

MatLab Marina: Web-Based Tutorials for Teaching Programming Concepts using MATLAB

Priya T. Goeser¹, Anthony Flett², Justin Kriske² and Charles Panter³

Abstract – Virtual learning environments are effective course supplements that are beneficial in reinforcing concepts from lectures and in the ability to meet diverse pedagogical needs. The objective of this work is to develop such an environment - MatLab Marina: a framework of web-based learning modules and tutorials dedicated to the improvement of student learning of programming concepts using MATLAB. MatLab Marina is used as a supplement in the course: Computing for Engineers. The main objective of this course is to introduce students to the foundations of computing with an emphasis on both the design and analysis of algorithms and the design and construction of programs for engineering problem solving. MATLAB is the computing language that is used to meet this objective. MatLab Marina is developed with this perspective to provide a balanced understanding of both algorithm development and programming using MATLAB. Further details including assessment results are discussed in the paper.

Keywords: programming for engineers, MATLAB, web-based tutorials, computing for engineers

INTRODUCTION

Current research on the effectiveness of virtual learning environments shows positive results including the reinforcement of concepts from lectures, exposure to practical applications and problems and the ability to meet diverse pedagogical needs. These are used as supplements in certain courses or in the engineering curriculum as a whole to improve students' understanding of fundamental concepts, student interest and student performance. Programming is a skill that is a crucial component in all engineering functions from problem solving to analysis and design. Hence, all engineering curriculums include programming courses, many of whom use MATLAB, a high-level, technical computing language [1]. Teaching and training students to become efficient and good programmers however, has been a challenge for many years. Engineering faculty have implemented several pedagogical approaches to address this challenge, including the use of virtual learning environments and/or online tutorials [2]-[9].

In a case study with University of Michigan and Carnegie Mellon University, a set of tutorials was developed for teaching students to use MATLAB in the context of Automatic Control [4]. The pedagogical basis on which these tutorials were built was that students learn most quickly when they are presented with material that they can quickly use to solve problems. In another paper, faculty at Georgia Southern University studied and present the results based on the feedback that students completed on a media tool that was used in the instruction of MATLAB/JAVA for computing for engineers [5]. From the survey results and student responses in this pilot study, it was concluded that media based instruction, in conjunction with traditional teaching methods, is preferred by students. In an effort to rationalize resources and improve cost-effectiveness of course delivery in online courses, the faculty from University of Sydney propose a blended approach to e-learning [6]. This paper explicitly presents the development and implementation of an e-learning tool that incorporates a Self-Practice Online Tool (SPOT). The tool has been

¹ Associate Professor, Engineering Studies Program, Armstrong Atlantic State University, 11935 Abercorn St. Savannah GA 31419, Priya.Goeser@armstrong.edu

² Undergraduate Student, Engineering Studies Program, Armstrong Atlantic State University, 11935 Abercorn St. Savannah GA 31419

³ Undergraduate Student, Computer Science, Armstrong Atlantic State University, 11935 Abercorn St. Savannah GA 31419

successfully used in a course that is made of two components that run in parallel: computer aided design and programming using MATLAB.

Background and Motivation

The Engineering Studies Program at Armstrong Atlantic State University (AASU) offers programs where students complete their freshmen and sophomore years of the engineering curriculum at AASU and transfer to the Georgia Institute of Technology to complete their degrees. Over the past ten years, the program has grown significantly from about 185 students to 350 students. Computing for Engineers (ENGR1371) is a freshmen engineering course at AASU wherein students study the fundamentals of programming and learn to solve engineering problems using MATLAB. Currently, the average enrollment for ENGR1371 is 60 per semester of which about 65% successfully complete the course (C grade or above). In addition, based on student performance in other courses such as Computational Modeling (ENGR2010) that use MATLAB extensively, approximately 70% of the above students (those that have successfully completed ENGR1371) are still not proficient with programming. Student performance and retention is clearly a significant problem. While several factors could be attributed to this problem, this paper presents one approach: the preliminary development and assessment results of MatLab Marina: a framework of web-based learning modules and tutorials dedicated to the improvement of student learning of programming concepts using MATLAB. This resource is used as a supplement in the course ENGR1371.

The following sections present a description of the course under consideration, an overview of MatLab Marina, assessment based on student surveys and website traffic analysis, and a few concluding remarks with an outline of future work.

