2020-2021 Assessment Report for Department: *Computer Science & Engineering -draft*

General Education Core Curriculum Area:

Undergraduate Major: *Computer Science*

Department Mission Statement:
The mission of the Computer Science Program is to produce computer science graduates who, trained in the design, implementation, and analysis of computational systems and skilled in technical communication, will contribute towards the advancement of computing science and technology.

Program Outcomes (*a.k.a.* *Student* Outcomes):
Owing to changes in ABET-CAC requirements, we altered our program (student) outcomes. The new outcomes are the following.

By the time of their graduation, the undergraduate academic program in Computer Science should enable our graduates 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; and
6. Apply computer science theory and software development fundamentals to produce computing-based solutions.

Note[1]: The above program / student outcomes are ABET outcomes.

[1] This addresses a comment we received from a past review.

Curricular Map:

Each numeric entry in the matrix below (between 1 and 3) represents the relative weight / contribution of a required course (row) towards a program / student outcome (column). *Note*[2]: This is the *final* matrix. It was obtained after trimming entries with smaller contributions (= 1) in an *initial* matrix in order to keep the assessment effort manageable. This is why there are no 1-entries and the row for CSE 101 is blank.

Required Courses	Program/Student Outcomes					
	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
CSE 101						
CSE 113		**2**				
CSE 122		**2**				
CSE 213		**2**				
CSE 221		**2**				
CSE 222		**2**				
CSE 241						**2**
CSE 324	**3**	**3**				
CSE 325	**3**				**3**	
CSE 326			**3**		**3**	**3**
CSE 331	**3**		**2**			
CSE 342						**3**
CSE 344						**3**
CSE 353	**2**	**3**				
CSE 382				**3**		
CSE 423	**3**	**3**				**3**

[2] 2 This addresses a comment we received from a past review.

Our process:

- The above curricular map was obtained by first considering all courses and assigning weights between 1 and 3 (inclusive) to reflect the strength of their contribution, with the following interpretation.

Contribution	*Interpretation*
1	Introductory / preliminary
2	Reinforcement / extension / application
3	Major component

Next, it was pruned keeping only the required courses, and then pruning it further by eliminating weights of 1 and 2 unless one of three criteria (omitted here) were met.

- For each Program (/Student) Outcome, we obtain a number between 1 and 4 through a weighted sum of scores from contributing courses as per the curricular map shown above. Our acceptance threshold for each student outcome is 3.0.
- The scores from contributing courses come from the respective instructors who are required to submit an assessment report for each offering of such courses. This report outlines the relation between the scores and the course learning outcomes.
- The report also provides comments regarding successful strategies and plans for future modifications. While we have a numeric acceptance threshold, the instructors' comments are always important.
- The numeric score for the j^{th} student outcome is a normalized weighted sum

