Teaching Modules for the Technical Skills Component of ABET 2000

Robert Leland[1] and John WIest[2], and Dave Arnold[2],

Abstract

This paper describes part of an effort by Engineering faculty at the University of Alabama to develop three hour instructional modules to teach some of the skills addressed in the ABET EC 2000 Criterion 3 (a)-(k). The goal was to produce modules that could be incorporated into existing classes with minimal preparation time by the instructor. In this paper we describe four modules developed to address the technical skills components: Computational Skills, Design Skills, Modeling Skills, and Problem Solving Skills. Individual faculty members, who worked in teams of three to provide critical review and suggestions, developed the modules during the 1999-2000 and 2000-2001 academic years. During the summer of 2001 these modules were classroom tested on classes of approximately 12 engineering students, representing a diversity of race, gender, major and year at the university. Pre and post module questionnaires were used to assess the effectiveness of the modules, and obtain student feedback, as well as pre and post tests for some modules. Each module includes learning objectives, justification, instructor's manual, homework assignments, and PowerPoint slides for the classes.

Introduction

The ABET EC 2000 criteria require that engineering curricula teach not only the traditional content areas, but also the set of skills described in Criterion 3 (a)-(k) [1]. Many of these skills contribute to a large number of engineering subjects, but are typically not addressed by a specific course. Rather than develop a separate course to address these skills, during the 1999-2000 and 2000-2001 academic year, a team of faculty in the College of Engineering at the University of Alabama developed instructional modules to cover these skills. Each module is designed for three 50-minute class periods, but can be broken up into single classes. The modules are designed to be incorporated into existing classes in any engineering discipline. Research suggests that if skills are taught in a separate class, they do not stick with the students as well as when they are integrated into classes in the discipline [8], [9]. The modules are also designed for minimal preparation time on the part of the instructor. To this end, homework assignments and Powerpoint slides for the lectures are included with each module, as well as an instructor's guide. A justification of the importance of each skill is also included with each module, to help students understand the importance of that skill to their professional practice.

Fifteen modules were developed and classroom tested. These modules were divided into four areas: Technical Skills, Communication Skills, Professional Skills, and Ethical-Social Skills. This paper describes four modules produced to address the ABET criteria for Technical Skills in the areas of Computational Skills, Design Skills, Modeling Skills, and Problem Solving Skills.

[1] Department of Electrical and Computer Engineering, University of Alabama, Box 870286 Tuscaloosa, AL 35487.

[2] Department of Chemical Engineering, University of Alabama, Box 870203 Tuscaloosa, AL 35487.

Procedure

A team of ten engineering faculty members worked over two years to develop a set of fifteen teaching modules that cover key skills from Criterion 3 (a)-(k) that are not specific to any one department. To maintain consistency, the team selected a set of specifications for all the modules to follow. Each module should:

- Fit into one week of classes (three 50 minute classes).
- Be appropriate for all engineering majors.
- Not require any special facilities, except perhaps a projection system with Powerpoint.
- Require a small amount of preparation time by the instructors.
- Fit into upper level (junior or senior) engineering courses.
- Use active or cooperative learning.

The team decided that each module should contain:

- Clear learning objectives.
- A clear justification to show why the student needs to learn the skill being taught.
- Student exercises.
- An instructor's guide discussing how to use the material.
- Powerpoint slides for the lectures.

The modules are available on the web site at www.ece.ua.edu/faculty/rpimmel/public_html/ec2000-modules in PDF format. Powerpoint files are available upon request.

The faculty involved were further grouped into smaller teams of three. These teams reviewed each other's materials and gave feedback. They also reviewed the materials of other teams and provided feedback.

