BS in Computer Science - Resolution of Recommendations arising from the 2017-2019 Assessment report

<u>Prepared by Jai Navlakha, Assessments Coordinator</u>   **March 10, 2020**

Based on its consideration of the Biennial Assessment Report of the BS in Computer Science program for 2017-2019 cycle which included recommendations made by the Subject Area Coordinators (SAC), the Undergraduate Committee (UGC) of the School of Computing and Information Sciences discussed them and made several recommendations of its own. These recommendations were evaluated by Dr. Shu-Ching Chen, the Associate Director and the Director of Undergraduate programs, who made his additional recommendations. Resolution of these three sets of recommendations concludes the 2017-2019 assessment cycle for the BS in Computer Science Program, and are summarized below:

A. Recommendations originated by the Subject Area Coordinators:

- <u>CAP 4630: Students complained that the course covers too much material</u>

  <u>Recommendation</u>**:** The SAC agreed with the students and recommended: (1) Remove the unit on game theory and multi agent systems, (2) Remove mixed integer programming, linear programming, and MDPs from unit two, and (3) Spread the remaining material across the allotted time. *UGC suggested the SAC to revise the syllabus and include learning outcomes in consultation with faculty members who teach COP 4630.*

  <u>Resolution:</u> *Agreeing with the recommendation of UGC, Dr. Chen assigned Prof. Leonardo Bobadilla to work with the SAC to make the desired changes.*

- <u>CDA 4625: Include</u> more hands-on and visual material as suggested by students.

  <u>Recommendation</u>**:**

  <u>Resolution:</u>

- **CGS 3095:** No changes are recommended. However, some students requested that the course material include more focus on the impacts of social media and destructiveness of tech startups.

  <u>Recommendation</u>**:** <mark>*UGC recommends the SAC to review the topics covered and if necessary, include suitable learning outcomes in consultation with faculty who teach this course.*</mark>

  <u>Resolution:</u> *The Associate Director concurs with this recommendation, and will convey this to the SAC.*

- **ENC 3249**: No changes are recommended. However, technical writing should be emphasized more in this course as instructors [of CGS 3095] found the students "deficient to adequate" in writing skills.

  Recommendation: *UGC refers this issue to the administrator to discuss with English Department faculty.*

-
- Resolution: *The UGC chair together with Associate Director will discuss this with English Department to emphasize increased technical writing in this course.*

- CDA 3103: It was observed that the use of interactive textbooks (Zybooks) improves student learning, and it should be continued in the new course CDA 3102.

  Recommendation: *UGC agrees with this remark and the SAC should convey to faculty who teach CDA-3102.*

-
- Resolution: Dr. Chen *concurred with UGC's suggestion and will ask SAC to convey it to faculty members who teach this class.*

-
- **CDA 4101**: An introductory lecture with online resources for Verilog at the beginning of the term is essential. For each group project, include peer evaluations among group members to address an important concern expressed by students about sharing the work in group projects and the credit for each team member.

- Recommendation: *UGC recommends no action since this course will be discontinued from Spring 2021.*

-
- Resolution: *The Associate Director concurred with UGC's recommendation. This course will be discontinued from Spring 2021 due to the curriculum changes.*

- **COP 4722**: One instructor consistently indicated that the objective "Object-Oriented Database" and "Spatial and Multimedia Databases" as inappropriate. It is recommended that two outcomes of this course need to be discussed and possibly readjusted.

  Recommendation: *UGC recommends the SAC to revise the syllabus appropriately.*

-
- Resolution: Dr. Chen will *inform the SAC to revise the syllabus according to the UGC's recommendation.*

- **COP 4534**: Instructors found students' preparedness for the class to be between deficient and adequate. An instructor comment suggested that students should have some basic knowledge of combinatorics, statistics, and probability before taking this course. This lack of preparation is observed in many courses in this subject area, and we need to come up

with some scheme of topic coverage in this and pre-requisite courses to alleviate this problem.

