AC 2008-1337: FRESHMEN ENGINEERING: THE INFLUENCE OF STUDENT FEEDBACK AND INVOLVEMENT ON A COURSE TEACHING MATLAB AND LABVIEW

David Illig, Clarkson University

David Illig is an Undergraduate Teaching Assistant in the Electrical and Computer Engineering Department.

John Hrynuk, Clarkson University

John Hrynuk is an Undergraduate Teaching Assistant in the Mechanical and Aeronautical Engineering Department.

Matthew Pennington, Clarkson University

Matthew Pennington is an Undergraduate Teaching Assistant in the Mechanical and Aeronautical Engineering Department.

John P. Dempsey, Clarkson University

John P. Dempsey is a Professor in the Civil and Environmental Engineering Department.

Freshmen Engineering: The Influence of Student Feedback and Involvement on a Course Teaching MATLAB and LabVIEW

Abstract

This paper describes the impact that undergraduate student feedback and involvement has had on Clarkson University's freshmen engineering course ES100: Introduction to Engineering Use of the Computer. ES100 provides students with an introduction to the MATLAB and LabVIEW programming languages, as well as introducing methods to solve engineering and science problems using MATLAB and LabVIEW. All undergraduate engineering majors are required to pass this class, which is taught by a team of faculty members from each of Clarkson's undergraduate engineering departments. In August 2006, Professor John Dempsey invited a group of sophomore engineering students who had just taken the class to attend a workshop on the course to share their experiences. This workshop resulted in the introduction of undergraduate teaching assistants (UTAs) in each ES100 classroom.

These UTAs provided, and continue to provide, input on revisions for many aspects of ES100, including course format, topics covered, and laboratory experiments. In particular, the UTAs were able to use their experiences in ES100 to assist in the redesign of course materials to be more consistent, uniform, and mainstream, assisting in Professor Dempsey's goal of making all engineering freshmen at Clarkson feel comfortable using MATLAB and LabVIEW. In this paper, the course revisions and their effects on the Spring 2007 offering of the course will be discussed. In Spring 2007, ES100 students were able to provide input on the course at the conclusion of each lecture and in a series of surveys. At the conclusion of each lecture, students were required to provide feedback on the course's Blackboard webpage regarding any difficulties encountered during that lecture or possible improvements to the lecture materials and exercises. Several short surveys were given during the semester, primarily to gauge feedback on the laboratory experiments. A more thorough survey was given at the end of the semester to evaluate how effective the course was in teaching students to use MATLAB and LabVIEW and to obtain suggestions on improvements to the course. The data from these three sources will be used to indicate the relative success of the revisions to lecture materials and laboratory experiments. These data have also shown further areas in which ES100 could be improved, and some of the adjustments implemented for the Spring 2008 offering of the course will be discussed.

1. Introduction and Course Format

The revisions to Clarkson's ES100 course were made possible by a CCLI A&I grant from NSF.[1] This proposal's intent was to promote a hands-on learning environment across the engineering curriculum, build self-confidence, promote teamwork and communication skills, and broaden the range of teaching styles to meet the needs of a diverse student population.[1] The Principal Investigators of this "Hands-On Learning in Engineering" project were Professors J. Dempsey, J. Carroll, J. Taylor, W. Wilcox, and A. Zander. The teaching methodology for the revised ES100 course adapted the 'integrated teaching and learning' paradigm pioneered and developed by Drs L.E. Carlson and J.F. Sullivan at the University of Colorado at Boulder.[2] The adaptation at

Clarkson is a combination of laboratory experience woven within an introductory computer course teaching both MATLAB and LabVIEW. Significantly, note that just recently (February, 2008), Drs. Sullivan and Carlson were awarded the prestigious 2008 Bernard M. Gordon Prize by the National Academy of Engineering for their ITL Program at Boulder.[3]

Clarkson began offering ES100 as an interdisciplinary course teaching MATLAB and LabVIEW during the Spring 2005 semester; prior to this point in time the course had been run separately by each of Clarkson's engineering departments, and each department had used its own software packages and focused on its own topics of interest. The reform of ES100 involved all four engineering departments and was actively supported by the administration of the Coulter School of Engineering. Two teams of faculty and graduate students were formed: one to work on the overall curriculum issues and one to develop multi-disciplinary laboratory experiments. The lecture portion of the course emphasized the use of MATLAB software to model and analyze simple systems. All sections were coordinated and involved instruction in a computer classroom with emphasis on hands-on exercises. Another of the desired changes to the course was to incorporate laboratory experiments that could be performed by the students to further increase their interest in engineering and to generate real-world data sets for processing, analysis and reporting. It was decided that the computer-based data acquisition system be controlled using LabVIEW software in order to provide the students with a foundation for higher-level classes where MATLAB and LabVIEW are used. During Spring 2005 and Spring 2006, considerable effort was expended on the design of suitable hands-on PowerPoint lectures for BlackBoard and on a novel fuel cell experiment.[4, 5]

Although these reforms showed improvements over the previously utilized course format, Professor Dempsey and PhD student John Bean (also Associate Professor, Paul Smiths College) found that the course was not running smoothly, as evidenced by negative feedback from students who had completed the course. In August 2006, Professor John Dempsey invited a group of sophomore engineering students to a workshop to share their experiences in Clarkson University's ES100 Spring 2006 course. The issues discussed at this workshop were the level of the materials presented in the course, the level of the laboratory experiments performed in the Spring 2006 course offering, and difficulty in getting help outside of class. The majority of the students present at this workshop agreed to become undergraduate teaching assistants (UTAs) for ES100, allowing them not only to continue modifying course materials but also to create an infrastructure for students to get more help outside of the classroom.[6]

