Assessment Driven Continuous Improvements in the B.S. in Computer Science Program at UL Lafayette

A. Student Outcomes

A.1. Assessment Process and Timeline

The Program evaluates student outcomes based on the data from its courses as well as inputs from current students, graduating students, alumni, faculty, employers of our graduates, and an advisory board. Results from peer faculty evaluations are also taken into consideration when appropriate.

Each semester, the Program gathers data for its core courses that contribute significantly to student outcomes. From each such course, the following materials are collected:

- copy of the syllabus with learning outcomes
- copies of assignment, project, and test/exam
- copies of the graded material for best, average, and lowest grades for each assignment, project, and test/exam
- copies of grading criteria/rubric for each assignment, project, test/exam
- data for each student on each parameter of the rubric/grading criteria

The mapping of Student Outcomes to courses for the Program is provided in Table 1 and the Student Outcomes are listed in Table 2.

Table 1: Course to Outcome Mapping for the Outcomes in Effect Through Spring 2019

Outcome → Course ↓	1	2	3	4	5	6	7
CMPS 310				**X**			
CMPS 340	**X**				**X**		
CMPS 341	**X**						
CMPS 351			**X**				
CMPS 430		**X**				**X**	
CMPS 450			**X**		**X**		
CMPS 453		**X**		**X**		**X**	**X**
CMPS 455	**X**	**X**					**X**
CMPS 460	**X**		**X**			**X**	**X**

Table 2: Student Outcomes of the Programs (In Effect Through Spring 2019)

Outcome Number	Outcome
1	Understand and use the fundamental principles of the science of computation, including those of algorithm analysis, software design, operating systems, and database.
2	Have the ability to analyze, design, implement, and test computer systems and components of varying complexity with respect to a variety of criteria relevant to the task.
3	Be proficient in more than one programming language on more than one computing platform.
4	Understand the professional, ethical, legal, and societal issues and responsibility as well as the global impact in the design and application of computer systems.
5	Possess the ability to solve problems using efficient algorithms, data structures, and appropriate design choices.
6	Communicate effectively, both orally and in writing.
7	Work effectively, both independently and in a team.

A.2. The Frequency of the Assessment Process

As described in A.1, the course data is collected every semester. Such a comprehensive gathering of data is done so that the best practices of assessment are followed by the Program on an ongoing basis. Availability of such data has helped the Program to gauge its level of success at a finer grain and kept instruction and learning at an excellent level. Additionally, such practice has fostered a culture of assessment and continuous improvement among the faculty in the Program.

Formal evaluation of course data is done based on an assessment plan which is shown in Table 3. It can be observed that the assessment plan shown in Table 3 is a subset of actual, comprehensive assessment done by the Program. Formal evaluation is done only at certain points (more specifically, synchronized with the University's evaluation cycle in Live Text). Since 2015, the Program has been using Live Text for its assessment reporting. Prior to that, the University used Weave which is not used any more.

Assessment based on graded material from courses such as tests, projects, presentations, and written reports are direct evidence of learning (direct measure) and they are used by the Program in conjunction with indirect measures which are based on surveys. Students provide their input about each course through Student Evaluation of Instruction (SEI) which is conducted by the University each semester. Faculty also conducts Peer

Evaluation of Instruction (PEI) each year such that all courses are peer evaluated. Surveys of alumni and employers as well as advisory board members are typically done once in five years and they are used to improve the Program as needed.

Table 3. Plan of Direct Assessment of Student Outcomes from Courses

Student Outcome (in effect through Sp 19)	Sp & Fa 2015	Sp & Fa 2016	Sp & Fa 2017	Sp & Fa 2018	Sp & Fa 2019
1		X		X	
2		X		X	
3		X		X	
4	X		X		X
5		X		X	
6	X		X		X
7	X		X		X

A.3. The expected level of attainment for each of the student outcomes

The expected level of attainment for each of the seven Student Outcomes is an average of 2.8 out of 4 on the surveys (i.e., from the indirect assessment). The expected level of attainment for direct assessment is explained next.

Direct assessment of an outcome is done per the plan in Table 3. A student outcome is considered achieved (or its goal is considered met) in a course if 70% of students receive 70% or more marks on relevant exams, tests, homework, projects, reports, or presentations in that course. A student outcome is considered achieved (or its goal is considered met), though the direct assessment, for the Program if all assessed courses meet their goals.

A.4. Summaries of the results of the evaluation process and an analysis illustrating the extent to which each of the student outcomes is being attained

The results of the evaluation are described next. Receiving unadjusted/raw score on relevant exams, tests, homework, projects, reports, or presentations in the 85%-100% range is termed *Exemplary*, in the 70%-84% range is termed *Developed*, in the 50%-74% range is termed *Developing*, and in the 0%-49% range is termed *Amateur*. A student

outcome is considered achieved (or its goal is considered met) from an assessed course if 70% or more students achieve Developed or Exemplary rating.

Results of Evaluation of Student Outcome 1 (Indirect Measure): *Understand and use the fundamental principles of the science of computation, including those of algorithm analysis, software design, operating systems, and database.*

The stakeholders of the Program were asked to evaluate the importance of Student Outcome 1 and to also evaluate to what degree the Student Outcome 1 was being achieved. The respective results are shown in Figures 1 and 2.

