

Using Mathcad Prime 2.0 to Restructure a Computer Applications Course

Kenneth P. Brannan¹, Kaitlin H. Marley², John A. Murden³

Abstract – Substantial restructuring of a computer applications course to use Mathcad Prime 2.0 instead of Mathcad 15; introduce an approach called the TOOLBOX; create a learning pyramid as a guide for sequencing course topics; restructure homework, assignments, and tests; expand the use of competitive learning activities; and incorporate a lifelong learning component appeared to improve attitudes toward Mathcad and improve motivation to solve problems in Mathcad. This observation is based on a comparison of surveys completed by students who took the restructured class in the fall of 2012 and by students who had taken the original course earlier. Survey responses of students who took the restructured course averaged 2.69 (mostly positive, based on Negative = 1, Neutral = 2, and Positive = 3) and responses of students who took the original course averaged 1.69 (slightly negative) on their attitudes towards solving problems in Mathcad. On the question, “How motivated did you feel in the course to solve problems using Mathcad?” those taking the restructured course averaged 4.06 out of 5 (with 5 representing very motivated) and those taking the original course averaged 2.14 out of 5. This paper discusses the details associated with restructuring the course and presents in more detail the results of the survey.

Keywords: Mathcad, programming, Toolbox, restructuring

INTRODUCTION

In the mid-1990s, the Department of Civil and Environmental Engineering Department at The Citadel selected Mathcad as its programming language for a computer applications class. Mathcad offers a comparable programming capability to previously used languages with additional advantages. These advantages include the ability to produce well-documented solutions, perform routine calculations, generate quality graphs with ease, and incorporate units as part of a computation. Since Mathcad’s introduction, instructors have worked to improve the learning environment to increase student enthusiasm and enhance understanding by team teaching, active learning, frequent tests and assignments, flowcharting, pseudocode, debugging features, and clickers. Recently, Mathcad released a new software version, Mathcad Prime 2.0. This version offers many new features that greatly simplify learning in the Mathcad environment. The instructors of the course chose to take advantage of the new version, and at the same time, worked to advance the overall course again. After extensive discussion with students, the instructors identified that students often struggled to distinguish between content that should be memorized and content that should be adapted to various situations. To address this difficulty, the instructors chose to introduce the idea of a TOOLBOX. This TOOLBOX served as a representation of exactly what must be memorized in order to be successful in the course. Additionally, the active learning component of the course was strengthened to build interest in mastering course goals. Finally, a lifetime learning module was added to promote continued learning

¹ The Citadel, Civil and Environmental Engineering Dept, 171 Moultrie St., Charleston, SC 29409, ken.brannan@citadel.edu

² The Citadel, Civil and Environmental Engineering Dept, 171 Moultrie St., Charleston, SC 29409, kmarley@citadel.edu

³ The Citadel, Civil and Environmental Engineering Dept, 171 Moultrie St., Charleston, SC 29409, john.murden@citadel.edu

beyond the engineering classroom. This paper discusses the integration of Mathcad Prime 2.0, the TOOLBOX, the other changes in the course, and the positive outcomes that resulted.

BACKGROUND

Programming in the engineering classroom has been implemented in a number of different ways in order to generate interest and enthusiasm, to help students better understand applications, to de-mystify the topic, and to facilitate student learning. Examples include using active learning techniques to promote the learning of syntax [1] or using a variety of different techniques other than the traditional lecture. Azemi and D'Imperio [2] describe a computer science course in which team-teaching, cooperative learning, the use of a hybrid delivery system including recording lectures, and assessment activities designed to help students in preparing for and participating in class are used. Students appreciated the approach which enhanced the performance of more motivated students. Another approach explores the use of a graphical language as an alternative to traditional programming languages. In one study [3] involving four traditional languages and two graphical languages, student perception was that more was learned with the traditional languages; however, the authors concluded that this may have resulted from the amount and type of exposure to the languages. A second study [4] also showed a slightly higher performance with a traditional language as compared to a graphical language, although both groups performed comparably using the same second language. Sun and Sun [5] described developing a core programming skillset using a modular programming approach. A hands-on approach with laboratory exercises [6] was used to enhance a computational methods course taught to first-year students. Students using a robotics kit (Parallax Boe-Bot) in the first-year curriculum at Louisiana Tech University [7] felt that their retention of subject matter and confidence were improved through the experience. Jeager, et al. [8] demonstrated that hands-on experiences can be successfully accomplished at low cost.

In addition to programming languages such as C/C++, MATLAB, Java, and BASIC, programming may be taught using a package such as Mathcad [9]. Mathcad not only provides typical programming constructs that can be used in the Mathcad worksheet, it also has a number of features that may be used to enhance learning in a variety of ways. These include graphing, units, symbolic manipulation, numerical solutions, and the capability to provide documented problem solutions or design computations. A variety of ways to use Mathcad in support of course learning objectives have been reported from the freshman through the senior years. For example, Swanborn, *et. al* [10] introduced Mathcad into a freshman experience course in which students designed and fabricated pumps. In this course, Mathcad was used to analyze a datapoint and Mathcad units were used in performing the analysis. Efimba and Smith [11] found Mathcad to be valuable in mechanics courses such as statics and mechanics of materials. The computer-based assignments along with teamwork and communication skills emphasized in these courses were designed to enhance student understanding of course concepts. An example of the use of Mathcad in a senior level course was reported by Hardin and Hodges [12], in which Mathcad was used for computer modeling in a materials course. Use of the computer was intended to strengthen student understanding of isotropic and orthotropic materials. In another senior level course, Mathcad modules were used to promote learning in a reinforced concrete design course [13]. Mathcad is a convenient and powerful tool that has been used for complex or repetitive calculations for analysis and design [14], [15]. Mathcad can also be valuable to instructors as well as students. For example, Pauley [16] uses it to develop different homework or exam problems in a Fluid Mechanics class.

