Assessment Report for the 2022-23 Academic Year

Department of Mathematics and Computer Science

Chair: Chris Camfield, Professor of Mathematics

Part I: Program Assessment

In our Assessment Plan, 2022-23 is the year to examine Mathematics Learning Goal 3 and Computer Science Learning Goal 4. Most of our department was able to meet after the semester in one-on-one meetings to discuss relevant elements of this assessment report and contribute to its construction.

We collect data for every learning goal each year, not just those being assessed. This enables us to have multiple years of data on hand when assessing each goal. The data is currently stored in a spreadsheet with a tab for each department learning goal.

MLG3: Master at least one field of mathematics to a depth beyond that typical of a single advanced undergraduate course in the topic.

Courses: MATH 420 and MATH 450.

For direct assessment of this goal, instructors rated each student's performance in relation to this learning goal. Data has been collected for three of the past six semesters. These courses are offered in alternating years, so only one is offered each year. In the 2022-23 academic year, MATH 450 was cancelled due to low enrollment. For this year, we are going to include direct assessment for MATH 350 instead.

- A total of 18 students across 3 MATH courses were assessed using grades from relevant assignments and exams. The students were rated according to the following distribution:

Strong	Satisfactory	Needs Growth	Unsatisfactory	N/A
4	11	5	0	0

- The initial observation here is that 83% of students are performing at a satisfactory level or better. Overall, we are satisfied with that number.
- This learning goal has been difficult to assess because we only have two upper-level sequences. Enrollment is always a little low in the second course, but it has been even lower during and since the pandemic. This past year has shown an increase in enrollment at the sophomore level, so we are optimistic that we will see better numbers in these courses going forward.
- We will be discussing this along with our other learning goals as part of our program review in fall 2023.
- Our program review will also be looking at our curriculum to see if our current structure with the two course sequences in Algebra and Analysis are the best way to achieve this learning goal.

For indirect assessment of this goal, students should have been asked their opinion about achieving this goal. In 2022-23, the course didn't make. In 2021-22, our visiting professor did not include the question in the feedback survey. In 2020-21, the response rate was low with 3 out of 10 students responding. For what it's worth, all three of those students strongly agreed that they achieved this goal. As mentioned above, the structure and assessment of upper-level depth will be an area of focus in our program review.

CSLG4: Employ mathematical ideas in a computing context.

Courses: CSCI 151, 285, 322, 335, 365, 380, 382, MATH 240, MATH 340.

For direct assessment of this goal, instructors rated each student's performance in relation to this learning goal. Data has been collected for the last six semesters.

- A total of 284 students across 15 courses were assessed based on project and lab grades. The students were rated according to the following distribution:

Strong	Satisfactory	Needs Growth	Unsatisfactory	N/A
113	82	50	35	4

For indirect assessment of this goal, a question was included in the course feedback survey. Data has been collected for the last six semesters.

- When asked about their perception of meeting this goal, 143 students across 14 courses responded according to the following distribution:

Strongly Agree	Agree	Neither Agree nor Disagree	Disagree	Strongly Disagree
90	43	8	2	0

Analyzing the direct and indirect assessment data together, we find:

- The direct assessment data indicates that 69% of students are performing at a satisfactory level or better. This confirms our informal observations that a significant minority of our students struggle with advanced mathematics topics pertinent to computer science.
- From the indirect assessment data, we find that students responded favorably (Strongly Agree + Agree) at a rate of 93%, which is significantly higher than the faculty's opinion of their performance. The wording of the question could be a reason for this discrepancy. "This course has increased my ability to apply mathematical ideas that are relevant in a computing context." While faculty are assessing students on their performance, students are assessing if their skills increased, and an increase in skills could still result in a performance level of Needs Growth or Unsatisfactory. We will have conversations in the fall about the potential to better align our indirect questions with our learning goals.
- The courses CSCI 335 *Artificial Intelligence* and CSCI 322 *Computing Systems Organization* have significant mathematical content, but the classroom teachers did not include the indirect assessment question on the course feedback form. Both classroom teachers will do so when the courses are offered in the upcoming academic year.
- We also did not properly communicate the need for the assessment question to the adjunct faculty member who most recently taught CSCI 285 *Scientific Computing*. Again, we will ensure that we do so the next time the course is offered.
- From our senior exit interviews, students routinely find CSCI 382 *Algorithms*, the course which most exemplifies this learning goal, to be a difficult but valuable course in the computer science curriculum.

