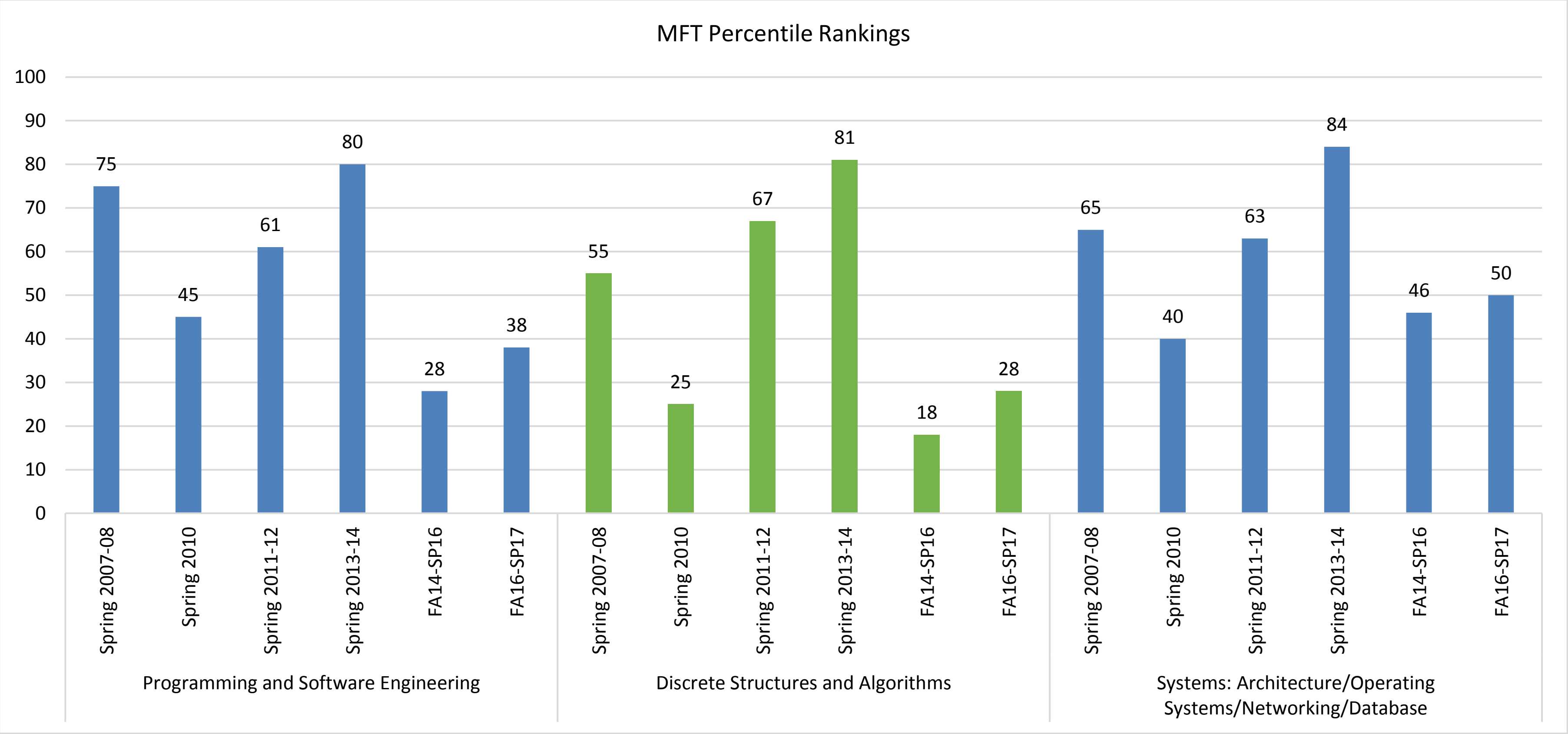
Program Assessment Plan Matrix

Academic Department/Program: Computer Science
Assessment Cycle: 2017-18
Date Submitted: 6/19/18

Contact Person: Robert Kasper (Chair)
Phone Contact: x3229
Email Contact: robert.kasper@mvnu.edu