Recommendation**:** <mark>UGC recommends SAC to consider adding (STA-3033 or MAD-4203) as an additional prerequisite.</mark>

Resolution: Dr. Chen: "*Let's be more careful before considering adding STA-3033 or MAD-4203 as a prerequisite. We don't want to create a long prerequisite chain and also have prerequisite from other departments. It is not that a prerequisite course is lacking for students in this class but the *quality* of the knowledge and skills they should have obtained from the courses they have taken.*" *Accordingly, Dr. Chen will speak with the SAC to deliberate this issue with faculty members who teach this course and its prerequisite courses.*

COP 4555: Instructors found that the student preparation for this class ranges from deficient to adequate. A few professor appraisal comments stated that students need better mathematics preparation to understand the essential concepts of functions, sets, and relations; better rigorous thinking and logical reasoning capabilities; and that the course be taught in a laboratory to practice programming in F#. Lack of preparation is observed in many courses in this subject area, and we need to come up with some scheme of topic coverage in this and pre-requisite courses to alleviate this problem.

Recommendation**:** <mark>UGC recommends SAC to add CDA-3102 as an additional prerequisite so that students would have exposure to computer architecture.</mark>

Resolution: The Associate Director as well as most of the instructors of this course *object this recommendation. Computer Architecture does not address the deficiency identified. Also assembly language does not have a major impact on this course since this course does not cover much of the program implementation and bypasses assembly language completely. Adding another pre-requisite lengthens the dependency chain and can create logistic problem for students and thus impact the graduation rate. The current coverage emphasizes the functional programming that is "declarative" and thus is very different from the "imperative" programming such as Java and C++ etc. Understanding functions, induction, recursion, and pattern matching are the key.*

COT 3100: Students' preparation for this course ranges from non-existent, deficient, adequate to good. A few professor appraisal comments included (1) student must develop stronger work ethics prior to enrolling in this course, (2) the number of course objectives is too high, (3) the outcomes related to programming should be compressed into a single outcome and make it be "familiarity" rather than implementation, (4) students have a very low level of math and logical reasoning and therefore it is very difficult for them to formalize problems and proofs, and (5) there is no time to properly cover some of the objectives related to program implementation. We need to address the lack of preparation for this course, too.

Recommendation**:** *UGC makes no recommendation: Since this is an introductory course to proofs, time is required for students to learn the basics. (1,4). UGC suggests SAC to clarify the distinction between course outcomes and learning objectives to instructors (2, 3, 5). UGC further recommends that this course need to be scheduled only for full terms (not in mini-terms). UGC recommends class size with low students to faculty ratio.*

A thought on this: Programming I (COP2210) is a co-req for this course, not a pre-req. The students need only have some college-level programming course (meaning even COP1000, which uses "graphics and animation in a media programming environment", can count). I sincerely question if that is enough. If we make Programming I a pre-req students will enter with more experience thinking *algorithmically*. Additionally as far as I can tell, the length of the flowchart would not be increased by this change. I would support it.

Resolution: Dr. Chen asserts that w*e will try our best to offer this course only for full terms (not in mini-terms). We will also try our best to have a low students to faculty ratio.*

COT 3541: Students suggested that more videos be used to explain the course material, to have more time on Prolog, to have a better textbook, and to have more examples. One comment of the online offering was to change discussion posts to classwork.

Students' preparation for this course was adequate. A few professor appraisal comments included this course has effectively challenged students to think and logic provides the unifying foundation for computer science. One suggestion was to explicitly cover propositional logic to help students have a consistent and systematic knowledge of various concepts in logic.

Recommendation**:** *UGC recommends SAC to find a suitable textbook and to suggest instructors to provide online resources and explore the possibility of recording lectures (with tools such as Momentos).*

More a comment, I saw the Syllabus and propositional logic is in the course description. I agree with what the UGC says here.

Resolution: *I concurred UGC's suggestion and will ask SAC to find a suitable textbook.*

COT 4521: Instructor suggested that Data Structures and Linear Algebra should be required pre-requisites. Further, more demos could also help student understanding.