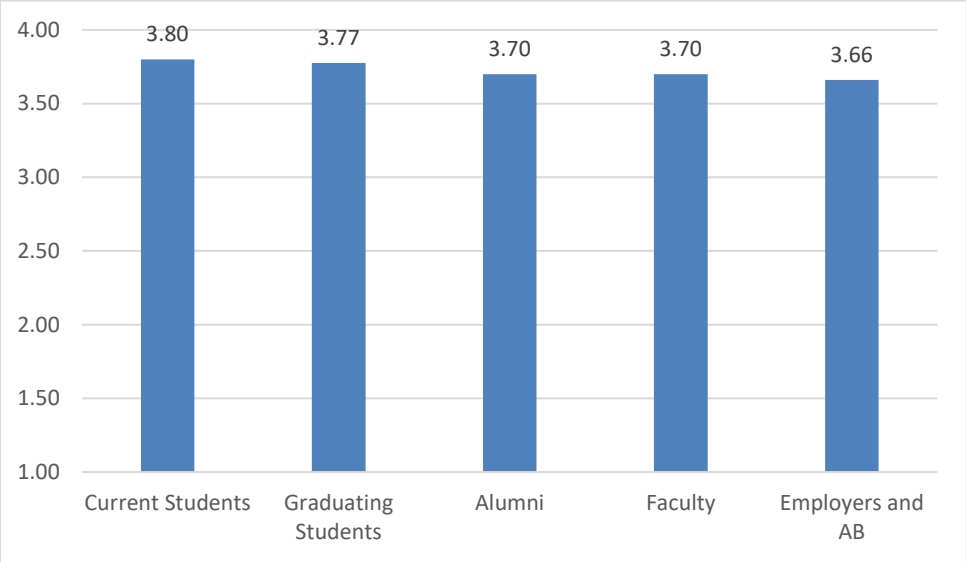

Figure 1. Importance of Student Outcome 1

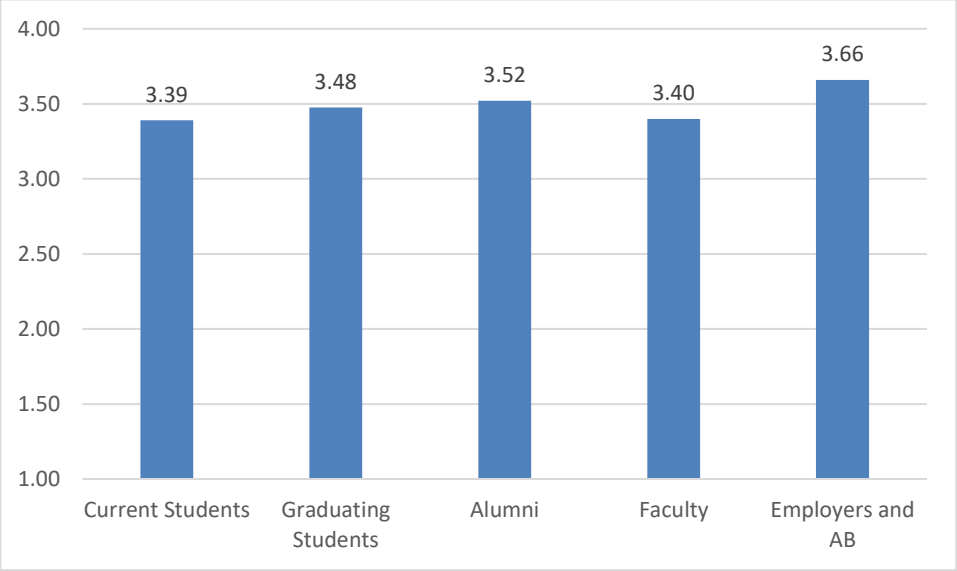

Figure 2. Success of Student Outcome 1

The goal for this outcome is to achieve an overall weighted average rating of 2.8 or above on the survey and it has been achieved.

Results of Evaluation of Student Outcome 2 (Indirect Measure): *Have the ability to analyze, design, implement, and test computer systems and components of varying complexity with respect to a variety of criteria relevant to the task.*

The stakeholders of the Program were asked to evaluate the importance of Student Outcome 2 and to also evaluate to what degree the Student Outcome 2 was being achieved. The respective results are shown in Figures 3 and 4.

