Computers & Education 84 (2015) 36–48

Contents lists available at ScienceDirect

Computers & Education

journal homepage: www.elsevier.com/locate/compedu

CALEE: A computer-assisted learning system for embedded OS laboratory exercises

Edward T.-H. Chu*, Chi-Wei Fang

Department of Computer Science and Information Engineering, National Yunlin University of Science and Technology, 123 University Road, Section 3, Douliou, Yunlin 64002, Taiwan, ROC

A R T I C L E I N F O

Article history:
Received 19 September 2014
Received in revised form
15 January 2015
Accepted 20 January 2015
Available online 30 January 2015

Keywords:
Applications in subject areas
Architectures for educational technology system
Cooperative/collaborative learning
Improving classroom teaching
Interactive learning environments

A B S T R A C T

With the popularity of embedded systems and the consequently rising demand in the job market for professionals well-versed in them, embedded operating systems (EOSs) have become one of the core courses in computer science studies across the world. The objective of EOS courses is to develop students' ability to port, modify, and customize an embedded operating system through a series of laboratory exercises. However, our teaching experience has revealed that beginners require a considerable amount of time to familiarize themselves with the development environment and the relevant processes, such as operating in a command line interface, setting environment variables, and kernel configurations. Furthermore, students need to constantly handle compiler error messages, malfunctions of the target EOS, and incompatibility issues related to development tools. These problems may frustrate and discourage students. A common strategy to address this problem is to dedicate more hours to teaching or to hire more teaching assistants to help students progress. However, none of these methods is suitable for institutions with limited resources. Therefore, in this paper, we develop a computer-assisted learning system called the Computer-assisted Learning Environment (CALEE) to assist students with their assignments and thus motivate them. CALEE consists of two parts: a self-learning assistant (SLAT) and a collaborative learning website (CLW). SLAT is a software application that provides a set of useful functions to help students perform EOS laboratory exercises, whereas the collaborative learning website seeks to encourage greater interaction among students. Our experiments show that CALEE expedites learning, improves students' motivation, and reduces the teaching load.

1. Introduction

An embedded operating system (EOS) is a specially designed operating system for embedded computer systems, such as smart phones, robots, industrial control systems, medical electronics, and so on. Unlike general-purpose operating systems, embedded operating systems are designed to be reliable, easy to port, and efficient at resource usage. Due to the popularity of embedded systems and an increasing demand in the job market (104 Job Bank, 2014), EOS has become a core course in computer science (CS) and related areas. The primary objectives of EOS courses include developing students' ability to implement embedded operating systems and motivating them to tackle advanced courses, such as modern multi-core computer architecture and real-time operating systems. As a result, EOS laboratory exercises, such as kernel compilation, kernel module implementation, device driver programming, and operating system porting, play a crucial role in providing students with hands-on experience in EOS courses.

EOS laboratory exercises are traditionally conducted by the lecturer or teacher. For each laboratory exercise, the teacher first prepares slides or notes with detailed instructions for students. All the students need to do is to read and understand the content of the slides to complete each assignment. However, not only does the development of an embedded operating system require knowledge of operating systems, it also requires an understanding of hardware. The learning process may become time-consuming because most students have little experience of system-level programming, e.g., OS porting, hardware configuration, interrupt handling, low-level language

* Corresponding author.
E-mail addresses: edwardchu@yuntech.edu.tw (E.T.-H. Chu), M10117025@yuntech.edu.tw (C.-W. Fang).

http://dx.doi.org/10.1016/j.compedu.2015.01.006
0360-1315/

programming, etc. Considering embedded Linux development as an instance, students should have sufficient knowledge of shell commands, environment variables, cross compilers, and hardware configuration before they can begin compiling their EOS kernels. They also have to handle various errors and warning messages generated by compilers or operating systems. Furthermore, the version of the host operating system and the kernel image can affect the development process and generate more error messages. It is difficult for the teacher to include all the details of each EOS laboratory exercise in the lecture slides or notes. Students have to continually solve the above-mentioned problems all the time, which might demotivate them and slow down their progress. Therefore, designing an effective tool and learning activities to guide and motivate students is an important issue for teachers responsible for EOS labs.