COURSE DESCRIPTION: COMPUTING FOR ENGINEERS

Computing for Engineers (ENGR1371) is a 3-credit hour course taken by freshmen mechanical and civil engineering students at Armstrong Atlantic State University. The main objective of this course is to introduce students to the foundations of computing with an emphasis on both the design and analysis of algorithms and the design and construction of programs for engineering problem solving. As in most engineering curricula, MATLAB is the technical computing language that is used as the tool to meet this objective.

The following are specific objectives of this course:

- Objective 1: To understand the general principles of abstraction, testing, debugging and problem solving.
- Objective 2: To develop and write algorithms and programs utilizing programming fundamentals such as arrays, execution control, functions, character strings, cell arrays and structures, principles of problem solving and recursion.
- Objective 3: To develop and write algorithms and programs using procedural programming principles such as plotting (2D and 3D), matrices, file input/output, processing images and sound, and numerical methods.
- Objective 4: To understand the concepts of queues, stacks, searching and sorting data.

An approach to objective 1 is through the use of these principles in the specific topics listed under objectives 2 and 3. Objectives 2 and 3 are met by in-class programming exercises, assignments and projects which are based on practical applications of the concepts covered. The concepts of dynamic data structures as listed under objective 4 are introduced in traditional lecture formats through the semester.

It has been observed that these are extensive topics that warrant a 2-semester course when covered in depth and detail. However, in a 1-semester session, instructors (including the 1st author) are forced to emphasize either teaching the use of the tool (MATLAB) or teaching programming concepts. While both are necessary, a good understanding of programming concepts and algorithm development can be applied to any programming language as well as be used for engineering problem solving [10], [11]. In addition, students view this course to be a collection of abstract concepts, rules and methods that are difficult to understand and apply to engineering problems. It is proposed that a supplement such as MatLab Marina can address this challenge.

Teaching Pedagogy

The course, which meets either as 3 x 50 minute or as 2 x 75 minute weekly sessions, is currently taught as a combination of short 10-15 minute lectures followed by in-class programming exercises, assignments and projects. This use of active learning has been found to be effective in the instruction and learning of programming concepts [12]. While the in-class programming exercises introduce basic syntax and concepts of algorithm writing, the

assignments and projects extend the application of these to engineering problem solving. The projects are designed to be team projects keeping in mind and encouraging the pedagogy behind pair programming – ‘an effective programming pair has an active relationship with both halves actively involved in the development of the code or design’ [13].

The instructors of ENGR1371 also continue to incorporate various modalities of delivering instruction consistent with modern teaching methods suggested by educational literature. Students have access to a course website where class notes, lecture slides, sample programs, projects, assignments and other relevant and useful links are posted. In addition to standard quizzes and exams, student performance measures have also included in-class exercises and team projects.

OVERVIEW OF MATLAB MARINA

MatLab Marina is a framework of web-based learning modules and tutorials dedicated to the improvement of student learning of programming concepts. This is currently being used as a supplement in one section of the course ENGR1371 (as a pilot study) and will continue to be used in subsequent semesters.

The overall organizational structure of MatLab Marina is shown in Figure 1. Currently there are two phases of MatLab Marina: Phase I - Beginning Programming and Phase II - Intermediate Programming. Phase I has four modules: Getting Started, Vectors, Matrices and Plotting (two-dimensional). The ‘Getting Started’ module is specific to MATLAB’s interface and consists of four 2-3 minute video tutorials. The modules on ‘Conditional Structures’ and ‘Iteration’ introduce these as algorithmic and computing concepts (execution control) and the concepts are reinforced with examples using MATLAB’s syntax and interface. The other modules are based on similar pedagogy. Currently, there are a total of about thirty three 2-4 minute video tutorials. For further details of MatLab Marina, please visit <http://engineering.armstrong.edu/priya/matlabmarina/main.html>.

