

William Paterson University
College of Science and Health - Department of Computer Science

Fall 2013 – Spring 2015 Assessment Cycle
Analysis of the Program’s Student Outcome Assessment Data

Program’s Student Outcome: S10: (revised)

Old version: Demonstrate competence in computer organization and architecture.

New Version: Demonstrate competence in computer organization.

ABET’s Related Student Outcomes: (c), (i).

Curriculum Committee Subgroup: Computer Organization, Systems, and Network

Members: John Najarian (Chair), Gilbert Ndjatou, Bogong Su

Date: February 22, 2016

Updated On: September 29, 2016

A. Analysis of the Assessment Data

For the assessment period Fall 2013 to Spring 2015, this student outcome was assessed in the courses CS2800 and CS3410.

In CS2800, we observe the lower tail-end category “Some Ability” over four semesters is: 0%, 0%, 10%, 10%.

Having 6% in the below adequate threshold is actually quite good. The next category above that one is consistently the maximal population case over all semesters (41%, 41%, 33%, 35%), though often tied as such.

In the “High Ability” tail, we observe 18%, 29%, 29%, 35% ; the Fall semester is lower than the Spring, with a ratio that is close. Could there be a complex (possibly hidden factor such as more class sessions) inducing this, a correlating factor?

In CS3410, we observe a recurrent theme, to quote: “nearly the entire semester was spent on Boolean algebra...”

This indicates that a size-able portion of students need better backgrounds. Some transfer students have Discrete Math (CS 2600) elsewhere, where Sum-of-Products circuit design is given cursory coverage (if at all!) Some discrete math texts have minimal coverage of Boolean Algebra. Secondly, notable are the high “Adequate Ability” statistics: 53%, 47%, 50%, 43%.. Since this is over the whole semester, these statistics add direct weight to the above recurrent theme.

B. Suggestions for Improvement

In CS2800, to address the Adequate Ability case's situation, more practice/hands-on sessions may provide improvement. These students may need more exercises for homework and hands-on in class to make them more capable.

Perhaps the “75-minute lecture on computer architecture” could be covered earlier in CS 2800 or distributed over a few sessions. Leaving it for the end does not work. An alternative is to add tutorial homework assignments which focus on this, reducing the lecture time requirement of this sub- subject. Students can do the readings outside class and there are excellent pedagogical support material on the web on computer architecture to accomplish this. Handouts are another method of augmenting lecture without using as much class time.

Also in CS 2800, adding some discussion on the effects of assembly instructions in terms of behavior in the context of the basic organization components would help introduce the hardware aspect in conjunction with the assembly language.

In CS 3410, considering to use a new textbook would definitely modify the general learning process along several parameters for the whole course. Actually, an alternative is to supplement the textbook with support material on computer organization. A few handouts or chapters on general organization should provide the general structures and behavior at the hardware level. Assigning it as take-home material (even a few introductory PowerPoint slides) and adding some basic homework problems would make it less burdensome on class time.

Of greater importance is the quoted “nearly the entire semester was spent on Boolean algebra, Boolean function minimization, combinational and sequential circuit design.” Greater coverage (and prerequisite coordination) in CS 2600 Discrete Structures may be needed, especially in the Disjunctive Normal Form material and basic circuit synthesis aspects of CS 2600. While it is mandatory material, increased course coordination can provide uniformity and enhancement in preparation. Transfers may need to be advised or directed to identify potential weakness in transferred Discrete Math coverage. Perhaps using a problem-set workbook (like the Schaums series) on this aspect of CS 3410, would provide shifting the predominance in in-class time to some active learning at home. Lecture time is at a premium, so we may need to be “more intrusive” on student home time. Also shifting towards a more recitation orientation would reduce the time pressure by focusing on problem-solving rather than exposition. Homework study groups may also help, preferably with a blend of at least one accomplished students in the each group, as a role model. Another solution is to have a student tutor to serve as that role model (say directly after or before the class session).