- We believe that the current pedagogical innovation of POGIL exercises in CSCI 382 *Algorithms* is improving both direct and indirect assessment scores from where they would be otherwise. While we do not have assessment data for CSCI 382 prior to the introduction of POGIL exercises, comments from students on the feedback forms testify strongly to its benefits to student learning.
- CSCI 382 has CSCI 151 and MATH 240 as prerequisites. We will next examine those two courses in detail.
- CSCI 151 *Data Structures* is a critical course in the Computer Science curriculum in general and with respect to CSLG4 in particular. It is where the concept of asymptotic analysis, explored in great depth in CSCI 382, is first introduced. The direct assessment data for this course has 44% of students exhibiting Strong performance on CSLG4, with an additional 32% Satisfactory. As a comparison, in that same course over the same time period with the same students, 53% exhibit Strong performance on CSLG1, with an additional 26% Satisfactory. This indicates that our coverage of CSLG4 in CSCI 151 is underperforming compared to our achievement of other learning goals.
- In addition, the fact that CSCI 151 is foundational to later courses suggests to us that improving overall performance for CSLG4 requires improvements to CSCI 151 in particular.
- Examining the assignments in CSCI 151, we find there is only one assignment in which students are assessed on their understanding of asymptotic analysis, and fewer than half of our students are showing mastery on that assignment. We identified two projects in which performing asymptotic analysis is a natural part of the assignment. Starting in Fall 2023, to receive full credit on those two projects, students will have to correctly employ asymptotic analysis.
- We believe the introduction of the Disco programming language in MATH 240 *Discrete Mathematics* is improving student outcomes. In Spring 2021, the direct assessment data shows 52% of students achieving Strong or Satisfactory performance. The Disco language was first introduced in Spring 2022. That semester, student performance fell considerably to 33% exhibiting Strong or Satisfactory performance. At that point, Disco was still under heavy development and determining appropriate pedagogical use of the language was a work in progress. In Spring 2023, the second iteration with Disco, performance skyrocketed to 71% of students exhibiting Strong or Satisfactory performance.
- While there may have been factors other than Disco that contributed to the outcomes described for MATH 240, in response to the course feedback question "Using Disco helped me learn the mathematical ideas in the course better", in Spring 2023 seven students Strongly Agreed, seven more students Agreed, and two students Disagreed. As student outcomes had improved as measured directly, and as students attributed to Disco a positive contribution to their learning, we conclude that its introduction has indeed improved the mathematical abilities of our students.
- We look forward to seeing improvements in outcomes for CSCI 382 as the above changes impact cohorts that continue to progress through the program.
- We did not examine the roles of the mathematical elective courses (CSCI 285 and 360, and MATH 340). This was in part because data from several of those sections was incomplete, and in part because the mathematical issues in the core required major courses were in greater need of immediate attention. Especially with the inclusion of MATH 270 as an additional option, assessment of those courses will be a priority the next time we assess CSLG4.

Part II: Actions Taken This Year

The following actions were taken this year to improve our programs.

