Mathematics & Computer Science Department
524 West 59th Street
New York, NY 10019

Douglas E. Salane, Chairperson
Tel. (212) 237-8920
dsalane@jjay.cuny.edu

Computer Science & Information Security (BS)

Program Learning Assessment

2011-2015 Assessment Period

March 11, 2016

1. Introduction

The Mathematics & Computer Science Department has assessed learning outcomes in its computer major each semester since 2011. This report summarizes assessment results for the 2011-2015 period and describes actions taken to ensure students achieve both program and course learning outcomes. The report also describes how assessment has evolved during this period and provides rationale for the current way assessment of the major is done. In addition, the report indicates future modifications to the assessment process which will help us gauge student satisfaction with the program.

During the 2011-2015 period, the Department modified the assessment process to conform to best practices and to accommodate significant changes in the major. Beginning in 2013, the Department put in place a multilevel assessment plan. Thus, in line with widely accepted practices, the Department assesses courses at various points in the curriculum. The new strategy is to assess selected entry level, mid-level and capstone courses. This allows us to track cohorts of students as they progress through the major. We are able to observe particular points where students fail to achieve course and program learning outcomes. Multilevel assessment also allows us to take targeted corrective actions at points in the program where students begin to fail to achieve learning outcomes. The strategy has proven effective, and we plan to continue using it during the 2016-2020 assessment period.

Significant changes to assessment were warranted during the period because we made substantial changes to the computer major. In 2014, the Department revised the original Computer Information Systems major established in 1987. The revised major, which is now called the Computer Science & Information Security Major, focuses, as the name indicates, on computer and information security while providing a firm foundation in computer science and security principles. In addition, the name change reflects the nature of the program core, which, as in the previous major, provides firm grounding in computer science. As noted, in revising the major, the entry level and core courses, common in most computer science programs, remain in place. The key changes are the addition of several mid-level courses to support the new major, e.g., Computer Networking (CSCI 379) and Cryptography (CSCI 360), and several capstone courses that focus on computer and network security as well as digital forensics. The revised major prepares students for graduate study in computing, particularly in digital security, and also for direct entry into the computing profession, especially for positions that require a background in digital forensics or security.

Given the hierarchical structure of the major, multilevel assessment is critical. For the past three years, we have assessed entry, mid- level and capstone courses in computing each semester. Entry (or gateway) level courses include CSCI 271 and CSCI 272 which introduce students to programming and basic computing principles. Mid-level courses assessed include, CSCI 373 and CSCI 377, courses that require a solid background in programming and build advanced algorithmic problem solving skills. Capstone courses include the capstone experience courses, CSCI 400 and CSCI 401, as well as advanced computer and network security courses, CSCI 411 and 412. These courses require solid programming skills as well as firm grounding in the principles and methods learned in the mid-level courses, e.g., operating system fundamentals, essential algorithms and computer networking protocols. Moreover, the capstone courses assume the student at this point is in a position to acquire new advanced technical knowledge on her own.

2. Learning Outcomes:

Before moving on to the assessment results and actions taken, we list the program learning outcomes. These are essentially the same as in the older Computer Information Systems program but with some minor modifications to accommodate the revised major. Graduates of the Computer Science and Information Security major will be able to do the following:

> L1: Use and critically evaluate the variety of theoretical approaches that are relevant to the fields of Computer Science and Information Security.
>
> L2: Use and critically evaluate the variety of practical/hands-on/research approaches that are relevant to Computer Science and Information Security.
>
> L3: Analyze the quality of the programs in Computer Science and Information Security to enable graduates to be successful in highly competitive, high technology industries and graduate programs.
>
> L4: Communicate effectively through integrating theory, research and policy in written reports and presentations.
>
> L5: Understand the ethical considerations and statutory requirements computer professionals encounter as care takers of sensitive data and designers and developers of systems that can impact the well-being of individuals and organizations.

Every discipline has its principles, methods, practices and tools. A closer look at the above program learning outcomes shows that each outcome concerns one of these. Students who complete the program should be familiar with basic principles that underlie the discipline, know how to apply principles to develop solutions, know the basic methods of the field – in this field these are the essential algorithms that underlie much of computing − and be familiar with a range of best practices that have been established as well as current tools. In the field of computing, this is a tall order since computing is a vast, eclectic field that borrows from many disciplines. Nonetheless, professional organizations such as ACM[1] and IEEE[2] constantly work to

[1] ACM is the Association for Computing Machinery, www.acm.org

define the field, recognize what is essential, and provide guidance as to what course content and learning outcomes should be.