ES100 is a required freshman engineering course, with no option to test out of the course and no "advanced" course for those students with previous computer programming experience. The course is offered each spring semester. There were six Instructors, eight UTAs, and 323 students enrolled in the course for the Spring 2007 semester. Two Instructors were affiliated with Clarkson's Civil and Environmental Engineering (CEE) department, two with the Mechanical and Aeronautical Engineering (MAE) department, and one Instructor each from the Chemical and Biomolecular Engineering (CBE) and Electrical and Computer Engineering (ECE) departments. Of the UTAs, one was in the CEE department, four were from the MAE department, and three were from the ECE department. Students enrolled in the course were enrolled in a heterogeneous manner, such that there was no correlation between the department of their major and the department of their Instructor or UTA. In Spring 2007, the course had 12 lecture sections. Lecture sections met twice a week for 75 minutes at a time in a computer lecture hall, where each student was provided with a computer. Each lecture began with a quiz on the previous lecture's material and ended with a "short answer" style survey, called the "Muddiest Point," to get student feedback on the lecture.[7] Seventeen PowerPoint lectures were presented on MATLAB, followed by three on LabVIEW. Homework assignments were submitted on the course's Blackboard webpage prior to the start of each lecture. Blackboard is a suite of software that allows Instructors to improve student engagement and increase assessment efficiency through the Internet.[8] Both an Instructor and UTA were present at all lecture meetings. In addition, there were four laboratory meetings during the semester, in which students used MATLAB and LabVIEW to perform and analyze the results of three experiments. Students worked in groups of four to complete their laboratory experiments and reports.

2. Lecture Modifications

Modifications to the ES100 lecture materials were motivated mainly by an attempt to have the course focus evenly on mainstream material, as opposed to more specialized knowledge, and thus allow course materials to be less intimidating to students who had no programming experience. As indicated by the title of the NSF-funded project that underpins the curriculum reform under discussion, there was also a keen desire to make the lectures more hands-on.[1] This was important, since it was felt that students who did not have practice from hands-on exercises with the programming concepts were unlikely to retain these new concepts for homework and quizzes.[9] The UTAs also felt that it was important to connect the PowerPoint lecture presentations to the textbook, so attempts were made to use examples directly from the textbook, and also to include references to pages in the textbook on specific PowerPoint slides.

In Spring 2006, when the UTAs were students in ES100, the course attempted to cover nearly every major concept in its textbook, which was Stephen J. Chapman's *MATLAB Programming for Engineers* (1st Edition).[10] This caused the course to rush through many of the programming fundamentals, effectively leaving behind any student who did not rapidly grasp those fundamentals. For the Spring 2007 offering of the course, the textbook was changed to Holly Moore's *MATLAB for Engineers* (1st edition), which was felt to be a more introductory textbook.[11] The effects of these changes can be seen by comparing the topic outline from the Spring 2006 syllabus to the Spring 2007 syllabus, as shown in Table 1. Most of the MATLAB material presented in lecture in Spring 2007 was covered in the first twelve meetings of the Spring 2006 offering of the course. Only one lecture was devoted to most separate concepts in Spring 2006, which illustrates why some students had difficulties in that year. The goal of the Spring 2007 offering of the course was to allocate more time for the presentation of each topic so that students would have more time to become comfortable with each concept before moving on to the next concept; students were also encouraged to study additional topics present in the textbook but not discussed in lecture. Recent experience at Clarkson University indicates that students in ES100 typically begin having difficulties in the course with the introduction of **if-statements** and **looping**. This caused the UTAs and Instructors to significantly extend the introduction of these materials. **If-statements** and the underlying logical concepts required to understand them were presented in two lectures and **looping** was given three lectures; this is in contrast to the one lecture given to each concept in the Spring 2006 offering of the course.

Table 1. Spring 2006 and Spring 2007 Topic Outlines

Class	Topic in Spring 2006	Topic in Spring 2007
1	Introduction to the Course and MATLAB	Introduction
2	Variables and Arrays	Variables and Arrays
3	Array Operations, Displaying Data	Matrices
4	Built-in Functions, Plotting, Debugging	Built-in Functions
5	Program Design, Flowcharts and Pseudo-Code	Importing/Exporting Data, M-files
6	The Logical Data Type	Plotting in 2-D
7	Branches: if and switch	More Plotting in 2-D, Curve Fitting
8	Plotting	Lab 1
9	Lab I Preview/ Exam 1	Symbolic Mathematics
10	Looping: for and while	User-Defined Functions
11	Logical Arrays	Lab 2
12	Functions	Exam 1
13	Lab 1 Review/ Lab 2 Preview/ Exam 2	User-Controlled Input and Output
14	Variable Passing and Optional Arguments	Logical Functions
15	Global and Local Variables	Selection Structures
16	Complex Data	For Loops
17	Lab 2 Review/ Lab 3 Preview	While Loops
18	String Functions Part I	Looping Practice
19	String Functions Part II	Function Practice
20	Additional 2D Plotting	Exam 2
21	Lab 3 Review/ Lab 4 Preview/Exam 3	Lab 3 lecture
22	3D Plotting	Lab 3 Part A
23	Cell Arrays Part I	Lab 3 Part B
24	Cell Arrays Part II	LabVIEW Introduction
25	Solving Linear Simultaneous Equations	LabVIEW Programming
26	Symbolic Math	LabVIEW Simulation
27	Review	Exam 3
28	Exam 4	Course Surveys