During the summer of 2001, the modules were classroom tested on classes of approximately twelve engineering students, representing a diversity of race, gender, major and year at the university. Each module was taught by a member of the team who was not the developer. For each class, a third member of the team was appointed as an observer. The observer and developer observed the class, and the observer filled out questionnaires to review the class presentation. The students filled out pre and post module questionnaires to assess the effectiveness of the modules, and provide student feedback. Pre and post module questionnaires were also used to obtain feedback from the instructor on the course material and instructors guide for the module.

Descriptions of the Modules

In this section we present descriptions of the Technical Skills modules, and the data obtained to assess their effectiveness with students. Four modules are described here: Computational Skills, Design Skills, Modeling Skills, and Problem Solving Skills.

Computational Skills Module

A three-hour Computational Skills module was developed. This module primarily supports EC 2000 Criterion 3 outcome (k): *"A student should develop an ability to use techniques, skills, and modern engineering tools"* [1], as it prepares the students to use computational tools. It also supports outcome (e) "*Develop an ability to identify, formulate and solve engineering problems*" [1], since students learn to take an engineering problem and develop an algorithm to solve it. This module can be taught in any classroom with Powerpoint projection, and does not require students to have access to a computer. Of course, a computer classroom would permit many valuable student exercises. After this module, each student should be able to:

1. Use Matlab to solve computational problems.

2. Give a step-by-step description of how to compute the solution to an engineering problem.

3. Determine the accuracy of computed results.

Modern engineering problems often require extensive computation. To adequately solve these problems, engineers need to be proficient with computational tools, and be able to precisely define their problems and call in the help of the computer to solve their problem. Matlab was used for these modules due to its simplicity and ease of use, and its widespread use in engineering programs. Each class is structured around one challenge problem.

In class 1, the challenge problem is to find the water level in a tank with a time varying flow rate into the tank. The flow rate is chosen so an analytical solution is not possible. This introduces the class to Matlab, numerical integration, Matlab vector computations, and plotting in Matlab.

In class 2, the challenge problem is to develop an algorithm to analyze the image of a flat component, and determine the surface area of one side. A typical algorithm might be:

1. Divide the image up into little squares.

2. Count the number of little squares that are at least half in the component.

3. Multiply by the area of one little square to obtain the component area.

This problem requires the students to address knowledge representation: 'How do I express the relevant information on a computer?'; algorithmic thinking: 'How do I solve the problem in a step-by-step manner?', and levels of abstraction: 'How do I think about the problem and its solution at varying levels of detail?'

In class 3, the challenge is to numerically solve a nonlinear differential equation. The accuracy of the solution will vary with the size of the time step. Euler's method is used to simplify the presentation, and clarify the role of the time step size. Computational accuracy is seen to decrease over time. This class introduces the students to issues of computational precision.

This module employs active learning exercises, where the students work as individuals, and a few cooperative learning exercises. The Powerpoint slides contain examples of programs in Matlab, which can be explained by the instructor. The module includes a pre-class exercise, which can be assigned as homework before the first class, and assignments for after each of the three classes.

The module was classroom tested for a class of twelve students. The instructor had no prior knowledge of Matlab. The students filled out questionnaires before and after the module to assess the impact of the modules. Students indicated 1 Strongly disagree, 2 Disagree, 3 Undecided, 4 Agree or 5 Strongly Disagree in response to the statement "I am confident that I can …" for each of the three module objectives. A control group of students from classes testing two other modules was also surveyed to rate their confidence in the computational skills objectives.

The percentage of students indicating Agree or Strongly Agree is shown in Figure 1 below. An increase in confidence is shown in each category, except using Matlab to solve computational problems. However the average score in this category increased from 3.29 to 3.67. The decrease/small increase in this category may be due to prior experience of the students with Matlab, or the need for additional classes in Matlab programming.

The instructor for the course had no prior knowledge of Matlab, but nevertheless indicated an average preparation time of 1 hour per class, thus the modules appear to be easy to use.