Student Learning Outcomes	Assessment Methodology	Target	Summary of Major Findings	Actions Taken to Improve Student Learning	Timeframe
SLO 1: Exhibit a fundamental knowledge of the origins of computing, the capabilities and limitations of computing, and the use of computer technology as a tool for problem solving.	Method 1: Senior Major Field Test		Findings Method 1: See MFT results table at end of matrix.	See analysis and plans at end of matrix after results table.	
SLO 2: Demonstrate problem solving and communication skills fundamental to a liberal arts education and the computing profession.	Method 1: Senior Major Field Test		Findings Method 1: See MFT results table at end of matrix.	See analysis and plans at end of matrix after results table.	
SLO 3: Demonstrate an awareness of computer technology's impact on society, and engage in responsible actions when developing and using computing systems.	Method 1: Case Study		Findings Method 1: Students have completed written responses to case studies in two courses, but this is currently only a qualitative assessment. An instrument or rubric needs to be developed to provide a measurable assessment.	Developing rubric to provide a measurable assessment.	2018-19
SLO 4: Develop practical, usable skills for employment in computer related professions as well as a foundation in computer science concepts as a basis for professional growth and further academic studies.	Method 1: Senior Major Field Test		Findings Method 1: See MFT results table and summary at end of matrix.	We revised the CSNE major to replace a requirement to earn professional certifications with a requirement to complete four courses that cover background and skills that are necessary for students to prepare for professional certification exams. The new courses are closely aligned with professional certification requirements. The new curriculum includes detailed assessments of student mastery of knowledge and skills that were not available in the past. Curriculum resources have been chosen to make use of online simulations that students may complete in a directed study format that is sustainable even when only a few students need to take the courses in a particular year.	Changes to CSNE major approved Feb. 2018, to take effect in fall 2018.
	Method 2: Certification Exam Results for Computer Systems and Network Engineering major.		Findings Method 2: Most students have successfully completed only one of 3 expected certification exams at time of graduation. The professional certification exams are administered by external companies, so detailed results are only directly available to the students. We have offered only zero credit elective courses to help students prepare for professional certifications, but as this activity does not earn academic credit, it becomes a relatively low priority for students		
	Method 3: Alumni data (employment/enrollment survey)		Findings Method 3: Not available this year. More detailed responses about adequacy of preparation are obtained in survey administered for each 5 year self-study.		
SLO 5: Transition effectively into computing professions, exhibiting a Christian world view and a commitment to life-long learning and service.	Method 1: Alumni data (employment/enrollment survey)		Findings Method 1: 2016 EES survey data showed 100% of Computer Science graduates employed, but the results are not considered statistically significant due to low response rates.	Office of Institutional Effectiveness continues to explore avenues to increase response rate.	

Computer Science Major Field Test Results (B.S. and B.A. majors)



Summary and Analysis:
The Computer Science Major Field Test, administered by ETS, has been taken by all seniors completing the Computer Science majors (both B.S. and B.A.), and the department has collected the results since 2000. While total score was reported for all years, the sub-scores for different areas within Computer Science were not reported by ETS for 3 years when fewer than 5 students took the exam (2003, 2006, and 2011). The sub-scores are more useful than the total for assessment of different parts of our curriculum, and more recently, the Test Center grouped two years together to allow full reporting with a sufficient cohort of students from Fall 2014 – Spring 2016.

The MFT results over the past decade indicate an average slightly above the 50th percentile for two of three sub-scores. However, results have fluctuated significantly from year to year, presumably due to the small size of most cohorts, and the lack of meaningful incentive for some students to take the exam seriously as it is not connected to a regular required course. Therefore, we have not observed any clear trends that suggested particular changes to the curriculum. The lowest MFT scores are consistently in Discrete Structures and Algorithms, the category that measures more theoretical knowledge. This result is not surprising, as most of our students tend to be oriented toward more practical aspects of the field related to software engineering and systems. Another factor related to the lower scores for Discrete Structures and Algorithms is the different requirements for the B.S. and B.A. in Computer Science. The B.A. major does not require two courses with more theoretical content (Advanced Discrete Math and Survey and

Organization of Programming Languages). We accept that the B.A. students do not cover as much theoretical background as the B.S. students, most other feedback from former students and their employers suggests that they are still adequately prepared for general work in software development. Therefore, we have not interpreted these results as a mandate to change the requirement for the B.A. major, which allows more flexibility for students to pursue any other subject as a minor or second major. We have noted that the average MFT scores were below average for the last two cohorts, and will monitor this, as it would be a concern if a downward trend continues.

Spring 2007-08 Cohort			
Spring 2010 Cohort			
Spring 2011-12 Cohort			
Spring 2013-14 Cohort			
FA14-SP16 Cohort	**10 students**		
FA16-SP17 Cohort	**7 students**		
Assessment Indicators		**Mean Percent Correct**	**Percentile (National)**
Programming and Software Engineering		66	75
Programming and Software Engineering		57	45
Programming and Software Engineering		62	61
Programming and Software Engineering		59	80
Programming and Software Engineering (FA14-SP16)		48	28
Programming and Software Engineering (SP17)		51	38
Departmental Average (2007-2017)		**57.2**	**54.5**
Discrete Structures and Algorithms		37	55
Discrete Structures and Algorithms		29	25
Discrete Structures and Algorithms		39	67
Discrete Structures and Algorithms		48	81
Discrete Structures and Algorithms (FA14-SP16)		33	18
Discrete Structures and Algorithms (SP17)		38	28
Departmental Average (2007-2017)		**37.3**	**45.7**
Systems: Architecture/Operating Systems/Networking/Database		47	65
Systems: Architecture/Operating Systems/Networking/Database		44	40
Systems: Architecture/Operating Systems/Networking/Database		46	63
Systems: Architecture/Operating Systems/Networking/Database		49	84
Systems: Architecture/Operating Systems/Networking/Database (FA14-SP16)		42	46
Systems: Architecture/Operating Systems/Networking/Database (SP17)		43	50
Departmental Average (2007-2017)		**45.2**	**58.0**