty advisor discusses the Academic Progress Report with the student which indicates clearly all the requirements that have been completed and flags the remaining requirements. The student and advisor draw up a plan to meet the remaining requirements in the coming two quarters.
        3. The Department staff updates the standing of the student to Graduation Check standing on CSNS. This allows the staff to check on the student’s remaining requirements in the coming quarters. The Department staff then forwards the graduation application to the Graduation Office for an official audit.
        4. The Graduation Office of the University Registrar’s Office has the sole authority to audit and certify that a candidate for graduation has fulfilled her/his approved program requirements. Graduation check audit results are sent by surface mail to the students prior to their final anticipated quarter. Students who are enrolled in the quarter they expect to graduate but do not complete all degree requirements will have their graduation application automatically transferred to the following quarter.
        5. Students who do not finish their requirements by the end of the second quarter must reapply and restart the graduation application procedure.
        6. Degree dates are posted at the end of the quarter in which all requirements are met. On completion of the degree requirements, the transcripts on GET indicate the following:

Degree : Bachelor of Science

Confer Date : xxxx-xx-xx

Plan : Computer Science

An example is shown below in Figure 1.3



**Figure 1.3: Degree Awarded on GET**

* + - * 1. After the degree is posted on GET, CSNS imports the list of all graduated students and updates their standing to an alumni standing.

# CRITERION 2. PROGRAM EDUCATIONAL OBJECTIVES

The program educational objectives stay the same in the quarter to semester transition process.

Program Educational Objectives are broader statements that will describe what graduates are expected to attain within a few years of graduation. CSULA Computer Science Undergraduate Program Educational Objectives are listed below:

1. Students who enter the workforce will have established themselves as effective professionals by having solved real problems through the use of their computer science knowledge and their communication, critical thinking, and problem-solving skills.

2. Students who continue in academia will have been successful in pursuing advanced degrees and in demonstrating their ability to master advanced areas of computer science.

3. Students will have demonstrated their ability to adapt to a rapidly changing environment by having learned and applied new skills and new technologies.

# CRITERION 3. STUDENT OUTCOMES

The program student outcomes stay the same in the quarter to semester transition process.

Student Learning Outcomes are specific skills that the students will possess on completion of the degree program. Cal State LA Computer Science Undergraduate Student Outcomes are:

1. Students will be able to apply concepts and techniques from computing and mathematics to both theoretical and practical problems.

2. Students will be able to demonstrate fluency in at least one programming language and acquaintance with at least three more.

3. Students will have a strong foundation in the design, analysis, and application of many types of algorithms.

4. Students will have a fundamental understanding of computer systems.

5. Students will have the training to analyze problems and identify and define the computing requirements appropriate to their solutions.

6. Students will have the training to design, implement, and evaluate large software systems working both individually and collaboratively.

7. Students will be able to communicate effectively orally and in writing.

8. Students will have the knowledge, skills, and attitudes for lifelong self-development.

9. Students will have the ability to analyze the local and global impact of computing on individuals and society.

10. Students will a fundamental understanding of social, professional, ethical, legal, and security issues in computing.

# CRITERION 4. CONTINUOUS IMPROVEMENT

Continuous improvement of BSCS program is achieved through the established program assessment processes. Q2S conversion provided an excellent opportunity for us to re-examine and strengthen our curriculum. During the curriculum conversion process, several modifications have been made based on the feedback from our main constituencies to enhance crucial student outcomes. This section summarizes the primary changes made in semester curriculum to achieve continuous improvement of BSCS program.