A common strategy to address the above problem is to provide students with detailed instructions for each assignment. However, preparing and maintaining detailed slides for every step of each assignment is extremely time-consuming because each open source component, such as cross compilers, libraries, device drivers, and the host OS, may vary from time to time, let alone different types and configuration of computers in a laboratory. Another common strategy is to increase the duration of classes and laboratory sessions, or to hire more teaching assistants to help students (Salminen, Vanhatupa, & Järvinen, 2011). However, both strategies are impractical for institutions with limited resources. Collaborative learning is another possible strategy to improve students' motivation and help them progress. Several researchers have designed collaborative learning activities for computer science (CS)-related courses. Regueras, Verdú, Verdú, and de Castro (2011) divided students into small groups, engaged them in discussions, and asked them to submit their results to an online forum. Barr and Gunawardena (2012); Tsai, Li, Elston, and Chen (2011) set several collaborative online tasks, such as debate and discussion, for students. The results of their research show that online forums or chat rooms are useful tools to motivate students to complete their homework assignments because they help students feel comfortable in sharing knowledge. However, discussion forums are neither efficient nor practical tools for in-class activities. Students need immediate feedback and assistance when they encounter problems during EOS laboratory exercises. Jong, Lai, Hsia, Lin, and Lu (2013) proposed a collaborative game-based learning activity in a course on operating systems. They divided students into several teams, and members of each team collaborated to interrogate other teams or to answer questions from them. Jong et al. (2013) concluded that the question-answer activities kept students engaged and motivated during the lecture. However, as mentioned above, students are discouraged from completing assignments because they have to address various error and warning messages generated by compilers or operating systems while performing EOS laboratory exercises. Thus, question-answer activities are not effective in motivating students to complete EOS laboratory exercises. Our research problem, then, is to design effective tools and learning activities to motivate students and expedite their learning.

1.1. Our contributions

Due to the high demand for professionals with EOS porting skills in the job market in Taiwan (104 Job Bank, 2014), we design a series of progressive laboratory exercises that build upon one another to provide students with hands-on experience in completing laboratory exercises. Our laboratory exercises focus on EOS porting-related laboratory exercises, such as Android 4.0 kernel compilation, kernel module development, kernel-level debugging tools setup, device driver porting, and boot loader modification. Based on the proposed laboratory exercises, we develop a computer-assisted learning system called Computer-assisted Learning Environment (CALEE) to help students complete lab assignments. CALEE consists of two parts: a self-learning assistant (SLAT) and a collaborative learning website (CLW).

SLAT is a software application that provides a set of useful functions for students in order to save development time when performing EOS lab exercises. For example, SLAT automatically stores shell commands so that students can quickly find the commands they need from the reference command list. If a student repeatedly types in an error command, SLAT will search the command history log and provide a list of recommended commands to the student. SLAT also provides a user-friendly programming environment for students to develop a shell script, which is a series of shell commands. Furthermore, SLAT allows students to post questions or store corresponding solutions on a remote server. Whenever students have difficulty in performing EOS lab exercises, they can search the remote server for the answers. On the other hand, the CLW is a website that aims to increase interaction among students. In our experiments, we grouped three students into each team and two teams into each study group. Every week, we assigned three lab exercises to each team, so that one member is responsible for an individual lab exercise. The three lab exercise had to be completed sequentially and in class. Whenever a team completed a lab exercise, it could download the next exercise from the CLW. Meanwhile, the CLW assigns teams a real-time ranking based on the progress of each team. In order to motivate students to engage in study group discussions, we graded each team according to the time taken by it as well as the time its associated study group spends to complete all assignments.

We applied CALEE to a course on the design and implementation of embedded operating systems (Spring semester, 2014) in the Department of Computer Science and Information Engineering at the National Yunlin University of Science and Technology, Taiwan. Thirty-three students were enrolled in this three-unit course. Every week in class, we first introduced a topic related to embedded operating systems in the first hour. We then made the students practice several assignments in the next two hours of class. We conducted a series of evaluations to assess the effectiveness of CALEE in increasing students' motivation to learn, expediting their learning, and reducing the teaching load. Our results show that CALEE is a useful assisted learning system that motivates students and helps them successfully complete embedded OS laboratory exercises.

2. Related work

A considerable amount of research has been dedicated to designing effective tools and learning activities to guide and motivate students in class. In this section, we first summarize research that focuses on designing embedded operating system labs. We then present research related to computer-aided learning systems for CS courses. Finally, we compare our work with existing collaborative learning strategies.

2.1. Embedded operating system labs

Several studies have been conducted on the design of embedded operating system course curricula. Most of these have focused on developing laboratory exercises or term projects in order to motivate students. Andrus and Nieh (2012); Wang et al. (2010) designed several