$$Score\_Outcome_j = \frac{\sum_i (n_{ij} * s_{ij})}{\sum_i n_{ij}}$$

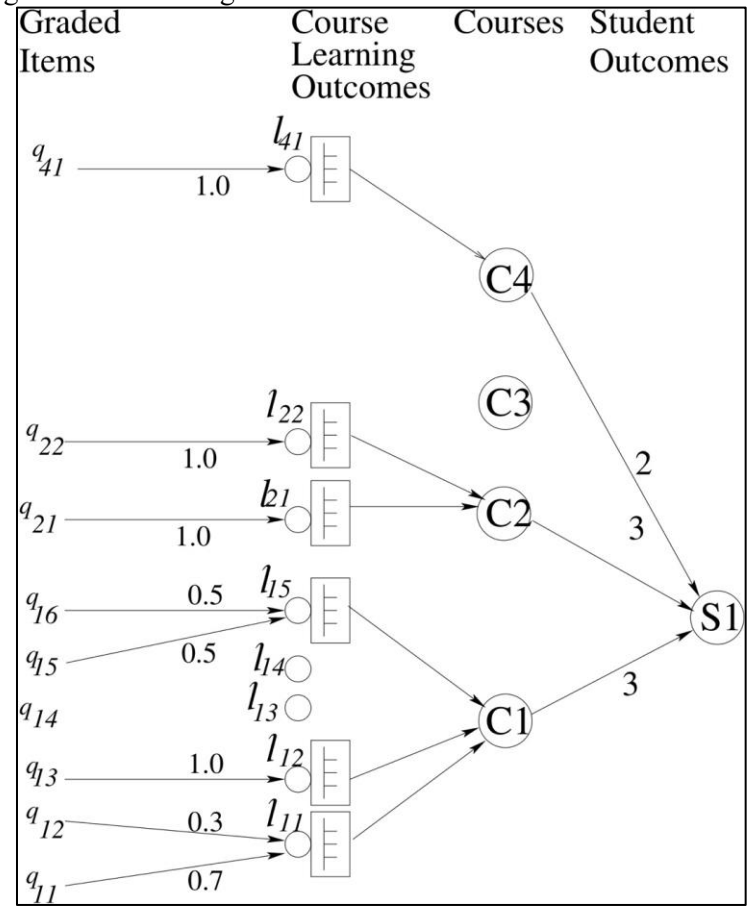
where the weights n_{ij} are the non-zero entries in the column for student outcome *j* in the curricular map, and each value S_{ij} is a score that comes from the assessment of the i^{th} course specifically for the j^{th} Student outcome. For example, the curricular map shows that Student outcome 3 (technical communication) will be measured using three courses CSE326 Software Engineering, CSE423 Compiler Writing, and CSE331 Computer Architecture, with impact factors of 3, 3, and 2 respectively. If the numeric scores assessed by those three courses are 3, 2, and 4 respectively, then the score computed for Student outcome 5 is given by (3*3 + 2*3 + 4*2)/(3+3+2), i.e., 2.88.

We limit the score S_{ij} (reported by a course *i* for a Student outcome *j*) to a number between 1 and 4 with the following interpretation.

Student Outcome score	*Interpretation*	
1	Unsatisfactory	
2	Marginal	
3	Satisfactory	
4	Excellent	

- The instructor of the i^{th} course computes a score s_{ij} for the j^{th} Student outcome as follows.

Example: computing the score for Program/Student Outcome S1 from relevant courses C1, C2, and C4.



The first step is to identify the largest disjoint set *L* of course learning outcomes corresponding to the Student outcome at hand at hand. For that set *L*,

1. The instructor decides on a performance metric to interpret an average score for a course outcome as unsatisfactory, marginal, satisfactory, or excellent, resulting in the basis for a four-point scale; this takes care of variations among courses in grading, e.g., relative versus absolute, partial credit versus all-or-none grading.
2. Each course outcome l in *L* is tied to a set of gradable items in the course, e.g., a project, specific questions in the final exam, a presentation, etc. The sets of items should be disjoint among learning outcomes. In the above figure, course outcome l_{11} would be tied to questions q_{11} and q_{12}.
3. Weights are assigned to these questions or items (in Figure 1, 0.7 and 0.3 for questions q_{11} and q_{12} respectively); using them, a formula is written to compute a normalized weighted sum from the scores for those questions or items;
4. From a table of scores of the students on those gradable items, one numeric score is computed for each student per course outcome l.
5. Those numeric scores are then averaged over the whole class to get one numeric score p_l for each course outcome l.
6. Using the performance metric, a number q_l is obtained by quantizing p_l to a four-point scale.
7. The above is repeated for each l in *L*.
8. The scores q_l (in the four-point scale) are averaged over all l in *L*.

The result is s_{ij}, the numeric score (between 1 and 4) from course i to student outcome j.