The format of each lecture was also modified from Spring 2006. In Spring 2006, text was presented on slides with few references to the textbook and few examples for students to work out on their own. These two issues were given particular attention in the revisions for the Spring 2007 offering of the course. References to the textbook pages covering certain concepts were placed next to the title on many PowerPoint slides. These allowed students to easily find more information on a concept that may have been confusing to them during the lecture. Some examples were also taken directly from the textbook and placed in the PowerPoint presentations. A major effort was put into making the PowerPoint lectures more interactive for the students.[1, 9, 12] Three modifications were made for this purpose. First, each PowerPoint lecture was reformatted to use a color coding scheme, indicating to students when important concepts or syntax were introduced and emphasizing warnings for common mistakes. Second, efforts were taken to ensure that many hands-on exercises were placed in each lecture.[9] These included both “yellow code” exercises where students typed and ran code given to them, as well as “Try This…” exercises where students were given a problem to solve. The distribution of these exercises throughout the course is shown in Table 2. Note that only lecture meetings are shown. Table 2 shows that although many exercises were provided in some lectures in Spring 2007, there were some lectures with fewer exercises. Third, each lecture was concluded with a “Your Turn” activity, in which a problem, typically open-ended, was given to the students to work on for 10-15 minutes. Students were required to complete these “Your Turn” activities in order to receive the password to the “Muddiest Point” survey at the end of class.

Table 2. Distribution of Hands-On Exercises in Spring 2007

Class	Topic in Spring 2007	Number of Hands-On Exercises
1	Introduction	5
2	Variables and Arrays	8
3	Matrices	12
4	Built-in Functions	12
5	Importing/Exporting Data, M-files	5
6	Plotting in 2-D	16
7	More Plotting in 2-D, Curve Fitting	11
9	Symbolic Mathematics	21
10	User-Defined Functions	7
13	User-Controlled Input and Output	9
14	Logical Functions	6
15	Selection Structures	4
16	For Loops	4
17	While Loops	5
18	Looping Practice	4
19	Function Practice	6
21	Lab 3 lecture	5
24	LabVIEW Introduction	2
25	LabVIEW Programming	6
26	LabVIEW Simulation	5

3. Laboratory Experiment Modifications

The laboratory experiments were found by the UTAs to be the most severe problem from the Spring 2006 offering of the course. In Spring 2006, four experiments were performed in which LabVIEW was used to provide virtual instruments (VIs) for interfacing with equipment such as a solar panel array, electrolyzer, fuel cell, water pump, and light bulbs. The later experiments in this sequence built from material presented in the first experiments. MATLAB was used by students, in groups of four, to analyze their results. However, students often found that they did not have the conceptual background to understand the tasks they were performing in these experiments, sometimes even leaving them unable to understand their results or write correctly about the experiment in their reports. These problems were further exacerbated by malfunctioning equipment, a lack of understanding by some on how experiments were to be run, and the lack of a clear grading rubric for the reports. As such, the UTAs advocated changing the structure of the experiments for the course, as well as providing more transparency in the grading process.

The main goal of the UTAs in modifying the laboratory experiments was to create experiments where the focus was on applying MATLAB skills to solve a problem, rather than experiments whose focus was the analysis of the results of the experiments.[12] There was also a strong effort to make the laboratory experiments align with the MATLAB skills that had been taught in the course up until the experiment date, since in Spring 2006 some skills were required for experiments that had not been taught.[13] Additionally, there was a desire to create experiments representing the interdepartmental aspect of ES100, as most of the previous experiments focused on applications of topics from electrical and chemical engineering. As a result, three new laboratory experiments were created for the Spring 2007 offering of the course. Lab 1 was a combination of the fuel cell and light bulb experiments from the Spring 2006 course offering, with particular focus on the use of MATLAB to create plots, solve equations, and import data from LabVIEW VIs. Lab 2 was an entirely new experiment with a civil engineering basis, in which students used a Garmin GPS unit to collect data from a route on the Clarkson University campus, with particular focus on the use of MATLAB to create 3-D plots and determine certain maximum and minimum values from the data. Lab 3 was another entirely new experiment, which focused on software engineering and program design for a model traffic light system. In the first week of lab 3, students were expected to write their own functions to model the behavior of the traffic light and sensors in the intersections. In the second week of lab 3, students were expected to put all of their functions together to create a working intersection. For all three laboratory experiments, students worked in groups, usually with a group size of four students. A group report was due after the completion of the experiment. The UTAs hoped that these new experiments would show students different areas in which MATLAB could be applied, and also avoid overpowering MATLAB concepts with the concepts behind the experiments.[12]

For the grading of laboratory reports, the UTAs developed a grading rubric for each laboratory experiment. These rubrics were not shown to students ahead of time, though they were provided to students when the students received their grades. All of the students in a given group received the same score for the report. Approximately half of the points for each report were allotted to the groups' MATLAB code for analyzing results, with the majority of the remaining points allocated to the written analysis of results. Points were also allocated for the introduction, conclusion, and formatting of the reports, although these were weighted significantly lower than the other parts of the report. No points were taken away for grammatical or spelling mistakes. When students received their grade, they also received a marked-up rubric showing where they lost points. Due to time constraints in the grading process, students did not receive personalized feedback on their reports.