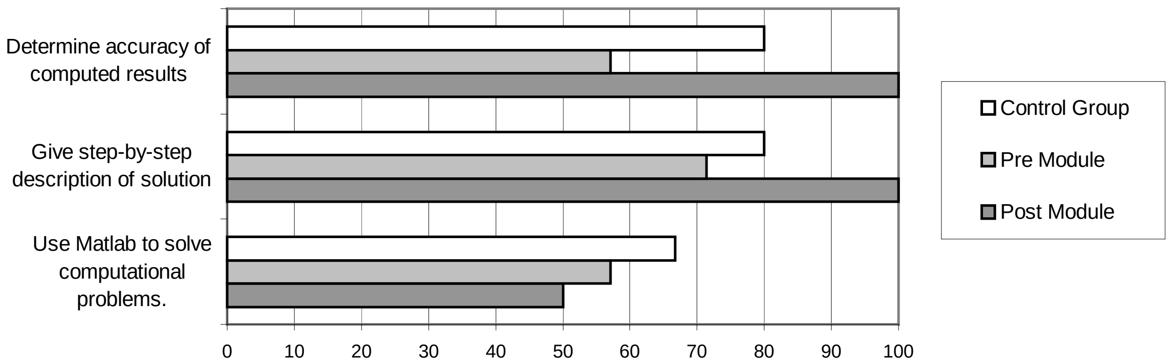

Figure 1. Percentage of Students Indicating Agree or Strongly Agree for Confidence in Computational Skills.

Design Skills Module

A three-hour Design Skills module was developed. This module primarily supports EC 2000 Criterion 3 outcome (c): "*Develop an ability to design a system, component, or process to meet desired needs*" [1]. It also supports Criterion 4:

> *The professional component requirements specify subject areas appropriate to engineering but do not prescribe specific courses. The engineering faculty must assure that the program curriculum devotes adequate attention and time to each component, consistent with the objectives of the program and institution. Students must be prepared for engineering practice through the curriculum culminating in a major design experience based on the knowledge and skills acquired in earlier course work and incorporating engineering standards and realistic constraints that include most of the following considerations: economic; environmental; sustainability; manufacturability; ethical; health and safety; social; and political. The professional component must include: . . . (b) one and one-half years of engineering topics, consisting of engineering sciences and engineering design appropriate to the student's field of study* [1].

This module can be taught in any classroom with Powerpoint projection, and does not require students to have access to a computer. After this module, each student should be able to:

1. Perform preliminary design step
2. Develop time schedule
3. Describe the design process

Each class is structured around one challenge. In class 1, the importance of design for engineers is justified. To get the students thinking about doing design in an intelligent way, the students are asked to write a ten-step design strategy. This work is done in teams, and the student teams present their results. To prepare for the next class on activity schedules, the students are asked to make a schedule for themselves.

In the second period students learn how to prepare activity time schedules. The first five steps in a ten-step design process are explained: 1. Identify a need. 2. Define the problem. 3. Search for information. 4. Identify constraints. 5. Develop criteria for evaluating the design. The students work in teams and choose design problems from a list provided by the module developer. The students must come up with an activity schedule for the tasks in their design.

In class 3, the students prepare design specifications. The last five steps in a ten-step design process are explained: 6. Identify alternatives. 7. Analysis. 8. Choose the best solution. 9. Complete the specifications. 10. Communicate your results.

This module employs active learning exercises, where the students work as individuals, and a few cooperative learning exercises. The module includes a pre-class exercise, which can be assigned as homework before the first class, and assignments for after each of the three classes.

The module was classroom tested for a class of twelve students. The instructor had experience teaching design. The students before and after the module to assess the impact of the modules filled out questionnaires. Students indicated 1 Strongly disagree, 2 Disagree, 3 Undecided, 4 Agree or 5 Strongly Disagree in response to the statement "I am confident that I can …" for each of the three module objectives. A control group of students from classes testing two other modules was also surveyed to rate their confidence in the design skills objectives. The results are shown in Figure 2 below, and show an increase in confidence in design skills after the module, and in comparison to the control group. The instructor for the course indicated an average preparation time of 1 hour per class, thus the modules appear to be easy to use.