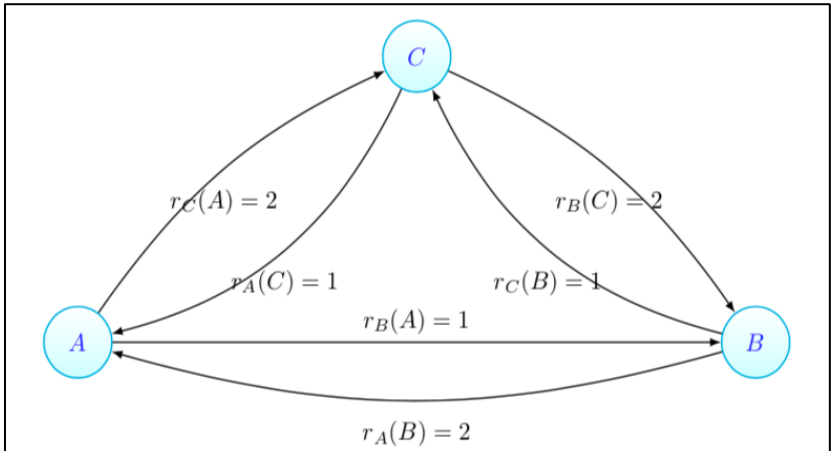
- *New assessment method for team (/group) work.*
We found it quite difficult to measure the effectiveness of team work using direct measures. Invariably, the instrument became corrupted with the quality of the projects or deliverables. The issues differed with the content of the courses; a workable solution for all concerned courses proved elusive. Beginning in Spring 2019, we decided to use an indirect measure using surveys following a suggestion of Dr. Gloria Rogers, the facilitator of an assessment workshop organized at NMT by AVPAA Peter Mozley. We adopted three yardsticks from her examples, and constructed the following computational scheme to obtain a numeric measure consistent with our scheme.

  First, we chose three team performance yardsticks:
  - fulfilling team role duties,
  - sharing of team work, and
  - listening to teammates.

  For each of those yardsticks, for each group, the following information was obtained. Each individual in the group was rated on that yardstick by each of his/her team-mates on a *0..M* scale.
  The information (for a given yardstick and given group) is modeled using a weighted directed graph. Each member of the group is represented by a vertex; each rating of a member *u* by member *v* is represented by an out-edge from *u* to *v*, the weight on that edge being the rating (a number in *0..M*.
  - A *member interaction* metric for each team member is then defined as the sum of the weights on the out-edges from the vertex representing that member divided by the maximum possible sum, which is *(N-1)*M*.

- The *group interaction* for a group is essentially an average over all its members. But since that average is in the closed interval [0,1], it is translated to 1..4 by multiplying by 3 and adding 1 (a linear transformation).
- The *interaction* on a given yardstick is the average over all the groups in the class.
- Finally, the averages of the three yardsticks is the *class team work* measure for the entire class.

The following example outlines the initial steps of our method. Suppose the following graph captures the result of a survey of a group with members *A, B,* and *C* for a given yardstick using a rating scale of 0..3. The sum of the two out-edges from *A* is 3 (A is rated 2 and 1 by C and B respectively) while the maximum such sum is 3*(3-1) = 6 (the scenario in which both would have rated *A* at 3); thus the interaction of *A* is 3/6 = 0.5. Similarly, the interaction of *B* and *C* are also 0.5; thus the interaction of the group (on this yardstick) is 0.5; that is transformed into 2.5 on our 1..4 scale.



Example of a team with $N = 3$ members A, B, and C. A rates B and C at 2 and 1 respectively.
$r_C(A) = C$'s rating of A's interaction
$r_B(A) = B$'s rating of A's interaction

Since A is rated 2 and 1 (using a rating scale from 0 to 3) by C and B respectively, the member interaction of A is given by:

$$memberinteraction(A) = \frac{(r_c(A) + r_B(A))}{(2 \times 3)} = \frac{(2+1)}{(2 \times 3)} = \frac{2}{3}$$

The group interaction (1..4) for an n-member group is defined as follows.

$$average\ member\ interaction = \frac{1}{n} \sum_{i=1}^{n} interaction(i)$$

$$group\ interaction = 1 + 3 \times average\ member\ interaction$$

Assessment Instrument for team (/group) work: We ask each participant to rate the others in the group on a 0 to 3 scale, enter the data in a spreadsheet, and compute the interaction score for each group using the above. The students are informed that the survey results will not impact their grades.

Assessment of Team Activity

Note: This will be used for course assessment, not for grading.