4. Student Response to Course Modifications

Student response to course modifications was gauged by a series of surveys. At the conclusion of each lecture, all students were required to complete a short answer survey called the "Muddiest Point." These short answer surveys were used to gauge student understanding of the concepts introduced in that lecture, as well as to determine if students felt there were things that could be

Table 3. Spring 2007 “Muddiest Point” Responses

“Muddiest Point” Topic	Overall % of Responses
arrays and array functions	1.8
computer skills required for homework submission and in-class participation	1.8
curve fitting	1.8
difficulty of homework assignments, quizzes, and in-class exercises	8.0
fprintf() and formatted output	2.0
getting used to LabVIEW environment	0.7
getting used to MATLAB environment	1.4
if statements	0.4
importing/exporting data	1.7
input from the user	0.1
LabVIEW easier than MATLAB	1.7
lecture format	0.7
lecture length	2.6
logical data type and associated concepts	0.5
looping	3.2
mathematical functions	1.0
MATLAB easier than LabVIEW	0.5
miscellaneous built-in functions in MATLAB	0.2
miscellaneous built-in functions in LabVIEW	0.2
miscellaneous programming concepts	0.8
nonspecific negative feedback	2.0
operator syntax	1.9
other	1.2
plotting	2.2
positive feedback	57.7
random number generation	0.5
symbolic toolbox	1.1
user-defined functions	2.7

improved in the lecture or the Instructor’s presentation of lecture materials.[8] The “Muddiest Point” surveys also allowed student feedback to affect the learning environment in ES100.[8, 9, 13, 14] Overwhelmingly, students used the “Muddiest Point” for two types of comments. The first were comments stating that all concepts had been “clearly” presented or that the lecture was “good.” The second type of comment is typified by the statement: “I had trouble understanding *this concept* in today’s lecture.” Students almost never provided suggestions for improvement in lecture materials, though interestingly they would often provide negative feedback if an Instructor “rushed” the lecture or tried to introduce topics “out of order” according to the syllabus. In addition, some students used the “Muddiest Point” to indicate their preference for

MATLAB or LabVIEW once LabVIEW was introduced in lecture. Table 3 shows a summary of the percentages of "Muddiest Point" responses from all twelve lecture sections for all twenty lectures given in ES100 in Spring 2007. Students' responses have been sorted by topic. Students typically expressed some difficulty in understanding a new concept in the lecture in which it was presented, but it seems that most students were able to become comfortable enough with that concept that it did not confuse them in further lectures.

4.1 Student Response to Lecture Modifications

The data presented in Table 3 indicate that the revisions to the course materials for the Spring 2007 offering of ES100 were largely successful. Nearly 58% of all comments were positive feedback, indicating that over half of the students in any given lecture did not have difficulties with the material presented in that lecture. Many of the other topics shown in Table 3 received fewer than 1% of all responses, implying that these were issues that either affected a small number of students or only affected students for a very small number of lectures. Some surprising results occurred, perhaps due to the direction of the efforts of the UTAs and Instructors to revise the course. Although over 3% of all responses indicated that **looping** was a problem, fewer than 1% of responses indicated difficulties with **if-statements**. This indicates that the efforts to revise the two **if-statement** lectures were very successful, with most students understanding the material. It also indicates that the revisions to the three **looping** lectures were not as successful. In addition to **looping**, the following lecture topics received at least 1% of all "Muddiest Point" responses: **arrays and array functions**, **curve fitting**, **fprintf() and formatted output**, **importing/exporting data**, **operator syntax**, **plotting**, the **symbolic toolbox**, and **user-defined functions**. This indicates that these topics caused students significant difficulty at different points in the course. The variation of the number of "Muddiest Point" comments related to these topics over the length of the course is shown below in Figure 1.

Of these nine topics, difficulties with **curve fitting**, **importing/exporting data**, and the **symbolic toolbox** were isolated to the lectures in which those topics were introduced. This indicates that the three lectures covering those topics need significant revisions so these concepts are better understood when first presented. Interestingly, students continued to have problems with **arrays and array functions** throughout the entire course, with negative comments on this topic occurring in eleven of the twenty lectures. This indicates that this topic may not have received adequate attention when course materials were revised, despite being introduced in the first several lectures of the course. **User-defined functions** received about 20% of the "Muddiest Point" responses in the lecture in which they were introduced, and received 5% or more of the responses in the three lectures following that. A table summarizing the "Muddiest Point" responses to the lecture in which **user-defined functions** were introduced is shown below in Table 4. **User-defined functions** also received close to 15% of the "Muddiest Point" responses in the "Function Practice" lecture. A table summarizing the "Muddiest Point" responses to the "Function Practice" lecture is shown below in Table 5. **Looping** was a consistent challenge for students, appearing as a problem in seven out of eight lectures after its introduction.

