Course Supplement Tools for Enhancing Students' Learning in ECE Freshmen Courses

Zekeriya Aliyazicioglu, Rajan Chandra

Electrical and Computer Engineering Department, California State Polytechnic University, Pomona, CA

Abstract

All freshmen students in our Electrical and Computer engineering take (1) Introductory Circuit Analysis and (2) Introduction to C++ Programming for Engineers. These courses are the gateway courses to both Electrical and Computer Engineering programs. They not only provide the essential problem solving skills and foundation that is needed for the two majors, but also have a great impact on retention for later engineering courses. Research indicates that students often get overwhelmed by these two classes and change his/her major. The goal of this paper is to describe a set of course supplement tools that we have used in these classes to improve the students’ learning and retention. Our freshmen programming class is usually populated with bimodal group of students—some with exceptional programming background while others are totally inexperienced in this area. In order to best utilize the class time and to make both groups of students happy a flipped mode type of instruction is used. Of which when providing students with additional materials, they could timely access before and after class lectures. These information includes PowerPoint slides, review questions, and video clips. For circuit classes there are some simulation tools that would help students to create, visualize, and understand a circuit and its sensitivity to circuit parameters. All of these supplement information are accessible from laptops, tablets, and smartphones at no additional cost to the students. Through surveys, student class presentations, and student interviews it showed evidence that these supplements have enhanced the students’ interest in course materials, problem solving skills, and retention.

Introduction

Students in Electrical and Computer Engineering (ECE) receive instruction in both lecture and laboratory settings. Laboratory exercises offer students an immersive experience which are specifically designed to encourage problem solving skills in a real-world environment. We notice that many of our freshmen students are unprepared for basic courses in circuits and C++ programming. Consequently, they are very frustrated in this kind of setting and thus this situation pushes them to change their majors. We also observed that our students are very much motivated
by practical examples and greatly benefited by lab sessions. This paper presents some of the remedial steps that we have taken in order to increase the retention in both classes.

Problems with introductory circuit analysis and C++ programming classes

Student success in electrical engineering is built on mastery of foundational circuit analysis concepts such as Kirchhoff's laws nodal analysis, Thevenin, and Norton equivalent circuits. However, the course in which these concepts are taught, comes very early in the student's baccalaureate career. Many students at this level have not yet developed sufficient skills such as effective note taking, building conceptual frameworks that integrate new ideas with existing knowledge, and the need to utilize concepts from prerequisite courses.

In our programming classes, two groups of students are encountered- the first group of student those who knows how to write code, and the second group where a student has no background whatsoever of programming. When these two groups are put together in a single class session, the second group students are very much overwhelmed and thus frustrated. Such a situation also poses a great challenge to an instructor with respect to course structuring. In other words, it is very difficult to satisfy this bi-modal student group. Also, the problems solved in a programming class is totally different from which is being used in a traditional courses. In a programming class students must understand basic structures such as loops and functions. Through our observation many students are unable to understand either of these concepts. Further, the first step in solving a programming problem is to conceive a solution and break this solution into a set of smaller sub-problems. Afterwards the student then codes each sub-problem and finally integrate these sub-problems together in order to obtain the final solution to the original problem. Many of our students are confused with respect to this form of problem solving (via making logical decisions).

Our approach

Because of modern technology almost all students have access to a Wi-Fi device such as a tablet, or a smartphone. Also, Wi-Fi connectivity is available for free in common areas such as student center, library and campus coffee shops where students work on their problems. This prompted us to search for a set of web based tools that could supplement traditional materials such as lecture notes and homework problem sets. At this point, we set a goal that the chosen web tools should be of low cost, and it should be accessible from anywhere at any time.

Circuit Course Supplements

Online web tools have been created in the HTML5 format because it is compatible with many different platforms and web browsers. Each example and problem are available in HTML and
PDF formats. At the end of some solutions, we provide a link to watch the solution on YouTube and to simulate the circuit with online circuit simulation tool. The online resources includes a powerful schematic editor equipped also with simulation tools which has a user-friendly interface, making it easily accessible to an aspiring engineering student.