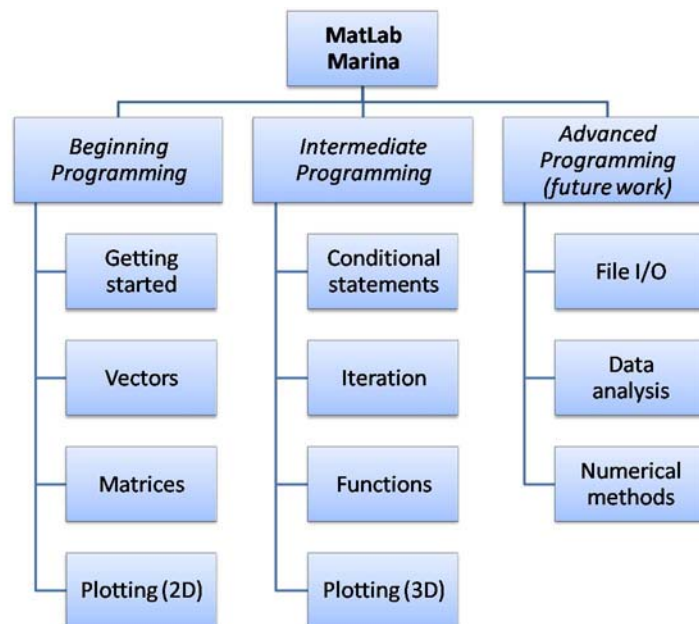


Figure 1: Organizational Structure of MatLab Marina

Note that the overall organization of the modules is similar to the chapters/topics covered in the textbook [14].

Video Tutorials

The tutorials in MatLab Marina were created with the objective to provide a balanced understanding of both algorithm development and programming using MATLAB’s interface. This dual approach leads to a complete and

better understanding of programming concepts that can be translated and applied to any programming language. This approach also helps improve student retention of concepts learned and the ability to apply these to varied problem solving scenarios.

Figure 2 shows a snapshot of a sample tutorial on iteration. The example uses a while loop to create a vector with elements equal to the cube of its indices (k^3), till a pre-defined maximum element is reached. The algorithm here is shown as a flowchart on the right along with the corresponding MATLAB program on the left. Dynamic arrows and text boxes simultaneously show the step by step working of the algorithm as well as the program. Written captions and a corresponding audio recording of the same provide brief explanations of the working process. It is noted here that the tutorials are designed to address the visual, auditory and typographical aspects of learning.

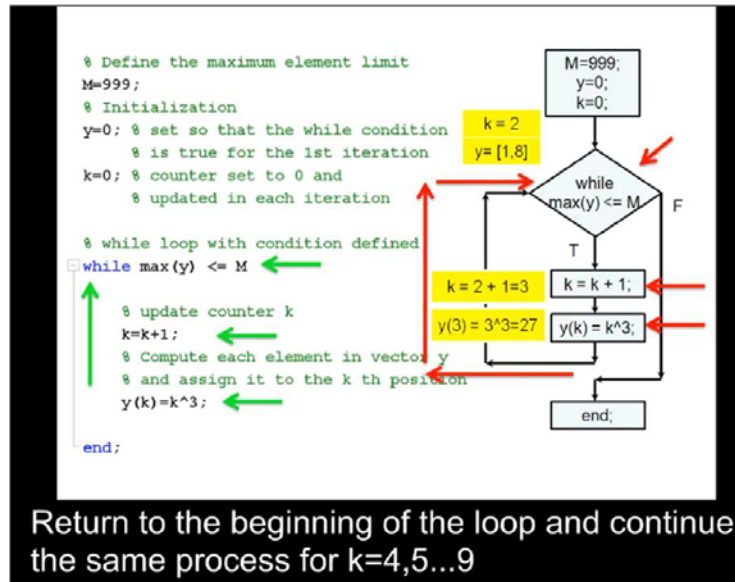


Figure 2: A snapshot of a tutorial on iteration

The tutorials were created using Camtasia Studio [15], a screen recording and video editing software and Natural Reader [16], a software that converts text to speech using natural voices. The videos are hosted on YouTube with direct links from the website so that viewers require no other software other than a web browser with a media player.

ASSESSMENT

To assess the effectiveness of the MatLab Marina, surveys were given to students (in the section that used this as a supplement) to complete and return anonymously. A summary of results of this survey is provided in Table 1. It is observed from the results, that 93.33% of the respondents in Fall 2011 strongly agreed or agreed that the use of supplemental online resources such as MatLab Marina in ENGR1371 helped them better understand the material. In addition, a majority of 86.66% of the respondents strongly agreed or agreed that the use of flowcharts and algorithms ‘side-by-side’ along with the programs in the tutorials helps them understand the concepts better. All of the respondents did note that the inclusion of several tutorials in the modules on vectors and plotting (2D and 3D) helped them in their understanding of these concepts.