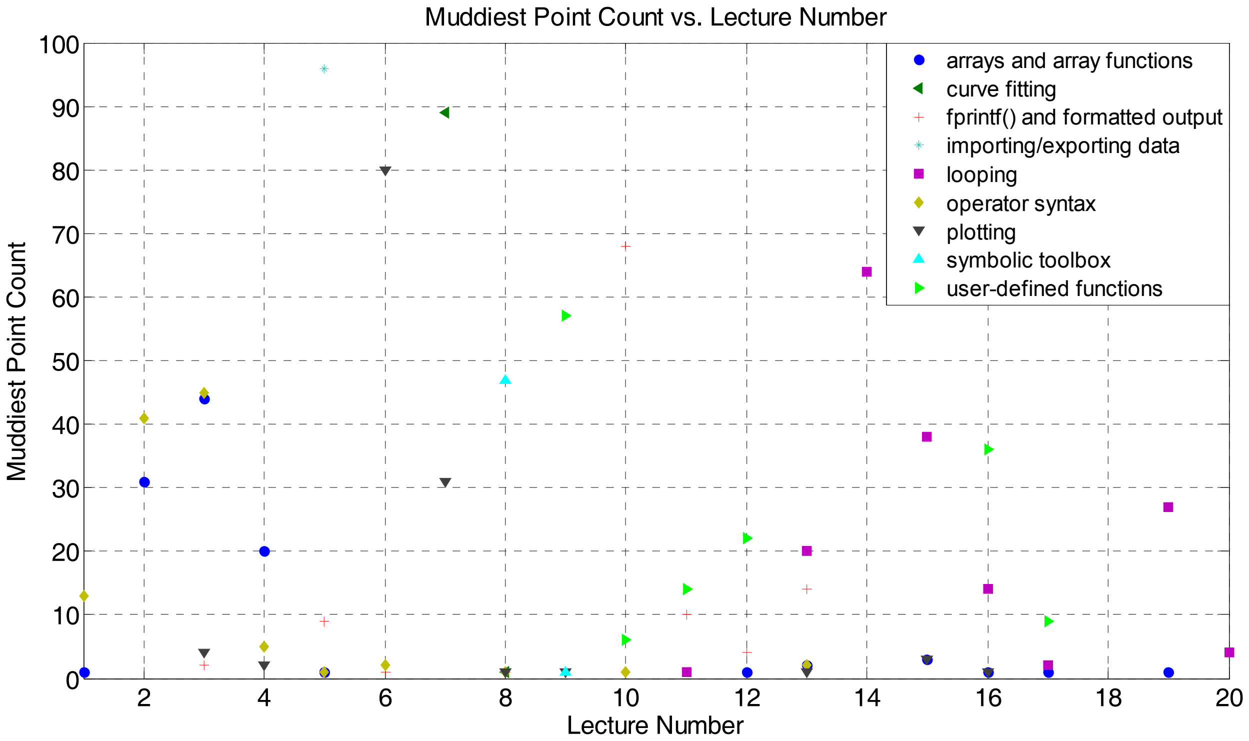

Figure 1. Variation of selected “Muddiest Point” over the length of the course

The other three topics which caused students significant problems were relatively unexpected. **Plotting** was believed by the UTAs and Instructors to be one of the easier topics to learn in MATLAB, and yet **plotting** was the third highest ranked problem area in Spring 2007. In addition to having over 25% and over 10% of the “Muddiest Point” responses in the two lectures introducing **plotting**, respectively, there continued to be difficulties with **plotting** in five of the fourteen lectures after the second **plotting** lecture. Tables 5 and 6, shown below, summarize the “Muddiest Point” responses to the two plotting lectures. **Operator syntax** was another topic which continued to be a problem throughout the course, perhaps indicating that too much of a focus was placed on conceptual understanding at times rather than on MATLAB syntax. Finally, the use of MATLAB’s **fprintf()** function to create formatted output gave 25% of students difficulty in the lecture in which it was introduced, and continued to give significant numbers of students trouble in the three lectures after that. The “Muddiest Point” responses for that lecture are summarized in Table 7, shown below. Strangely, some students also reported difficulty in understanding **fprintf() and formatted output** prior to the introduction of these topics in Lecture 10, indicating that one or more of the Instructors may have tried to introduce these topics before the agreed upon point in the course syllabus. This irregularity may help to explain some of the difficulty students experienced with **fprintf() and formatted output**, but there were still enough students reporting difficulty with these concepts after their formal introduction to indicate that the material was not well presented.

Table 4. Lecture 09: User-Defined Functions

“Muddiest Point” topic	Count	%
computer skills required for homework submission and in-class participation	6	2.3
difficulty of homework assignments, quizzes, and in-class exercises	16	6.1
getting used to MATLAB environment	6	2.3
lecture format	3	1.2
lecture length	7	2.7
mathematical functions	3	1.2
miscellaneous built-in functions in MATLAB	1	0.4
miscellaneous programming concepts	2	0.8
nonspecific negative feedback	5	1.9
other	4	1.5
plotting	1	0.4
positive feedback	149	57.1
symbolic toolbox	1	0.4
user-defined functions	57	21.8

Table 5. Lecture 16: Function Practice

“Muddiest Point” topic	Count	%
arrays and array functions	1	0.4
difficulty of homework assignments, quizzes, and in-class exercises	35	12.9
lecture format	3	1.1
lecture length	3	1.1
logical data type and associated concepts	1	0.4
looping	14	5.2
nonspecific negative feedback	7	2.6
other	1	0.4
plotting	1	0.4
positive feedback	165	60.9
random number generation	4	1.5
user-defined functions	36	13.3