A computer applications class in which students use Mathcad as an environment to learn programming as well as learn the basic features of Mathcad was launched in The Citadel's Department of Civil and Environmental Engineering in 1996. The package has served the department well over the years as a programming environment for the course (CIVL 210 is the current course designation) and for subsequent student use as they progress to upper level courses. The primary topic in CIVL 210 which the instructors have noted needs improvement is the use of looping and subscripted variables. In the fall of 2006, new debugging features associated with version 13 of Mathcad were incorporated into the course to help improve student performance in this area. While students' responses in a survey indicated that these features were valuable, the debugging features did not prove to be as helpful as teaching methods that were already being employed [17]. In addition, while the benefits could easily be seen, students did not appear to be enthusiastic about spending additional time to learn the syntax of the debugging features. In fall of 2008, a more interactive approach using clickers was implemented to help with teaching programming concepts. An assessment of student feedback [18] demonstrated that clickers helped students stay on task, maintain interest, and retain course material. In contrast to the study on debugging features, students ranked

2013 ASEE Southeast Section Conference

clickers higher than most of the teaching techniques that had been used previously. Clickers were again used in the 2009 fall semester, but for a variety of reasons, clicker use declined in the following years. It should be noted that while there was a high degree of student satisfaction with clicker use, it was not possible to determine in the 2008 study if a significant difference in retention of course material could be attributed to the use of clickers in the course. For these reasons, the course instructors began to consider changes that could be made during the 2012 fall semester that would positively impact student learning and enthusiasm in the course.

COURSE RESTRUCTURING

In March of 2012, PTC released Mathcad Prime 2.0. Mathcad Prime 2.0 included a number of significant changes from Mathcad 15, the version used in CIVL 210 the previous semester. Many of these changes were advantageous; therefore, the instructors of CIVL 210 chose to incorporate Mathcad Prime 2.0 into the overall course upgrade for the 2012 fall semester. The instructors reviewed and adjusted the course goals, analyzed the course progression, consulted current and past students, and discussed the course among faculty at length. As a result of these efforts, considerable adjustments were made to the course content and structure. Mathcad Prime 2.0 became the primary software of the course. CIVL 210 was also reorganized to account for proper progression of course content, and the TOOLBOX idea (discussed later in this section) was introduced. The course assignments were updated to supply more time to process new concepts while still providing frequent feedback. Active learning activities were increased, and lifelong learning was incorporated directly into the curriculum to build student confidence when working with unknown information.

Switch to Mathcad Prime 2.0

The switch to Mathcad Prime 2.0 offered many advantages. The most significant advantage was the upgrade of the user-interface of Mathcad Prime 2.0. Previous versions of Mathcad employed a drop-down menu user-interface whereas Mathcad Prime 2.0 employs a ribbon user-interface based on Microsoft® Fluent UI [19]. The ribbon user-interface of Mathcad Prime 2.0 simplifies learning of the Mathcad environment for users unfamiliar with the program. Students enrolling in CIVL 210 often have little exposure to Mathcad, and are using it for the first time in the course. For these students, the ribbon user-interface proved useful, and it allowed students to learn basic operations in Mathcad more quickly than students taking the course using previous versions of Mathcad.

Along with the ribbon user-interface, Mathcad Prime 2.0 included a number of other new features that enhanced the CIVL 210 course. Table 1 includes a summary of these changes and additions. The table also indicates whether the additions improved student learning primarily through upgrades to the user interface or through adjustments to programming methods. For example, with the introduction of if statements in previous versions of Mathcad, students were initially confused as to whether the condition or the instruction should be placed first. This confusion did not arise with Mathcad Prime 2.0 because the if statement was broken into two lines. Students easily remembered that the first line of the if statement required the condition input and the second line required the instruction input. The grid was a nice addition for two reasons. It was not necessary to spend much time on discussing alignment in early classes and students appreciated being able to print out well-organized problem solutions on paper that looked like engineering paper. The placeholder for graph units was a very significant addition. Using previous versions of Mathcad, it was not uncommon for students to forget to divide axis expressions by the correct units, or even worse to multiply instead of divide. Because Mathcad Prime 2.0 has a placeholder, this was never an issue during the fall semester of 2012. These changes and additions among the others documented in Table 1 all helped to improve the overall quality of the CIVL 210 course in the fall of 2012.

2013 ASEE Southeast Section Conference

Table 1 - Mathcad Prime 2.0 Additions Sorted by Order of Significance

Addition	Description	User Interface	Programming
Ribbon User-Interface	Drop-down menus of previous versions of Mathcad were replaced with a Ribbon-User Interface in Mathcad Prime 2.0.	x	
If Statement Format	In Mathcad Prime 2.0, the first line of an if statement is used for the condition. The second and following lines are used for instructions. In previous versions of Mathcad, the condition and the instruction were included in the same line if there was only one required instruction.		x
Two Vertical Bars Enclose Multi-Line Functions	In previous versions, vertical bars on the left-hand side of a multi-line function indicated the level of the function. In Mathcad Prime 2.0, there are bars on the right-hand side as well. The bars on the right are useful for editing.		x
Labels and Formatting Automatically Distinguish Variables, Units, Constants, and Functions	Mathcad Prime 2.0 automatically assigns a specific label and format to an entry based on its type (variable, unit, constant, or function).	x	
Grid	Mathcad Prime 2.0 features an optional grid in the background. All elements (text or math) are assigned to a corner of the grid, simplifying alignment.	x	
Placeholder for Graph Units	In previous versions, expressions in graph axes had to be divided by the units the user wished to display on the graph. The desired units may be input directly into a placeholder in Mathcad Prime 2.0.	x	
Evaluation Equals Sign Does Not Default to Assignment Equals Sign	In recent versions of Mathcad, an evaluation equals sign changed automatically to an assignment equals sign if the user had not previously defined the variable. In Mathcad Prime 2.0 an error is generated.	x	x
Additional Markers on Graphs	Mathcad Prime 2.0 allows greater than two markers per axis per graph.	x	
Automatic Closing Parenthesis	When a left parenthesis is entered, a right parenthesis is automatically generated.	x	
Square Symbol on Vector/Matrix Subscripts	A square bracket symbol appears on a Vector/Matrix subscript when the user selects a subscripted variable, differentiating it from a literal subscript.	x	x
Automatic Formatting of Range Variables	Mathcad Prime 2.0 provides placeholders to be filled in directly for a range variable as soon as the user enters the initial value for the range. This minimizes the number of mistakes entering range variables for beginning and experienced users.	x	