When the new major was put in place, the Department revised the L5 ethics learning outcome. The revised outcome reflects the special responsibility computing professionals bear in an age where many are charged with overseeing or processing sensitive personal information. In fact, a new ethics course is now required in the major and supports this program outcome.

Although the major has changed, the remaining program learning outcomes remain the same for the revised major. Corresponding course outcomes, however, have been modified to reflect the new security focus of the major as well as changes in the field and recommendations of professional organizations. Of course, new course learning outcomes were put in place for new capstone and upper level courses developed for the revised major.

4. Observations

In this section we summarize key findings revealed in the program learning assessment done during the past five years. The attached data summaries show the following:

i. At each level assessed (entry, mid-level or capstone course), the majority of students are meeting or exceeding the learning outcomes.

ii. At each level assessed (entry, mid-level or capstone), we observe a significant dichotomy in student achievement of learning outcomes. For example, in the assessment of mid-level courses CSCI 373 and 377, rigorous courses that require solid programming skills, a significant number of students are either approaching or not meeting one or more course learning outcomes.

iii. Data collected by instructors also show that students who do not achieve one outcome often are likely not to achieve other outcomes as well.

iv. Correspondingly, students who approach or achieve one learning outcome often are very likely to achieve other outcomes.

v. Students who only approach learning outcomes in lower level courses often fail to meet them in mid and higher level courses.

[2] IEEE is the Institute for Electronics and Electrical Engineers, www.computer.org

vi. In some capstone courses, certain students had difficulty achieving learning outcomes that required highly specific technical skills, for example, familiarity with practical OS tools such as Linux command line utilities.

vii. On the other hand, we a majority of students in most capstone courses assessed achieving the learning outcomes and developing significant analytical capabilities that make use of the principles, methods and practices they have learned.

4. Actions Taken or in Progress

This section describes some of the key actions taken or planned based on assessment data. In fact, many of the revisions in the major put in place in 2013 reflected the need to offer courses consistent with program learning outcomes in a computer major whose focus is security and digital forensics. For example, addition of the courses in computer networking, cryptography, advanced computer and network security, and the capstone experience courses support learning outcome L3, i.e., help prepare students for advanced study or entry into the computer profession in the area of cyber security. Based on assessment results, the following are some of the key actions taken or in progress.

i. A committee of computer science faculty members is now overseeing the regularization of programming exercises in the gateway level programming courses CSCI 271 and 272. In order to complete these courses, all students must be able to solve a standard set of programming exercises that ensure adequate programming skills for upper level courses.

ii. In order to ensure all students have the practical skills needed for capstone courses, a new course, Linux System Administration and Security has been developed. The course helps students not familiar with Linux operating systems be prepared for capstone level security courses. The course has been offered in fall 2015 and spring 2016 semesters and has been extremely popular with students.

iii. As noted, in order to support ethics program learning objective L5 a new course Ethics and Information Technology was developed by the Philosophy Department and is required of all majors.

iv. The Department now offers almost all major courses in labs to allow students to obtain practical skills. In rare cases where a course is not in a lab, laptops are delivered to the classroom or students bring their own.

v. A new freshman level course that introduces students to computing and programming is being developed by Professor Aftab Ahmad and Hunter Johnson. The course, which will be available to non-majors, will allow the general student population to gain exposure to computing and coding in a course appropriate for a general audience. Currently, many students take CSCI 271, the first course in the major, simply to gain some exposure to computing or satisfy the Scientific World Requirement. These students often are not prepared for the intricacies of a programming language such as C++, a language usually restricted to computer majors.

vi. Instructors now refer students to the Computer Science, Statistics and Data Resources Lab, which provides expanded support for programming, advanced statistics courses and upper level project work. Many students are taking advantage of the expanded services in the lab and appear to be doing much better in advanced courses. The group learning promoted in the lab appears to be particularly helpful.

vii. The Department has improved advisement by adding additional advisors and developing an enhanced web site helps that helps students be sure they take courses in proper sequence.

viii. The Department is devising a policy that will discourage students from progressing in the major if they do not achieve a grade of C or better in the CSCI 271/272 gateway courses. We are currently discussing with Undergraduate Studies the correct way to implement the policy.

The actions implemented certainly are having an impact. However, we will need additional semesters of assessment to quantify the results. Also, it will be a least two semesters before we can fully implement all the actions mentioned.

