Computer Science, BS (CS) Baccalaureate Degree

REPORT PREPARED by: Meyer, Karen

ACADEMIC YEAR COVERED BY THIS REPORT: 2020-2021

I. PROGRAM LEARNING OUTCOMES

The BS in Computer Science program has been continually accredited by the ABET Computing Accreditation Commission (CAC) since 1987. ABET Criterion 3 Student Learning Outcomes used for Assessment Data Collection Students who complete the BS in computer science will have (a)an ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline (b) an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution (c) an ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs (d) an ability to function effectively on teams to accomplish a common goal (e) an understanding of professional, ethical, legal, security and social issues and responsibilities (f) an ability to communicate effectively with a range of audiences (g) an ability to analyze the local and global impact of computing on individuals, organizations, and society (h) recognition of the need for and an ability to engage in continuing professional development (i) an ability to use current techniques, skills, and tools necessary for computing practice (j) an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices (k) an ability to apply design and development principles in the construction of software systems of varying complexity ABET released new Student Learning Outcomes that will be used for the next accreditation visit in 2022. They are being reviewed by department faculty members. Graduates of the program will have an ability to 1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions. 2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline. 3. Communicate effectively in a variety of professional contexts. 4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles. 5. Function effectively as a member or leader of a team engaged in activities

appropriate to the program's discipline. 6. Apply computer science theory and software development fundamentals to produce computing-based solutions. [CS] Relationship of Student Outcomes to Program Educational Objectives The attached table specifies how each PEO is supported by multiple (and in some cases overlapping) student outcomes. PEO Expert requires students to gain both general and specific proficiency with mathematical and scientific principles and engineering processes used in computing, and to be able apply these principles to productively solve engineering problems as computing professionals. Well-prepared computing professionals must be not only well versed technically, but must also be capable communicators able to work in modern team-based environments. PEO Agile requires both foundational engineering skills and the broad/¬contemporary knowledge of changing issues in the profession necessary to remain current in practice. PEO Engaged requires students to go beyond technical capability and demonstrate appreciation of their role as a member of an organization, profession, and society. PEO Focused requires that students possess the software construction skills expected of modern Computer Scientists. Refer to the supporting documents to see how the program PEOs are mapped to the CAC Student Learning Outcomes. programPEOmaptoStudentLearningOutcomes.docx

II. PROCEDURES USED FOR ASSESSMENT

A. Direct Assessment

Assessments of retained relevant knowledge (Prerequisite quizzes) Formative data is collected in online multiple-choice prerequisite quizzes to assess retained prerequisite knowledge required for courses in the mandatory program introductory core. This data is used both by the UGSC to monitor program quality as well as by individual instructors to help identify students that are struggling with the required knowledge/skills so that they can receive appropriate support or advice. The formative assessments focus on discipline-specific knowledge and skills that are expected to be retained as students progress along the prerequisite introductory core course sequence. As such, these assessments focus on • Prior CS (CAC) learning objectives assessed abcijk • Current CS (CAC) learning objectives assessed 1, 2, 6 This assessment strategy utilizes GRE-style multiple choice quiz questions to directly assess formative learning objectives. These assessment instruments are not delivered in the courses where the topics are taught. Instead, the program instruments achievement of student outcomes by assessing the ability of students to demonstrate mastery of prerequisite knowledge engendered in precursor courses. The direct assessment questions are delivered to students in the form of an on-line "assessment of prerequisite knowledge" in each of their core program courses. At a minimum, the program expects each outcome to be measured in at least two places over the course of every three-year period. In practice, however, the ease of delivery of the on-line assessment allows us to target the significantly higher goal of directly assessing every student, every term, in every core course. The pre-course assessment tools attempt to formatively assess learning objectives dealing with knowledge of computer science principles and relevant concepts from other disciplines, the ability to analyze data, design/construct computational solutions to problems, recognition of design/algorithmic trade-offs, and knowledge of contemporary tools. We do not