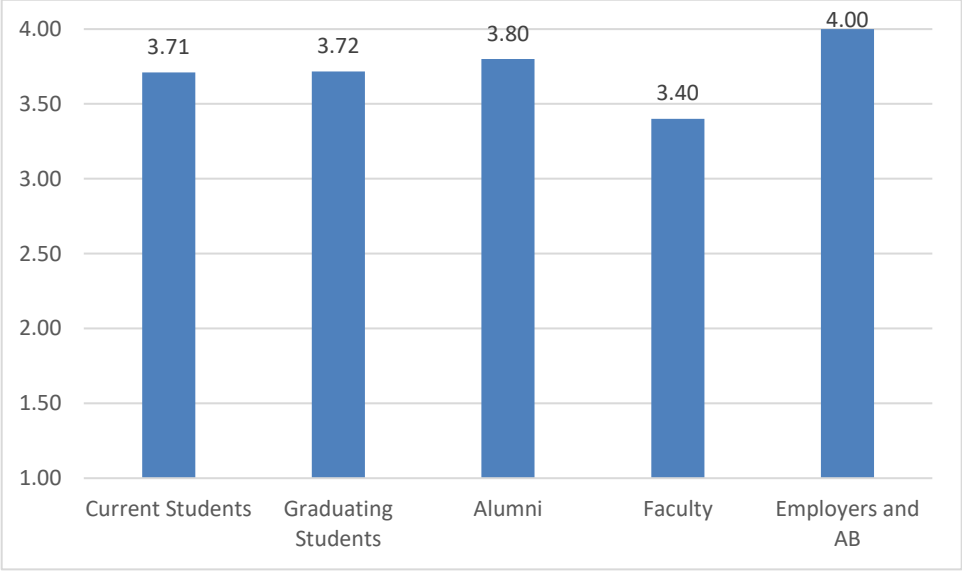

Figure 3. Importance of Outcome 2

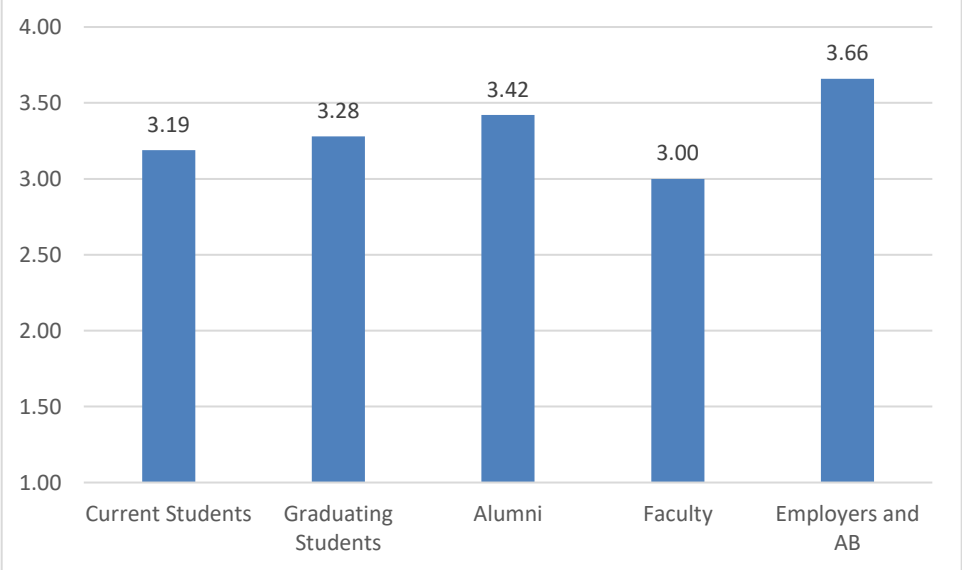

Figure 4. Success of Outcome 2

The goal for this outcome is to achieve an overall weighted average rating of 2.8 or above on the survey and it has been achieved.

Results of Evaluation of Student Outcome 3 (Indirect Measure): *Be proficient in more than one programming language on more than one computing platform.*

The stakeholders of the Program were asked to evaluate the importance of Student Outcome 3 and to also evaluate to what degree the Student Outcome 3 was being achieved. The respective results are shown in Figures 5 and 6.

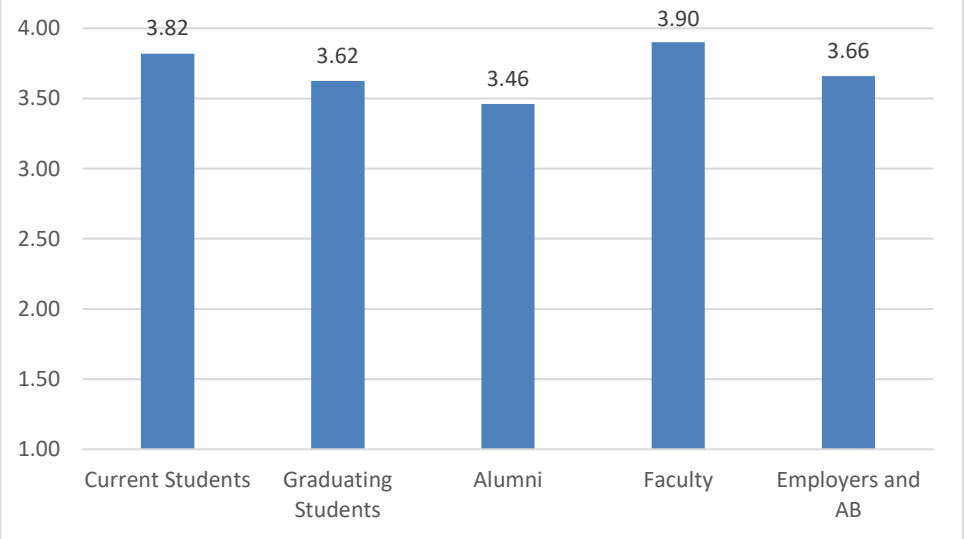

Figure 5. Importance of Outcome 3

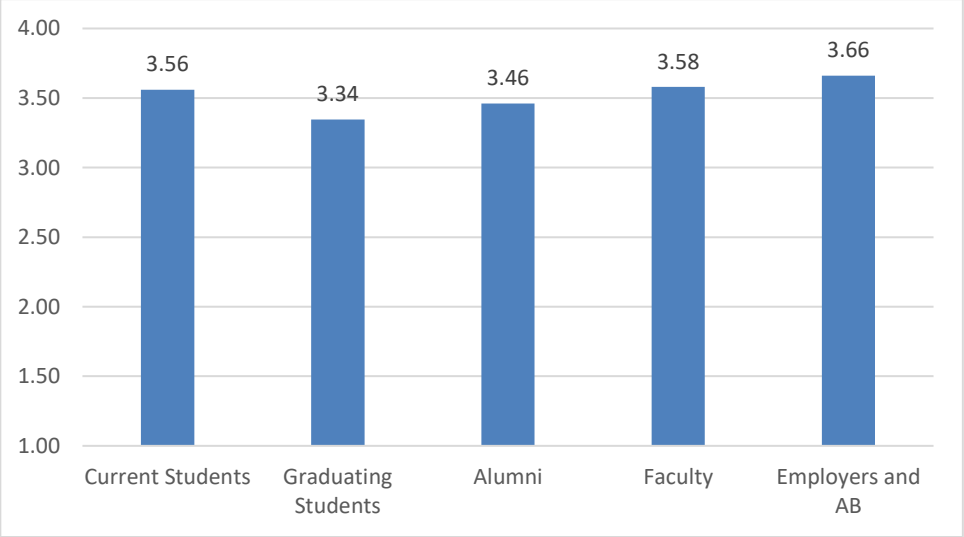

Figure 6. Success of Outcome 3

The goal for this outcome is to achieve an overall weighted average rating of 2.8 or above on the survey and it has been achieved.

Results of Evaluation of Student Outcome 4 (Indirect Measure): *Understand the professional, ethical, legal, and societal issues and responsibility as well as the global impact in the design and application of computer systems.*

The stakeholders of the Program were asked to evaluate the importance of Student Outcome 4 and to also evaluate to what degree the Student Outcome 4 was being achieved. The respective results are shown in Figures 7 and 8.