Questions	Options	Fall 2011 # of responses (% of N=15)
1. The use of supplemental online resources such as MatLab Marina in this course ENGR1371 helped me to better understand the course material.	Strongly agree Agree Disagree Strongly disagree Not applicable	6 (40.00%) 8 (53.33%) 1 (6.67%) 0 0
2. The use of flowcharts and algorithms 'side-by-side' along with the programs in the tutorials helps me understand the concepts better.	Strongly agree Agree Disagree Strongly disagree Not applicable	8 (53.33%) 5 (33.33%) 2 (13.33%) 0 0
3. The inclusion of several tutorials in MatLab Marina covering concepts such as vectors, arrays and plotting (2D and 3D) helped me understand these concepts better.	Strongly agree Agree Disagree Strongly disagree Not applicable	5 (33.33%) 10 (66.67%) 0 0 0
4. The inclusion of additional tutorials (to be done) in MatLab Marina covering concepts such as iteration, functions, structures, etc. will help me understand these concepts better.	Strongly agree Agree Disagree Strongly disagree Not applicable	8 (53.33%) 4 (26.67%) 1 (6.67%) 1 (6.67%) 1 (6.67%)
5. I would recommend the use of MatLab Marina to other students asking for a refresher in programming concepts using MATLAB.	Strongly agree Agree Disagree Strongly disagree Not applicable	11 (73.33%) 4 (26.67%) 0 0 0

Table 1: Summary of results from the survey for Fall 2011.

Student's general comments are shown in Table 2. While these comments certainly support the efficacy of this supplement, they do show the need to expand the existing learning modules and include additional ones in the future.

Other Comments (written comments submitted as a part of the same survey)	# of similar responses
Load more for the advanced programming section	5
Include more complicated programs	2
More iteration examples would be nice.	2
A question/answer section might be convenient	1
I liked MatLab Marina. It helped me score very well on the first exam.	1
The key to my learning and understanding in this course was examples and the breakdown of each line of these examples.	1

Table 2: Summary of other comments from students in Fall 2011.

In addition to the surveys, relative student performance has also been evaluated to assess the effectiveness of the supplement on student learning. It was observed by the instructor that 90% of the students developed an excellent understanding of vectors and arrays and a good competency in plotting. This can be attributed to the learning modules on these concepts which students were required to include in their reading material (outside of class time).

Usage Statistics

The website itself is hosted on AASU's server while the tutorials are hosted on YouTube. Google Analytics and YouTube's video analytics tools are currently being used to track website and video traffic on the site. Figure 3 shows the number of page views and unique page views (main page and 10 other pages with highest number of views) that have been tracked over the past month (January 5th 2012 to February 1st 2012). The main page serves as the home page or landing page for all the other pages that house the learning modules and tutorials. Hence this page shows the maximum number of views.

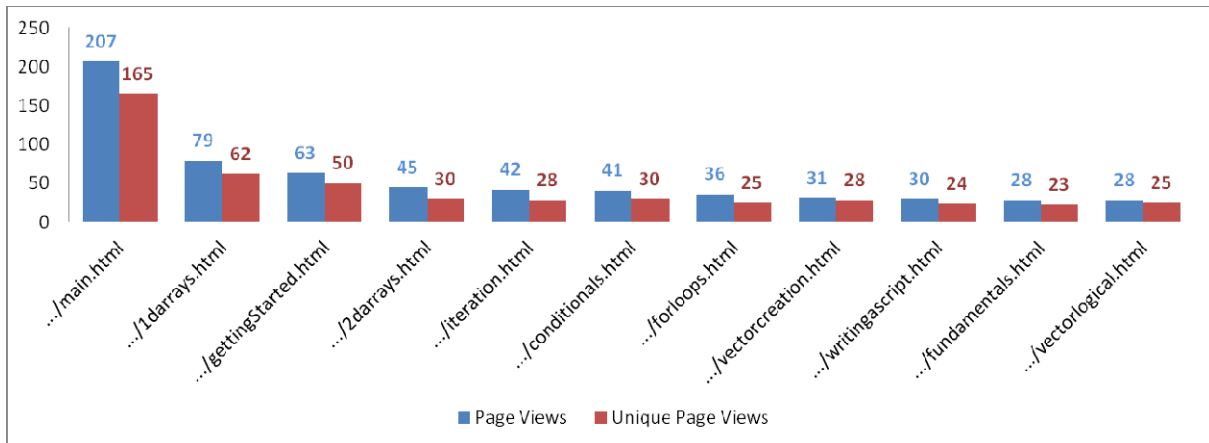


Figure 3: Number of Page Views from 01/05/12 to 02/01/12 (data from Google Analytics)

