

Extended Abstract – Development of an Interactive, Self-Directed Learning Tool for First-Year Programming

Victoria Goodrich, Jaclyn Nord, Kerry Meyers, Leo McWilliams and Jay Brockman
 University of Notre Dame, vfroude@nd.edu, jnord@nd.edu, kmeyers1@nd.edu, lmcwilli@nd.edu, jbb@nd.edu

Abstract - In this work, we present the “Docens Learning Tool”, an automated study tool developed for use in the MATLAB environment to provide numerous practice problems with real-time evaluation. This learning tool provides students with an opportunity to read a problem description, write and debug a code segment, and submit the program for evaluation. Docens Tool was created completely within the MATLAB environment, the programming environment taught in the first-year engineering course.

This work focuses largely on the features of the Docens “Learning” Tool, including future expansions to other instructional tool development. In addition, we will review the initial launch of the tool in the spring 2012 semester of the first-year engineering course at the University of Notre Dame. This class contains approximately 400 students across all engineering disciplines with a wide variety of programming backgrounds. In short, students were provided with the tool as a study aid several weeks before a live, timed programming exam in the course. Participation was voluntary, and it was not tied to student grades in any way.

Index Terms – Introductory Programming, MATLAB, Self-directed learning

INTRODUCTION

Mathworks MATLAB programming environment has seen increasing use in engineering courses even over more traditional programming languages such as C, Fortran, and Java [1,2]. Because MATLAB has applications in many engineering fields, it extends well into a first-year engineering course sequence where all engineering majors are represented. However, despite the extensive pre-made functions and tools, students are still burdened with learning basic programming constructs (such as loops, conditionals, and functions), and faculty are challenged to provide the most efficient instruction to a student population with diverse programming skills. The best method of delivering these fundamental building blocks has been an ongoing discussion in engineering education [3,4]. However, one common method is to introduce fundamental techniques and

syntax early through traditional lectures, followed by providing students with redundant practice opportunities [5]. Unfortunately, in large, introductory classes where programming skills are often introduced, assessment is most often limited by the staff size. In fact, due to staffing limitations, very few assessed programming opportunities were provided outside of homework assignments. During the timed programming challenges given each year, it was determined that many students lacked the practice required for programming in an effective and efficient manner. Several schools and organizations have developed automated systems for MATLAB programming [6,7] which allows for additional assessment that requires minimal staff time. Following a similar model, a fully automated learning tool was proposed and built to provide students with a multitude of practice problems with instant feedback, creating an opportunity for self-motivated learning and assessment. In this paper, we introduce the Docens Tool, a learning tool encompassed completely within MATLAB, to provide added practice and evaluation opportunities.

COURSE BACKGROUND

MATLAB is taught throughout both the fall and spring semester in a first-year undergraduate engineering design course sequence. It is first introduced in the fall semester, but is a primary focus during the spring semester through a semester long, interdisciplinary project.

Each semester, the course sequence has enrollments of approximately 350-450 first-year engineering students. The course involves large group lectures that introduce the theory and background for projects followed by small group (25-35 students) learning centers that focus on hands on learning. For this paper, we will focus on the Spring 2012 semester, where Docens Tool was initially launched. During the semester students complete a 3-5 person, open-ended group project with a large MATLAB requirement. Well-structured code and efficient programming are stressed through lecture, learning centers, and projects. Assessments are conducted through weekly homework assignments and deliverables, two exams, project requirements, and two timed programming challenges.

A sequence of MATLAB lectures walk students through a significant number of basic programming building

Session F4C

blocks over the course of 10 weeks. First, students develop skills for creating and manipulating vectors, arrays, and strings, which includes presenting and plotting data. Next, basic programming constructs are introduced with a focus on algorithmic thinking. Loops, conditionals, and functions are each evaluated in the following weeks. Finally, instruction focuses on creating a graphical user interface (GUI) in MATLAB's Graphical User Interface Design Environment (GUIDE) as a means of explaining and exploring their projects.

In addition, students attend weekly learning centers where they follow highly structured activities that guide them through the programming concepts related in lecture. Students are required to complete a learning center deliverable, as well as separate homework assignments

focused on basic MATLAB skills for the first 8 weeks of the course.

DOCENS “LEARNING” TOOL

In order to provide supplemental practice and real-time feedback opportunities for students, an automated learning tool was developed and launched in the Spring 2012 semester. Docens “Learning” Tool, shown in Figure 1, is hosted within a graphical user interface (GUI) in the MATLAB environment. The tool consists of a large button matrix of seven rows and nine columns, a “Start Button” to display the problem statement, an “Evaluate Problem” to check code outputs, and a text box to provide immediate feedback or instructions for using the tool.