Table 6. Lecture 6: Plotting in 2-D

"Muddiest Point" topic	Count	%
computer skills required for homework submission and in-class participation	1	0.4
difficulty of homework assignments, quizzes, and in-class exercises	19	6.6
fprintf() and formatted output	1	0.4
getting used to MATLAB environment	2	0.7
lecture format	2	0.7
lecture length	15	5.2
mathematical functions	7	2.4
miscellaneous programming concepts	4	1.4
nonspecific negative feedback	1	0.4
operator syntax	2	0.7
other	1	0.4
plotting	80	27.9
positive feedback	152	53.0

Table 7. Lecture 7: More Plotting in 2-D, Curve Fitting

"Muddiest Point" topic	Count	%
curve fitting	89	34.5
difficulty of homework assignments, quizzes, and in-class exercises	18	7.0
lecture format	1	0.4
lecture length	9	3.5
nonspecific negative feedback	7	2.7
other	1	0.4
plotting	31	12.0
positive feedback	102	39.5

Table 8. Lecture 10: User-Controlled Input and Output

"Muddiest Point" topic	Count	%
computer skills required for homework submission and in-class participation	6	2.1
difficulty of homework assignments, quizzes, and in-class exercises	10	3.6
fprintf() and formatted output	68	24.4
getting used to MATLAB environment	3	1.1
input from the user	5	1.8
lecture length	4	1.4
nonspecific negative feedback	13	4.6
operator syntax	1	0.4
other	7	2.5
positive feedback	157	56.1
user-defined functions	6	2.1

4.2 Student Response to Laboratory Experiment Modifications

Students were also presented with surveys at the end of each of the laboratory experiments. However, the main focus of these surveys was to allow students to provide feedback on group dynamics rather than to provide feedback on the experiments themselves. These surveys did contain a short answer question for students to give feedback on the experiments, but relatively few students took advantage of this. Even fewer provided constructive comments on the experiments. This may be because the surveys on Labs 1 and 2 were presented over two months after the completion of the experiments, thus students' reactions towards these two experiments may have been forgotten by the time of the surveys. Hence, it is difficult to determine representative student opinions on the experiments, and even more difficult to determine what percentage of students these opinions may represent. Student feedback did make clear that some students felt the experiments still had the same problems that the UTAs had set out to fix: a lack of conceptual background to perform the experiments, a disconnect to the MATLAB concepts taught in classes leading up to the experiments, and confusing grading policies. Despite efforts to minimize the conceptual background needed to understand the procedure of Lab 1, students still complained that they did not feel comfortable performing the experiment. Some students also did not see the connection between the experiment and MATLAB. These problems were not as severe with Lab 2, in which students recorded data on a GPS unit and analyzed that data. This may be because many of the MATLAB concepts used in Lab 2 were refinements of those required for Lab 1. A mix of opinions was provided in the feedback for Lab 3. Some students felt that Lab 3 was far too difficult for an introductory freshmen level course, whereas other students claimed that they enjoyed the programming challenges presented in the experiment. Finally, for all three experiments students complained about the grading system. They felt that different members in groups should receive different grades based on their group participation. Students also believed it unfair to base the entire grade on the report for the experiment, since the efforts required to perform the experiment and write the report could be significantly different. Some students even felt that having to write a report on the experiments was too much effort for an introductory freshman course.

4.3 Student Responses on End of Course Survey

Students were presented with a comprehensive survey on the last day of class, presented in addition to Clarkson University's required course evaluation survey. This was a 70 question survey, consisting primarily of Likert scale questions with some short answer questions. This survey was created to obtain information related more closely to the course than the information assessed by Clarkson University's required course evaluation survey.[13] Of these questions, 21 questions were meant to determine the proficiency and helpfulness of the Instructor and UTA in a given lecture section. There were also 13 multiple choice questions included to allow students to compare the three laboratory experiments to each other, but no questions were included for students to evaluate the experiments individually. These 34 questions thus provide little useful feedback on student opinions of the course, as they do not objectively analyze the course. Fortunately, many of the remaining questions do assess specific areas of the course, thus providing useful feedback.

Of the students enrolled in the course, 234 students took the survey, or about 72% of students enrolled in the course at the start of the semester. Some of the more useful questions from the survey were: how often students typed in examples, how helpful students felt that examples were, and how well students felt that homework assignments prepared them for quizzes. The majority of students surveyed, approximately 85%, said that they typed in lecture examples "often" or better. Given that a major focus of revisions for the Spring 2007 course offering was to make the course more hands-on, this is very helpful feedback. However, 70% of students surveyed said that the lecture examples were "helpful" or better. This indicates that although the majority of students were typing in examples, a significant number of students did not find the examples to be a meaningful reinforcement of the concepts introduced in the lecture. In addition, 31% of students found the "Your Turn" exercises to be "somewhat helpful" or better, which is in contrast to the much larger percentage of students who found lecture examples to be helpful. Also, 65% of students surveyed indicated that homework assignments were useful preparation for the lecture quizzes. Although revision of homework assignments was not a major focus of the revisions for the Spring 2007 offering of the course, this feedback indicates that more effort should have been put into correlating the homework assignments with the lecture quizzes.