attempt to assess the ability to work in teams, ethics, communication, or leadership qualities in the formative stages. These student outcomes are assessed at the summative level in Senior Design. The course is listed below, then the broad topic areas for prerequisite questions, then the CAC student outcomes. (the table was not preserved in this application) CS1181 Computer Science II Fundamental Programming Concepts, Mathematical Preparedness 1, 2, 6 CEG2350 OS Concepts & Usage First-year programming concepts 1, 2, 6 CS3100 Data structures and algorithms First-year programming concepts, computer organization, fundamental operating systems concepts, discrete structures, 1, 2, 6 CS3180 Comparative Languages First-year programming concepts, computer organization, data structures and algorithms, discrete structures, CAC only 1, 2, 6 CEG3310 Computer Organization First-year programming concepts, 1, 2, 6 CEG4110 Software Engineering Data structures 1, 2, 6 CEG4350 OS Internals & Design Data structures concepts 1, 2, 6 No less often than once every three years the UGSC considers summaries of this data, collected each term. In practice, this data is automatically summarized and examined every term and any flagged concerns can be brought forward as soon as they are detected. The Prerequisite Data Summary Report for Fall 2019 is attached. We are having difficulty with the tool used to summarize the data for Spring 2020. We are working on a solution to the problem. Senior Design. Summative data is collected in the senior design experience (CEG 4890/4891 Team Projects I and II). These courses, generally taken in the senior year, are the most convenient place to assess overall student learning achievement. Team Projects I assesses all learning objectives, except for the outcomes that are difficult to assess individually in a team-based course and are already well assessed elsewhere in the program. • Team Projects I, Prior CS (CAC) learning objectives assessed (a) – (k) • Team Projects I, Current CS (CAC) learning objectives assessed 1 – 6 Team Projects I The Team Projects (senior design) course requires a number of student deliverables targeted to reinforce and to serve as performance indicators for assessment of mastery of critical program learning objectives. This assessment process is shared by the electrical engineering, computer engineering, and computer science programs. Data is disaggregated by program when appropriate. Performance Indicators / Deliverables CAC student outcomes D01 Short Form White Paper Students are required to produce a "short form" proposal that outlines her/his personal ideas, including requirements and constraints, about how to approach one of the product modules OR argues for one that the instructors may not have suggested. Students may submit may submit up to THREE of these, with the highest grade being recorded. Faculty will use these short form proposals as an information source to form student groups. 1, 3, 4, 6 D02 Personal Resume/Cover Letter Students will prepare a resume and cover letter aimed at getting "hired" onto one of the teams producing a design for a specific module of the whole class product. These deliverables will be graded and used as an information source to form student groups. 3, 4 D03 Briefings & Charts At various times in the course, you (as an individual) will create briefing charts and be asked to brief these to selected audiences. You will also be required to give impromptu and extemporaneous briefings without charts or other visuals. 4, 5 D04 Document Reviews At various times in the course, you (as an individual) will be asked to perform reviews of documents produced by other groups or by external authors. Some of these, to be randomly selected, will be graded and you will receive points based on the quality of the feedback you provide your peers in other groups or of your evaluation of external documents. 1, 3, 4, 5, 6 D05

Quizzes Quizzes are given regularly, through Pilot, within the first 15 minutes of class. Quiz material is typically drawn from contemporary course matter. The points you receive for all quizzes will be computed by scaling the number ten by the ratio of questions you answered correctly across all quizzes. Quizzes given during unexcused absences will be scored as zero and there will be no make-ups. 1, 4, 6 D06 GROUP Proposal Your group will respond to the provided RFP to produce a "short-form" proposal that synthesizes the best ideas from each member. This activity is designed to give initial experience in managing group activity and in persuasive writing. The instructors, after reading this document, should feel your basic approach is feasible. 1, 3, 4, 5, 6 D07 GROUP Requirements/Constraints/Standards List Your group will develop a detailed list of specific requirements, a detailed list of realistic constraints, and a detailed list of applicable design/code standards associated with your proposed approach. 1, 2, 4, 5, 6 D08 GROUP Requirements Specification Document Your group will conduct a detailed analysis of your specific requirements and constraints and produce a formal, and equally detailed, standards-compliant requirements specification. 1, 2, 4, 5, 6 D09 GROUP Design Specification Document Once your requirements analysis is complete, each group will be expected to produce a formal design document that clearly and specifically connects your specific requirements and constraints to your formal design with explicit compliance to applicable design standards. 1, 2, 3, 4, 5, 6 D010 GROUP Presentation/Poster At the end of the semester, your group will be required to produce a short briefing that argues for adopting your design. 1, 2, 3, 4, 5, 6 See the three attached files - Fall 2019 data for Overview of Team Projects - Spring 2020 data for Overview of Team Projects - Overview of Team Projects (senior design)