> Recommendation**:** *UGC recommends to add Linear Algebra as a Foundation group elective and as a prerequisite to this course.*

Resolution: *We already submitted the request to add Linear Algebra as a Foundation group elective to the university and are waiting for the approval.*

A general note: There are a few common problems in the Foundation Area courses listed above: (1) deficiency of students' preparation in math and logical thinking and (2) how to help students to better understand course materials and prepare for exams. The offering of COT-3100 discrete structures may alleviate problem (1) for some other courses, but itself encounters the same problem. To address problem (2), homework grading criteria need to be changed to discourage homework copying and encourage student efforts; and in-class practices and quizzes are used to improve students understanding of fundamental concepts and performance on exams. Several observations include low student evaluation responses and missing appraisal comments from several instructors consistently, which need to be addressed to improve learning.

> Recommendation**:** *UGC recommends the administrators to follow-up on Instructor Course Appraisal process and to support academic integrity effort.*

Resolution: *A committee has been formed to address the academic integrity issue. This committee will provide suggestions and guidelines to faculty to deal with this issue.*

MAD 2104 & MAD 3512: Neither student evaluations nor instructor appraisals are available for these courses. No changes are recommended.

Subject Area: Programming (SAC: Tim Downey)

COP 2210: Continue to evaluate the effectiveness of the math prerequisite, but do not change the prerequisite at this time. Continue to urge instructors to cover all the outcomes of the course. No change is needed on the course outcomes or syllabus.

> Recommendation**:** *UGC recommends administrators to conduct an ongoing evaluation on the correlation of student performance in COP-2210 and MAC-1105.*

Resolution: *Our school will continue working with UGC to conduct an ongoing evaluation on the correlation.*

COP 3337: The school has instituted a new design for COP2210, with fewer sections and a common exam. This should address the concern of students having diverse preparation for the

course. The low coverage in some semesters is not a problem with the structure of the course, but with the presentation of the material. All instructors should be encouraged to cover all the material in a meaningful way. A review of the outcomes should be made to assess if removing some of the outcomes would maintain the content of the course and allow more time for other topics.

Recommendation: *UGC reviewed the course outcomes and they all seem to be appropriate.*
UGC recommends adding learning outcomes in the syllabus to help instructors to understand the coverage of each topic.

Resolution: *I concurred this suggestion.*

COP 3530: COP3337 instructors should ensure that all course outcomes are met. No change is needed in the course outcomes or syllabus.

Recommendation: *UGC recommends adding learning outcomes in the syllabus to help instructors to understand the coverage of each topic.*

Resolution: *I concurred this suggestion and will work with UGC to help instructors to add learning outcomes in the syllabus.*

COP 4226: Remove database connectivity from the outcomes because a database course is not a pre-requisite for this course.

Recommendation: *UGC recommends no change to the syllabus and course outcomes.*
UGC recommends adding learning outcomes in the syllabus to help instructors to understand the coverage of each topic in particular for database connectivity.

Resolution: *I concurred this suggestion and will work with UGC to help instructors to add learning outcomes in the syllabus*

COP 4338: The low coverage in some semesters mentioned by students is not a problem with the structure of the course, but with the presentation of the material. All instructors should be encouraged to cover all the material in a meaningful way. Instructors should be asked if there is enough time to cover the advanced material in the course while providing introductions to UNIX, pointers, and C data structures.

Recommendation: *UGC recommends adding learning outcomes in the syllabus to help instructors to understand the coverage of each topic.[include Bill's input on all course outcomes [3]] I have no issues with the course outcomes for this course. At some point, if students were introduced to C and/or Unix in prev courses, then less time could be spent on outcomes O1 & O2, and more time could be spent on O5 & O6, in keeping with the course name change to Systems Programming.*

This needs to be incorporated in the syllabus when the course gets redesigned. Consider using UNIX environment for some projects in COP-3337.

Resolution: *I concurred this suggestion and will work with UGC to help instructors to add learning outcomes in the syllabus*

Subject Area: Software Engineering (SAC: Masoud Sadjadi)