5. Concluding Remarks

Multilevel assessment is now a key component of the revised computer major. Each semester we assess gateway, mid-level and capstone courses. Since even at each level courses are sequential, we typically assess pairs of courses during the year in sequence, for example, introduction to computing I and II, CSCI 271/272, the data structures and algorithms courses, CSCI 373 and 377, and capstone I and II courses, CSCI 400 and 401. We expect to maintain this schedule in the coming assessment period.

We currently focus on student achievement of course learning outcomes. In the future, we plan to assess the student experience as well and gauge the effectiveness of the support services we offer such as tutoring and lab facilities. We also will assess auxiliary activities, for example, talks hosted by the Center for Cybercrime Studies and other organizations, which introduce students to key issues in computing, security and current research in the field.

Assessment has become a critical part of the Computer Science & Information Security major and now plays an important role in helping us improve the program and move it forward. In the future, we plan to update our assessment practices so they are consistent with assessment practices required for professionals accreditation for the program.

6. Acknowledgments

A number of faculty members contributed to the assessment effort during the 2011-2015 period. Their names appear below. We also thank Dr. Virginia Moreno, Director of Outcomes Assessment, for her guidance and advice in helping us to implement assessment and use it to improve the program.

Eman Abdu

Aftab Ahmad

Spiridon Bakiras

Sven Dietrich

Ping Ji

Hunter Johnson

Shaobai Kan

Bilal Khan

Jin Woo Kim

Evan Misshula

Michael Puls

Douglas Salane

Shamik Sengupta

Peter Shenkin

Computer Science and Information Security (BS)

Program Learning Assessment. Key findings and proposed actions

(2011-2015)

Program Learning Goals

1: Use and critically evaluate the variety of theoretical approaches that are relevant to Computer Information Systems.

2: Use and critically evaluate the variety of practical/hands-on/research approaches that are relevant to Computer Information Systems.

3: Analyze the quality of Computer Science and Information Systems.

4: Communicate effectively through integrating theory, research and policy in written reports and presentations.

5: Demonstrate proper business ethics and professionalism in the proposal or evaluation of solution to Computer Information Systems applications.

Outcomes Assessment

Program Learning Goal #	% Meet / Exceed[1]	Key Findings	Proposed Actions (Semester Implemented)	Was action effective?[4] Follow-up assessment	
				Sem.Year	% Meet / Exceed
2011					
***Assessment Context*[2] : MAT 324 (n=10), *Tool*[3] : Project**					
2	90	Most students did a good job using mathematical symbolism in writing their model. For example, the decision variables were clearly labeled. However, most students did not explain what the optinal solution was, of how management should implement it.	Spend more in class time explaining how to write a report more suitable for reading by upper level management. In particular, write out what the optimal solution is and how to implement it.		
2012					
***Assessment Context* : MAT 400 (Pilot), *Tool* : Project**					
1 2 3 4		This assessment exercise used the final project report. It serves as a starting point assessing the learning outcomes of the CIS major. Overall, students are meeting departmental expectations of learning objectives.	Augment the assessment using items from the final exam.		
***Assessment Context* : MAT 323 (n=16), *Tool* : Final Exam**					
2	50	Overall, students are meeting expectations. There is wide variation within the class in terms of preparedness, there are not many students in the middle in terms of achievement.	Spend more time during first week of class reviewing the more important mathematical concepts needed to be successful in MAT323.	2013	67 ↑

(1) Percent represents ratio of students who met or exceeded expectations. Where scores represent mean performance, the mean score and highest scale value are indicated (e.g., 3.3 out of 4). (2) Assessment context may relate to comprehensive program review, specific academic setting (e.g., course #, capstone, internship), class standing (e.g., seniors, transfers, alumni), post-graduation outcomes (e.g., placement, further education, employers ratings of employee skills), or indicators of learning progress. (3) Examples of tools include exams, portfolios, research projects, lab reports, papers, essays, surveys, licensure tests, performances, presentations. (4) Re-assessment of learning follows the implementation of actions to determine their effectivenes in improving learning outcomes.