Reordering of Course Content

The goal of CIVL 210 is to show students how to use Mathcad as both an engineering tool and a programming tool. Historically, the most challenging component of the course for students has been writing original programs. In particular, students struggle to write programs that require nested loops and subscripted variables. This prompted the CIVL 210 instructors to develop a programming learning pyramid that visually illustrates which course topics must be mastered before other topics can be fully understood. As shown in Figure 1, understanding Nested Loops is the ultimate programming goal of the course, and many topics must be mastered prior to successfully using Nested Loops. Only course topics directly associated with programming are shown in Figure 1; topics such as professional and ethical considerations and lifelong learning are not included.

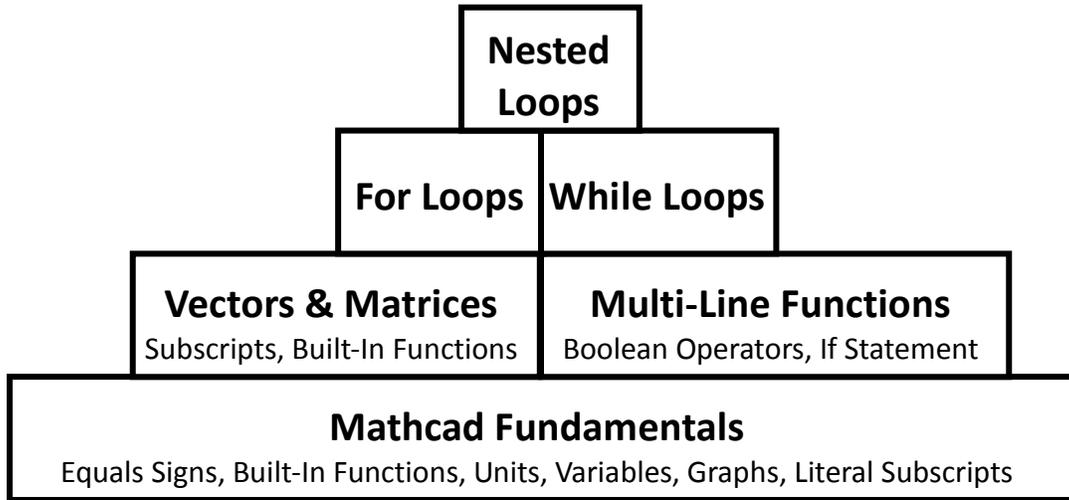


Figure 1 – CIVL 210 Programming Learning Pyramid

The learning pyramid in Figure 2 was compared to the order the topics were covered in the fall of 2011. Through this process, it was discovered that Vectors and Matrices were introduced after For Loops and While Loops. This progression of topics did not match the progression of the programming learning pyramid. Further, the instructors observed that students traditionally struggled to understand subscripted variables, and believed the course progression may have contributed to this problem. Therefore, the course content for the fall of 2012 was reorganized to properly follow the progression of the pyramid. Specifically, Vectors and Matrices were covered before For Loops and While Loops. Figure 3 illustrates the order of the course content for the fall of 2012 as it compares to the programming learning pyramid.