CEN4010:

o Observations:
  - The irrelevance of Net-Centric course as one of the pre-requisites for this course is rightfully questioned by the instructor and students of the course.
  - There is a request for adding agile software development approaches to this course to better serve the senior project.
  - Lack of enough teamwork experience is evident in some cases. Our professors would like our students to perform better in their groups.
  - Lack of enough exposure to software development tools such as version control (e.g., git).
  - Our students expect to learn more about the real-world problems and the state-of-the-art software engineering practices being used in industry.
  - They do not want to be bugged down with plenty of homework assignments and extra documentations that would be of no use to them in the future.
o Recommendations:
  - Prerequisite and Preparedness
    ❖ Net-Centric should be removed from the list of prerequisites for this course.
    ❖ Opportunities for teamwork experience in prior courses should be explored.
    ❖ Opportunities to expose students to software development tools such as version control should be explored in prior courses.
  - Agile and Scrum software development approaches should be included in the syllabus of this course.
    ❖ State-of-the-art practices of software development from industry should be adopted in this course.
    ❖ An Agile/Scrum textbook should be included as a reference, if not the main textbook of the course.
    ❖ Class lecture times should be spent more on practicing agile software engineering development than just giving lectures.
  - Learning by example and practice is the best way to transfer the knowledge and experience from the professor to the students.

  Recommendation**:**

The prerequisite for CEN-4010 has already been changed to "CGS 3095 and COP 3337" and CGS-3095 has group projects. UGC suggests providing online introduction to github using the school systems. UGC leaves the implementation of agile/scrum to the SAC. Additionally, the UGC suggests to include individuals with industry experience to give talks, particularly about software development tools currently used in industry, and to capitalize on opportunities during PAW (Panther Alumni Week).

UGC recommends adding learning outcomes in the syllabus, and to include UML in the learning outcomes.

Resolution: *I concurred this suggestion and will ask SAC to work with instructors to add learning outcomes to the syllabus and to include UML in the learning outcome.*

CEN 4021: The lack of UML knowledge is an indication that some professors might have not put enough emphasis on learning and practicing UML diagrams in CEN 4010 for the sake of adding some Agile/Scrum concepts. This should not be the case. Adding agile is a great improvement to CEN 4010, but it should not mean dropping the ball on the UML diagrams.

Recommendation**:** *UGC recommends adding learning outcomes in the syllabus to help instructors to understand the coverage of each topic.*

Resolution: *I concurred this request.*

CEN 4072: The following recommendations are made.

o Bring the syllabus of this course up to speed with the state-of-the-art practices in industry, test-driven development is one of the popular agile software development practices in industry. Students should be exposed to this approach.
o Debugging should stay in the syllabus as testing without debugging would not help with improving the quality of the software solution.
o Give students some hands-on experience, a good portion of the lectures time should be spent more on practicing the testing/debugging methods using state-of-the-art tools. Alternatively, some online tutorials can be suggested to the students to do some self-learning.

Recommendation**:** *UGC recommends the SAC to revise the syllabus with these recommendations.*

Resolution: *I concurred this suggestion..*

CIS 4911: The following observations and recommendations are made by the SAC.

- Students should be better prepared for this class.
    - Add Agile/Scrum software development approaches to CEN 4010. Also, they should learn and experience how to be a good team member in a self-organizing Agile/Scrum development team.
    - Adding Agile to the syllabus of CEN 4010 should NOT mean that learning of UML diagrams should be dropped or taken lightly. Our students must know how to read/create the most popular UML diagrams. Unfortunately, this is not the case for many of our students.
    - Provide students with a compressed Agile/Scrum online training at the beginning of the semester so that those of them who are lacking some knowledge in this area can catch up before the work on their senior projects starts.
    - An eligibility test should be taken at the beginning of the semester so that students are well prepared to perform in a project. This would avoid issues with their teammates during the semester.

        Recommendation: *UGC agrees for the SAC to create the Agile/Scrum online training. This course has been redesigned with a two-course sequence and this should alleviate the need for eligibility test.*

Resolution: *I concurred this suggestion.*

- The product owners should be better prepared for this class.
    - The product owners of approved projects must go through a short crash course on how to be a good product owner for our students.
    - They must commit to be available to answer our students' questions daily and be available to review/evaluate their work every other week and provide them with enough work for the following sprints ahead of time.

        Recommendation: *UGC agrees for the SAC to create and supervise crash course for product owners.*

Resolution: *I concurred this suggestion.*

- Expectations from the students should be clearly communicated to them.
    - The instructor of the class must provide clear breakdown of the points and provide students with bi-weekly updates on their status.

