AC 2007-341: DESIGN OF AN INTRODUCTORY MATLAB COURSE FOR FRESHMAN ENGINEERING STUDENTS

Darryl Morrell, Arizona State University

Design of an Introductory MATLAB Course for Freshman Engineering Students

1 Introduction

This paper describes the design and implementation of a one credit-hour MATLAB course for freshman engineering students in the new multi-disciplinary engineering program at Arizona State University at the Polytechnic campus. The course was initially offered in the Spring 2006 semester; based on the spring experience, the course was significantly redesigned for the Fall 2006 semester. The goal of the course design was to teach specific MATLAB and programming concepts within a broader context of developing problem solving skills for real world engineering applications.

1.1 Context

This course was designed in the context of a newly-developed four-year multi-disciplinary engineering program. In this program, all students learn a common body of engineering foundation material in their freshman and sophomore years, and then specialize through a primary and secondary concentration in their junior and senior years. Students typically begin a calculus sequence in their first or second semester in the program; the calculus courses use MATLAB extensively. Informal student feedback indicated that lack of familiarity with MATLAB was a difficulty for many students in the calculus courses. In addition, familiarity with MATLAB, while not a requirement of the freshman engineering courses, was felt by the faculty to be a useful skill. Thus, the MATLAB course was designed as an optional one-credit-hour class for students in their freshman year.

The development of several other MATLAB courses for freshman students has been described in the literature. A course developed at West Virginia University for second-semester freshmen[1] is based on the Object Scaffolding approach to teaching MATLAB.[2,3] This approach includes presenting MATLAB within a technical problem solving context, collaborative interaction among students, and progressive coverage of first scalars, then vectors, and then matrices. An initiative to introduce MATLAB to students in the Mechanical and Aerospace Engineering Department (MAE) at University of Florida was undertaken, and the effectiveness of several different formats for and timing in the curriculum was evaluated.[4] It was discovered that "low risk" courses (e.g. a course that is not perceived as a "weed-out" course) that relate directly to students' other coursework are better received by students.

The course that we have developed differs from previous MATLAB courses found in the literature in several ways. First, it is a one credit-hour course rather than all or part of a three credit-hour course and thus covers significantly less content than courses found in the literature. Secondly, it is offered in the context of a multi disciplinary engineering program, rather than as part of a discipline specific engineering program or as a freshman level service course.

1.2 Course Design Challenges

Many of the challenges in the design of the course stem from issues faced by all engineering freshmen. A significant fraction of students do not yet employ a systematic approach to problem solving; most of the students cannot clearly articulate their problem solving approach. Students

Table 1: Student learning objectives.

1.	The student employs a problem solving process.
2.	The student uses the computational capabilities of MATLAB.
3.	The student uses simple MATLAB script files.
4.	The student graphically presents computed results following appropriate engineering conventions.

are also inexperienced in communicating quantitative information using appropriate engineering conventions (e.g. plots and graphs). Many are making the academic transition to the university environment; learning to manage independence and time is often an issue.[5]

The range of prior student programming experience varies dramatically. Most students in the class are novice programmers, although a few have significant prior programming experience; to date, none have had prior experience with MATLAB. Thus, many students face challenges stemming from issues faced by all novice programmers, which are described in *Natural Language Tutoring and the Novice Programmer*[6] and the references therein; beyond learning the specific elements of a given programming language, most of the challenges are related to managing complexity and developing problem solving skills and schemas. Specific challenges include the lack of a "library of schemata" (i.e. a collection of structures and concepts) from which to draw problem solution components; difficulty managing the decomposition, composition, and goal/sub-goal processes inherent in programming; and a tendency to begin coding without adequate (or often any) planning.

2 Course Design

Our process of developing the MATLAB course addressed the course design challenges by focusing on problem solving and simple programming schemas (e.g. loops and conditionals).

Most of the students in the course are relatively novice problem solvers. It is unrealistic to assume that students will become expert problem solvers within one semester. Indeed, we assume a developmental model similar to that used by Alverno College.[7,8] In this model, the first developmental step is to become aware of and able to explicitly elucidate problem solving processes that are actually used. Thus, we focus much of the problem solving activities on elucidation of the processes used by students to solve programming assignments. A significant component of most homework assignments is to describe and evaluate the problem solving process used to complete the assignment. The course materials also provide students with a model problem solving process, many of the steps of which are adapted from a general engineering problem solving process;[9] students are encouraged to use this process to address the more complex assignments.