B. Scoring of Student Work

Formative data is collected in online multiple-choice prerequisite quizzes to assess retained prerequisite knowledge required for courses in the mandatory program introductory core. These assessment instruments are not delivered in the courses where the topics are taught. Instead, the program instruments achievement of student outcomes by assessing the ability of students to demonstrate mastery of prerequisite knowledge engendered in precursor courses. The direct assessment questions are delivered to students in the form of an on-line "assessment of prerequisite knowledge" in each of their core program courses. At a minimum, the program expects each outcome to be measured in at least two places over the course of every three-year period. In practice, however, the ease of delivery of the on-line assessment allows us to target the significantly higher goal of directly assessing every student, every term, in every core course. Questions are taken from professionally-developed assessment instruments, such as the GRE subject test in computer science, and scored according to the provided answer keys associated with these instruments.

C. Indirect Assessment

Exit surveys are used to collect graduating student assessment of self-efficacy for each of the learning outcomes. For each outcome students are asked to rate their own level of ability/achievement. For FA19 and SP20, student responses generally indicated a high self-assessment of abilities related to program outcomes. These self-efficacy ratings did not exhibit any concerning trends relative to previous years. See attached documents, ExitSurveyF19.pdf and ExitSurveyS20.pdf, for full details including descriptive statistics and distribution plots.

III. ASSESSMENT RESULTS/INFORMATION:

Prerequisiste Quizzes, see attached file, PrerequisiteDataSumaryReport F19.pdf
Team Projects Capstone, see attached file,

See graphs in PrerequisiteDataSumaryReport F19.pdf

Student learning outcome j was flagged as a concern. The UGSC is reviewing the data. No other actions are indicated.

IV. ACTIONS TO IMPROVE STUDENT LEARNING

This data is used both by the UGSC to monitor program quality as well as by individual instructors to help identify students that are struggling with the required knowledge/skills so that they can receive appropriate support or advice. The UGSC faculty and student advisors review all program courses every three years to ensure that course pre-requisites are relevant, student learning objectives are accurate and sequential courses are aligned. Technical electives are reviewed and new courses created as needed to ensure coverage of emerging technologies. Student grades are analyzed to flag courses where students are having difficulties so course coordinators can identify any problems and adjust the course content or learning strategies. A student advisory board that consists of a cross-section of students from all CSE majors and levels meet with the undergraduate program director at least once a year. Plans are shard and students give input on what is working and suggestions for improvements. Similarly, an external advisory board comprised of business leaders and alumni meet with the department chair and program faculty to give program suggestions and feedback. CSE was the first department to build active learning classrooms. We have found that students, especially in introductory level courses, embrace this learning style. This is reflected in the course comments as well as student advisory board meetings. We continue to convert classrooms to support the active learning pedagogy. Most recently with the transition to remote learning, many faculty use the breakout rooms to continue to incorporate active learning in distance mode. New faculty members and adjuncts are assigned to faculty mentors. The department system administration team periodically meets with the UGSC and

department chair to discuss the state of the labs to offer courses in modern, up-to-date laboratories that meet the needs of the ever changing technological requirements. Our systems team stood up a Discord server to foster communication between students, faculty and advisors during the quarantine. We have received high praise as students use the site for study groups, help room assistance and to connect with their peers.

V. SUPPORTING DOCUMENTS

Additional documentation, when provided, is stored in the internal Academic Program Assessment of Student Learning SharePoint site.