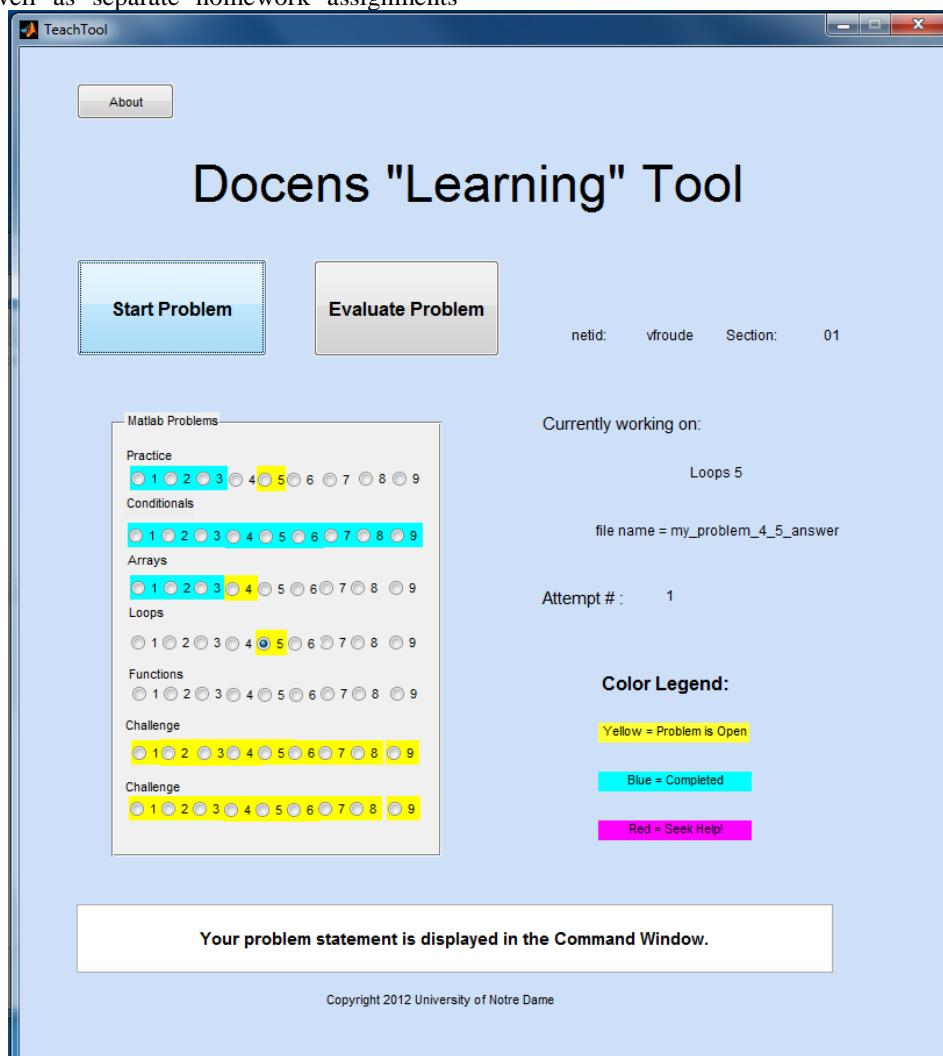


FIGURE 1. GRAPHICAL USER INTERFACE OF THE DOCENS “LEARNING” TOOL

Docens Tool was provided to all students through a university managed dropbox system. Although not necessary for this type of tool, using the dropbox system allowed for progress to be recorded and availability from any computer with access to the dropbox system. In order to

use the tool, students only had to launch the GUI file, while all other files were contained in protected background files created by the course instructors.

To use the tool, a student selects a radio button and then hits “Start Problem”. The problem statement appears in

Session F4C

MATLAB's Command Window and a new, correctly named script is created in the Editor Window and saved in a specific folder location that is unique to each student. If a student has already attempted the problem, the most recently saved script will appear; therefore, no previous work is ever lost. Students complete their code segment in the appropriate script file, and after saving, can evaluate their code by pressing the "Evaluate Problem" button. A textbox in the GUI will provide feedback about the problem and the radio button color may update, as described in the following section.