The expectation is that a typical student in a one-credit hour course will spend three hours per week on that course. Since this course is taught in an environment that values "learning by doing," two of the three hours per week are spent in a mediated classroom in which every student sits in front of a computer. This allows the students to work on the MATLAB assignments during class and provides Internet access for help and course resources; the bulk of class time is spent by students

Table 2: Student competencies.

1.	Attend at least 80% of classes
2.	Prepare for and participate in class
3.	Clearly describe a problem solving process.
4.	Effectively use a problem solving process.
5.	Use MATLAB to create graphs of computed results.
6.	Graphs follow appropriate engineering conventions.
7.	Use the computational capabilities of MATLAB to solve an engineering problem.
8.	Use the symbolic capabilities of MATLAB to solve an engineering problem.
9.	Correctly implement MATLAB functions.
10.	Correctly implement a "for loop" or equivalent vector operation to solve a problem.
11.	Correctly implement a "while loop" to solve a problem.
12.	Correctly implement an "if statement" to solve a problem.
13.	Select a proper programming construct to solve a problem.
14.	Use an abstract representation (flow chart or pseudo code) to design an m-file script.
15.	Combine programming constructs (loops and conditionals) to solve a problem.

working with MATLAB to solve problems. Students use the hour outside of class to complete assignments and to document and assess their work.

2.1 Course Learning Objectives

The course learning objectives were designed to address the challenges described above. They focus on problem solving and on the use of MATLAB to solve simple engineering problems. The student learning objectives for the course are shown in Table 1.

The student learning objectives were further refined into fifteen competencies; the competencies are shown in Table 2. In addition to the problem solving competencies and those relating to MATLAB, two competencies addressing attendance and participation are included to encourage students to be actively engaged in the class. A mastery-based grading approach[10,11] is used to assess student achievement of each objective. Each homework assignment provides opportunities to demonstrate several competencies, and students are given multiple opportunities to demonstrate each competency. Students must demonstrate a competency at least once in a homework assignment or the final exam to receive credit for the competency.

2.2 Assignments

Homework was assigned and submitted weekly. Students use a significant portion of the course meeting time to work on assignments, and typically finish the assignment and the associated self reflection out of class before the assignment is due. Most of the assignments require the student to

Table 3: MATLAB skills and associated problems.

MATLAB Skill	Associated Problem
Simple Calculation	Use MATLAB to compute distance from the voltage output by an infrared distance sensor and to compute the terminal velocity of a skydiver.
Graphical Presentation of Data	Use MATLAB to compute and graph the equivalent resistance of two parallel resistors and to compare measured velocity with the theoretical velocity of a falling body.
Problem Solving	Model the relationship between volume and weight of uncrushed aluminum cans.
For Loops	Create graphs for several different parameter values, and compute velocity and displacement from acceleration data for a railgun projectile.
Conditionals	Develop a cost analysis in which costs depend on number of units manufactured.
Symbolic Computation	Solve simultaneous non-linear equations describing velocity and position of two objects.
While Loops	Compute the balance, finance charge, and minimum payment when one semester's tuition is charged on a credit card and determine the time needed to pay off the bill by paying only the minimum payment each month.
Functions	Modify the program for the previous topic to use functions.
Integration of Concepts	Implement a control strategy for a simulated robot "capture the flag" game.

describe their problem solving process. For every assignment, the student is asked to self-assess whether they have demonstrated the competencies associated with the assignment and to describe the evidence that supports their assessment. Every assignment also requires a short reflective statement, which includes what was learned and what was still unclear. Most of the assignments require students to create a MATLAB solution (often MATLAB script files) from scratch-no sample or partially completed solutions are given as part of the assignments.

Each assignment is based on a more-or-less real-world problem that is appropriate for the analytical skills of freshman-level students. The MATLAB skills addressed in the assignments and associated problems used in the Fall 2006 semester are shown in Table 3.

2.3 Course Materials

There is a significant number of textbooks on MATLAB for engineering students.[12,13,14,3,15,16] Most of these texts are designed to provide a fairly comprehensive treatment of MATLAB's extensive capabilities, and are somewhat daunting to the student in a one-credit-hour course. As an alternative, course specific materials have been developed and made available online through the Connexions Project. These materials include instruction in MATLAB programming concepts and structures, drill exercises, sample problems with some solutions, and problems to be used in home-

work assignments; these materials are available at http://cnx.org/content/col10325/latest/. Students are also encouraged to discover and use available help resources (e.g. the MATLAB help command and online tutorials).