Using the following rubric, rate each of your team mates (do not rate yourself) in a **0** to **3** scale on three yardsticks:

1. Fulfilling team role duties
2. Sharing of team work
3. Listening to teammates

Your Name:

	Fulfilling team role duties	Sharing of team work	Listening to teammates
0 = Unsatisfactory	does not perform any duties of assigned team role	always relies on others to do the work	never allows others to speak
1 =Developing	inconsistently performs assigned duties	rarely does assigned work: often needs reminding	usually does most of the talking; rarely allows others to speak
2 = Satisfactory	performs assigned duties needs reminding	usually does assigned work	listens most of the time
3 = Exemplary	performs all duties assigned and actively assists others	always does assigned work without needing reminders	consistently listens and responds to others appropriately
Name of team-mate	0 1 2 3	0 1 2 3	0 1 2 3
Name of team-mate	0 1 2 3	0 1 2 3	0 1 2 3
Name of team-mate	0 1 2 3	0 1 2 3	0 1 2 3

(circle your choices)

Rubric for assessment of teamwork.

#1 Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions;

<table>
<tr><th>Student Learning Outcomes</th><th>Assessment Procedures</th><th>Assessment Results</th><th>Assurance</th></tr>
<tr><td>Learning Outcomes of the Program—Students will be able to:</td><td>Process/Instrument used:
Direct measures. Graded items are weighted and linked to courses; courses are weighted, aggregated, and linked to student outcomes. Covered Fall 2019, and Spring 2020.</td><td>What were your findings?
Score range: 1 (unsatisfactory),
2 (marginal),
3 (satisfactory), and
4 (excellent).</td><td>Our department believes we fulfill this Learning Outcome because:
(state evidence in 30 words or less)</td></tr>
<tr><td>1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions</td><td>Direct Measure:

Quantitative Assessment Procedure on CSE 324, 325, 331, 353, and 423.</td><td>
<table>
<tr><th>Course</th><th>Score</th><th>Weight</th><th>Overall</th></tr>
<tr><td>CSE 324 S21</td><td>4</td><td>3</td><td rowspan="5">3.5</td></tr>
<tr><td>CSE 325 F20</td><td>3.8</td><td>3</td></tr>
<tr><td>CSE 331 S21</td><td>3.57</td><td>3</td></tr>
<tr><td>CSE 353 F20</td><td>3.17</td><td>2</td></tr>
<tr><td>CSE 423 S21</td><td>3</td><td>3</td></tr>
</table>
</td><td>the overall scores for both Student outcome 1 is higher than 3.0, our acceptance threshold.</td></tr>
</table>

Adjustment/Improvement
CSE 324: The class **on-line** delivery profoundly affected the students' performance, which is reflected on the class outcomes 2, 3, and 4 lower scores, yet it was not that bad, I was able to keep the student attention in the class, by asking questions and rewarding extra points for first three correctly answered. Moreover, I made sure about the fairness of the exams (with no chance to search for the answers while answering the exams). CSE 325: Due to the time limit, the topic of security and protection mechanisms was not covered in Fall 2020. I plan to drop this outcome as the department will offer a new security course. Lab6 (Bigger files for xv6) is not assigned to the student due to the time limit and pandemic, I will assign this lab to students in Fall 2021. CSE 331: Due to the COVID-19 pandemic, I didn't have enough time to cover all topics of instruction level parallelism. Some topics such as Tomasulo's algorithm and loop unrolling were not taught this semester. One lab was not given to students due to the shortened semester. It is expected that the future offering of the course will go back to normal when the pandemic has passed. CSE 353: I plan to further improve the performance of outcome 6 by giving more in-class examples and assignments. Due to the time limitation, I was not able to offer a project in socket programming as planned. This will be considered in the next offering of the course. CSE 423: The course is difficult. The instructor will increase the amount of team/group work in future offerings to enable more software engineering aspects to be learned and evaluated.