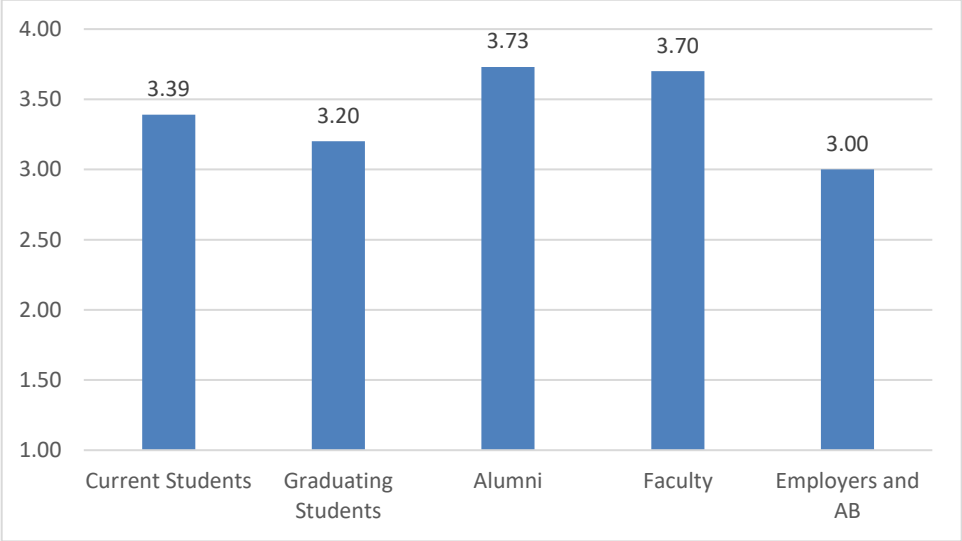

Figure 7. Importance of Outcome 4

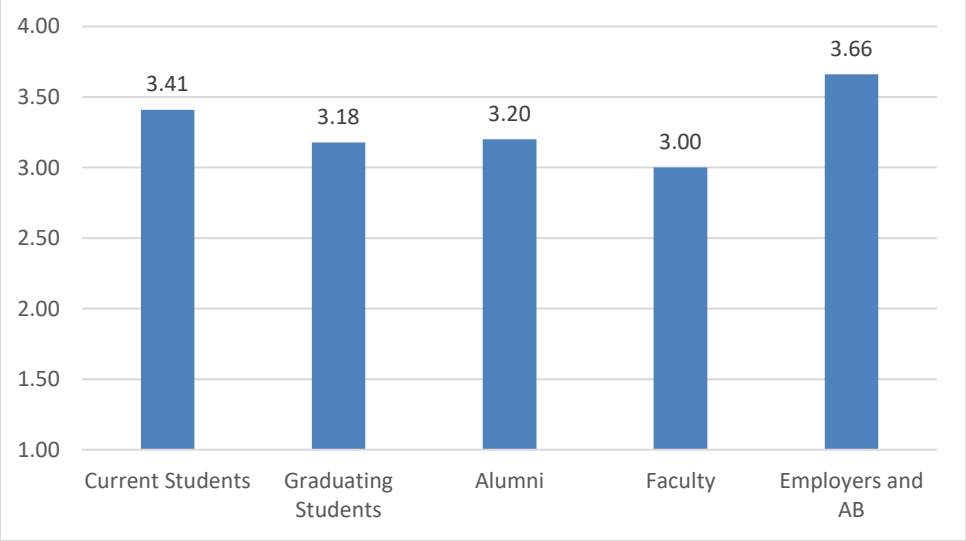

Figure 8. Success of Outcome 4

The goal for this outcome is to achieve an overall weighted average rating of 2.8 or above on the survey and it has been achieved.

Results of Evaluation of Student Outcome 5 (Indirect Measure): *Possess the ability to solve problems using efficient algorithms, data structures, and appropriate design choices.*

The stakeholders of the Program were asked to evaluate the importance of Student Outcome 5 and to also evaluate to what degree the Student Outcome 5 was being achieved. The respective results are shown in Figures 9 and 10.