3 Assessment of Course Effectiveness

All of the difficulties experienced by novice programmers identified in Section1.2 were observed in both semesters. Also, the effectiveness of the students' application of the structured problem solving process varied widely, as did their ability to describe the problem solving process that they actually used.

Student course and instructor evaluations were generally positive; in particular, there were positive comments about the use of realistic problems as the foundation for homework assignments. There were some negative comments about the effort required to describe the problem solving process. This reinforces informal classroom observations; in particular, several of the students with prior programming expertise had difficulty making their problem solving process explicit.

With a few exceptions, students generally provided positive but shallow assessments of their mastery of the course competencies, rarely explicitly citing supporting evidence that they had achieved these competencies. To some extent, this reflects a lack of experience on both the part of the students and the instructor in implementing effective self evaluation processes; one area for improvement is to improve the self assessment process.

The expectation that students solve problems using MATLAB commands and scripts developed from scratch without any exemplars or partially solved examples may have been detrimental to student learning. Several times during the Fall 2006 semester, a significant fraction of the students had difficulty working through the problem solving process at the same time they were attempting to understand basic programming concepts and establish a library of programming schemata. This may be related to the significant cognitive load imposed by both problem solving and programming. One approach that may improve student learning is to reduce the cognitive load, perhaps by changing some assignments so that they do not require developing MATLAB commands and scripts from scratch but instead require modifying and analyzing existing MATLAB code.

4 Conclusions and Future Work

We have developed a one credit-hour MATLAB course for freshman engineering students in a multi-disciplinary engineering program. The course has been reasonably successful at providing students with basic facility with MATLAB and problem solving. The primary issue still requiring improvement is the development of problem solving skills while still imparting MATLAB content knowledge. Another area requiring improvement is the development of more effective self assessment processes; the practices identified at Alverno College[17] may form a basis for this improvement.

References

[1] Robin Hensel and Ye Sun. Application of object scaffolding to develop a hands-on, problem-centered, and project-based freshman MATLAB course. In *ASEE Annual Conference and Exposition*, 2006.

[2] J. Sticklen, M. Amey, T. Eskil, T. Hinds, and M. Urban-Lurain. Application of object-centered scaffolding to introductory MATLAB. In *ASEE Annual Conference and Exposition*, Salt Lake City, Utah, June 2004.

[3] J. Sticklen and M. Taner Eskil. *An Introduction To Technical Problem Solving with MATLAB V. 7*. Great Lakes Press, Inc., 2005.

[4] Raluca Rosca. Learning MATLAB–just-in-time or freshman year? In *ASEE Annual Conference and Exposition*, 2006.

[5] Dan D. Budny and Cheryl A. Delaney. Working with students and parents to improve the freshman retention. In *31st ASEE/IEEE Frontiers in Education Conference*, pages T3E5–T3E10, Reno, NV, October 10 - 13 2001.

[6] H. Chad Lane. *Natural Language Tutoring and the Novice Programmer*. PhD thesis, University of Pittsburgh, 2004.

[7] *Student Assessment-as-Learning at Alverno College*. Alverno College Institute, 1994.

[8] Marcia Mentkowski and Associates. *Learning That Lasts: Integrating Learning, Development, and Performance in College and Beyond*. Jossey-Bass, San Francisco, 2000.

[9] H. Scott Fogler and Steven E. LeBlanc. *Strategies for Creative Problem Solving*. Prentice-Hall, Upper Saddle River, NJ, 1995.

[10] Thomas R. Guskey. *Implementing Mastery Learning*. Wadsworth Publishing, 2nd edition, 1997.

[11] Thomas R. Guskey and Jane M. Bailey. *Developing Grading and Reporting Systems for Student Learning*. Corwin Press, 2001.

[12] Joe King. *Matlab 6 for Engineers: Hands-On Tutorial*. RT Edwards, 2001.

[13] David Kuncicky. *MATLAB Programming*. Prentice-Hall, 2003.

[14] Delores Etter, David Kuncicky, and Holly Moore. *Introduction to MATLAB 7*. Prentice-Hall, 2005.

[15] Edward B. Magrab, Shapour Azarm, Balakumar Balachandran, James Duncan, Keith Herold, and Gregory Walsh. *An Engineer's Guide to Matlab*. Prentice-Hall, 2007.

[16] Holly Moore. *MATLAB For Engineers*. Prentice-Hall, 2007.

[17] Georgine Loacker, editor. *Self Assessment at Alverno College*. Alverno College Institute, 2000.