Figure 4 shows a summary of the daily total number of views (of the video tutorials) from August 15th 2011 (beginning of the Fall 2011 semester) to February 2nd 2012 (4 weeks into the Spring 2012 semester). As is observed, the website and tutorials are extensively used with identifiable 'peak days' or days when they are used most often. These 'peak days' can be traced back to weeks during which topics such as vector and arrays, iteration and plotting were covered in class. It is noted that currently these are the modules which are almost complete and house most of the tutorials.

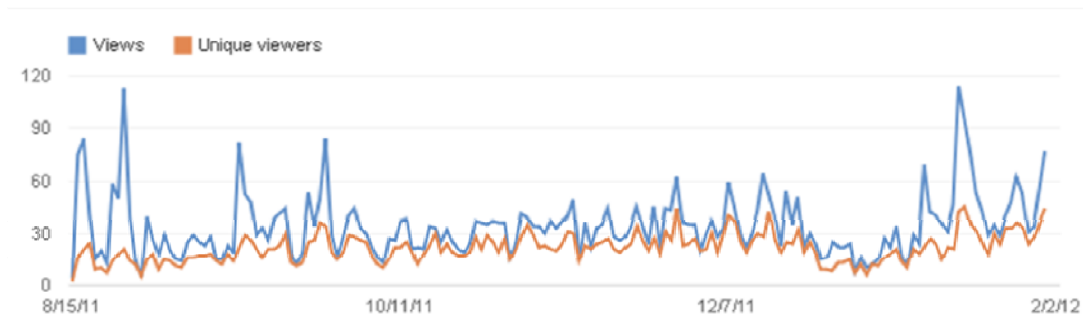


Figure 4: Number of Video Views (data from YouTube Analytics)

According to YouTube's statistics, the tutorial on 'Plotting a 3-D surface plot in MATLAB' is the most viewed one with a total of 634 views and that on 'Using a for loop to compute cosine of x using the Taylor series in MATLAB' has been viewed a total of 546 times. Figures 5 and 6 show further data for these two videos - a measurement of 'relative audience attention', which is a subjective tool that is useful for measuring student's attentiveness. According to YouTube, this tool shows the video's ability to retain viewers during playback relative to all YouTube

videos of similar length. As can be observed from Figures 5 and 6, the relative audience attention is average or above average for both these tutorials.

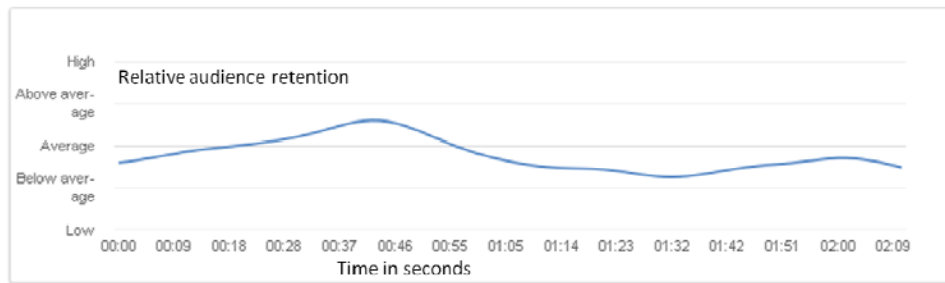


Figure 5: Relative Audience Retention for tutorial on ‘Plotting a 3-D surface plot in MATLAB’ (data from YouTube)

