2018 Assessment Document
Department of Mathematics and Computer Science
Hendrix College

Narrative of Strengths

Both of our departmental programs are characterized by the following strengths:

- Strong employment and graduate-school outcomes for our major graduates.
- Graduates reporting that our curricula have prepared them well for those outcomes.
- Our seniors are well-prepared for their capstone experiences, completing them at a rate of nearly 100%. The only completion failures in recent years came from one Mathematics senior who switched to a Psychology major, and another who switched to a Physics major.
- Departmental faculty show consistently strong classroom performance.
- Departmental faculty also show strong performance in their scholarship. This past year included multiple peer-reviewed publications as well as the publishing of a textbook that was many years in the making.
- Strong student-faculty interactions and relationships. This includes significant contributions by departmental faculty to Odyssey projects.
- Departmental faculty are well-known across campus for their contributions of service to the Hendrix community through their committee work.

Key areas of work for 2018-19:

Calculus Placement Study and the Precalculus Course: Professor Lars Seme has undertaken on behalf of our department over the last several years a statistical study in regards to our placement recommendations for Calculus. Our hypothesis was that the Math ACT is useful for predicting performance in Calculus. He confirmed in his study several years ago that a Math ACT below 27 is statistically associated with a drop in performance, and a MATH ACT below 24 is associated with a further drop in performance.

We decided that for students below a Math ACT below 27, we would examine their secondary school record to determine whether they were prepared for Calculus, with a recommendation to take MATH 120 *Functions and Models* if we deemed them unprepared. For students with a Math ACT below 24, we decided to recommend that they all take MATH 120 *Functions and Models*.

Over the last year, Professor Seme has further studied the degree to which MATH 120 is improving student outcomes in Calculus. He has found that MATH 120 yields a relatively small improvement in observed Calculus performance. This led to a discussion among the three current and future instructors of the course (Prof. Seme, Dr. Downes, and Dr. Camfield) about aspects of the course that could be altered to improve the situation. The collective observation was that students taking the course have a wide variety of backgrounds. Some of them are prepared for the course. Others are manifestly incompetent in applying and understanding the

most basic of algebraic operations. The tendency of our instructors is to try to meet the needs of all of these students simultaneously. What they concluded in the light of the data analyzed by Prof. Seme is that the attempt to meet the needs of all students results in meeting the needs of none of them.

What we have concluded as a department is that MATH 120 needs to be refocused on its principal task of preparation for Calculus. This refocusing will be accompanied by a retitling of the course to *Precalculus* and significant changes to its catalog language. The instructors will assume basic competence in algebra among the enrolled students and teach accordingly. This will enable them to focus in more depth on key precalculus concepts.

After these reforms are in place (in the 2019-2020 academic year), we intend to study once again the degree to which MATH 120 successfully prepares students for Calculus. This will be a combination of a statistical study along the lines Prof. Seme previously performed, in combination with observations from the Calculus instructors.

Another related issue is that we have realized that, although we have taught Calculus in common among multiple instructors for many decades, we have never formulated common learning goals for the Calculus course. This is an issue we plan to address in 2018-19. We plan to include those common learning goals in our 2019 report.

Capstone Projects: The Department of Mathematics and Computer Science hosts two majors with distinct curricula: a major in Mathematics and a major in Computer Science. Nevertheless, both majors share a common capstone concept of students completing an independent project. Over the years, it has become clear that our criteria and objectives for these projects are vague at best. To remedy this situation, we have agreed upon the following common learning goals for all of our departmental capstone projects:

1. Apply MATH/CSCI concepts and techniques from multiple classes or experiences to investigate a substantive problem.
2. Demonstrate “technical mastery” and “significance” of the topic in a clear and well-organized written thesis.
3. Effectively communicate project scope and achievements in a concise oral presentation.
    - Demonstration of competency & mastery in response to questions
4. Independence of work; self-awareness
    - Selecting an interesting research question to start with
    - Shows proper response to feedback from pertinent audience
5. Demonstrated consistent engagement with project topic/problem for the duration of the project period.
6. Demonstrate membership in disciplinary community
    - Follow appropriate presentational conventions.
    - Proper use of technical terminology

We have created a preliminary rubric based on these learning goals that we will employ in assessing our 2018-19 capstone projects. We will present the final form of this rubric and some preliminary assessment results in our 2019 report. We also need to relate these capstone learning goals back to the program learning goals for each of our majors.