* Introduction to Higher Education for Computer Science Majors: During Q2S conversion, a new 3-unit CS1010 course is developed to replace the 2-unit CS101 course in quarter system. The new course contains a 2-unit lecture and 1-unit lab, which will expose the incoming freshman students to the skills required for the computing profession; It will also introduce hands-on projects to explore the computing disciplines.CS1010 also meet the GE block E (life-long learning) requirements defined in the newly established university GE policy for semester system.
* 2+1 (3 unit) courses to facilitate project-based learning: Project-based learning has proved to be an effective way to develop students’ design skills. During the Q2S conversion, this pedagogy is extended to multiple lower division and upper division required courses.
* Introductory program sequence: To emphasize the importance of the introductory programming sequence, the three quarter programming sequence (CS201, CS202, CS203) is expanded to a three semester programming sequence (CS2011, CS2012, CS2013).
* ENGL2030 (Introduction to Professional and Technical Writing): This course is co-developed by ENGL and engineering departments to meet the second composition course outcomes while equipping students with essential technical writing skills. In this course, students will learn various writing strategies; advance their knowledge on common persuading techniques; perform extensive literature reviews on subjects within and beyond engineering; and learn proper structure, format, and requirements in technical writing. Students will need to conduct research and work on weekly writing assignment to practice the writing strategies, persuading techniques, formatting, citation, and surface control.
* Discrete Structures: The quarter system Discrete Mathematics (MATH248) has been redefined as Discrete Structures (CS2148). The mathematical concepts are covered in a way to make it more suitable for Computer Science majors. CS2148 will be a better prerequisite for CS3112.
* Communication skills (both writing and oral communication) are critical for Computer Science and Engineering majors. The college of ECST fully endorses the “Writing across curriculum” principle proposed by GERS and has embedded writing instruction and writing practice into our curriculum. Students will receive university-level writing instruction in ENGL1010 (or equivalent). They will enhance their writing skills and receive advanced instruction in professional writing in ENGL 2030. These two English courses will meet the General Education stated composition outcomes. In addition, students will receive additional writing instruction and will practice their writing skills in both their lower division and upper division courses.
* Increase number of units in the Senior Design sequence (CS 4961/CS 4962): Students will have the additional training to analyze problems and identify and define the computing requirements appropriate to their solutions. Students will have the additional training to design, implement, and evaluate large software systems working both individually and collaboratively. These courses have been revised to improve both oral and written communication skills and be capable of communicating their ideas to a broad audience.
* Societal and ethical issues in computing: The course (CS3801) has been expanded to meet the GE Humanity outcomes in new campus GE policy for semester system.

# CRITERION 5. CURRICULUM

### Program Curriculum

The Computer Science curriculum provides students with basic knowledge, training, discipline, and skills, as defined by the Computer Science Program Student Learning Outcomes. Through its lower division (CS1000 & CS2000 level) required courses, the curriculum provides students with the basic mathematical and science framework. Through its upper division (CS3000 & CS4000 level) required courses, the curriculum builds upon the fundamental principles of computer science for more advanced study. Through its upper division technical electives, students gain additional breadth and/or depth in computer science by an appropriate selection of courses. Through its Capstone courses students demonstrate their abilities to apply the knowledge and skills they acquired. The curriculum is thus consistent with the defined Program Educational Objectives and Student Learning Outcomes.

**A.1 Course requirements**

The requirements for the major are described below:

**Requirements for the Major (93 units)**

A grade of "C" or better is required for all prerequisite courses in the major.

**Lower Division Required Courses (39 units)**

CS 1222 Introduction to Relational Databases (3)

CS 2011 Introduction to Programming I (3)

CS 2012 Introduction to Programming II (3)

CS 2013 Programming with Data Structures (3)

CS 2148 Discrete Structures (3)

ENGL 2030 Introduction to Technical Writing (3)

MATH 2110 Calculus I (4)

MATH 2120 Calculus II (4)

MATH 2550 Introduction to Linear Algebra (3)

PHYS 2100 General Physics I (5)

PHYS 2200 General Physics II (5)

**Upper Division Required Courses (33 units)**

CS 3035 Programming Language Paradigms (3)

CS 3112 Analysis of Algorithms (3)

CS 3186 Introduction to Automata Theory (3)

CS 3220 Web and Internet Programming (3)

CS 3337 Software Engineering (3)

CS 3801 Societal and Ethical Issues in Computing (3)

EE 3445 Computer Organization (3)

CS 4440 Introduction to Operating Systems (3)

CS 4961 Software Design Laboratory I (3)

CS 4962 Software Design Laboratory II (3)

CS 4963 Computer Science Recapitulation (3)

**Electives (21 units)**

**Mathematics Electives (3 units)**

Select 3 units of lower division or upper division course(s) in the Mathematics area with prior approval of the Computer Science undergraduate adviser.