A video solution for each example and practice problems is currently available from a link sending the user to YouTube. We intentionally kept each videos five minutes and less to encourage students to watch the entire presentation. The video presentation is made by a faculty member who taught the course for many times. This kind of environment makes a student feel that the professor is guiding the students to solve the problem for them personally. Our survey has indicated that this is the most popular element in our toolkit. In the fall 2014 the National Instrument’s myDAQ device has been introduced for the first time. The exit survey has indicated that students enjoyed using this device and they also prefer to use this device in the other courses. Also, by including the National Instrument’s myDAQ measurement and instrumentation device to Introductory Circuit Analysis lab in Fall 2014 our survey results given at the end of the quarter shows that the students enjoyed using it and also prefer to using this device in the future courses.

**Programming Course Supplements**

We used blackboard in order to house lecture notes, videos, review questions, quizzes and homework and lab exercises. It has the ability to contain links to video lectures in the YouTube. These lectures are also five to ten minutes long and is given by programming professionals. They are free and can be accessed from anywhere as they are public domain.

During lab sessions students have to work on practical problem for around three hours (some of which as a group), of which are supervised by graduated students. Student are also encouraged to work on a final group project of which the size is limited to two or three students. The students may choose instructor defined projects or may work on their own project which are approved by the lab instructor. Students are required to present the project in the final week of the quarter (before finals) to their peers and they are graded by their peers and the instructor(s).

At the end of the academic year, we expect to finish the following additional supplemental items:

a. A customized website that provides introduction to any topics in the C++ environment with many examples to support the chosen topic.

b. An online tool that will allow students to debug, run, and study the behavior of their codes. This tool will accomplish the goal by accessing a C++ compiler on a remote server or Chromebook.
Performance Assessment and Students’ Feedback

In Fall of 2014, exactly 136 students took “C++ Programming for Engineers” classes in five different sections. Two sections of the classes were taught using the new design supplemental tools. At the end of the quarter a survey was given in these two sections to obtain student feedback. The following questions were asked in the both sections:

1. This class made you interested in programming
2. The PowerPoints are appropriate and helped me understand the subject
3. YouTube examples are helpful to understand the concepts
4. The review sessions were helpful in understanding the materials
5. The lab and lecture complemented each other
6. The quizzes were reasonable in the sense it tested me how well I understood the class materials
7. The online resources presented improved my learning of C++ Programming for Engineers materials and be a helpful reference for future courses.
8. Which topics are difficult to understand and needs more help?
9. What topic did you find most frustrating?
10. Any suggestions for improvement.
11. Would you prefer to take the class without the lab?

Answers for questions 1 through 7, fall into one of the five groups SD = Strongly Disagree; D = Disagree; N = Neutral; A = Agree; SA = Strongly Agree.

The survey results are summarized in Fig.1. It shows that 76 % students a strongly agree that YouTube examples are helpful to understand the concepts. Also, 90% students agree or strongly agree that these class sections made students interested in programming.
Most students found that arrays, pointers, and recursion are the hardest areas to learn. By student’s suggestions, we need more examples, practice programming exercises, and quizzes to be given on a weekly basis (after the topic covered in the class). Also, 85% of students prefer to take the class with a lab. Labs provide the much needed practice to learn and thus helped many students understand the material better.

The grade distribution is given in Fig. 2. There is a small improvement in lower graders. 6.3% of students received less than C- in the non-design sections and 3.8% students received less than C- in course redesign sections. The distribution doesn’t show too much the grades improvement in higher grades. The following reasons may be attributed to this trend:

a. Redesigned class met once a week for three hours. Students were not in favor of such long lecture sessions.

b. Non-redesigned sections are primarily taught by part-timers. Part-timer professors are very lenient when it comes to grading.
We offered five sections of our Basic Circuit Analysis class, the ECE 109 use course redesign supplements. One section used the course redesign supplements with 35 students. The grade distribution is given in Fig.3. There is a big improvement in grades lower side of the grade spectrum. In the non-redesigned section 20% of the students have received less than C. On the other hand, only 3% of the students of the redesigned section have received less than C.

Conclusion

In this paper, a set of supplements for improving students’ performance in introductory circuit analysis and C++ programming are presented. These supplements include blackboard, video lectures, online circuit analysis applications and online applications that allow a student to debug and run their C++ code. All of these tools are freely accessible and these tools pose no restrictions on a student regarding location. Our surveys indicated that students in programming classes prefer short lecture sessions and long lab sessions. Specifically, they are strongly in favor of a terminal group programming project and presenting the results to their peers. In introduction to circuit analysis classes students want to see more examples on difficult topics and many practice problems and automated question bank that will test a student’s understanding of the materials.

References