Computer Science Major: Staffing constraints, including the newly introduced department chair course release and the impending retirement of Dr. Barel, have forced the department to reduce its total number of course offerings. We have greatly expanded our Computer Science elective offerings in recent years, thanks especially to the new ideas and contributions of Dr. Goadrich and Dr. Yorgey. But we have observed that we are offering enough such courses that our elective offerings are starting to compete with each other for enrollments. From this observation and our staffing pressure, we made the decision to reduce by one the number of CSCI courses we offer each year.

To achieve this goal, we carefully considered the role of each course in our curriculum. By examining what we are actually doing in practice, we significantly reformulated our program learning goals. Our new program learning goals are as follows.

Students completing a major in Computer Science at Hendrix College are expected to be able to do the following:

1. Create and demonstrate software that correctly solves realistic problems with open-ended scope.
2. Use empirical methods to analyze computational systems and models.
3. Employ multiple levels of algorithmic and data abstraction to manage system complexity.
4. Employ mathematical ideas in a computing context.
5. Create, implement, and evaluate software abstractions that model complex phenomena.
6. Create, apply, and understand the software abstractions that manage interactions with hardware.
7. As part of a team, develop robust software artifacts that successfully enable their users to achieve their goals.
8. Employ written and oral communication in both technical and nontechnical settings.
9. Understand the social and ethical context of computing.

Based on these reformulated goals, we are revising the CSCI major. Two currently required courses that are offered annually, CSCI 230 and CSCI 352, will become electives that are offered in alternate years. This achieves our needed course reduction. CSCI 230 *Computing Systems Organization* covers material relating to how modern computers are built, from the circuit level all the way up to the operating system level. Changes to the ACM Curriculum Guidelines have moved this material out of the Core - Tier 1 body of knowledge. By requiring students to choose between this course and CSCI 320 *Operating Systems & Concurrency*, students should study sufficient material in the low-level systems area to help satisfy the Tier 2 guidelines from ACM. This requirement is embodied by Learning Goal 6 above.

CSCI 352 covers important topics in commercial software development, focusing on the development of desktop software. We plan to change the focus of the course to mobile device development, an area in which we do not currently offer any coursework. This reflects the employment outcomes of our graduates, many of whom now work in the mobile development area. In particular, a major local employer, First Orion, who has hired many of our graduates, works exclusively in the mobile domain. As this course is now becoming an elective, we have recognized that two of our other courses, CSCI 340 *Database and Web Systems* and CSCI 370 *Interactive Game Development* share some similar learning goals. In particular, all three courses are focused on teaching students the necessary skills to be productive in delivering customer-facing software. Each student, then, will be required to take one of these three courses. This requirement is embodied by Learning Goals 1 and 7 above.

Customer-facing software faces particular constraints. In particular, there is a strong emphasis on using existing tools, libraries, algorithm implementations, and infrastructure to create solutions for customers. We concluded that our students also need to have experience directly implementing software that models complex phenomena, the type of software routinely used as libraries by commercial developers. This should enable every graduate to understand the logic behind the types of libraries they will be routinely using in practice. To this end, we are adding a major requirement for students to take a course in which they implement such software, embodied by Learning Goal 5. Current courses meeting this requirement are CSCI 335 *Artificial Intelligence* and CSCI 360 *Programming Languages*.

We will retain in the revised major our current requirement for a mathematically-oriented course, chosen among CSCI 365 *Functional Programming*, CSCI 285 *Scientific Computing*, CSCI 380 *Theory of Computation*, and MATH 340 *Combinatorics*. We will also retain the senior capstone course (CSCI 410), the Algorithms course (CSCI 382), our MATH requirements (Calculus and Discrete Mathematics), and our two-course introductory sequence (CSCI 150 and 151). Two open-ended CSCI electives will allow students to pick a particular area of focus.

In addition to the above major revisions, we will also need to rework our VSL learning goal mapping and curriculum mapping for CSCI that we performed in 2017, given our new learning goals. We plan to include these new mappings in our 2019 report.

Statistics: We have two major concerns to address with MATH 215 *Statistics*:

- What is the proper technology to use to support the course? R? Excel? SPSS?
- Can we achieve a common set of learning goals with the comparable courses taught by other Hendrix departments?

Textbook Pricing: Textbook prices for courses in our disciplines are generally (and correctly) considered absurd. We plan to investigate ways in which we might adopt books with a lower cost that still meet our goals. In regards to Calculus, this provides a further incentive to actually

formulate our common goals. We have already made progress in this area with regards to the Statistics course, adopting the relatively inexpensive ($48) 12th edition textbook by Chris Spatz.