**Computer Science Electives (18 units)**

Select 15 units of upper division Computer Science (CS3xxx/CS4xxx) courses.

Table 5.1 describes the suggested plan of study (by year and semester term) for students in the computer science program. Many of the above courses have a built in laboratory component. All courses are delivered using a formula where one unit means one hour of lecture or three hours of laboratory per week.

Table 5.2 describes the program conversion from the old quarter system to the new semester format.

Figure 5.1 is a flowchart that illustrates the prerequisite structure of the program’s courses.

Program Name: Computer Science

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Course  (Department, Number, Title)  List all courses in the program by term starting with first term of the first year and ending with the last term of the final year. | | | Indicate Whether Course is Required, Elective or a Selected Elective by an R, an E or an SE.1 | *Subject Area (Credit Hours)* | | | | | | Last Two Terms the Course was Offered:  Year and,  Semester, or  Quarter | Average Section Enrollment  for the Last Two Terms the Course was Offered2 |
| Math & Sciences | Computing Topics  Mark with an F or A for Fundamental or Advanced | | General Education | | Other |
| **1st Semester** | | |  |  |  | |  |  | |  |  |
| MATH 2110 Calculus I | | | R | 4 |  | |  |  | |  |  |
| ENGL 1010 Written Communication | | | R |  |  | | 3 |  | |  |  |
| COMM 1100 Oral Communication | | | R |  |  | | 3 |  | |  |  |
| GE B2/B3 (Biological or Interdisciplinary Physical/Biological Science Elective) | | | R | 3 |  | |  |  | |  |  |
| CS 1010 Introduction to Higher Education for Computer Science Majors | | | R |  |  | | 3 |  | |  |  |
| **2nd Semester** | | |  |  |  | |  |  | |  |  |
| MATH 2120 Calculus II | | | R | 4 |  | |  |  | |  |  |
| HIST 2020 US History | | | R |  |  | | 3 |  | |  |  |
| POLS 1000 Gov. & Amer. Soc. | | | R |  |  | | 3 |  | |  |  |
| CS 1222 Introduction to Relational Databases | | | R |  | 3(F) | |  |  | |  |  |
| CS 2011 Introduction to Programming I | | | R |  | 3(F) | |  |  | |  |  |
| **3rd Semester** | | |  |  |  | |  |  | |  |  |
| MATH Elective | | | SE | 3 |  | |  |  | |  |  |
| MATH 2550 Introduction to Linear Algebra | | | R | 3 |  | |  |  | |  |  |
| PHYS 2100 General Physics I | | | R | 5 |  | |  |  | |  |  |
| CS 2012 Introduction to Programming II | | | R |  | 3(F) | |  |  | |  |  |
| GE C1 (Humanities Elective) | | | R |  |  | | 3 |  | |  |  |
| **4th Semester** | | |  |  |  | |  |  | |  |  |
| GE D1 (Social Science Elective) | | | R |  |  | | 3 |  | |  |  |
| ENGL 2030 Introduction to Technical Writing | | | R |  |  | | 3 |  | |  |  |
| PHYS 2200 General Physics II | | | R | 5 |  | |  |  | |  |  |
| CS 2013 Programming with Data Structures | | | R |  | 3(F) | |  |  | |  |  |
| CS 2148 Discrete Structures | | | R | 3 |  | |  |  | |  |  |
| WPE (Writing Proficiency Exam) | | | R |  |  | | 0 |  | |  |  |
| **5th Semester** | | |  |  |  | |  |  | |  |  |
| CS 3035 Programming Language Paradigms | | | R |  | 3(F) | |  |  | |  |  |
| CS 3112 Analysis of Algorithms | | | R |  | 3(F) | |  |  | |  |  |
| CS 3220 Web and Internet Programming | | | R |  | 3(A) | |  |  | |  |  |
| CS 3337 Software Engineering | | | R |  | 3(F) | |  |  | |  |  |
| EE 3445 Computer Organization | | | R |  | 3(A) | |  |  | |  |  |
| **6th Semester** | | |  |  |  | |  |  | |  |  |
| CS 3186 Introduction to Automata Theory | | | R |  | 3(A) | |  |  | |  |  |
| CS 3801 Societal and Ethical issues in Computing | | | R |  | 3(A) | |  |  | |  |  |
| CS 4440 Introduction to Operating Systems | | | R |  | 3(A) | |  |  | |  |  |
| CS Elective | | | SE |  | 3(A) | |  |  | |  |  |
| GE D1 (Social Science Elective) | | | R |  |  | | 3 |  | |  |  |
| **7th Semester** | | |  |  |  | |  |  | |  |  |
| CS 4961 Software Design Laboratory I | | | R |  | 3(A) | |  |  | |  |  |
| CS Elective | | | SE |  | 3(A) | |  |  | |  |  |
| CS Elective | | | SE |  | 3(A) | |  |  | |  |  |
| CS Elective | | | SE |  | 3(A) | |  |  | |  |  |
| **8th Semester** | | |  |  |  | |  |  | |  |  |
| CS 4962 Software Design Laboratory II | | | R |  | 3(A) | |  |  | |  |  |
| CS 4963 Computer Science Recapitulation | | | R |  | 3(A) | |  |  | |  |  |
| CS Elective | | | SE |  | 3(A) | |  |  | |  |  |
| CS Elective | | | SE |  | 3(A) | |  |  | |  |  |
|  | | |  |  |  | |  |  | |  |  |
| *Add rows as needed to show all courses in the curriculum.* | | | | | | | |  | |  |  |
| TOTALS-ABET BASIC-LEVEL REQUIREMENTS | | | | 30 | 24 (F)  39 (A) | 27 | |  | |  |  |
| OVERALL TOTAL CREDIT HOURS FOR COMPLETION OF PROGRAM | 120 |  | | | | | | | | | |