Outcomes Assessment

Program Learning Goal #	% Meet / Exceed	Key Findings	Proposed Actions (Semester Implemented)	Was action effective? Follow-up assessment		
				Sem.Year	% Meet / Exceed	
2012 (cont.)						
***Assessment Context*: MAT 272 (n=14), *Tool*: Exam II, Final Exam**						
1	58.1	Overall, students are meeting expectations. However, some students have trouble in C++ programming, especially in the topics of function. On the project, some students had trouble in creating a perfect program that meets the exact needs the problem asks for.	Instructors need to repeatedly address challenging topics.	2013	68.1	↑
2	76.7			2013	77.8	↑
4	55.0			2013	62.5	↑
***Assessment Context*: MAT 379 (n=10), *Tool*: Exam, Project**						
1	70	Students met departmental expectations. Although the level of students' peformance of the two exams does not change significantly, students performed better in the final exam.	More emphasis on Network Security in our classes, ideally having two separate and consecutive courses, one on the fundamentals and the advance course on security and legal perspectives of computer networking.			
2	70					
3	70					
4	70					
***Assessment Context*: MAT 324 (n=14), *Tool*: Midterm, Final Exam**						
2	64.3	Overall, students successfully met departmental expectations for learning outcomes two, three and four. However, there are some problems with algebraic manipulation and effective communication using spreadsheets.	Enforce prerequisites for enrollment in MAT 323. Provide supplemental handouts on best practices with respect to effective communication with spreadsheets.			
3	50.0			2013	66.0	↑
4	54.5					
2013						
***Assessment Context*: MAT 272 (n=14), *Tool*: Exam II, Project II**						
1	68.1	Most students successfully passed Exam II. However, some students still have trouble in C++ programming, especially in the topics of function. Of the students assessed for Project II, 80% of them show at least 'meet expectations' work in every evaluation category.	Difficult topics need to be addressed repeatedly by the instructor.	2014	78.0	↑
2	77.8			2014	80.0	↑
4	62.5			2014	69.0	↑
***Assessment Context*: MAT 323 (n=15), *Tool*: Midterm Exam**						
2	67	Percent of students who met expectations on LO 2 increased from Spring 2012. Somes reasons for this increase could be that the instructor spent more time during the first week of class reviewing some mathematical topics important for MAT 323. Students also met departmental expectations on LO 3.	Collect more data on LO 2 to see if improvement is sustainable, or a statistical exception. Continue to collect data on LO 3 to determine why 'approaching expectations' is the most popular outcome (Sp 13). Design more homework problems to deepen understanding of the concept of shadow price (F13).			
3	47					

Outcomes Assessment

Program Learning Goal #	% Meet / Exceed	Key Findings	Proposed Actions (Semester Implemented)	Was action effective? Follow-up assessment		
				Sem.Year	% Meet / Exceed	
2013 (cont.)						
***Assessment Context*: MAT 324 (n=31), *Tool*: Midterm Exam**						
2 3	64.5 66.0	Significant improvement over last year midterm exam, where 50% of students met expectations on LO2. Some possible reasons for this difference is the differences in the problems used for assessment. Sharp contrast to last year scores on LO3 when 50% met expectations.				
***Assessment Context*: MAT 400 (n=11), *Tool*: Final Project Report**			LO3: Require attendance in one or more local technology-related professional meetups, or graduate colloquium presentations. LO2: Provide hands-on experience in source version control using GIT. LO4: Introduce additional UML documentation techniques. LO5: Introduce students to the management of open source management using sourceforge (F13).			
1 2 3 4 5	81.8 77.3 36.4 45.4 54.5	Overall, students are meeting departmental expectations of learning outcomes 1 to 5.				
2014						
***Assessment Context*: MAT 204 (n=16), *Tool*: In-Class Exams, Take Home Exam**						
1 2 3	56.0 46.0 47.0	Distribution of expectations for the three learning outcomes assessed is almost the same. A closer examination of the data shows the same students can usually be found in each category. Generally, students who do not meet expectations appear to lack fundamental academic skills.	Revise syllabus and choose new text that focuses primarily on relations and sets (LO1), combinations (LO2) and mathematical logic (LO3). Revised syllabus will eliminate applications in current text which do not directly support learning outcomes I, II and III (Sp 15).	2015	68.0	↑
***Assessment Context*: MAT 272 (n=31), *Tool*: Final Exams, Project**						
1 2 4	78.0 80.0 69.0	Almost 70% of students are meeting expectations for the three learning outcomes. In addition, 50% exceeded expectations. In other words, they were able to produce a perfect, and well thought out, solution of the problem. Results show that students do much better on project work than they do an in-class finals.	Additional exercises to reinforce students coding skills (F15).			

