A Process for Updating Computer Science Lab Exercises for Non-Majors

David Kaczynski and Melis Öner

Central Michigan University
Department of Computer Science
Mount Pleasant, MI 48859, USA
(kaczy1da, oner1m)@cmich.edu

ABSTRACT

Computers permeate society deeper and more thoroughly as computer science and adjacent fields progress. Students who graduate from institutions of higher education are expected to adapt to and utilize computer technologies, regardless of major. Many universities already offer computer science courses for non-majors, but little research exists on methods to continuously update existing curriculum. We propose a method for updating computer science curriculum for non-majors with IT key qualifications that teach students how to adapt to an evolving technological ecosystem. This framework addresses how to update existing course objectives and materials using feedback from metrics, such as the students' attitudes, performance on examinations, and the time to prepare course materials. Our research finds that universities with existing computer science curriculum for non-majors may be able to utilize the methods outlined in this paper to teach students skills that foster confidence in their relationships with computers in the present and future.

Keywords

computer science education, non-majors, IT key qualifications, curriculum, computers and society

1 INTRODUCTION

Central Michigan University offers a computer science course for non-majors, CPS 100: Computers and Society. The course has an enrollment of several hundred students each year, largely due to being a compulsory course in several degree programs. The course covers a wide range of topics, including:

- history of computer hardware and software,
- societal impacts of computer technology,
- proficiency in navigating the Internet, and
- proficiency in Mirosoft Office and Microsoft Windows software.

CPS 100 is taught with equal parts of lecture and lab periods. In the case of the lab, the exercises have not been updated for several years. This has lead to many lab exercises being inoperable due to outdated instructions for user interfaces for software products, such as Google and Windows. We decided to take this as an opportunity to overhaul the lab exercises with modern computer science objectives for non-majors. The resulting course objectives can be described as applying ACM / IEEE international curricular guidelines for undegraduate programs in computing in the context of IT key qualifications as described by Dörge and Schulte. In other words, the goal of this course is to teach the social and professional issues recommended by IEEE / ACM. These topics are presented through lab exercises that require the students to apply key qualification skills, such as the ability to cooperate, flexibility, creativity, autonomy, and more.

For the purposes of this research, we choose to define a process for updating computer science curriculum for non-majors and measuring the progress of the transition. The process is summarized here:

1. Collect available metrics, such as grade distributions, completion rates, and attitude surveys.
2. Define the scope of the course (history, ethics, software proficiency, etc.).
3. Define two sets of course objectives: key qualifications, and social or professional issues. Define criteria for successful mastery of course objectives.
4. For all existing lab exercises, attempt to create an optimal relationship to course objectives. An optimal relationship is defined by the following two attributes:
    - A lab exercise addresses exactly one key qualification and one social or professional issue.
    - Each key qualification and social or professional issue is addressed by no more than one exercise.
5. Create lab exercises for the remaining course objectives, attempting to create optimal relationships when possible.
6. Implement updated lab exercises.
7. Collect available metrics and analyze the results.
8. Repeat the process, refining the scope, course objectives, and mastery criteria as needed.

The following sections include details of key qualifications and social or professional issues, the benefits of mastering such course objectives, and methods for applying the previousy defined process.

2 BACKGROUND AND LITERATURE REVIEW

In this section, we discuss adjacent fields of research and summaries of what we found to be important points of interest.

2.1 Computer Literacy

2.1.1 Productivity Software

2.1.2 Computer Security

2.2 Computer History

2.3 Computers and Society

2.4 IT Key Qualifications

2.5 Course Design Process