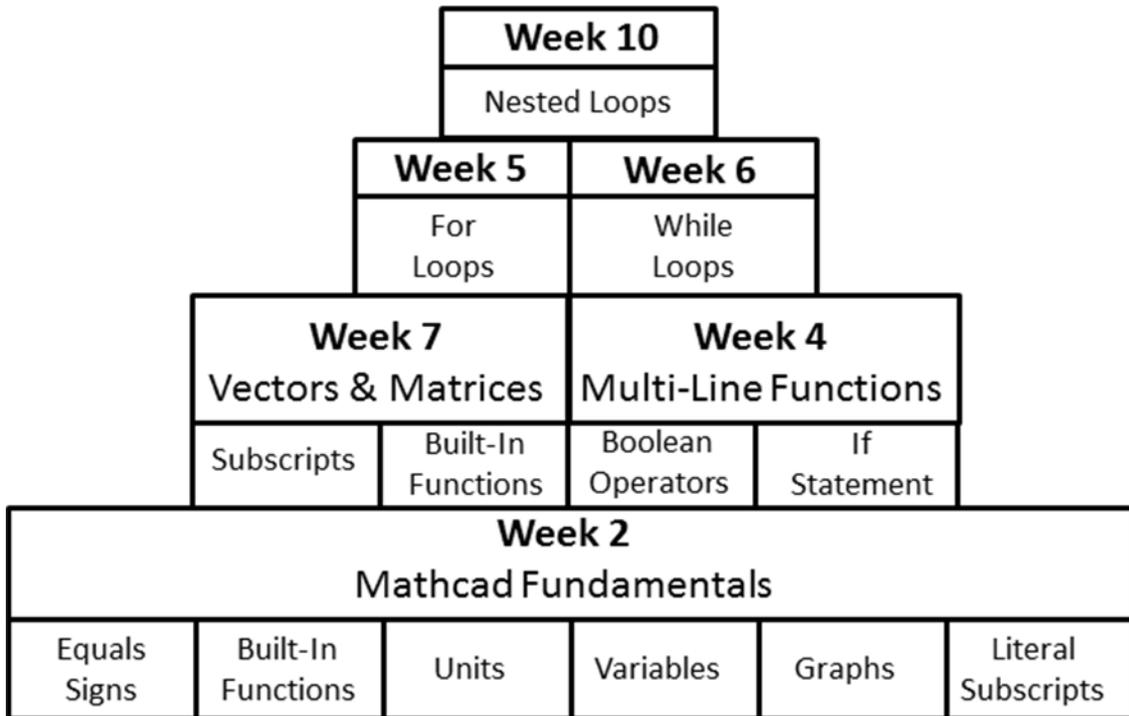


Figure 2 – CIVL 210 Programming Learning Pyramid with Fall 2011 Course Sequence. Note: Weeks 3, 8, and 9 are omitted because the course content those weeks was not relevant to programming.

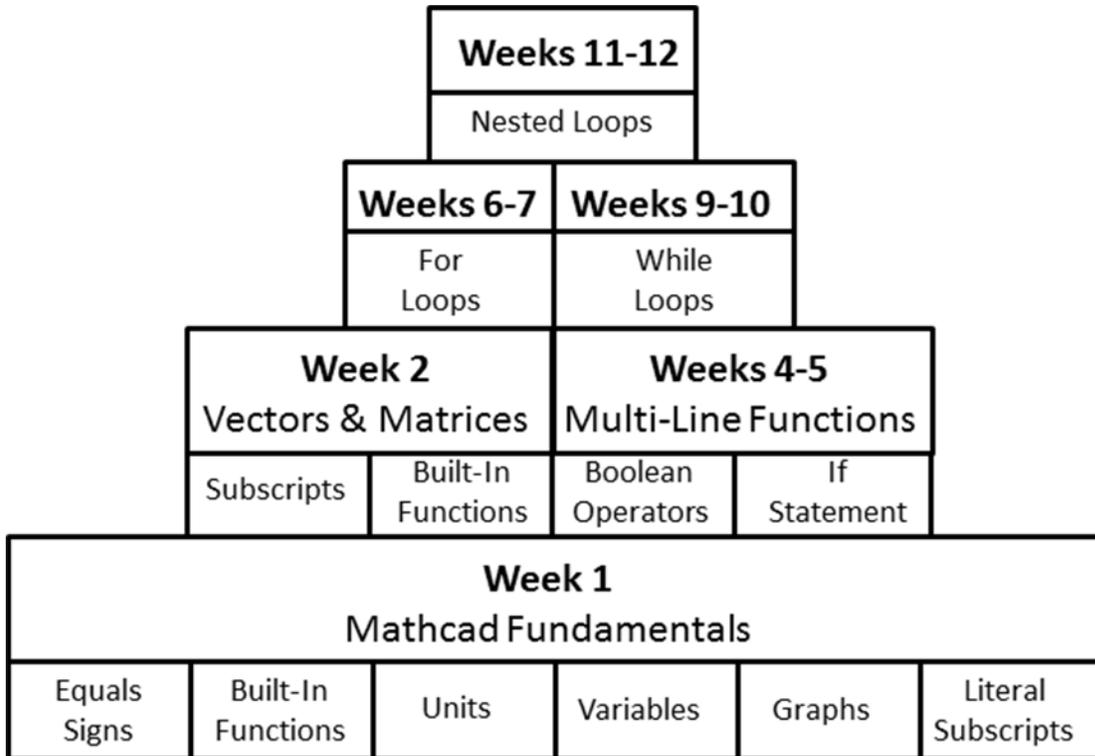


Figure 3 – CIVL 210 Programming Learning Pyramid with Fall 2012 Course Sequence. Note: Weeks 3 and 8 are omitted because the course content those weeks was not relevant to programming

Introduction of the TOOLBOX

Through years of observing and discussing with many students who completed the computer applications course, it became apparent that students struggled to distinguish between content that must be memorized and content that must be adapted to various situations. This distinction is important in a programming course because students must memorize certain information, such as syntax, but also understand that programming is generally a more creative process. Therefore, the idea of a TOOLBOX was introduced.

Each student was provided one blank sheet of paper. This sheet of paper was called the TOOLBOX. When a new definition, function, or concept was introduced in the class it was recorded in the TOOLBOX. Therefore, the TOOLBOX provided a record of all the content covered in the course. The students used the TOOLBOX when writing the programs, and understood that it contained all the content available to them that should be memorized.

Restructuring of Assignments

Research shows that frequency of testing may have a positive impact on student performance. In Tuckman's [20] study, higher testing frequency improved average test scores of procrastinators. Using data from a Principles of Marketing Class, Deck [21] showed that a weekly testing frequency produced higher average test scores during the semester than a monthly testing frequency. In this study, however, there was no significant difference between the two groups' final exam scores, indicating that retention was not improved with greater frequency testing. In addition, Mays, *et al.* [22] suggested that higher testing frequency has the potential not only for improving overall grades but in improving student moral.

In previous course offerings of the computer applications course, students received weekly feedback on assignments and tests. However, the weekly course format was (a) introductory lecture on Monday, (b) follow-up and assignment of a challenging Mathcad problem on Wednesday, and (c) submittal of assignment and quiz on Friday. Between Monday and Wednesday, homework was not consistently assigned, graded, and returned because of the strong emphasis on assignments and quizzes toward the end of the week. During the fall of 2012, the instructors wanted to provide the students with a more balanced coursework schedule with consistent feedback. Therefore, following most classes, homework was assigned and graded. Additionally, the weekly tests used prior to 2012 were adapted to weekly quizzes. The content of the homework and the quizzes was carefully aligned to encourage student motivation on homework assignments, quizzes, and throughout the course in general. Two tests were also given during the semester to help students connect the information learned over a number of weeks. The weekly assignments given in previous years were replaced with mini-projects that the student had approximately two weeks to complete. The mini-projects were identical to the assignments in most ways, but the students had more time to understand the information, attempt the solution, and ask questions if necessary.