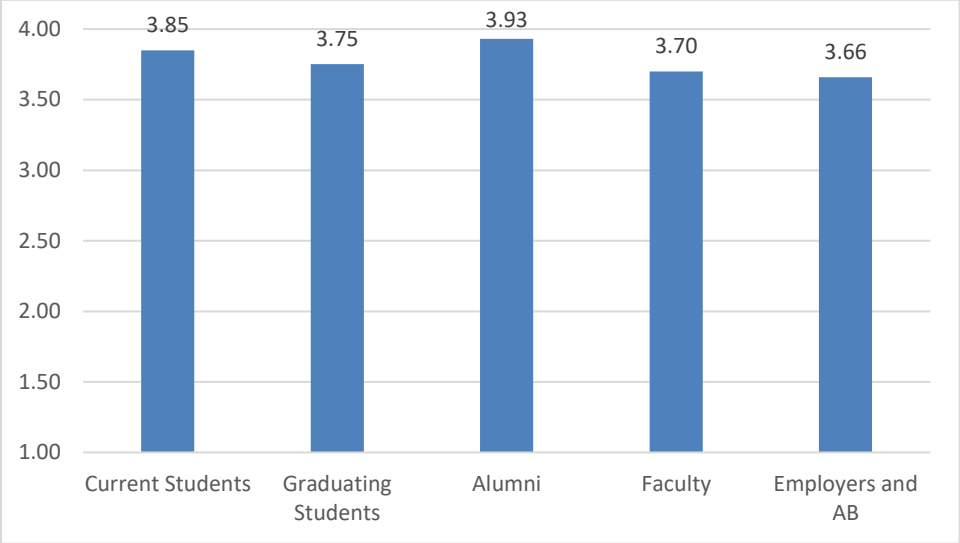

Figure 9. Importance of Outcome 5

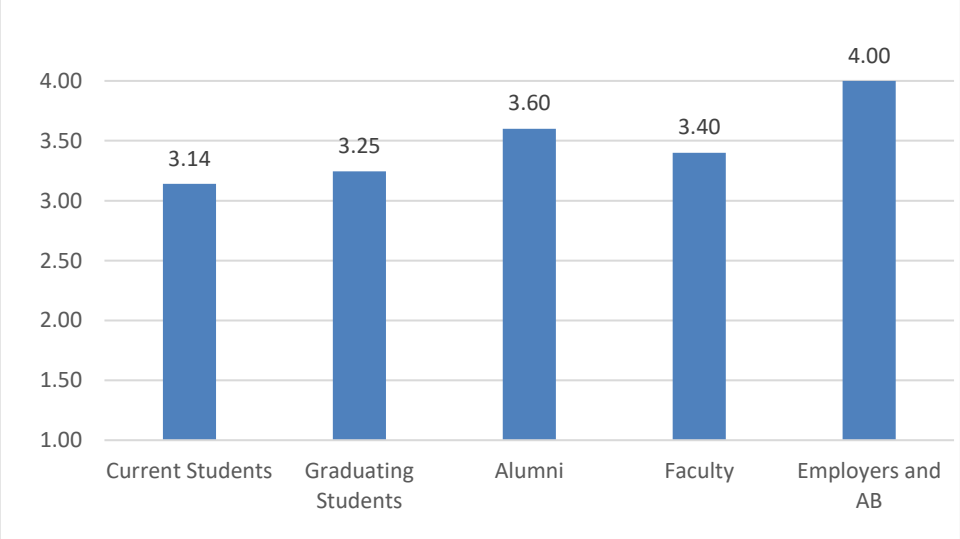

Figure 10. Success of Outcome 5

The goal for this outcome is to achieve an overall weighted average rating of 2.8 or above on the survey and it has been achieved.

Results of Evaluation of Student Outcome 6 (Indirect Measure): *Communicate effectively, both orally and in writing.*

The stakeholders of the Program were asked to evaluate the importance of Student Outcome 6 and to also evaluate to what degree the Student Outcome 6 was being achieved. The respective results are shown in Figures 11 and 12.

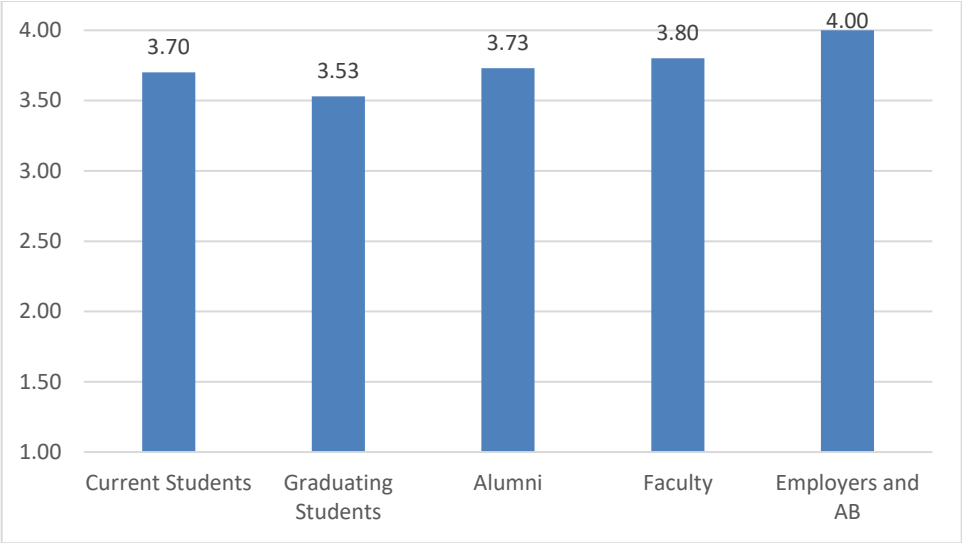

Figure 11. Importance of Outcome 6

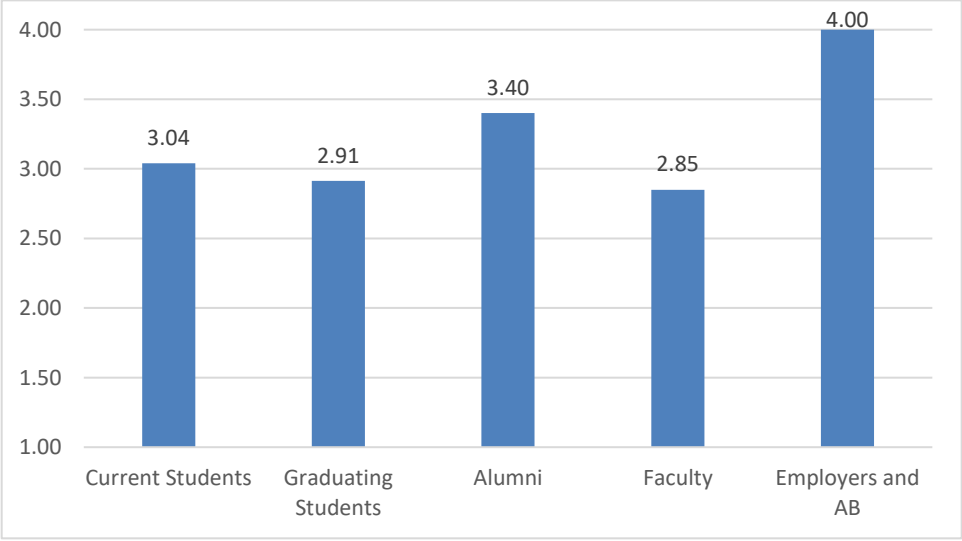

Figure 12. Success of Outcome 6

The goal is for this outcome was to achieve an average rating of 2.8 or above on the survey and it has been achieved.

Results of Evaluation of Student Outcome 7 (Indirect Measure): *Work effectively, both independently and in a team.*

The stakeholders of the Program were asked to evaluate the importance of Student Outcome 7 and to also evaluate to what degree Student Outcome 7 was being achieved. The respective results are shown in Figures 13 and 14.