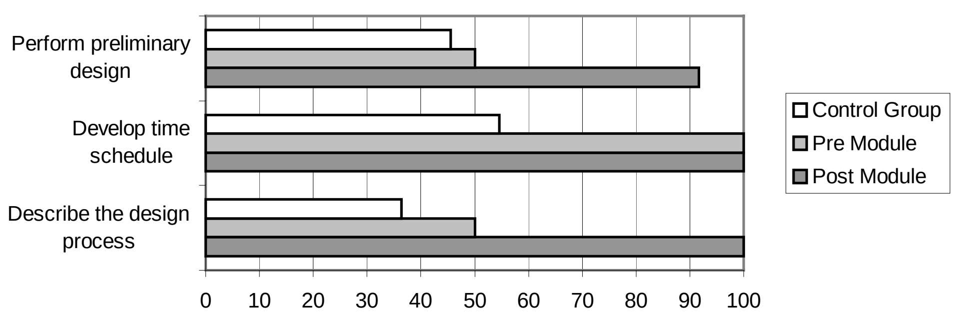

Figure 2. Percentage of Students Indicating Agree or Strongly Agree for Confidence in Design Skills.

Modeling Skills Module

Engineering analysis, design, and optimization commonly require that a process, product, or system be described quantitatively. Therefore, it is essential that engineering graduates be able to express the problems or concepts that they address in mathematical language — as some type of model. Although the general scientific principles that describe a process, product, or system are usually well known and understood, exact mathematical descriptions are frequently too complicated to be of use. Hence, it is necessary that the engineering graduate be able to develop simplified, useful models while simultaneously recognizing the limitations implied by simplifying assumptions.

Furthermore, with the increasing use by practicing engineers of a variety of computational and modeling tools — and as those tools become more complex and sophisticated — it is important that engineering graduates develop both an awareness of the tools and an appreciation for their capabilities and limitations.

In addition, the ABET criteria for many specific engineering disciplines (notably electrical and computer engineering and engineering mechanics) require that graduates be able to use mathematical and computational tools to analyze, model, and design systems [1].

To address these issues, a three-hour Modeling Skills module was developed. The primary objectives of the module are to provide students with the ability to:

1. Express an engineering problem or concept in mathematical language.

2. Identify assumptions invoked in or implied by mathematical formulation.

3. Identify and specify constraints and conditions on the mathematical formulation.

4. Specify and understand the limits of a model.

5. Identify contemporary (discipline specific) modeling tools.

The module assumes that the students have preparation in first semester calculus (with differential equations preferable) and elementary (first semester) physics. In addition, access to spreadsheet and mathematical software (*e.g.*, MathCad, Maple, Matlab, Mathematica) will be helpful but is not required.

The first class period is devoted to definition of modeling, development of a basic strategy for building models, and having the students work in groups to develop a very basic model. The second class period focuses on more complex models and the pitfalls that modeling can involve. The third class period is devoted to 'object oriented' modeling in which models for very complicated systems are built up from models for smaller parts. Group homework assignments are provided for after every class.

The module was tested on a group of eight students of widely varied background. In after-module survey 7 of the 8 students indicated that, as a result of the module, their modeling skills ability had at least doubled, with 4 of the 8 indicating that it had tripled. The percentage of students answering Agree or Strongly Agree for pre and post module confidence levels are shown in Figure 3.