Increase in Active Learning Activities

In engineering generally and programming specifically, students must practice solving problems themselves in order to master concepts. Therefore, to engage students in lecture and allow for student practice, many competitive active learning activities were introduced to the course. Students worked weekly in pairs and groups of three to solve problems in Mathcad. The competitive learning activities usually had small prizes as further motivation. Students engaged in competitive learning activities to create flowcharts using notecards, to produce the output expected from Mathcad, to write user-defined functions, and to write functions that used looping and subscripted variables.

Incorporation of Lifelong Learning

Lifelong learning was incorporated into the class for two reasons. First, lifelong learning is a critical skill that all engineers must possess. Second, the process of lifelong learning involves students solving problems for which they do not possess all of the required information. In many ways, this process mirrors the solution of programming problems. Therefore, discussing lifelong learning provided the students confidence to attempt to solve problems without immediate answers, which helped them develop the potential to become more capable programmers.

SURVEYS

To help assess the effectiveness of the course restructuring, students who were taking the computer applications class (CIVL 210) in the fall semester of 2012 and students who had previously taken the course completed the surveys included in the Appendix. The survey entitled "Current Student Survey – CIVL 210" was completed by

2013 ASEE Southeast Section Conference

sophomore students who were enrolled in the course during the fall semester of 2012. The survey entitled “Past Student Survey – CIVL 210” was completed by junior and senior students who had taken the computer applications class prior to 2012. During the Fall of 2011 all three authors taught a section of the course. During the Fall of 2012, Dr. Kenneth Brannan and Professor Kaitlin Marley each taught a section. The third author, Dr. John Murden, participated in the planning, assessment, and restructuring of the course. The surveys focus primarily on student attitudes and confidence associated with the course.

Results of the surveys are in Table 2. The Question Number shown in Table 2 was included in the table only for purposes of discussing the student responses and does not correspond with the numbers shown on the actual surveys. The student responses are discussed in the subsections below.

Effectiveness of Mathcad Prime 2.0

To assess the overall effectiveness of the switch to Mathcad Prime 2.0, the students were asked Questions 13 and 14 in Table 2. These questions targeted the difficulty students had learning to use the basics of Mathcad and learning to write programs in Mathcad. With a rating of 5 representing very difficult, the students that used Mathcad Prime 2.0 responded with ratings of 1.44 and 2.16 for Questions 13 and 14, respectively. The students that used Mathcad 15 when they took the computer applications class responded with ratings of 2.37 and 3.27 to the same questions. This suggests that the switch to Mathcad Prime 2.0 did facilitate a new user’s ability to learn the fundamental information and program within the Mathcad environment.

Effectiveness of the TOOLBOX

Students were asked in Questions 4 through 11 to evaluate the usefulness of teaching techniques in CIVL 210 designed to improve understanding. A rating of 5 represented a technique perceived to be very useful. The TOOLBOX feature received an average rating of 3.75. Of the eight techniques the students ranked, this represents the second lowest score (flowcharts received the lowest rating). A rating of 3.75 out of 5 does indicate usefulness, but the TOOLBOX did not score as high as the instructors anticipated. It is possible that although the students did not perceive the TOOLBOX as highly useful, that the existence of the TOOLBOX may have been valuable. The instructors believe that the TOOLBOX helped to reinforce the idea that a program cannot be generated from memorization of a previous program, and that it solved many problems previous classes experienced.

Effectiveness of Restructuring of Assignments

The survey results suggest that the restructuring of assignments was very effective. First, preparing daily homework, which had not been required in the same form in previous semesters, received the highest average usefulness rating of all techniques (4.28 out of 5). Further, students taking the course during the fall of 2012 rated preparing homework, preparing mini-projects, and taking tests 0.75-1.00 points higher in usefulness than students who took the course prior to the fall of 2012. Therefore, through the restructuring of the assignments, the students found them to be significantly more useful.

Effectiveness of Increase in Active Learning Activities

Although the usefulness of competitive active learning activities received a lower score than many of the other teaching methods (3.84 out of 5), the average student rating still indicates that the active learning component of the course was useful to the students. The standard deviation associated with this rating was 1.22. This is one of the highest standard deviations associated with the response to any question on the current student survey. The instructors believe that while these activities engaged and excited many students, they were less helpful to the students who are more introverted or reflective learners. Therefore, some students ranked the competitive active learning activities very high and some students ranked the competitive learning activities very low. This resulted in the high standard deviation. Overall, the competitive learning activities appeared to make the class more fun and exciting. There are many additional activities (homework and mini-projects) that target the reflective learners, and the instructors believe that the use of different types of activities that appeal to a variety of learning styles is important. It should be noted that while there were some active learning exercises for students taking the course prior to 2012, there were less of these activities and very few were designed to be competitive. For this reason, students taking the course prior to 2012 did not have a survey question on this technique.