- In response to our to-do list from last year:
    - MATH 270 Linear Algebra was added as an elective for the computer science major.
    - We completed detailed rubrics for learning goals CSLG4, CSLG8, and CSLG9.
- We had an unexpected staffing turnover this year when our visiting professor did not return after the fall semester. A lot of energy was devoted to covering courses that suddenly needed staffing. We conducted a successful tenure-track search, and we look forward to welcoming Dr. Allie Ray to our department in the fall. This tenure-track hire should provide some long-needed stability for the mathematics program.
- An external reviewer has been selected. We look forward to visiting with Dr. Russell Howell from Westmont College as he reviews both programs in the fall.
- We completed a renovation of 314 MC Reynolds from a computer lab to an active-learning classroom. Many of our classes employ group work, and this room is designed to facilitate that. We are excited to start using this room full time in the fall.
- Energy was devoted to program building and reestablishing some community that was lost during the pandemic. We distributed department stickers, and updated literature about the majors and minors. We also inducted students into Pi Mu Epsilon and had a math cookout for the first time in a few years. In partial response to our efforts, MATH 270 and MATH 290 will have near record enrollments next year.

Part III: To-Do List for 2023-24

- Rewrite mathematics learning goals using feedback from external program review.
- Evaluate options for the future of the mathematics capstone experience.
- Align assessment plan and the way we conduct senior exit surveys in order to make better use of them and to make sure we are collecting the data we need about our program.
- For courses that are required for majors outside our department, inquire about what assessment is needed for those programs.
    - The following courses are required for majors outside our department:
        - MATH 130: Economics, Chemistry, Biochemistry/Molecular Biology, Physics, Chemical Physics
        - MATH 140: Chemistry, Physics, Chemical Physics
        - MATH 215: Politics
        - MATH 260: Physics
        - CSCI 150: Study of the Mind
    - The following courses are elective options for majors outside our department:
        - MATH 215: Biology, Health Science, Study of the Mind, Environmental Studies, Sociology/Anthropology
        - CSCI 151, 270, 285, 335: Study of the Mind

- There is some overlap between students who enroll in the Mathematics and Computer Science programs. On several occasions, the MATH 240 prerequisite for CSCI 382 has been substituted by MATH 290. We will examine the possibility of making this substitution a formal curricular change.

Appendix A: Rubric for Direct Assessment of Computer Science Learning Goals

Learning Goal	STR	SAT	NG	UNSAT	Assessment Tool
CSLG1: Create and demonstrate software that correctly solves realistic problems with open-ended scope.	Course project demonstrates an innovative solution to a challenging, realistic problem.	Course project demonstrates a solution to a realistic problem.	Course project is operational but partially incorrect in its solutions to problems.	Course project is largely incorrect in its attempt to solve a realistic problem, or the problem is not realistic.	Large course project(s)
CSLG2: Use empirical methods to analyze computational systems and models.	Analysis of solution convincingly shows correctness and/or time and space performance.	Analysis of solution shows correctness and/or time and space performance.	Analysis of solution mostly shows correctness and/or time and space performance but has significant flaws.	Analysis of solution does not show correctness and/or time and space performance.	Course assignments with an analysis component
CSLG3: Employ multiple levels of algorithmic and data abstraction to manage system complexity.	Functions, classes, objects, and/or polymorphism manage project complexity in an innovative way.	Functions, classes, objects, and/or polymorphism manage project complexity in a competent way.	Functions, classes, objects, and/or polymorphism help manage project complexity, but are misused in some way.	Functions, classes, objects, and/or polymorphism fail to manage project complexity.	Complex course assignments involve use of functions, classes, objects, and/or polymorphism to manage complexity.
CSLG4: Employ mathematical ideas in a computing context.	Student insightfully employs pertinent mathematical ideas.	Student shows reasonable competence in employing pertinent mathematical ideas.	Student shows tentative understanding of pertinent mathematical ideas, but falls short in employing them.	Student shows no understanding of pertinent mathematical ideas.	Assignments employing asymptotic analysis; programming projects employing nontrivial mathematical modeling.
CSLG5: Create, implement, and evaluate software abstractions that model complex phenomena.	Complex phenomena are modeled in an innovative way.	Complex phenomena are modeled in a competent manner.	Modelling of complex phenomena captures some aspects but is inadequate in others.	Model fails to capture any essential elements of the modeled complex phenomenon.	Course assignments in which complex phenomena are modeled with data structures.
CSLG6: Create, apply, and understand the software abstractions that manage interactions with hardware.	Student code demonstrates a comprehensive understanding of the pertinent hardware.	Student code demonstrates a solid understanding of the pertinent hardware.	Student code demonstrates an understanding of some aspects of the hardware and a lack of understanding of other aspects.	Student code does not demonstrate any significant understanding of the pertinent hardware.	Course assignments in which student-authored code directly interacts with hardware.
CSLG7: As part of a team, develop robust software artifacts that successfully enable their users to achieve their goals.	Team develops software that enables users to achieve their goals, including implicit goals.	Team develops software that enables users to achieve their explicit goals.	Team develops software that enables users to achieve some but not all their explicit goals.	Team fails to develop software that enables users to achieve their explicit goals.	Large course project(s) conducted as part of a team
CSLG8: Employ written and oral communication in both technical and nontechnical settings.	Student communicates ideas clearly, succinctly, and professionally.	Student communicates ideas effectively, but with room to improve in clarity, succinctness, or professionalism.	Student communicates ideas somewhat competently, but with significant deficiencies.	Student does not communicate effectively.	Course assignments requiring writing and/or oral presentations.
CSLG9: Understand the social and ethical context of computing.	Student demonstrates a thorough understanding of the issues.	Student demonstrates a modest understanding of the issues, but falls short in some way.	Student demonstrates familiarity with the issues, but at most superficial understanding.	Student is unable to demonstrate any understanding of the issues.	Course assignments and/or discussions on social/ethical topics.