C. Improvement Implemented (in Fall 2016)

- In CS2800:
 1. Number of homework increased from 6 to 9.
 2. Tracing the execution of assembly language programs were emphasized both in homework and lab projects, which helped students to understand hardware aspect in conjunction with the assembly language.
 3. 75-minute lecture was given at the end of semester which covered some important concepts of computer architecture, such as RISC vs CISC, performance metrics and benchmarking, Instruction level parallelism and tread level parallelism.
- In CS3410:
 1. PPT files of computer organization contents were posted on BB
 2. Move some contents such as Boolean function complementing and simplification to CS260 “Discrete structure” was under consideration, which may provide few more class time for computer organization contents.
 3. Increase to 4 credits was under consideration.

D. List all the “performance level/frequency/percentage” tables and their sources

a. Faculty Course Assessment Report: CS2800, Fall 2013

Data Collected: Each student’s level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	0	0 %
Adequate Ability	7	41%
More than Adequate Ability	7	41 %
High Ability	3	18 %

Observations: 1) A 75-minute lecture on computer architecture was delivered in class but no test was given on the subject matter; 2) One student attended class once only and another transfer student has very weak background who even did not take college algebra, those two students must retake the course.

b. Faculty Course Assessment Report : CS2800, Spring 2014

Data Collected: Each student’s level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	0	0 %
Adequate Ability	7	41%
More than Adequate Ability	5	29 %
High Ability	5	29 %

Observations:

1) A 75-minute lecture on computer architecture was delivered in class but no test was given on the subject matter;
2) After improving contents and pedagogy average final exam grade increased.

c. Faculty Course Assessment Report: CS2800, Fall 2014

Data Collected: Each student's level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	2	10 %
Adequate Ability	7	33%
More than Adequate Ability	6	29 %
High Ability	6	29 %

Observations:

- 1) A 75-minute lecture on computer architecture was delivered in class but no test was given on the subject matter;
 - 2) Some students have low ability due to their weak background.
-

d. Faculty Course Assessment Report: CS2800, Spring 2015

Data Collected: Each student's level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	2	10 %
Adequate Ability	7	35%
More than Adequate Ability	4	20 %
High Ability	7	35 %

Observations:

- 1) A 75-minute lecture on computer architecture was unable to be delivered in class due to class cancellation
 - 2) Some students have low ability of assembly level programming.
-

e. Faculty Course Assessment Report: CS3410, Fall 2013

Data Collected: Each student's level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	0	0 %
Adequate Ability	8	53 %
More than Adequate Ability	4	27 %
High Ability	3	20 %

Observations: 1) As a three-credit course, nearly the entire semester was spent on Boolean algebra, Boolean function minimization, combinational and sequential circuit design. As a result, there was little time to learn computer organization; 2) Some students failed this course in previous semesters but worked very hard and passed this semester.

f. Faculty Course Assessment Report : CS3410, Spring 2014

Data Collected: Each student's level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	0	0 %
Adequate Ability	9	47 %
More than Adequate Ability	4	21 %
High Ability	5	26 %

Observations:

1) As a three-credit course, nearly the entire semester was spent on Boolean algebra, Boolean function minimization, combinational and sequential circuit design. As a result, there was little time to learn computer organization.

2) Two students gave up this semester with their own reasons.

g. Faculty Course Assessment Report: CS3410, Fall 2014

Data Collected: Each student's level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	1	6 %
Adequate Ability	6	33%
More than Adequate Ability	4	22%
High Ability	7	39%

Observations:

As a three-credit course, nearly the entire semester was spent on Boolean algebra, Boolean function minimization, combinational and sequential circuit design. As a result, there was little time to learn computer organization. I am considering to use a new textbook in F2015 semester.

h. Faculty Course Assessment Report: CS3410, Spring 2015

Data Collected: Each student's level of performance on homework assignments, two tests, final exam and a digital logic design project.

Method of Collection: Each student is given a score on the homework, tests, design project, and the final exam.

Performance Levels	Frequency	Percentage
Some Ability	0	0 %
Adequate Ability	10	43%
More than Adequate Ability	7	30%
High Ability	6	26%

Observations:

As a three-credit course, nearly the entire semester was spent on Boolean algebra, Boolean function minimization, combinational and sequential circuit design. As a result, there was little time to learn computer organization. I am considering to use a new textbook in F2015 semester.