2013 ASEE Southeast Section Conference

Table 2 – Summary of Student Responses (n = 83)

	Question	Average Rankings of Students		Standard Deviation of Rankings of Students	
		During Fall 2012	Before Fall 2012	During Fall 2012	Before Fall 2012
1	How well are you able to solve typical engineering problems in Mathcad? (5 = Very Well)	4.28	3.25	0.81	1.21
2	How well are you able to create a user-defined function (program) in Mathcad that uses 1 loop? (5 = Very Well)	4.56	2.78	0.67	1.26
3	How well are you able to create a user-defined function (program) in Mathcad that uses nested loops? (5 = Very Well)	4.00	2.34	0.98	1.39
	How useful were the following techniques or features in helping you understand loops in general? (5 = Very Useful)				
4	Flow Charts	2.97	3.04	1.20	1.23
5	Playing Computer	4.22	3.41	1.13	1.19
6	Preparing for Weekly Quizzes	3.91	N/A	1.23	N/A
7	Preparing for Tests	4.06	3.28	0.98	1.08
8	Preparing Homework	4.28	3.34	0.99	1.15
9	Preparing Mini-Projects	4.16	3.22	0.88	1.12
10	Competitive Active Learning Activities	3.84	N/A	1.22	N/A
11	The Toolbox	3.75	N/A	1.22	N/A
12	Rate your ability to perform lifelong learning as it applies to Civil Engineering. (5 = Very Strong)	4.22	3.12	0.75	1.12
13	How difficult was it for you to learn how to use the basic features of Mathcad (built-in functions, subscripts, vectors and matrices, headers and footers, text, etc.)? (5 = Very Difficult)	1.44	2.37	0.80	1.12
14	How difficult was it for you to learn how to write user-defined functions using loops (programs) in Mathcad? (5 = Very Difficult)	2.16	3.27	1.22	1.20
15	Select the word that best describes your attitude towards solving problems in Mathcad. (1 = Negative, 2 = Neutral, 3 = Positive)	2.69	1.69	0.47	1.16
16	Select the word that best describes your attitude towards writing user-defined functions (programs) in Mathcad. (1 = Negative, 2 = Neutral, 3 = Positive)	2.69	1.56	0.47	0.79
17	How interesting/exciting did you find the Mathcad course? (5 = Very Exciting)	3.97	2.40	0.95	0.89
18	How motivated did you feel in the course to solve problems using Mathcad? (5 = Very Motivated)	4.06	2.14	0.85	1.20

2013 ASEE Southeast Section Conference

Effectiveness of Incorporation of Lifelong Learning

In Question 12, students rated their ability to perform lifelong learning as it applies to civil engineering, based on a rating of 5 representing a very strong ability. Students taking the course during the fall of 2012 who were exposed to a new lifelong learning component in the course rated themselves over 1 point higher in their ability to perform lifelong learning than students taking the course prior to 2012. Incorporating a significant lifelong learning component seems to have strengthened the student belief that they can successfully engage in lifelong learning.

Overall Effectiveness of Course Restructuring

Overall, the effectiveness of the course restructuring can be seen in improved attitudes. Students taking the course during the 2012 fall semester averaged 2.69 out of 3.00 (mostly positive) on their attitude towards Mathcad, as shown in Questions 15 and 16, respectively. Students taking the course prior to 2012 had 1.69 and 1.56 averages (slightly negative) on their attitudes as shown in Question 15 and 16, respectively. The course restructuring seems to have adjusted attitudes. Additionally, students taking the course during the 2012 fall semester rated their motivation to solve problems in Mathcad as 4.06 out of 5, with 5 representing very motivated. Students taking the course prior to 2012 rated their motivation to solve problems in Mathcad as a 2.14 which is nearly a two point difference. Clearly, students taking the course in 2012 had more positive attitudes toward the course than students who took the class earlier.

CONCLUDING REMARKS

Based on the surveys, the new course format and content seems to have been highly successful. To determine if the apparent gains made in the course extend into the junior and senior years, it will be necessary to continue to obtain feedback from the students who took CIVL 210 during the fall semester of 2012. In addition, it may be valuable to look at exam performance to see if learning, and not just attitude, is improving. Another consideration is that students who were taking the course at the time the surveys were taken may have responded more favorably than students who took the course in an earlier year because they feel more confident about material that they are currently studying.

During future offerings of the course, it may be possible to improve the course by improving the quality of the active learning exercises in an effort to increase the student usefulness rating. It may also be possible to improve the value of the TOOLBOX to the students by emphasizing it more during the semester.

REFERENCES

- [1] Hansen, Steven, "Extended Active Learning as a Means to Learn Syntax in Programming Languages," *Proceedings of the 2008 Annual ASEE Conference*, Pittsburgh, PA, 2008.
- [2] Azimi, Azad and Nannette D'Imperio, "New Approach to Teaching an Introductory Computer Science Course," *Proceedings of the 2009 Annual ASEE Conference*, Austin, TX, 2009.
- [3] Garrett, Jeremy and Thomas Walker, "Student Attitudes towards the Use of Graphical Programming Languages," *Proceedings of the 2008 Annual ASEE Conference*, Pittsburgh, PA, 2008.
- [4] Harper, Kathleen, Richard Freuler, Stuart Brand, Craig Morin, Patrick Wensing, John Demel, "Comparing the Use of a Graphical Programming Language to a Traditional Text-Based Language to Learn Programming Concepts in a First-Year Course," *Proceedings of the 2009 Annual ASEE Conference*, Austin, TX, 2009.
- [5] Sun, Wangping, and Xian Sun, "Teaching Computer Programming Skills to Engineering and Technology Students with a Modular Programming Strategy," *Proceedings of the 2011 Annual ASEE Conference*, Vancouver, British Columbia, Canada, 2011.
- [6] Gustafson, Michael, Rebecca Simmons, W. Neal Simmons, Michael Ehrenfried, Tod Laursen, "A Hands-On Approach to Computational Methods in Engineering," *Proceedings of the 2009 Annual ASEE Conference*, Austin, TX, 2009.
- [7] Reed, Anthony, Taylor Creekbaum, Matthew Elliott, David Hall, Davis Harbour, "Utilizing Robotics to Facilitate Project-Based Learning: A Student Perspective," *Proceedings of the 2008 Annual ASEE Conference*, Pittsburgh, PA, 2008.
- [8] Jaeger, Beverly K., Susan F. Freeman, and Richard Whalen, "Programming is Invisible Or is It? How to Bring a First-Year Programming Course to Life," *Proceedings of the 2012 Annual ASEE Conference*, San Antonio, TX, 2012.