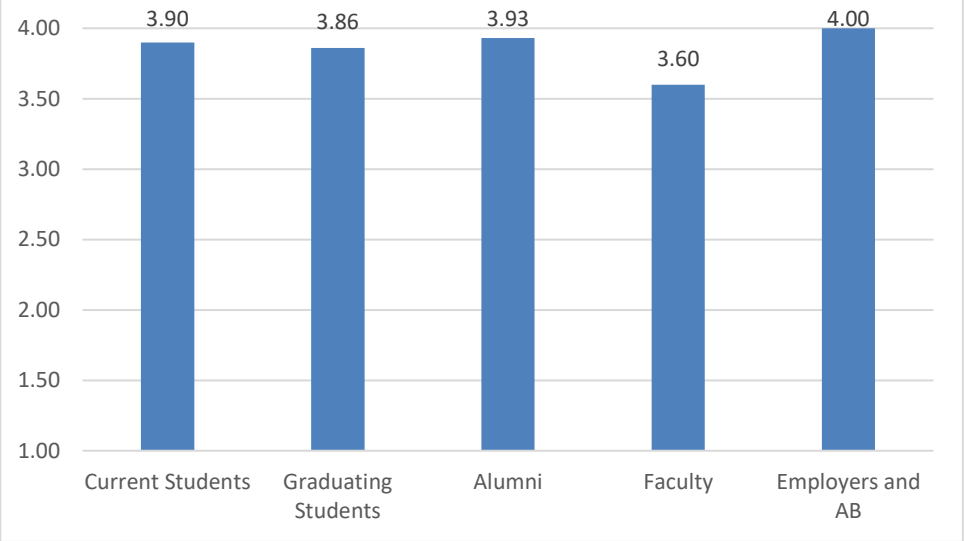

Figure 13. Importance of Outcome 7

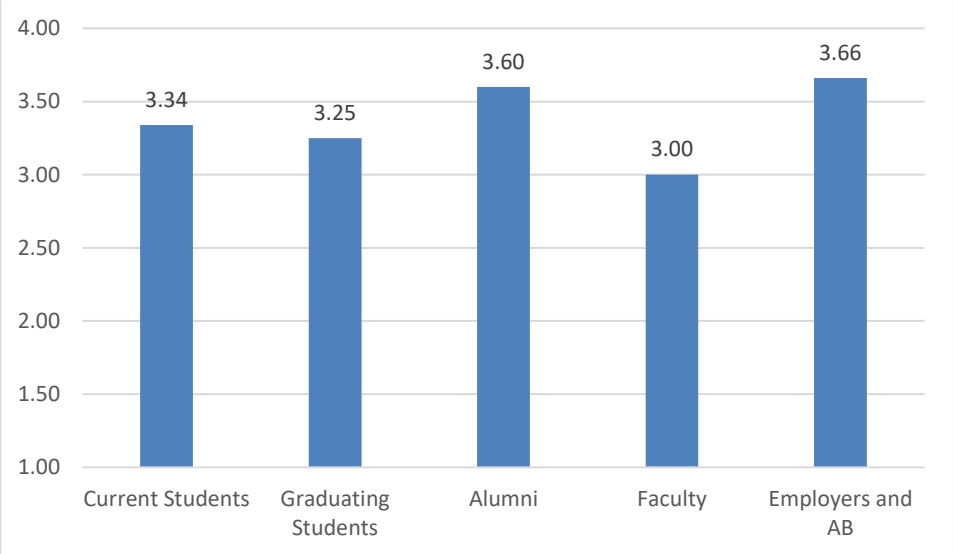

Figure 14. Success of Outcome 7

The goal for this outcome was to achieve an average rating of 2.8 or above on the survey and it has been achieved.

Results of Evaluation of Learning Outcomes: Direct Measure

As mentioned in section A.2 of this chapter, direct measure based assessment and evaluation is done based from the core courses' tests, exams, projects, reports, and presentations. A summary of evaluations is presented next.

Spring and Fall 2015 (2015-2016 Assessment Cycle in Live Text)

Outcome 4 (310 only)

310/Fall 2015: The outcome was assessed through CMPS 310 in Fall 2015. Twenty eight (28) out of 37 students (75%) achieved the rating of Developed or Exemplary on the rubric for this outcome. Goal met.

Overall: The program goal was met for this student outcome.

Outcome 6 (430, 453, 460)

This outcome was assessed for this cycle through the following courses: CMPS 430 (Fall 2015), CMPS 453 (Fall 2015), and CMPS 460 (Spring 2015). The following are the results.

430/Fall 2015: 12 out of 14 students (i.e., 85%) received Developed or Exemplary rating. Goal met.

453/Fall 2015: 40 out of 42 students (i.e., 95%) received Developed or Exemplary rating. Goal met.

460/Spring 2015: 52 out of 57 students (i.e., 91%) received Developed or Exemplary rating. Goal met.

Overall: The program goal was met for this student outcome.

Outcome 7 (453, 455, 460)
This outcome was assessed for this cycle through the following courses: CMPS 455 (Spring 2015), CMPS 453 (Fall 2015), and CMPS 460 (Spring 2015). The following are the results.

455/Spring 2015: 45 out of 50 (i.e., 90%) students received Developed or Exemplary rating. Goal met.

453/Fall 2015: 40 out of 42 students (i.e., 95%) received Developed or Exemplary rating. CMPS 460 (Spring 2015): 52 out of 57 students (i.e., 91%) received Developed or Exemplary rating.

Overall: The program goal was met for this student outcome.

Spring and Fall 2016 (2016-2017 Assessment Cycle in Live Text)

Outcome 1 (340, 341, 455, 460)

340/Fall 2016:
38% of all students achieved Developed or Exemplary rating on evaluation while 52% of the students achieved Developing rating. It was observed that most students were very close to achieving Developed rating. Goal not met.

Action: It was planned to introduce more applications and examples of different algorithm design techniques for a future offerings. It was also decided that instructors and teaching assistants would more actively offer their help to students with understanding the material in a subsequent offering. No change in the course content was devised.

341/Fall 2016: 71.3% of all students achieved Developed or Exemplary status on evaluation. Goal met.