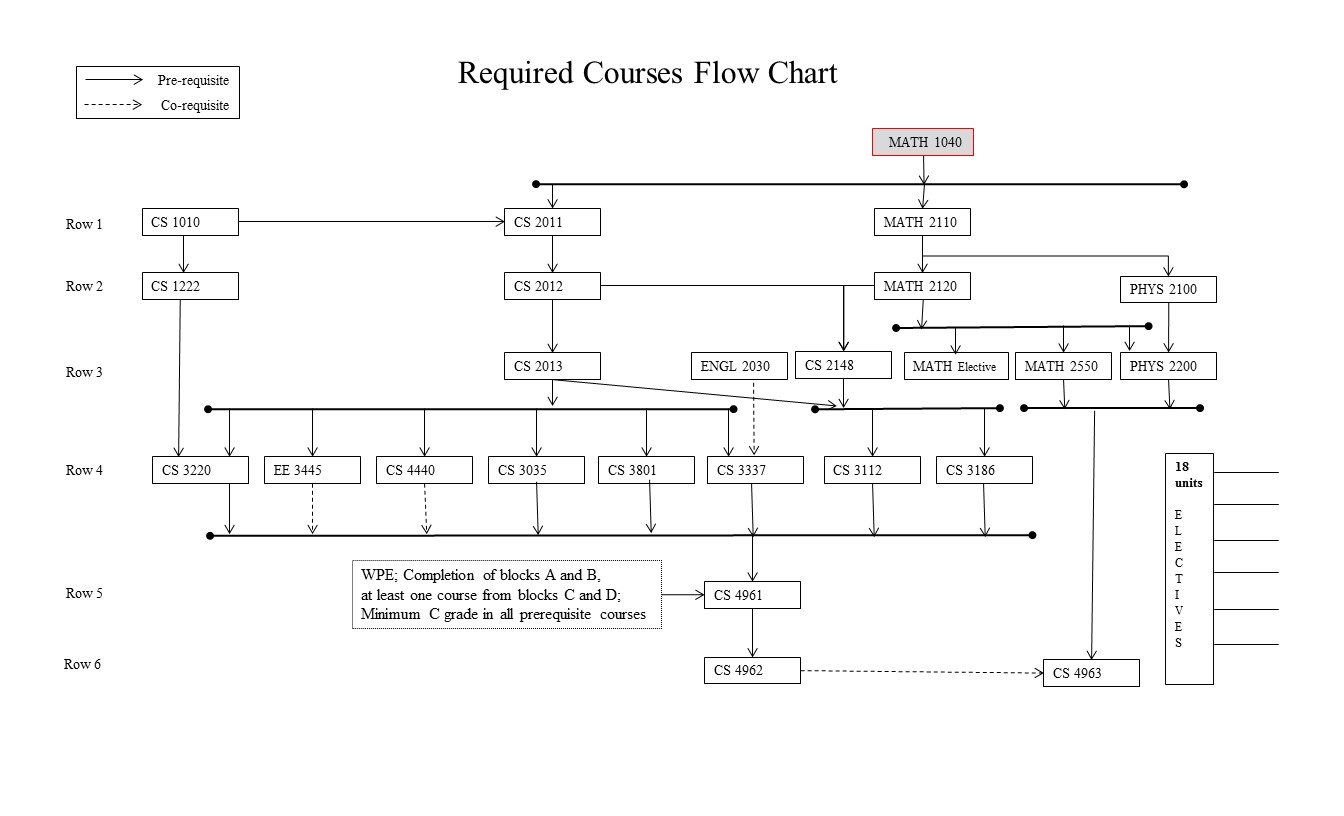
1. **Required** courses are required of all students in the program, **elective** courses (often referred to as open or free electives) are optional for students, and **selected elective** courses are those for which students must take one or more courses from a specified group.
2. For courses that include multiple elements (lecture, laboratory, recitation, etc.), indicate the maximum enrollment in each element. For selected elective courses, indicate the maximum enrollment for each option.