2013 ASEE Southeast Section Conference

- [9] Murden, John A. and Kenneth P. Brannan, "Ten Years Later: Teaching Mathcad as a Non-Traditional Programming Language," *Proceedings of the 2006 ASEE Southeast Section Conference*, Tuscaloosa, AL, 2006.
- [10] Swanborn, David Hall, and Kelly Crittenden, "Centrifugal Pump Design, Fabrication and Characterization: A Project-Driven Freshman Experience," *Proceedings of the 2008 Annual ASEE Conference*, Pittsburgh, PA, 2008.
- [11] Efimba, E. and Tori Rhoulac Smith, "The Use of Computer-Based Team Assignments as an Effective Tool in Teaching Mechanics," *Proceedings of the 2012 Annual ASEE Conference*, San Antonio, TX, 2012.
- [12] Hardin, Jon-Michael and Timothy Hodges, "Composites in Undergraduate Education: An Improved Understanding of Mechanics," *Proceedings of the 2010 Annual ASEE Conference*, Louisville, KY, 2012.
- [13] Das, Nirmal, "Teaching Reinforced Concrete Design with Mathcad Application," *Proceedings of the 2008 Annual ASEE Conference*, Pittsburgh, PA, 2008.
- [14] Estrada, Hector, and Luke Lee, "Challenges of Teaching Earthquake Engineering to Undergraduates," *Proceedings of the 2009 Annual ASEE Conference*, Austin, TX, 2009.
- [15] Wiggins, Edwin, "Teaching COGAS with Mathcad," *Proceedings of the 2008 Annual ASEE Conference*, Pittsburgh, PA, 2008.
- [16] Pauley, Laura L., "Adaptable Homework Solutions in Fluid Mechanics Using Mathcad," *Proceedings of the 2009 Annual ASEE Conference*, Austin, TX, 2009.
- [17] Brannan, Kenneth P. and John A. Murden, "Using Mathcad Debugging Functions as a Teaching Tool," *Proceedings of the 2007 ASEE Southeast Section Conference*, Louisville, KY, 2007.
- [18] Brannan, Kenneth P., John A. Murden, and Edward L. Hajduk, "A New Meaning to 'Click Here' in a Computer Class," *Proceedings of the 2009 ASEE Southeast Section Conference*, Marietta, GA, 2009.
- [19] PTC, "Mathcad: Version Comparison Chart," Retrieved November 20, 2012 from http://www.ptc.com/WCMS/files/136996/en/7358_Mathcad_Chart_TS_EN.pdf
- [20] Tuckman, Bruce W., "Using Tests as an Incentive to Motivate Procrastinators to Study," *The Journal of Experimental Education*, Heldref Publications, 1998.
- [21] Deck, D.W., *The Effects of Testing on College Students in a Principles of Marketing Course*, Ph.D. Dissertation, Virginia Polytechnic Institute and State University, Blacksburg, VA., 1998.
- [22] Mays, T. W., J. K. Plemmons, J. A. Murden., and K. P. Brannan, "Fine-Tuning the Frequency of Testing," *Proceedings of the 2003 ASEE Southeast Section Conference*, Macon, GA, 2003.

Kenneth P. Brannan

Kenneth P. Brannan is Professor and Head of Civil and Environmental Engineering Department at The Citadel. He was President of the Southeastern Section of ASEE during 1998-1999, co-recipient of the Thomas C. Evans Instructional Paper Award for 1990, and co-recipient of the Best Paper Award at the 2005 ASEE Annual Conference. He earned B.C.E. and M.S. degrees from Auburn University and the Ph.D. from Virginia Tech. A registered professional engineer, Dr. Brannan has interests in freshman engineering education, computers in engineering education, water supply, and wastewater treatment systems.

Kaitlin H. Marley

Kaitlin Marley is an Instructor in the Department of Civil and Environmental Engineering at The Citadel. She earned her Bachelor of Science degree in Civil Engineering from North Carolina State University in 2008, and her Master of Science degree in Geotechnical Engineering from the University of California at Berkeley in 2009. Prior to her employment at The Citadel, she helped develop textbooks and courseware for mathematics courses. She currently serves as an assistant advisor for the Society of Women Engineers at The Citadel.

John Alden Murden

Since 1989, J. A. Murden has served on the faculty in Civil and Environmental Engineering at The Citadel and currently holds the rank of Associate Professor. He earned his B.S., M.S. and Ph.D. degrees in Civil Engineering from Clemson University in 1977, 1984 and 1987 respectively. Dr. Murden's interests include modeling civil engineering systems, experimental mechanics, computer applications, and improving courses in civil engineering. He brings industrial experience in aerospace structures, nuclear engineering and shipboard structures into his courses.

APPENDIX

Current Student Survey

CIVL 210

Please respond to each of the following survey questions you felt on the last day of class in Mathcad (i.e. assume that you know everything about Mathcad that you knew on that day).