460/Sp 16: 81% of all students achieved Developed or Exemplary status on evaluation. Goal met.

455/SP 2016: Only 59.3% of all students scored 70 or more on Test 2. Goal not met.

Action: Explain deeper concepts with more examples in the class and also review the material over two classes before the tests. It was also observed that Test 2 was very hard for most of the students. No change in the course content was devised.

Overall: The program goal was partially met for this student outcome.

Outcome 2 (430, 453, 455)

430/Fall 2016: 76.09% of all students achieved Developed or Exemplary rating on evaluation. Goal met.

453/Fall 2016: 89% of all students achieved Developed or Exemplary rating on evaluation. Goal met.

455/Spring 2016: 99% students achieved Developed or Exemplary rating on the projects. Goal met.

Overall: The program goal was met for this student outcome.

Outcome 3 (351, 450, 460)

351/Fall 2016: 83% of all students achieved Developed or Exemplary rating on evaluation. Goal met.

450/Fall 16: Only 51% of all students achieved Developed or Exemplary status on evaluation. **Goal not met.**

Action: It was observed that students needed more practical examples particularly for functional and logical programming. Therefore, in a subsequent offering, it was planned to demonstrate functional and logical programming on the computer and solve programing problems in the classroom. As a result, the ideas and programming in different programming paradigms would be more understandable for students.

460/Fall 2016: 77.4% of all students achieved Developed or Exemplary rating on evaluation. Goal met.

Overall: The program goal was partially met for this student outcome.

Outcome 5 (340, 450)

450/Fall 16: Only 51% of all students achieved Developed or Exemplary status on evaluation. Goal not met.

Action: Similar to the observation for outcome 3, it was observed that students needed more practical examples particularly for functional and logical programming. Therefore, in a subsequent offering, it was planned to demonstrate functional and logical programming on the computer and solve programing problems in the classroom. As a result, the ideas and programming in different programming paradigms would be more understandable for students.

340/Fall 16: 80% of all students achieved Developed or Exemplary status on evaluation. Goal met.

Overall: The program goal was partially met for this student outcome.

Spring and Fall 2017 (2017-2018 assessment cycle in Live Text)

Outcome 4 (310, 453)

310/Fall 17: 87% of all students achieved Developed or Exemplary rating on evaluation. Goal met.

453/Fall 17: 71.4% of all students achieved Developed or Exemplary rating on evaluation. Goal met.

Overall: The program goal was met for this student outcome.

Outcome 6 (430, 453, 460)

453/Fall 17: 99% of all students achieved Developed or Exemplary status on evaluation. Goal met.

430/Fall 17: 83% of the students achieved Developed or Exemplary rating from evaluation of their research report. Goal met.

460/Spring 17: 98% of all students achieved Developed or Exemplary status on evaluation. Goal met.

453/Fall 2017: 99% of all students achieved Developed or Exemplary status on evaluation. Goal met.

Overall: The program goal was met for this student outcome.

Outcome 7 (453, 455, 460)

453/Fall 17: 99% of all students achieved Developed or Exemplary status on evaluation. Goal met.

460/Spring 17: 96% of all students achieved Developed or Exemplary status on evaluation. Goal met.

455/Spring 17: i) 80% of all students scored 70% or more on the projects. Goal met.
ii) On the peer review form over 90% students scored 70% or more. Goal met.

Overall: The program goal was met for this student outcome.

Spring and Fall 2018 (2018-2019 assessment cycle in Live Text)

Outcome 1 (340, 341, 455, 460)

340/Spring 18: 63% of all students achieved Developed or Exemplary status on evaluation. Goal not met.

Action: It was observed that there was more than 10% increase in the number of students achieving Developed or Exemplary rating with respect to the evaluations in 2016. It was decided to continue the practice of explaining with more examples and provide support to students.

341/Fall 18: 66% of all students achieved Developed or Exemplary status on evaluation. Goal not met.

Action: Based on the assessed data and evaluation, it was decided that providing more real-world applications of graphs in computing would possibly help students to understand the graph concepts and algorithms better. Also, possibly doing more and deeper examples in the class will help students on understanding graphs and graph algorithms. A similar action would be taken for explaining mathematical functions by relating them more to functions in programming in a future offering of the course.

Update: **341/Spring19:** Evaluation of assessed data for CMPS 341 for Spring 2019 showed that 91.2% of the students achieved Developed or Exemplary rating for outcome 1 thus providing strong support for the best practices put in place.

460/Fa 18: 94.3% of all students achieved Developed or Exemplary status on evaluation. Goal met.

455/Fa 18: 71.4% of all students achieved Developed or Exemplary status on evaluation. Goal met.

Overall: The program goal was partially met for this student outcome.

Outcome 2 (430, 453, 455)

430/Fall 18: 100%of all students achieved Developed or Exemplary status on evaluation. Goal met.
453/Fall 18: 74.9% of all students achieved Developed or Exemplary status on evaluation. Goal met.

455/Fa18: 83.3% of all students achieved Developed or Exemplary status on evaluation. Goal met.

Overall: The program goal was met for this student outcome.

Outcome 3 (351, 450, 460)

351/Fall 18: 71.7% of all students achieved Developed or Exemplary status on evaluation. Goal met.

450/Fall 18: 76% of all students achieved Developed or Exemplary status on evaluation. Goal met.

Note: It was observed that instructional improvements put in place after the last assessment and evaluation for this course worked really well.

460/Fall 18: 82.8% of all students achieved Developed or Exemplary status on evaluation. Goal met.

Overall: The program goal was met for this student outcome.

Outcome 5 (340, 450)

340/Spring18: 62% of all students achieved Developed or Exemplary status on evaluation. Goal not met.

Action: It was decided that instructors would solve more complex problems on the whiteboard in the class in a future offering and also provide more focused homework to strengthen students learning and performance on this outcome. Goal met.

Update: **340/Spring19:** Evaluation of assessed data for CMPS 340 for Spring 2019 showed that 86% of the students achieved Developed or Exemplary rating for outcome 5. Goal met.