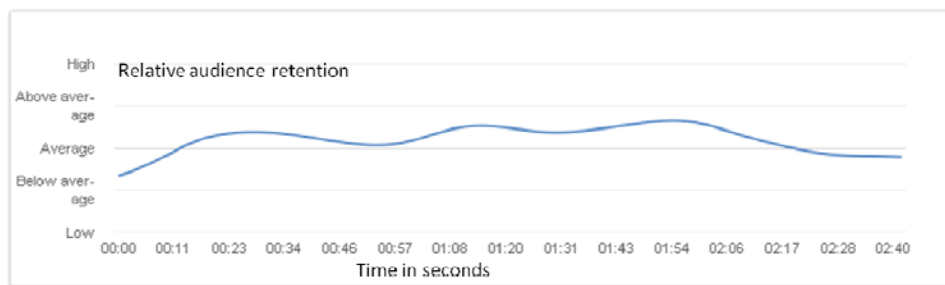


Figure 6: Relative Audience Retention for tutorial on ‘Using a for loop to compute cosine of x using the Taylor series in MATLAB’ (data from YouTube)

Discussion

Though these results represent a pilot study with preliminary data, a few observations can be made from them. First, the website, learning modules and tutorials in MatLab Marina have been used relatively consistently and extensively by students through the semester. Secondly, learning modules on vectors, arrays and plotting have been beneficial and effective in student learning. Thirdly, the feedback from the students does emphasize the need to extend this work to include other phases and learning modules.

CONCLUDING REMARKS AND FUTURE WORK

In this paper, the authors present the design and overview of MatLab Marina - a framework of web-based learning modules and tutorials dedicated to the improvement of student learning of programming concepts. MatLab Marina is a vital supplement to the course ENGR1371 and will improve student’s understanding of concepts, student performance and retention rate. Assessment results and student comments show that the tutorials have been beneficial to and well received by students. The primary advantage of these tutorials is that they provide a balanced understanding of both algorithm development and the use of MATLAB’s interface for programming. Pedagogically, MatLab Marina represents current trends in education where traditional teaching methods are complemented with online learning environments to meet the needs of the next generation.

There are three sections of ENGR1371 taught every semester at AASU with an average enrollment of 120 students a year. The tutorials developed have been in use since Fall 2011 and will continue in subsequent years. Future assessments including student surveys and relative student performance will be used to evaluate and assess the impact of this work. The success rates of the course will be compiled and studied to identify potential improvements. Traffic on the website is also being monitored using tools such as Google Analytics. A new phase

on topics in Advanced Programming will be developed in the future. Additional tutorials for the more difficult topics in Intermediate Programming such as iteration and functions will also be included in the recent future. It is also proposed that self-assessment tools such as interactive quizzes and practice exercises can be included in each learning module. This supplement will also benefit other courses in the engineering curriculum such as Computational Modeling (ENGR2010) and Signal Processing (ENGR2030) that are based on MATLAB programming. After successful implementation, MatLab Marina will be made available to the global engineering community.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the Armstrong Atlantic State University's Teaching and Learning Grant 2010-2011 P.I. Goeser, P.T., and the Talented Researcher in Engineering (TRIE) College Program 2011, NASA Georgia Space Grant Consortium, P.I. Coates, C. for providing the required funding for this work.