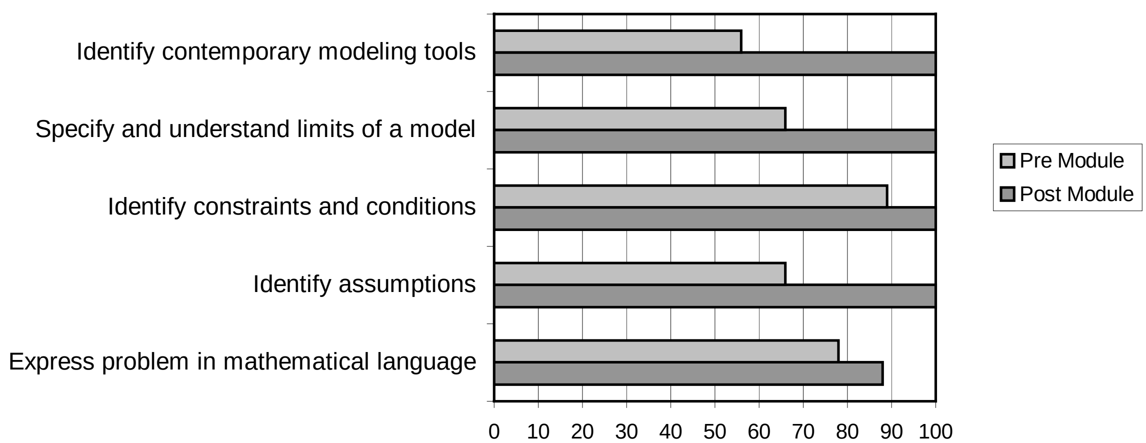

Figure 3. Percentage of Students Indicating Agree or Strongly Agree for Confidence in Modeling Skills.

Problem Solving Skills Module

A three-hour Problem Solving Skills module was developed. This module primarily supports EC 2000 Criterion 3 outcome (e) *"an ability to identify, formulate, and solve engineering problems"* [1]. This module can be taught in any classroom with Powerpoint projection, and does not require students to have access to a computer. Of course, a computer classroom would permit many valuable student exercises. After this module, each student should be able to:

1. Solve routine problems (exercise-solving)

2. Solve novel, out-context, problems where the approach is not obvious

3. Critique problem solving

Good problem solving is the essential engineering skill. In class 1, the students develop problem-solving strategies and learn to brainstorm. In class 2, the students solve simple problems and develop a rubric for problem solving. In class 3, the students work complex problems in teams and critique the problem-solving process.

This module employs active learning exercises, where the students work as individuals, and a few cooperative learning exercises. The module includes a pre-class exercise, which can be assigned as homework before the first class, and assignments for after each of the three classes.

The module was classroom tested for a class of twelve students. The students filled out questionnaires before and after the module to assess the impact of the modules. Students indicated 1 Strongly disagree, 2 Disagree, 3 Undecided, 4 Agree or 5 Strongly Disagree in response to the statement “I am confident that I can …” for each of the three module objectives. A control group of students from classes testing two other modules was also surveyed to rate their confidence in the problem solving skills objectives. The results are shown in Figure 4 below, and show an increase in confidence in problem solving skills after the module, and in comparison to the control group.

The instructor for the course had no prior experience teaching problem solving skills, but nevertheless indicated an average preparation time of 1 hour per class, thus the modules appear to be easy to use.