#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

<table>
<tr><th>Student Learning Outcomes</th><th>Assessment Procedures</th><th>Assessment Results</th><th>Assurance</th></tr>
<tr><td>Learning Outcomes of the Program—Students will be able to:</td><td>Process/Instrument used:
Direct measures. Graded items are weighted and linked to courses; courses are weighted, aggregated, and linked to student outcomes. Covered Fall 2020, and Spring 2021.</td><td>What were your findings?
Score range: 1 (unsatisfactory),
2 (marginal),
3 (satisfactory), and
4 (excellent).</td><td>Our department believes we fulfill this Learning Outcome because:
(state evidence in 30 words or less)</td></tr>
<tr><td>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</td><td>Direct Measure: Quantitative Assessment Procedure (see below) on CSE 113, 122, 221, 222, 324, 353, and 423.</td><td>
<table>
<tr><th>Course</th><th>Score</th><th>Weight</th><th>Overall</th></tr>
<tr><td>CSE 113 F20</td><td>2.33</td><td>1</td><td rowspan="9">3.3</td></tr>
<tr><td>CSE 113 S21</td><td>2.33</td><td>1</td></tr>
<tr><td>CSE 122 F20</td><td>3</td><td>1</td></tr>
<tr><td>CSE 122 S21</td><td>3</td><td>1</td></tr>
<tr><td>CSE 221 F20</td><td>3.33</td><td>2</td></tr>
<tr><td>CSE 222 S21</td><td>3.67</td><td>2</td></tr>
<tr><td>CSE 324 S21</td><td>3.7</td><td>3</td></tr>
<tr><td>CSE 353 F20</td><td>4</td><td>3</td></tr>
<tr><td>CSE 423 S21</td><td>3</td><td>3</td></tr>
</table>
</td><td>the overall score is higher than 3.0, our acceptance threshold.</td></tr>
</table>

Adjustment/Improvement

CSE 113: The Women in Computer Science program in the department decided to have two sections which separating experienced and inexperienced students: CSE 113 Silver is the section for students who have no prior programming experience, and CSE 113 Blue is one for students who have learnt any programming language before. Students took the placement test to help instructors understand students' past programming experience and find the best section for students. Students who are not in the right section can decide whether they would like to stay in the current section or switch to the appropriate section. We hope this change can build confidence of students who have less programming experience and give more challenge for students who have more programming experience.

However, we are under the coronavirus pandemic and the classes are in hybrid mode. It is hard to determine whether or not the two sections can help students. The major goal in this semester is to setup the online learning environment to conquer the difficulties of teaching during pandemic:

1. A department Discord server is setup for students exchanging information and establishing connection.
2. The pair-programming activities in this class was in-person activities. All the activities are moved on the Discord voice channels with Go Live function on Discord and the Live Share extension on VSCode. Students can easily see each other's screens and edit the same files.
3. The tutoring service is also moved on Discord. A chatbot is developed by the TA and help moderating the tutoring request.

Although we made changes for the pandemic, the performance was still dropping down. There were about 8 students among 55 who did not take the final exams and failed in the class. We hope after one semester, students are more used to be in the online class and have a better performance.

The instructors also figure out that there are more non-CS students taking the CS core courses. Students who are majoring Biology with Bioinformatics concentration have to take this class in order to take CSE 373 (Introduction to Database Systems). We are planning to design questions in the lab assignments that is more closed to Biology area so it can engage their interests.

CSE 122: I redesign the whole class after F20. A coding assignment (Homework 4) about heap sort, binary search, permutation, and combination is added into the course in S21. More coding assignments will be added into the course.

CSE 221: More topics related to the computer architectures should be added like basic concept of cache.

CSE 222: The instructor did not provide adjustments/improvements.

CSE 324:

CSE 353: See comments related to SO #1.