Appendix B: CSLG4 Assessment Data

					Direct Assessment						*Indirect Assessment*				
Term	**Course**	**Course Name**	**Instructor**	**Count**	**STR**	**SAT**	**NG**	**UNSAT**	**N/A**	**Resp**	**SA**	**A**	**N**	**D**	**SD**
2022 2S	CSCI 151	Data Structures	Ferrer	23	10	8	3	2	0	16	7	6	2	1	0
2022 2S	MATH 240	Discrete Mathematics	Yorgey	31	13	9	5	3	1	16	13	3	0	0	0
2022 1S	CSCI 151	Data Structures	Goadrich	12	4	5	2	1	0	11	6	3	2	0	0
2022 1S	CSCI 285	Scientific Computing	Wilson	0						0					
2022 1S	CSCI 382	Algorithms	Yorgey	15	8	1	3	3	0	7	5	1	1	0	0
2022 1S	MATH 340	Combinatorics	Sutherland	0						0					
2021 2S	CSCI 151	Data Structures	Goadrich	17	4	9	4	0	0	9	6	3	0	0	0
2021 2S	CSCI 322	Comp Systems Org	Yorgey	26	14	12	0	0	0	0					
2021 2S	CSCI 335	Artificial Intelligence	Ferrer	23	8	3	5	7	0	0					
2021 2S	CSCI 365	Functional Prog	Yorgey	8	2	4	1	1	0	4	1	3	0	0	0
2021 2S	MATH 240	Discrete Mathematics	Yorgey	18	3	3	5	7	0	9	3	3	2	1	0
2021 1S	CSCI 151	Data Structures	Goadrich	12	7	2	3	0	0	9	6	3	0	0	0
2021 1S	CSCI 382	Algorithms	Ferrer	16	5	2	7	2	0	14	11	3	0	0	0
2020 2S	CSCI 151	Data Structures	Goadrich	19	9	5	3	2	0	11	8	3	0	0	0
2020 2S	MATH 240	Discrete Mathematics	Yorgey	23	8	4	4	4	3	9	7	2	0	0	0
2020 1S	CSCI 151	Data Structures	Goadrich	15	9	2	4	0	0	11	5	6	0	0	0
2020 1S	CSCI 285	Scientific Computing	Goadrich	26	9	13	1	3	0	14	9	4	1	0	0
2020 1S	CSCI 382	Algorithms	Yorgey	0						3	3	0	0	0	0
2020 1S	MATH 340	Combinatorics	Sutherland	0						0					