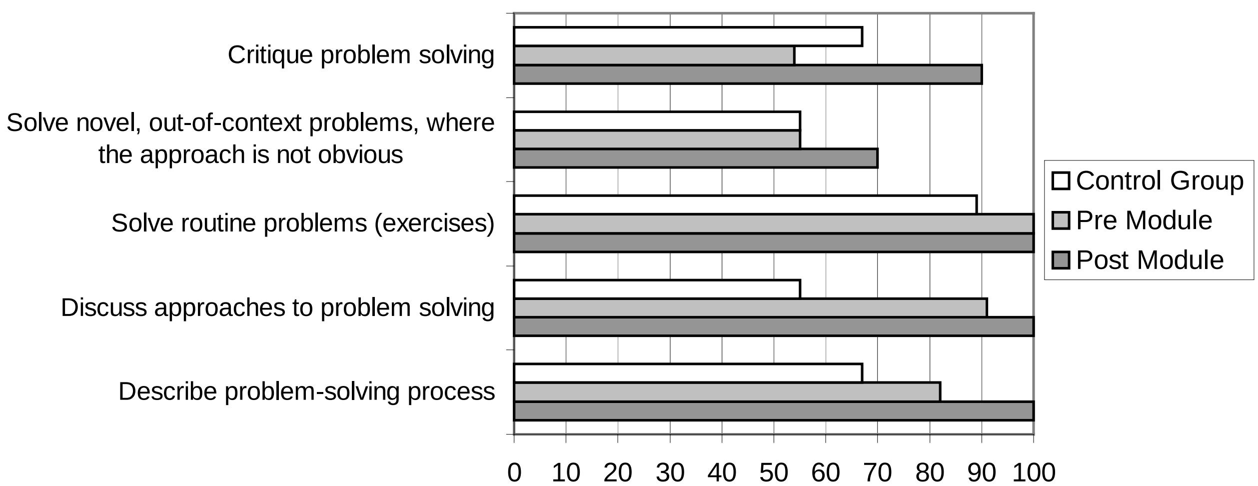

Figure 4. Percentage of Students Indicating Agree or Strongly Agree for Confidence in Problem Solving Skills.

Conclusion

This paper described four instructional modules for teaching Technical Skills to help satisfy the ABET Criterion 3 (a)-(k) outcomes. The modules fit within three standard 50-minute classes. On the average they required about one hour of preparation time for the instructor. Comparing the pre and post-module questionnaire results showed the main effect of the modules was to move almost all of the students into the Agree or Strongly Agree confidence levels. Many of the technical skills modules contained material that is currently taught in the standard curriculum, so the students had considerable prior exposure, which may account for the relatively small changes in student confidence levels for some topics. In the non-technical modules, which treated subjects the students were less familiar with, the increases in confidence levels after the module were often much larger. Although student self-evaluation of confidence levels is very subjective, it does give a strong indication of how much the students benefited from the modules. The content of the modules and the instructors' manuals are available on the web site at www.ece.ua.edu/faculty/rpimmel/public_html/ec2000-modules in PDF format. Powerpoint slides are available upon request.

Acknowledgements

This work was supported by the Engineering Education Program of the National Science Foundation under Award Number EEC-9802942.

References

1. "Engineering Criteria 2000," Accreditation Board for Engineering and Technology, Inc, 1997.

2. Bodner, G.M., (1991) "A View from Chemistry" in Smith, M.U. (ed), *Toward a Uniform Theory of Problem Solving, Views from the Contents Domain*, Lawrence Erlbaum, Associates, Hillsdale, NJ.

3. Fogler, H. S. And LeBlanc, S. E. (1995) *Strategies for Creative Problem Solving,* Prentice Hall, Englewood Cliffs, NJ.

4. C. J. Egelhoff, D. M. Blackketter, J. L. Benson (1999) "Algorithms for Solving Nonlinear Equation Systems Assist Students to Become Better Problem Solvers," *Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference*, p. 12a4-17 – 12a4-22.

5. Lumsdane, E. and Lumsdaine M. (1995) *Creative Problem Solving, Thinking Skills for a Changing World,* McGraw Hill, Inc., New York, NY.

6. J. Ramos, C. Yokomoto (1999) "Making Probabilistic Methods Real, Relevant, and Interesting Using MATLAB," *Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference*, p. 13b4-2 – 13b4-7.

7. P. R. Turner (1999) "Teaching Scientific Computing Through Projects," *Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference*, p. 11b4-1 – 11b4-6.

8. Woods, D. R., et al, "Developing Problem Solving Skills: The McMaster Problem Solving Program," *Journal of Engineering Education*, 86:75-91, 1997.

9. Woods, D. R., et al, "The Future of Engineering Education: Part 3 Developing Critical Skills," *Chemical Engineering Education*, 34:108-117, 2000.