- More in-depth feedback should be provided to the students both by the product owner and the instructor of the class on an ongoing basis and when requested specifically by the students.
- All the requirements and guidance for the class should be easily accessible by the students. Even if some requirements and guidance may be required to be in different systems, there must be one starting point from which everything is accessible.

Recommendation: *UGC agrees with the above suggestions and recommends the SAC to implement them.*

Resolution: *I concurred these suggestions.*

- Need for professional system staff support.
    - The project management tools adopted for this class in some cases had been hacked and the server went down.
    - There should be one or more system staff at SCIS assigned to this course to manage the support software tools for the students.

Recommendation: *UGC asks the administration to explore the feasibility of allocating staff for this course.*

Resolution: *It will depend on the school budget. Currently, we don't have the budget to hire a professional system staff. We will see whether we can ask current IT staff to support it.*

A. Recommendations of the Assessments Coordinator

1. Course Related:

AC-01: The Course Outcomes Surveys for MAD 3512 are not conducted. This is a continuing concern. If possible, some other assessment mechanism should be employed for MAD 3512 on a regular schedule. Further, only 10 out of 33 students (36.36%) performed at an expected level or slightly below. We need to examine the reasons for this inadequate performance of students, and then coordinate the content and delivery of this course better with the Department of Mathematics and Statistics.
See the UGC comment in the Foundations SAC section above (for Discrete Structures).

AC-02: As expected in this assessment cycle, course outcomes for MAD 2104 are substituted by those for COT 3100, the course offered and controlled by SCIS. In the Course Embedded Direct Assessment for COT 3100, 22 out of 29 students (75.86%) demonstrated proficiency in Discrete Structures. This is significantly better than the evaluation for MAD 2104 (20 out of 36 = 55.56%). Students recommend to be assigned more in-class practice problems that should be considered by the instructors as a committee.
UGC requires further clarification on Foundations SAC recommendation

AC-03: All courses in the newly created Subject Area "Applications" except one require no changes. Artificial Intelligence (CAP 4630) was taught by an instructor who is no longer with us, and he did not do a good job at all. The course bites too much, and needs to be simplified as suggested by the SAC.
See the UGC comment in the Applications SAC section above.

AC-04: In this assessment cycle, a Math pre-requisite was introduced for the first Programming Course, COP 2100. Students still seem to be deficient in their mathematical preparation for this course. It is suggested that a committee of instructors examine the effectiveness of the math pre-requisite.
See the UGC comment in the Programming SAC section above.

AC-05: Students in the second Programming Course, COP 3337, were found deficient in their overall preparation for the course. The School has instituted a new design for the first Programming Course, COP 2210, with fewer sections and a common exam. This should begin addressing this concern. It is suggested that a committee of instructors examine the Course Outcomes to assess if removing some of the outcomes would maintain the content of the course and allow more time for other topics. The instructors should ensure that all Course Outcomes are met.
See the UGC comment in the Programming SAC section above.

AC-06: For COP 4226, one of the Course Outcomes includes "database connectivity" when no database course is pre-requisite for this course. Remove "database connectivity" from the Course Outcomes.

See the UGC comment in the Programming SAC section above.

AC-07: For the series of courses in the Subject Area "Foundations (COT 3100, COT 3541, COT 4521)," students' preparation in mathematical and logical thinking was observed to be deficient. I suggest that we wait for the next assessment cycle to gauge the effect of COT 3100 towards this pre-requisite preparation of students for these courses. Also, it seems that students require increased assistance to better understand the course material and prepare for exams. It is suggested that more in-class practice problems be assigned to students along with quizzes to gauge their level of understanding.
==UGC agrees with AC and also see UGC comment in the Foundations SAC section above.==

AC-08: It appears that there is mixed feeling among instructors of COP 4722 about including "Object-Oriented Database" and "Spatial and Multimedia Databases" in the Course Outcomes. The SAC suggests that this be discussed and possible adjusted; I concur with that suggestion.
==See the UGC comment in the Computer Systems SAC section above.==

AC-09: To address student and instructor concerns for courses in Subject Area "Software Engineering," the following suggestions are made for CEN 4010.