- | | | | | | |
|---|---|---|---|---|---|
| 1) How well are you able to solve typical engineering problems in Mathcad on a scale of 1 to 5 (5 is very well)? | 1 | 2 | 3 | 4 | 5 |
| 2) How well are you able to create a user-defined function (program) in Mathcad that uses 1 loop on a scale of 1 to 5 (5 is very well)? | 1 | 2 | 3 | 4 | 5 |
| 3) How well are you able to create a user-defined function (program) in Mathcad that uses nested loops on a scale of 1 to 5 (5 is very well)? | 1 | 2 | 3 | 4 | 5 |
| 4) How useful were the following techniques or features in helping you to understand loops in general (5 is the most useful)? | | | | | |
| a. Flow Charts | 1 | 2 | 3 | 4 | 5 |
| b. Playing Computer | 1 | 2 | 3 | 4 | 5 |
| c. Preparing for Weekly Quizzes | 1 | 2 | 3 | 4 | 5 |
| d. Preparing for Tests | 1 | 2 | 3 | 4 | 5 |
| e. Preparing Homework | 1 | 2 | 3 | 4 | 5 |
| f. Preparing Mini-Projects | 1 | 2 | 3 | 4 | 5 |
| g. Competitive Active Learning Activities | 1 | 2 | 3 | 4 | 5 |
| h. The Toolbox | 1 | 2 | 3 | 4 | 5 |
| 5) ASCE defines lifelong learning in BOK 2 as, “the ability to acquire knowledge, understanding, or skill throughout one’s life.” Using a scale of 1 to 5, rate your ability to perform lifelong learning as it applies to Civil Engineering. (5 is a very strong ability to perform lifelong learning). | 1 | 2 | 3 | 4 | 5 |

2013 ASEE Southeast Section Conference

- | | | | | | | |
|-----|---|----------|---|---------|---|----------|
| 6) | How difficult was it for you to learn how to use the basic features of Mathcad (built-in functions, subscripts, vectors and matrices, headers and footers, text, etc.)?
(5 is very difficult) | 1 | 2 | 3 | 4 | 5 |
| 7) | How difficult was it for you to learn how to write user-defined functions using loops (programs) in Mathcad
(5 is very difficult)? | 1 | 2 | 3 | 4 | 5 |
| 8) | Select the word(s) that best describe your attitude towards solving problems in Mathcad. | Negative | | Neutral | | Positive |
| 9) | Select the word(s) that best describe your attitude towards writing user-defined functions (programs) in Mathcad. | Negative | | Neutral | | Positive |
| 10) | How interesting/exciting did you find the Mathcad course
(5 is very interesting/exciting)? | 1 | 2 | 3 | 4 | 5 |
| 11) | How motivated did you feel in the course to solve problems using Mathcad (5 is very motivated)? | 1 | 2 | 3 | 4 | 5 |
| 12) | Rate the Mathcad course overall (5 is excellent). | 1 | 2 | 3 | 4 | 5 |

Past Student Survey

CIVL 210

Please respond to each of the following survey questions you felt on the last day of class in Mathcad (i.e. assume that you know everything about Mathcad that you knew on that day).

- | | | | | | |
|--|---|---|---|---|---|
| 1) How well are you able to solve typical engineering problems in Mathcad on a scale of 1 to 5 (5 is very well)? | 1 | 2 | 3 | 4 | 5 |
| 2) How well are you able to create a user-defined function (program) in Mathcad that uses 1 loop on a scale of 1 to 5 (5 is very well)? | 1 | 2 | 3 | 4 | 5 |
| 3) How well are you able to create a user-defined function (program) in Mathcad that uses nested loops on a scale of 1 to 5 (5 is very well)? | 1 | 2 | 3 | 4 | 5 |
| 4) How useful were the following techniques or features in helping you to understand loops in general (5 is the most useful)? | | | | | |
| a. Flow Charts | 1 | 2 | 3 | 4 | 5 |
| b. Playing Computer | 1 | 2 | 3 | 4 | 5 |
| c. Doing an assignment in Mathcad | 1 | 2 | 3 | 4 | 5 |
| d. Preparing for Weekly Tests | 1 | 2 | 3 | 4 | 5 |
| e. Preparing Homework | 1 | 2 | 3 | 4 | 5 |
| f. Preparing Assignments | 1 | 2 | 3 | 4 | 5 |
| 5) ASCE defines lifelong learning in BOK 2 as, "the ability to acquire knowledge, understanding, or skill throughout one's life." Using a scale of 1 to 5, rate your ability to perform lifelong learning as it applies to Civil Engineering. Remember to rate your ability as if it were the LAST DAY of Mathcad class (Fall Sophomore year). (5 is a very strong ability to perform lifelong learning). | 1 | 2 | 3 | 4 | 5 |

2013 ASEE Southeast Section Conference

- | | | | | | | | |
|-----|--|-------------------|---|---------|---|------------|--|
| 6) | How difficult was it for you to learn how to use the basic features of Mathcad (built-in functions, subscripts, vectors and matrices, headers and footers, text, etc.)? (5 is very difficult) | 1 | 2 | 3 | 4 | 5 | |
| 7) | How difficult was it for you to learn how to write user-defined functions using loops (programs) in Mathcad (5 is very difficult) ? | 1 | 2 | 3 | 4 | 5 | |
| 8) | Are you currently using Mathcad in any of your classes? | YES | | | | NO | |
| 9) | If you responded YES to Q8, are you using Mathcad Prime 2.0 or Mathcad 15? | Mathcad Prime 2.0 | | | | Mathcad 15 | |
| 10) | Select the word(s) that best describe your attitude towards solving problems in Mathcad. | Negative | | Neutral | | Positive | |
| 11) | Select the word(s) that best describe your attitude towards writing user-defined functions (programs) in Mathcad. | Negative | | Neutral | | Positive | |
| 12) | How interesting/exciting did you find the Mathcad course (5 is very interesting/exciting) ? | 1 | 2 | 3 | 4 | 5 | |
| 13) | How motivated did you feel in the course to solve problems using Mathcad (5 is very motivated) ? | 1 | 2 | 3 | 4 | 5 | |
| 14) | Rate the Mathcad course overall (5 is excellent) . | 1 | 2 | 3 | 4 | 5 | |