CSE 423:

#3 Communicate effectively in a variety of professional contexts

<table>
<tr><th>Student Learning Outcomes</th><th>Assessment Procedures</th><th>Assessment Results</th><th>Assurance</th></tr>
<tr><td>Learning Outcomes of the Program—Students will be able to:</td><td>Process/Instrument used:
Direct measures. Graded items are weighted and linked to courses; courses are weighted, aggregated, and linked to student outcomes. Covered Fall 2020, and Spring 2021.</td><td>What were your findings?
Score range: 1 (unsatisfactory),
2 (marginal),
3 (satisfactory), and
4 (excellent).</td><td>Our department believes we fulfill this Learning Outcome because:
(state evidence in 30 words or less)</td></tr>
<tr><td>3. Communicate effectively in a variety of professional contexts</td><td>Direct Measure: Quantitative Assessment Procedure (see below) on CSE 326, and 331.</td><td><table><tr><th>Course</th><th>Score</th><th>Weight</th><th>Overall</th></tr><tr><td>CSE 326 S21</td><td>3.5</td><td>3</td><td rowspan="2">3.7</td></tr><tr><td>CSE 331 S21</td><td>4</td><td>2</td></tr></table></td><td>the overall score is higher than 3.0, our acceptance threshold.</td></tr>
</table>

Adjustment/Improvement
CSE 326: CSE 331: It is expected that the future offering of the course will go back to normal when the pandemic has passed.

#4 Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles

<table>
<tr><th>Student Learning Outcomes</th><th>Assessment Procedures</th><th>Assessment Results</th><th>Assurance</th></tr>
<tr><td>Learning Outcomes of the Program—Students will be able to:</td><td>Process/Instrument used:
Direct measures. Graded items are weighted and linked to courses; courses are weighted, aggregated, and linked to student outcomes. Covered Fall 2020, and Spring 2021.</td><td>What were your findings?
Score range: 1 (unsatisfactory),
2 (marginal),
3 (satisfactory), and
4 (excellent).</td><td>Our department believes we fulfill this Learning Outcome because:
(state evidence in 30 words or less)</td></tr>
<tr><td>4. [Applications:] exposure to one or more computer science application areas;</td><td>Direct Measure: Quantitative Assessment Procedure (see below) on CSE 382.</td><td><table><tr><th>Course</th><th>Score</th><th>Weight</th><th>Overall</th></tr><tr><td>CSE 382 F20</td><td>3.8</td><td>4</td><td>3.8</td></tr></table></td><td>the overall score is higher than 3.0, our acceptance threshold.</td></tr>
</table>

Adjustment/Improvement
CSE *382: Instructor also provided course learning outcomes pertaining to communications that might be used as inputs for SO #3.*

#5 Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline

<table>
<tr><th>Student Learning Outcomes</th><th>Assessment Procedures</th><th>Assessment Results</th><th>Assurance</th></tr>
<tr><td>Learning Outcomes of the Program—Students will be able to:</td><td>Process/Instrument used:
Indirect measure. Team members were surveyed on three yardsticks. A score for interaction effectiveness for the entire class was computed.
Covered Fall 2020, and Spring 2021.</td><td>What were your findings?
Score range: 1 (unsatisfactory),
2 (marginal),
3 (satisfactory), and
4 (excellent).</td><td>Our department believes we fulfill this Learning Outcome because:
(state evidence in 30 words or less)</td></tr>
<tr><td>5. [Tech Comm:] technical communication skills in written and oral form;</td><td>Direct Measure: Quantitative Assessment Procedure (see below) on CSE 325, and 326.</td><td>
<table>
<tr><th>Course</th><th>Score</th><th>Weight</th><th>Overall</th></tr>
<tr><td>CSE 325 F20</td><td>4</td><td>3</td><td rowspan="2">4</td></tr>
<tr><td>CSE 326 S21</td><td>4</td><td>3</td></tr>
</table>
</td><td>the overall score is higher than 3.0, our acceptance threshold.</td></tr>
</table>

Adjustment/Improvement
CSE 325: See earlier comments about CSE 325. CSE 326: See earlier comments about CSE 326.

#6 Apply computer science theory and software development fundamentals to produce computing-based solutions.