- Prerequisite and Preparedness
  - ❖ Net-Centric should be removed from the list of prerequisites for this course.
  - ❖ Opportunities for teamwork experience in prior courses should be explored.
  - ❖ Opportunities to expose students to software development tools such as version control should be explored in prior courses.
- Agile and Scrum software development approaches should be included in the syllabus of this course.
- State-of-the-art practices of software development from industry should be adopted in this course.
- An Agile/Scrum textbook should be included as a reference, if not the main textbook of the course.

==See the UGC comment in the Software Engineering SAC section above.==

AC-10: As suggested by some instructors of CEN 4072 "the details of debugging" is removed from the Course Outcomes. Some instructors believe that Debugging should stay in the syllabus as testing without debugging would not help with improving the quality of the software solution. I suggest that the instructors of this course come together and take a firm action on this issue. It is also suggested that students should be exposed to the test-driven development which is one of the most popular agile software development practices in the field.

==UGC agrees with AC and also see UGC comment in the Software Engineering SAC section above.==

AC-11: Most of the projects undertaken in CIS 4911 use Scrum. Accordingly, Scrum should be used in the pre-requisite course CEN 4010, or at the least, reviewed in CIS 4911. Additionally, students should be taught how to be a good team member for team projects.

AC-12: Beginning in Spring 2020, two courses, CDA 3103 and CDA 4101, will be replaced by CDA 3102. Since the use of interactive textbooks (Zybooks) was helpful in improving student learning in CDA 3103, <u>it is suggested that its use be continued in CDA 3102 also.</u>
UGC agrees with AC and also see UGC comment in the Computer Systems SAC section above.

2. Procedure Related:

AC-13: For a few courses, the Instructor Course Appraisals are not filed in. <u>The Associate Director (or designee) should ascertain that these are filled by the instructors every term.</u>
UGC agrees with the AC.

AC-14: We have now used the Employer Survey to measure attainment of Program Educational Objectives of our students for the third time. This is wonderful. However, the number of response (5 responses per question although 9 participated) was very low. <u>It is recommended that meaningful steps be taken in the future to increase this response rate. This is a continuing concern.</u>

UGC agrees with the AC and the administration may consider engagement with employers during the Senior project Day.

AC-15: In this assessment cycle, student participation in the Course Evaluation System was quite poor. This is mainly due to discontinuing our practice of taking netbook computers to every class and making students fill in these surveys. If that is not doable now for difficulties in its implementation, then <u>we must find other mechanism to improve this participation. May be, we should look into giving students some incentive to complete these surveys.</u>

UGC agrees with the AC.

AC-16: Very soon in the future, we will need to supply data to ABET for their evaluation of the online degree, BA in Computer Science, along with continuing application for BS in CS degree. Accordingly, beginning now, we should consider collecting Student and Instructor Evaluation of Course Outcomes separately for our online offerings.
UGC agrees with the AC.

3. General:

AC-17: It is challenging to perform meaningful assessment of Student Outcome a) ***Ability to apply knowledge of Computing and Mathematics*** using the rubric of the Senior Project class because there are essentially no projects attempted by students that address the relevant topics. <u>The point is made for discussion only; no recommendations are made</u>.
UGC agrees with the AC.

AC-18: The quality, the variety, and the number of activities performed by our student clubs (teaching new subjects through workshops, providing opportunities to improve through technical activities, providing outreach to communities by helping students in middle and high schools, and so on) has increased by a substantial percentage as compared with their past activities. Programming Team is well supported by Ultimate Software for many years and by FaceBook starting in 2019. SCIS administrators should continue to support them in whatever way possible, including providing more space for their activities.

UGC agrees with the AC.

AC-19: For a few years now, the meeting of the Industrial Advisory Board is conducted at the end of the Fall and Spring semesters when selected students present their Capstone Projects. Members have been suitably impressed with their work in the past and continue to be impressed now. Many members act as mentors and/or judges for these projects. This has proven to be very beneficial for the students. We should continue to find more and better ways to engage the Board members in student activities.

UGC agrees with the AC.

AC-20: To improve the response rate of Employers in their survey, we need to engage the Industrial Board members to respond themselves as they do hire many of our graduates, and entice other employers through their connections. May be, a letter from the Chairperson of the Board, specifying the importance of participation in this survey, should be attached with the survey instrument.

UGC agrees with the AC.

-