Outcomes Assessment

Program Learning Goal #	% Meet / Exceed	Key Findings	Proposed Actions (Semester Implemented)	Was action effective? Follow-up assessment	
				Sem.Year	% Meet / Exceed
2014 (cont.)					
***Assessment Context* : MAT 373 (n=37), *Tool* : Multiple Choice Test, Short Answer Questions**					
1	70	Most students have average or above-average understanding of the fundamentals of stacks, queues, and binary trees.	Upgrade the CSCI 271 and 272 courses to include additional coding exercises and projects to reinforce student skills in this area (F15).		
***Assessment Context* : MAT 404 (n=18), *Tool* : Multiple Choice, Coding Questions**					
3	50.0	Students who approached expectations had a difficult time with the coding questions. This indicates students need additional lab exercises to build Android programming skills. This brief assessment was conducted to get a better idea how students in the program will fare when challenged with advanced programming topics.	Revise course as a capstone experience to emphasize projects in computer security and forensics that utilize knowledge gleaned throughout the program (F14).		
2015					
***Assessment Context* : MAT 204 (n=35), *Tool* : Final Exam**					
1	68.0	Overall, students are meeting departmental expectations of learning outcome 1. They were able to show the basis step and state the induction hypothesis. However, they could not completely follow through with the algebra to successfully complete the last step.	Revise syllabus and choose new text (F15).		
***Assessment Context* : CSCI 271 (n=9), *Tool* : Final Exam, Project I**					
1 2 4	66.0 100.0 100.0	Results seem pretty good and this is not surprising since at this point in C++ computer programming students have done a good number of lab exercises and homework assignments. Important to mention the role of tutors willing to offer help outside the classroom when students experience challenges in their projects.			
***Assessment Context* : CSCI 272 (n=26), *Tool* : Midterm Exam, Project II**					
1 2	65.0 88.0	Performance better in the project, attributed to a lot of practice sessions. Lack of uniform experience in prerequisite could have played a role in students not meeting expectations in the first instrument.	Course before CSCI 271 should be initialized to be the service course for non-majors and can be taken by transfer students and computer science majors who would like to first focus on lower level skills with programming.		

Outcomes Assessment

Program Learning Goal #	% Meet / Exceed	Key Findings	Proposed Actions (Semester Implemented)	Was action effective? Follow-up assessment	
				Sem.Year	% Meet / Exceed
2015 (cont.)					
***Assessment Context* : CSCI 274 (n=23), *Tool* : Midterm Exam, Programming Assignment**					
1 2	83.0 70.0	Results point to the heterogeneous nature of Computer Architecture subject. It requires to learn computer 'math' and be able to work very close to the hardware components.	A computer science class that introduces to all computer topics, including the diference between higher and lower level languages would be helpful for students.		
***Assessment Context* : CSCI 373 (n=16), *Tool* : Multiple Choice Exam, Short Answer Questions**					
2	19.0	Large majority did not meet the course's expectations. All students had a difficult time with the coding of questions.	Enforce a restriction that does not allow students to pursue our major unless they get a grade of B or better in both CSCI 271 and 272. Assessment of 271 and 272 should be very strict and uniform, so that only students that are competent in C++ programming are given a grade of B or better.		
***Assessment Context* : CSCI 377 (n=23), *Tool* : Final Exam, Project I**					
1 2	70.0 70.0	Distribution of expectations for learning outcomes is not much different. Data shows that the same students can usually meet expectations in each category. Students who do not meet expectations appear to lack fundamental academic sklls.	More problem-solving sessions should be provided to enhance understanding of theory.		
***Assessment Context* : CSCI 379 (n=22), *Tool* : First and Final Exam**					
1 2 3	54.0 49.0 59.0	Results give an indication of the disparity in the skills and basic academic abilities of students.	Identify students weak in comprehension and writing skills to suggest they seek appropriate support services. Lower performing students need to have frequent assignments that are due every few days to ensure they stay on track. Establish measures that keep students informed of their progress in the program.		
***Assessment Context* : CSCI 400 (n=25), *Tool* : Labs, Project**					
3	52.0	Half the class is only approaching expectations. These students generally have a difficult time with the cybersecurity questions based on lack of skills.	Students need additional lab exercises to build Unix/Linux and networking skills. Department currently revising this course as to further integrate this course into the new cybersecurity curriculum. Continue to emphasize the need for students to work in teams. Develop additional learning outcomes and rubrics to assess this student capability.		