When launching the tool for the first time, the student will find all of the MATLAB problems have the blank (grey) color associated with each problem's radio button. Once students have opened a problem, the color of the button is changed to yellow. Button color is used as an indication of progress on each problem. A yellow button indicates that the student has opened and/or evaluated a problem, but has not yet correctly addressed the problem statement. In addition to the yellow button color, the number of evaluations is recorded as the "Attempt #". If a student evaluates a problem 10 times without providing an accurate code segment, the button color is changed to red and the feedback textbox directs students to get help from a course instructor or student assistant. Finally, if a student has correctly answered the problem, the button color is turned to blue. Attempts are recorded and carried over such that each time the tool is launched, students can view their progress.

Here, Docens Tool consists of 64 problems arranged by problem type and with increasing difficulty. However, this structure provides a flexible backbone that can be adapted for any course requirements. The radio button group can be changed to any size to match the number of problems available for students. Each radio button has a problem statement file as well as a problem solution file associated with it, and when the student evaluates a problem the outputs between the problem solution file and the student file are compared for accuracy. Because of this structure, problems can be updated or replaced by simply overwriting their original files, allowing instructors to adapt the tool as their courses evolve. Finally, because the tool is contained completely within MATLAB, students can use the tool on their own laptops without any additional programs installed.

USAGE RESULTS

Docens Tool was launched to all as an optional study aid for all students in the class specifically in preparation for a timed, live programming exam. Course time was dedicated to introducing the tool and initial instructions in how to use it. No grades were recorded based on use of the study aid; however, instructors monitored the progress and usage of students.

Instructors were able to track student use throughout the course of the semester. Overall, ~160 of the 411 students in the class (~39%) opened the tool at least once. Approximately 70 students (~17%) were classified as

regular users, a designation assigned for any student that attempted more than 5 problems. While the number of regular users was somewhat low, plans for future years hope to increase the number of students taking advantage of this as a study aid. However, there are no plans to make Docens Tool a graded assessment in the course.

Although not completed at this time, course instructors do plan on launching a research study into the usage and engagement in the tool as well as the impact on student grades beginning in the 2012-2013 academic year.

CONCLUSIONS

Docens "Learning" Tool is a flexible and valuable study aid for initial MATLAB programmers by providing ample opportunity to practice basic programming skills. By grading only the outputs, students can use their own creatively designed code segments to solve any of the problem statements. Therefore, the focus is on finding any algorithm that will complete the task rather than a single "correct" algorithm designed by the instructors.

During its initial launch, the Docens Tool was well received by students and course faculty. Due to the overall success, two additional learning tools are being launched during the Fall 2012 semester. A more basic learning tool will launch in October 2012, during the first semester of the course sequence, and a tool for transfer students who did not take the introductory course sequence will be made available in early August 2012. The Docens Tool pictured in this work will be released to the first-year engineering class in Spring 2013 as well.

REFERENCES

- [1] Haubold, A, "Matlab for First-year College Engineers", *ASEE/IEEE Frontiers in Education Conference*, 2007, October 10-13, 2007, pp. F1H-7 – F1H-12.
- [2] Azemi, A, Stoock, C, "Utilizing MATLAB in undergraduate electric circuits courses" *Frontiers in Education Conference Proceedings*, 1996, pp. 592-602.
- [3] Leutenegger, S, Edgington, J, "A Games First Approach to Teaching Introductory Programming", *SIGCSE '07*, March 7-10, 2007, pp. 115 – 118.
- [4] Avouris, NM, Tselios, N, Tatakis, EC, "Development and evaluation of a computer-based laboratory teaching tool" *Computer Applications in Engineering Education*, 2001, 9(1), pp 8-19.
- [5] Jermann, W, H, "The Freshman Programming Course: A New Direction", *Proceedings 1996 ASEE Annual Conference*, June 1996, Washington D.C.
- [6] Bowden, J,D, Price, C,E, "An Automated Grading System for Teaching MATLAB to Freshman Engineers", *Proceedings of the 2003 ASEE Annual Conference*, 2003, 2193.
- [7] MATLAB Cody. Accessed Online: July 20, 2012.
<http://www.mathworks.com/matlabcentral/cody/>

AUTHOR INFORMATION

Victoria Goodrich co-Director, First-Year Engineering Program, University of Notre Dame, vfroude@nd.edu

Session F4C

Jaclyn Nord Student, Electrical Engineering, University of Notre Dame, jnord@nd.edu

Kerry Meyers Director, First-Year Engineering Program, University of Notre Dame, kmeyers1@nd.edu

Leo McWilliams Director, Minority Engineering Program, University of Notre Dame, lmcwilli@nd.edu

Jay Brockman Associate Dean for Academic Programs, University of Notre Dame, jbb@nd.edu