Students also provided feedback regarding the course textbook, grading system, and quizzes and exams. Of the students surveyed, 36% felt that the textbook was "useful" or better for the course. Additionally, 29% of students claimed that they had "never used" their textbook. From these results, one can infer that lecture PowerPoint presentations were designed well enough that students simply did not feel it necessary to read additional material in the textbook.[15] On the other hand, one might infer from these results that not enough efforts were made to connect the textbook to the lecture materials.[12] Many students used the "short answer" question on grading to express their feeling that the grading used in ES100 was "harsh." Unfortunately, students did not provide information indicating whether they felt the grading for a particular item was "harsh" or if they were referring to the course as a whole. The average grade in ES100 in Spring 2007 was a B+, so it seems unlikely that students were dissatisfied by the grading system used in the course. Several students commented in the "short answer" on quizzes and exams that they felt there were "obscure references" to the textbook present in some questions. This may be more reflective of the number of students who did not use the textbook than an actual problem with quiz and exam questions.[15] Students also felt that some quiz and exam questions were "poorly worded," but they did not provide examples nor did they point out these problems at the time that the given quiz or exam was presented.

5. Overview of Changes for Spring 2008 Based on Student Feedback from Spring 2007

Several changes were made for the Spring 2008 offering of ES100 based on student feedback from Spring 2007. Budgetary constraints prevented the offering of another workshop session. Feedback from the surveys administered in Spring 2007 was used to determine areas that needed improvement for Spring 2008.[14] Student feedback indicated that the following lecture topics needed revisions: **arrays and array functions**, **curve fitting**, **fprintf() and formatted output**, **importing/exporting data**, **looping**, **operator syntax**, **plotting**, the **symbolic toolbox**, and **user-defined functions**. Student feedback also indicated that the laboratory experiments had the following problems: a lack of conceptual background to perform the experiments, a disconnect between the MATLAB concepts taught in classes leading up to the experiments and those required for the experiments themselves, and confusing grading policies. Finally, student feedback in the course survey indicated that a disconnect existed between lecture materials and the homework assignments, textbook, and "Your Turn" exercises.

As mentioned above, it appeared that student difficulties with **curve fitting**, **importing/exporting data**, and the **symbolic toolbox** were largely confined to the lectures in which those three topics were introduced. These three lectures were heavily modified in the hopes of making them more easily understood by students. Multiple side topics were removed from each of these lectures to allow for more examples on central concepts. Different approaches were taken for each of the other six topics that posed problems for students in Spring 2007. In the case of **arrays and array functions**, the majority of examples were modified to include a graphical representation of the array operation(s) being performed, in the hopes that this would assist students in understanding what was going on in MATLAB when they performed a given operation. In addition, more examples using arrays were included in the lectures succeeding the introduction of arrays, with the goal of reinforcing student understanding of arrays. For the introduction of **fprintf() and formatted output**, examples were revised so that one small feature was added to each example throughout the lecture. It is hoped that this incremental approach to introducing **fprintf() and formatted output**, combined with additional examples in subsequent lectures, will prevent students from having as many difficulties with these concepts. In the case of **looping**, the lectures introducing **for-loops** and **while-loops** were modified to be more fundamental. In addition, the "Looping Practice" lecture was replaced with a concept-based lecture to introduce some of the more advanced uses of **looping** structures which were removed from the introductory **looping** lectures. To cope with the difficulties that students had with **operator syntax**, examples were modified to introduce new operators such that mathematical or written descriptions of the operator were placed alongside MATLAB syntax. In the case of **plotting**, written descriptions were added to all examples, so that students will receive reinforcement of what the various **plotting** commands do. With **user-defined functions**, some of the more advanced topics were removed from the lecture so that more attention could be given to the fundamental structure of a **user-defined function**. Subsequent lectures were also modified so that at least one **user-defined function** example appeared in each lecture after their introduction, which unfortunately had not been done in Spring 2007. It is hoped that this approach will help students to retain information about **user-defined functions**.

In the case of the laboratory experiments, both large and small changes have been implemented to deal with student comments. The biggest complaints from students were centered on Lab 1, and many of these were the same issues that were brought to light by the UTAs in their August 2006 workshop. Thus, it was decided that this experiment be discontinued. A brand new experiment will be used for Lab 1, in which students will use a LabVIEW virtual instrument to simulate a kinematics experiment. The simulation will allow students to include such effects as friction and randomized measurement error. Students will then use MATLAB to analyze the results of their simulations and compare these results to the modeling equations given to them for the kinematics experiment. Stronger efforts have been made to ensure that all three laboratory experiments reflect the MATLAB concepts taught in the lectures leading up to the experiments. Lab 2, the GPS experiment, has been delayed so that **looping** procedures can be included in this experiment. Finally, for all three experiments the grading system has been revised so that points are awarded for four separate components: a pre-lab exercise, the experiment itself, the report on the experiment, and a post-lab survey. It is hoped that subdividing the points for the experiment in this way will create a grading system that is both fairer to students and more efficient for the graders.