450/Fall 18: 87% achieved Developed or Exemplary (final exam). Goal met.

Overall: The program goal was met for this student outcome.

A.5. How the results are documented and maintained

The results are documented, stored, and maintained in the Live Text system of UL Lafayette. Paper copies are kept in appropriate binders and stored in a dedicated office, room 229 of Oliver Hall, called the *Accreditation Room*. Course binders are also stored in the Program's Conference Room in 222 Oliver Hall for easy access to faculty members during faculty meetings for discussions and deliberations on assessment, evaluation, and continuous improvements.

B. Continuous Improvement

In addition to inputs from courses, alumni, employers and advisory board, the Program also uses the inputs from the following to improve its instruction and support and to make changes to its curriculum.

- Course-dependency based feedback (a course and its prerequisite course informing each other periodically)
- Peer evaluation of instruction by faculty
- Student graduation data
- ACM/IEEE curriculum guidelines
- Changes taking place in the field of computing
- ABET's guidelines from workshops such as IDEAL
- The state of Louisiana and UL Lafayette's requirements

The following changes in the curriculum have taken place based on such inputs.

- Instruction of CMPS 453 (Software Methodology) included principles of ethical computing included in the joint IEEE-ACM code of ethics, in order to better educate students in the area and also to invigorate assessment of the relevant learning outcome. This has been in place since Fall 2016.

- Instructional changes in CMPS 351 triggered by students' evaluation of instruction, exit survey and interviews of graduating students, and peer evaluations. Starting Fall 2017, a full-time instructor with a Ph.D. has been hired to teach CMPS 351. Both the quality of instruction and students feedback has greatly improved since the new instructor has begun teaching the course. Additionally, new contents have been added in CMPS 351 to better prepare students for CMPS 455 for which CMPS 351 is a prerequisite.

- Course changes in CMPS 453, CMPS 455, and CMPS 460 based on inputs from the accrediting agency (ABET). Starting Fall 2019, CMPS 453 and CMPS 455 will cover aspects of secure computing and protection.

- New course CMPS 432 on Distributed and Cloud Computing is added starting Fall 2019. This change was recommended by the program curriculum committee and it is in keeping with the changes in the field of computing. All students beginning their computer science program effective Fall 2019 will be required to take CMPS 432 in order to graduate.

- New course CMPS 413 on Computer Communications and Networks is added starting Fall 2019. This course has been offered as a special topics course (CMPS 499) for over 5 years and was recommended by the program curriculum committee to become a core course. This is also in keeping with ABET requirements. All students beginning their computer science program effective Fall 2019 will be required to take CMPS 432 in order to graduate.

- New course on algorithms, CMPS 340, added based on the recommendation of ABET. CMPS 340 is a core course for CMPS students. It has been taught and assessed every semester since Fall 2015.

- Starting Fall 2019, CMPS 490 (Senior Project) will become a core course, in compliance with the changes in student outcomes suggested by ABET.

- STAT 454 was added back as a core course in Fall 2013, in compliance with the ABET's suggestion.

- STAT 454, starting Fall 2019, will become a concentration elective for all concentrations. The program still meets the ABET's updated curriculum requirements and the curriculum committee recommended to make it an optional course starting Fall 2019.

- The prerequisite for CMPS 261 was changed from CMPS 260 and MATH 270 to CMPS 260 and MATH 110. MATH 270 is a prerequisite for both CMPS 340 and CMPS 341. This has resulted in increased retention of students in the program.

- With the usage of the new system, Banner, co-requisites have been removed for CMPS 150, CMPS 260, and CMPS 261 as it became impossible to effectively enforce co-requisites. The program does not use co-requisites anymore.

- Starting Fall 2017, ENGL 365 has been fulfilling the communications requirement for general education, and students are not required to take another communications course. This has added 3 hours to the free electives for students in the program.

- Cognitive Science concentration has been removed, effective Fall 2018. This was recommended by the program curriculum committee.

- Information Technology concentration is being replaced as Cloud Computing effective Fall 2019.

Throughout this accreditation cycle, the program has been meeting the ABET's curriculum requirement in all areas.

C. Additional Information

Assessment instruments and materials including rubrics, actual pieces of graded work, and evaluation data as well as all the other items mentioned in this report will be available for the visiting team. All data has been organized in properly labeled binders and are easy to peruse.

In response to the accrediting agency, ABET's new requirements, the Program is changing its Student Outcomes effective Fall 2019. The new outcomes and their course-mapping and plan of assessment and evaluation are shown in Tables 4, 5, and 6.

Table 4. Student Outcomes of the Program Effective Fall 2019

Outcome	Outcome
SO 1	Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
SO 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.
SO 3	Communicate effectively in a variety of professional contexts.
SO 4	Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.
SO 5	Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
SO 6	Be proficient in more than one programming language on more than one computing platform.
SO 7	Use computer science theory and software development principles to produce software solutions.

Table 5: Course to Outcome Mapping Effective Fall 2019

Course	SO 1	SO 2	SO 3	SO 4	SO 5	SO 6	SO 7
CMPS 310				X			
CMPS 340	X	X					
CMPS 341	X						
CMPS 351						X	
CMPS 430		X	X		X		
CMPS 450	X					X	
CMPS 453		X	X	X	X		X
CMPS 455	X	X			X	X	
CMPS 460	X	X	X		X	X	
CMPS 490			X				X

Table 6: Proposed Plan of Assessment Effective Spring 2020

Student Outcome (effective Spring 2020)	Sp & Fa 2020	Sp & Fa 2021	Sp & Fa 2022	Sp & Fa 2023	Sp & Fa 2024	Sp & Fa 2025
SO 1	**X**		**X**		**X**	
SO 2	**X**		**X**		**X**	
SO 3	**X**		**X**		**X**	
SO 4		**X**		**X**		**X**
SO 5	**X**		**X**		**X**	
SO 6		**X**		**X**		**X**
SO 7		**X**		**X**		**X**