Instructional materials and student work verifying compliance with ABET criteria for the categories indicated above will be required during the campus visit.

**Table 5.1: Curriculum (Required courses)**



**Table 5.2: Major Requirements (Conversion Table)**



**Figure 5.1: Required Courses (Prerequisites structure)**

The requirements indicated above translate to a road map as indicated in Table 5.3. This ideal roadmap provides a pathway for students to complete their undergraduate requirements in four years.

|  |  |  |  |
| --- | --- | --- | --- |
| **year**  **1** | **summer** | **fall** | **spring** |
|  | MATH 2110 | MATH 2120 |
|  | ENGL 1010 | HIST 2020 |
|  | COMM 1100 | POLS 1000 |
|  | GE B2/B3 | CS 1222 |
|  | CS 1010 | CS 2011 |

|  |  |  |  |
| --- | --- | --- | --- |
| **year**  **2** | **summer** | **fall** | **spring** |
|  | MATH Elective | GE D1 |
|  | MATH 2550 | ENGL 2030 |
|  | PHYS 2100 | PHYS 2200 |
|  | CS 2012 | CS 2013 |
|  | GE C1 | CS 2148 |

|  |  |  |  |
| --- | --- | --- | --- |
| **year**  **3** | **summer** | **fall** | **spring** |
| WPE | CS 3035 | CS 3186 |
|  | CS 3112 | CS 3801 |
|  | CS 3220 | CS 4440 |
|  | CS 3337 | CS Elective |
|  | EE 3445 | GE D2 |

|  |  |  |  |
| --- | --- | --- | --- |
| **year**  **4** | **summer** | **fall** | **spring** |
|  | CS 4961 | CS 4962 |
|  | CS Elective | CS 4963 |
|  | CS Elective | CS Elective |
|  | CS Elective | CS Elective |
|  |  |  |