Efforts have also been taken to adjust for the apparent disconnect between lecture materials and homework assignments, the textbook, and "Your Turn" exercises. In Spring 2007, 53% of problems assigned as MATLAB homework assignments came from Moore's textbook, and 100% of problems assigned as LabVIEW homework assignments came from a National Instruments manual.[16] This indicates that there was in fact a significant disconnect between MATLAB homework assignments and other lecture materials. For the Spring 2008 course offering, 93% of problems assigned as MATLAB homework assignments came from Moore's textbook, and 100% of problems assigned as LabVIEW homework assignments came from a National Instruments manual. This increased focus on problems from the textbook will improve the connection between homework assignments and lecture materials. Regarding the textbook itself, in Spring 2007 about 33% of slides in the lecture PowerPoint presentations referenced page numbers in the Moore textbook, giving evidence that there was indeed some lack of connection between the textbook and lecture materials. However, for Spring 2008, 100% of slides in the lecture PowerPoint presentations reference page numbers in the textbook. Additionally, the textbook *Essential MATLAB for Engineers and Scientists* (3rd edition) by Brian Hahn and Daniel Valentine has been added as a required textbook for the course, with all lecture materials including page references to this textbook as well.[17] Students who may not find Moore's textbook to be a useful reference for their learning style may be able to use the Hahn/Valentine book instead, and vice-versa. It is hoped that these modifications will make a stronger connection between the textbooks and lecture materials, as well as minimize the number of students who claim to have never used the textbook. The "Your Turn" exercises at the end of each lecture have been modified to be comprehensive for all of the major topics introduced in each lecture, whereas in Spring 2007 some "Your Turn" exercises only referred to the last major topic introduced in a given lecture. It is hoped that this will make the "Your Turn" exercises more useful for students, since the goal of these exercises is to help students practice materials learned in lecture.

6. Conclusions

Clarkson University's ES100: Introduction to Engineering Use of the Computer is a course that has evolved over time due to student and Instructor feedback. The creation of the UTA position in August 2006 has created a group of former students dedicated to improving ES100 and analyzing the large quantity of student feedback obtained from the course. These UTAs helped motivate many changes for the Spring 2007 offering of ES100. Lessons learned from student feedback to the changes implemented for Spring 2007 resulted in new modifications to make to ES100 for the Spring 2008 offering. ES100 will continue to evolve in response to student feedback, with an ultimate goal of helping all engineering freshmen to become comfortable with the fundamentals of MATLAB programming.

Acknowledgements

The authors wish to acknowledge the dedicated efforts of James Carroll (ECE), Jubum Kim (CEE), Daniel Valentine (MAE), and William Wilcox (CBE). The authors also wish to acknowledge PhD student John Bean for his contributions to the instruction and revision of ES100. In addition, the authors wish to acknowledge the contributions of the Clarkson undergraduate students who participated in the August 2006 workshop and/or served as UTAs: Kyle Ashley (CEE), Sarah French (CEE), Kayla Kuzmich (MAE), Nora O'Loughlin (MAE), Michelle Turk (ECE), and Greg Zenger (ECE). This CCLI A&I project is being supported in part by the National Science Foundation through the Department of Undergraduate Education Grant 0311075 and in part by both the Coulter School of Engineering and the President's Office at Clarkson University.

Bibliography

1. Dempsey, J. P., Carroll, J. J., Taylor, J., Wilcox, W. R., and Zander, A. K. "Hands-On Learning in Engineering." NSF CCLI A&I Grant DUE-0311075.
2. Carlson, L.E. and Sullivan J.F. "Hands-On engineering: Learning by Doing in the Integrated Teaching and Learning Program," *International Journal of Engineering Education* **15** (1999) 20-31.
3. http://www.nae.edu/nae/awardscom.nsf/JMAN-7A4L7N?OpenDocument
4. Bean, J., Carroll, J., Dempsey, J.P., Strong, A.H. and Wilcox, W.R. "Adding a Hands-On Laboratory Experience to the Freshman Engineering Programming Class at Clarkson University." Proceedings 2006 Annual Conference, Paper #2251, 9p.
5. Bean, J., Carroll, J., Dempsey, J.P., Valentine, D.T. and Wilcox, W.R. "Collaboration between Engineering Departments at Clarkson University for a Freshman-Level Engineering Programming Course Including an Experimental Lab Experience." CIEC 2007 Conference, 4p.
6. Martin, J. K. "Experiences combining technology, assessment, and feedback to improve student learning in mechanical engineering thermal science courses." *Proceedings – Frontiers in Education Conference* (2005) T2F-18-T2F-23.
7. Angelo, T. A. and Cross, K. P. *Classroom Assessment Techniques*. 2nd ed. San Francisco: Jossey-Bass, 1993. pp. 154-8.
8. *Blackboard Academic Suite*. Blackboard Inc., 2008.
9. Catalano, G. D., Catalano, K. "Transformation: From teacher-centered to student-centered engineering education." Journal of Engineering Education 88 (1999) 59-64.
10. Chapman, S. J. *Essentials of MATLAB Programming*. 1st ed. Ontario: Thompson Publishing, 2005.
11. Moore, H. *MATLAB for Engineers*. 1st ed. Upper Saddle River, New Jersey: Pearson Prentice Hall, 2006.
12. McWilliams, L. H., Silliman, S. E., Pieronek, C. "Modifications to a freshman engineering course based on student feedback." *ASEE Annual Conference Proceedings* (2004) 9977-9989.
13. Chang, J. L., Piket-May, M.J., and Avery, J. P. "Using Active Student Feedback in the Learning Environment." Journal of Engineering and Applied Science 2 (1998) 643-646.
14. Moskal, B. M. "Using student feedback to improve instruction in Engineering Calculus." *Frontiers in Education Conference Proceedings* 2 (2000) F3A-1-F3A-7.
15. LaBranche, D. F. "Scaffolding – Getting students engaged in learning." Journal of Professional Issues in Engineering Education and Practice 132 (2006) 96-98.
16. National Instruments. *LabVIEW Basics I*. Austin, Texas: National Instruments Corporation, 2000.
17. Hahn, B. and Valentine, D. *Essential MATLAB for Engineers and Scientists*. 3rd ed. Italy: B-H Elsevier, 2007.