<table>
<tr><th>Student Learning Outcomes</th><th>Assessment Procedures</th><th>Assessment Results</th><th>Assurance</th></tr>
<tr><td>Learning Outcomes of the Program—Students will be able to:</td><td>Process/Instrument used:
Direct measures. Graded items are weighted and linked to courses; courses are weighted, aggregated, and linked to student outcomes.
Covered Fall 2020, and Spring 2021.</td><td>What were your findings?
Score range: 1 (unsatisfactory),
2 (marginal),
3 (satisfactory), and
4 (excellent).</td><td>Our department believes we fulfill this Learning Outcome because:
(state evidence in 30 words or less)</td></tr>
<tr><td>6. [Team:] the capacity to work as part of a group</td><td>Direct Measure: Quantitative Assessment Procedure (see below) on CSE 241, 326, 342, 344, and 423.</td><td>
<table>
<tr><th>Course</th><th>Score</th><th>Weight</th><th>Overall</th></tr>
<tr><td>CSE 241 F20</td><td>3</td><td>2</td><td rowspan="5">3.3</td></tr>
<tr><td>CSE 326 S21</td><td>4</td><td>3</td></tr>
<tr><td>CSE 342 S21</td><td>3</td><td>3</td></tr>
<tr><td>CSE 344 F20</td><td>3.6</td><td>3</td></tr>
<tr><td>CSE 423 S21</td><td>3</td><td>3</td></tr>
</table>
</td><td>the overall score is higher than 3.0, our acceptance threshold.</td></tr>
</table>

Adjustment/Improvement
CSE 241: The online offering of this course as planned out was a failure. I had spent the summer trying to learn how to conduct a black-board based, practice heavy class online. However, the outcome was not a success. There were several specific reasone the class failed - zoom was unable to handle the class size - especially in encouraging the students to answer questions. If a student paused, I was unable to judge if they are finding the right word, or need help: this stalled interaction, which was essential the way the course was structured. I had experimented polls etc. with a small number of zoom audience, but at full capacity, it failed. The upshot was that I was unable to gauge the understanding of students, and could cover less material than planned - which had a cascading effect of having to restructure homework almost every week, etc. Overall, I was unable to go into depth of recursion and induction. Other material was covered in the same depth as before, Also, the 0-credit lab that was introduced could not be conducted as designed – it ended up being an exercise session. If the class is offered online again, I plan to make short videos and use the class time for exercises. CSE 326: The improvement made on Outcome 3 in 2021 was noteworthy. This seems to be the results of more design-related homework assignments including design patterns. CSE 342: This year's course structure was determined by covid-19. I decided on short(-ish) videos embedded with quizzes posted each week that the students are to watch. One other thing I tried - which was mostly well received - was to give addition online resources not created by me. This was especially useful to run DFA and Turing machines. I now have videos that I can use as reserve material in future classes.

CSE 344: Hopefully, next Fall, the class will be taught face-to-face. That would encourage greater student engagement in class which, in turn, should benefit weaker students, who tended to avoid class involvement. Furthermore, examinations and proctoring would be less restricted. Overall, this instructor is unhappy with the use of Zoom in a large class because student engagement was lower than usual. Hopefully, reverting to traditional instruction modalities will help us resume continuous improvement.

CSE 423: ?

Concluding Comments:

Based on ABET requirements, we continue to use last year's set of program / student outcomes and curricular map.

This is the first report to assess ethics based on the course CSE 382 taught by philosophy professor Christopher ChoGlueck.

Overall, a major challenge is the increase in the number of students, which is negatively impacting the amount of interaction and help available to students. We need to increase the number of faculty and teaching assistants in order to continue to provide quality education.

This report has been read and accepted by the department faculty on Nov 8, 2021.

All course assessment reports can be supplied on request.

Submitted by: Clinton Jeffery Department Chair: Clinton Jeffery Date: 11/8/2021

Reviewed by Assessment Director/Director Signature: Date: (by 10/01)
Comments:

Reviewed by Faculty Senate Assessment Committee/Committee Chair Signature: Date: (by 11/01)
Comments:

Reviewed by Associate VP of Academic Affairs/AVPAA Signature: Date: (by 11/15)
Comments:

Submitted to Vice President of Academic Affairs/Date: (no later than 9/15)