**Table 5.3: Road map (Ideal)**

### 

For a variety of reasons many students formulate their own schedules with varying loads in any given term. Depending on the summer schedules, students can elect to spread the load over the Summer term as well. An advisor is available to help students plan or adjust their schedules. Students select courses for the General Education (GE) Blocks (B,C,D) from a variety of courses. Similarly, students select upper division (CS3000/CS4000) courses as electives. Therefore, the roadmaps described in Table 5.3 should be used as an ideal guide.

It is essential that every student should see a faculty academic advisor and complete an Individualized Advisement Planner (IAP) described earlier. This should be updated if any situation changes down the road.

**A.2. Course prerequisite(s) requirements**

All major requirements have a designated set of prerequisites. For example, CS2012 has a prerequisite of CS2011. A grade of "C" or better is required in all prerequisite coursework. Prerequisites are strictly enforced on the GET system at the time of registration. The prerequisite flowchart is shown in Figure 5.1.

**A.3. Capstone courses**

Students are required to complete a senior design project sequence (CS4961/4962) and to take a recapitulation course (CS4963) during their senior year. These two senior capstone experiences are used as a primary instrument in program assessment.

CS4961/4962

This is a two-semester senior design laboratory project sequence in which each student participates in a group project under the supervision of a faculty advisor. The goals of the course are

* To improve students’ ability to undertake complex software projects.
* To require students to learn new technologies and concepts on their own.
* To improve students’ oral communication skills through presentations and interaction with project stake holders.
* To improve students’ written communication skills through the writing of project documents.

These course goals contribute to the success of Student Learning Outcomes (SLO):

* SLO5. Students will have the training to analyze problems and to identify and define the computing requirements appropriate to their solutions.
* SL06. Students will have the training to design, implement, and evaluate large software systems working both individually and collaboratively.
* SLO7. Students will be able to communicate effectively orally and in writing.
* SLO8. Students will gain the knowledge, skills, and attitudes for lifelong self-development.

At the end of CS4962, the teams show off their projects at Senior Design Day—an event organized at the College level across all disciplines. Each project team prepares a poster and presents its project (<http://csns.calstatela.edu/projects.html>). The event is attended by the members of the Department’s Industry Advisory Board and the liaisons of the industry sponsored projects. This gives students a sense of what it means to work in a professional environment where customers expect the delivery of development products. At the same time the projects are crafted to meet the educational goals of the capstone design course, which runs the full senior year.

CS4963

This is a recapitulation course covering the primary concepts of Computer Science. Students enhance their problem solving and presentation skills as they prepare solutions to several conceptual questions covering the core course topics in the undergraduate curriculum:

* Theory (CS2148, CS3112, CS3186)
* Programming (CS2011-2013, CS3035, CS3220)
* Algorithms (CS2013, CS3112, CS3660, CS3035)
* Systems (CS1222, CS4440, EE3445)

These course goals contribute to the success of Student Learning Outcomes (SLO):

* SLO#1: Students will be able to apply concepts and techniques from computing and mathematics to both theoretical and practical problems.
* SLO#2: Students will be able to demonstrate fluency in at least one programming language and acquaintance with at least three more.
* SLO #3: Students will have a strong foundation in the design, analysis, and application of many types of algorithms.
* SLO #4: Students will have a fundamental understanding of computer systems.

At the end of CS4963, students take a standardized Major Field Test ([MFT](http://www.ets.org/mft/)) conducted by the Educational Testing Service ([ETS](http://www.ets.org/)) and the results are archived on CSNS at <http://csns.calstatela.edu/assessment/mft/viewMFTScores.html>.

CS4963 provides the capstone experience to recapitulate the major concepts, analyze and solve problems, understand relationships and interpret material from the field of Computer Science.