REFERENCES

- [1] The Mathworks, Inc., <http://www.mathworks.com/>
- [2] Wickert, J. and Beuth, J., "Web-Based Computer-Aided Engineering Tutorials across the Mechanical Engineering Curriculum", *American Society for Engineering Education Annual Conference*, Salt Lake City, UT, June 2004.
- [3] Campbell, C., "On-Line Video-Based Training for MATLAB, MATHCAD and MAPLE", *American Society for Engineering Education Annual Conference*, Austin, TX, June 2009.
- [4] Tilbury, D. and Messner, W., "A Case Study in Software Instruction over the World Wide Web: The Michigan-CMU Control Tutorials for MATLAB", *American Society for Engineering Education Annual Conference*, Milwaukee, WI, June 1997.
- [5] Thomassian, J.C., Kumazawa, R. and Kinnicutt, P., "A Study of Freshmen Students' Outlook to Media Based Tutorials of MATLAB/JAVA in Computing for Engineers", *American Society for Engineering Education Annual Conference*, Honolulu, HI, June 2007.
- [6] El-Zein, A., Langrish, T. and Balaam, N., "Blended Teaching and Learning of Computer Programming Skills in Engineering Curricula", *Advances in Engineering Education, A Journal of Engineering Education Applications*, Winter 2009, Vol. 1 Issue 3.
- [7] Baibak, T. and Agrawal, R., "Programming Games to learn Algorithms", *American Society for Engineering Education Annual Conference*, Honolulu, HI, June 2007.
- [8] Wang, E., LaCombe, J. and Vollstedt, A.M., "Teaching Structured Programming using Lego Programmable Bricks", *American Society for Engineering Education Annual Conference*, Honolulu, HI, June 2007.
- [9] Lehr, S. and Grant, C., "Utilizing Programming Projects in a Freshmen Programming Course", *American Society for Engineering Education Annual Conference*, Honolulu, HI, June 2007.
- [10] East, J.P., Thomas, S. R., Wallingford, E., Beck, W. and Drake, J., "Pattern-based Programming Instruction", *American Society for Engineering Education Annual Conference*, Washington D.C., June 1996.
- [11] Shaykhian, L. and Shaykhian, G. A., "Improving Problem-Solving Skills Through Adapting Programming Tools", *American Society for Engineering Education Annual Conference*, Honolulu, HI, June 2007.
- [12] Gottfried, B.S., "Teaching Computer Programming Effectively Using Active Learning", *American Society for Engineering Education Annual Conference*, Milwaukee, WI, June 1997.
- [13] Wiebe, E.N., Williams, L., Petlick, J., Nagappan, N., Balik, S., Miller, C. and Ferzli, M., "Pair Programming in Introductory Programming Labs", *American Society for Engineering Education Annual Conference*, Nashville, TN, June 2003.
- [14] Smith, David M., *Engineering Computation with MATLAB*, Addison-Wesley, Georgia Institute of Technology, Atlanta, GA, 2010, ISBN-13: 9780136080633
- [15] TechSmith Corporation, <http://www.techsmith.com/>
- [16] NaturalSoft Limited, <http://www.naturalreaders.com/index.htm>

Priya T. Goeser

Priya T. Goeser is an Associate Professor of Engineering Studies at Armstrong Atlantic State University, Savannah GA. She received her Ph.D. in mechanical engineering from the University of Delaware. Her current research interests are structural health monitoring, functionally graded materials and innovative teaching methods in engineering education.

Anthony Flett

Anthony Flett is an undergraduate (sophomore) student pursuing a B.S. degree in Mechanical Engineering through Georgia Institute of Technology's Regional Engineering Program (GTREP) at Armstrong Atlantic State University, Savannah GA. His current interests include automotive systems and components which, after 10 years of working as an automotive technician, was the primary reason he returned to college to advance his career in the industry. His other key interests include hybrid technology and safety systems.

Justin Kriske

Justin Kriske is an undergraduate (junior) student pursuing a B.S. degree in Biology with a minor in Engineering Studies at Armstrong Atlantic State University, Savannah GA. His current interests include environmental conservation, micro- and macroscopic ecology, and alternative energy.

Charles Panter

Charles Panter is an undergraduate (sophomore) student pursuing a B.S. degree in Computer Science at Southern Polytechnic State University, Marietta, GA. He completed this freshmen year at Armstrong Atlantic State University, Savannah GA during which he worked on this project.

Appendix

Screen shots from MatLab Marina: <http://engineering.armstrong.edu/priya/matlabmarina/main.html>.

The screenshot shows the MatLab Marina website in a Windows Internet Explorer browser. The page features a dark red header with the logo "MatLab Marina" and the tagline "Navigating the high seas of programming using MATLAB". Below the header, there is a navigation menu on the left with options: Home, Fundamentals of Programming (with sub-links for Beginning, Intermediate, and Advanced Programming), and Books. The main content area is titled "Overview" and contains text about virtual learning environments (VLEs) and the development of a Virtual Learning Center (VLC) for programming for engineers using MATLAB. It also mentions current courses at Armstrong Atlantic State University.

The screenshot shows the MatLab Marina website displaying a page titled "Iteration: for-loops". The page includes a navigation menu on the left and a main content area with the following sections:

- Execution control: iteration** and **Iteration: while-loops** (both as links).
- The following tutorials show how to write simple for loops.** The corresponding algorithms (flowcharts) are shown alongside the MATLAB script in the tutorials. Note that the tutorials only highlight the specific portion of the script that uses loops. It is assumed that the other commands in the scripts are familiar to the user. The MATLAB programs (m-files) are also available for use (only for learning/educational purposes). To use in MATLAB, right click and save as a *.m file.
- Examples** and **Tutorials** columns:

Examples	Tutorials	MATLAB programs
Write a statement N times		forloop_writeNtimes.m
Build a vector		forloop_buildvector1.m