The MFT is designed to measure the critical knowledge and understanding obtained by students in Computer Science. The exam produces a normed score for each student. In addition, the results provide three departmental summaries of assessment indicators which tie closely to the Student Learning Outcomes.

**A.4. Course coverage of required areas**

The required courses for the B.S in Computer Science degree program (See Table 5.1) includes:

63 units of major Computer Science requirements

45 units of Computer Science core courses

18 units of Computer Science electives

20 units of Mathematics courses

10 units of Science courses

27 units of general education courses.

The program meets all the specific requirements as indicated by the ABET category content for the Computer Science as described below:

**Computer science**

*Coverage of the fundamentals of algorithms, data structures, software design, concepts of programming languages and computer organization and architecture*

CS 1222 Introduction to Relational Databases(3)

CS 2011 Introduction to Programming I (3)

CS 2012 Introduction to Programming II (3)

CS 2013 Programming with Data Structures (3)

CS 2148 Discrete Structures (3)

ENGL 2030 Introduction to Technical Writing (3)

CS 3035 Programming Language Paradigms (3)

CS 3112 Analysis of Algorithms (3)

CS 3186 Introduction to Automata Theory (3)

CS 3220 Web and Internet Programming (3)

CS 3337 Software Engineering (3)

EE 3445 Computer Organization (3)

CS 4440 Introduction to Operating Systems (3)

*An exposure to a variety of programming languages and systems*.

CS 1222 Introduction to Relational Databases(3)

CS 2011 Introduction to Programming I (3)

CS 2012 Introduction to Programming II (3)

CS 2013 Programming with Data Structures (3)

CS 3035 Programming Language Paradigms (3)

CS 3220 Web and Internet Programming (3)

CS 3337 Software Engineering (3)

CS 3801 Societal and Ethical issues in Computing (3)

EE 3445 Computer Organization (3)

*Proficiency in at least one higher-level language and familiarity with at least three more*

CS 1222 Introduction to Relational Databases(3)

CS 2011 Introduction to Programming I (3)

CS 2012 Introduction to Programming II (3)

CS 2013 Programming with Data Structures (3)

CS 3035 Programming Language Paradigms (3)

CS 3220 Web and Internet Programming (3)

*Advanced course work that builds on the fundamental course work to provide depth*.

CS 3035 Programming Language Paradigms (3)

CS 3112 Analysis of Algorithms (3)

CS 3186 Introduction to Automata Theory (3)

CS 3220 Web and Internet Programming (3)

CS 3337 Software Engineering (3)

CS 3801 Societal and Ethical issues in Computing (3)

EE 3445 Computer Organization (3)

CS 4440 Introduction to Operating Systems (3)

CS 4961 Software Design Laboratory I (3)

CS 4962 Software Design Laboratory II (3)

CS 4963 Computer Science Recapitulation (3)

CS3000/4000 upper division computer science electives - (18)

***Science and mathematics***:

*Mathematics*

MATH 2110 Calculus I (4)

MATH 2120 Calculus II (4)

MATH 2550 Introduction to Linear Algebra (3)

CS 2148 Discrete Structures (3)

Mathematics Electives (3)

* + - Select 3 units of lower division or upper division course(s) in the Mathematics area with prior approval of the adviser.

*Science*

PHYS 2100 General Physics I (5)

PHYS 2200 General Physics II (5)

General Education Science requirement (3)

* + - Select one course from Biological science OR Interdisciplinary Physical-Biological science

**A.5. Attainment of Student Learning Outcomes**

The new semester program curriculum described in Section 5A (Table 5.1) includes all the necessary math, physics, computers science and general education courses. As described in Table 5.2 the new program is mapped directly from the old existing program. Since the Student Learning Outcomes have not changed and they have been shown to be attained by our existing quarter curriculum, the new course structure supports the attainment of the same Student Learning Outcomes.

# CRITERION 6. FACULTY

Q2S conversion does not impact this criterion.

# CRITERION 7. FACILITIES

Q2S conversion does not impact this criterion.

# Appendix- Sample Individualized Advisement Plan

