Chapter 2

Assessment of Challenges of Teaching and Learning Computer Programming in Tanzanian Higher Education: The Case of Sokoine University of Agriculture

Baraka M. Kagombe  
Sokoine University of Agriculture, Tanzania

Michael P. J. Mahenge  
Sokoine University of Agriculture, Tanzania

Safari Timothy Mafu  
St. John’s University, Tanzania

Camilius Aloyce Sanga  
Sokoine University of Agriculture, Tanzania

ABSTRACT

This chapter emanates from a study which sought to investigate challenges of teaching and learning computer programming in higher education. The study was conducted at Sokoine University of Agriculture. The study had three specific objectives: first, to identify learners’ prior knowledge on computer programming at the time of joining the university; second, to investigate learners’ self-efficacy in computer programming course; the third objective was to evaluate the learning styles used by learners in the computer programming course. The study adopted a quantitative research method, grounded in experiential learning theory. The data was collected from respondents using questionnaires and the analysis of the data was done using statistical software. The findings indicate that inadequate computer laboratories, lack of competent staff in ICT-based instructional design, inadequate teaching and learning materials, and students’ lack of prior knowledge on computer programming at the time of joining the university are the main challenges.

DOI: 10.4018/978-1-5225-5915-3.ch002
BACKGROUND INFORMATION

Many students in the world grumble about complexity of learning computer programming (Austin, 1987; Duveskog et al., 2003; Tan et al., 2009; Apiola & Tedre, 2012). The fact that computer programming seems to be a difficult course for students is evident when students manifest incompetency in applying knowledge and skills gained from the course (Fang, 2012). This is a universal problem to students pursuing a Bachelor’s degree programme in Computer Science or Information Technology or Information Systems or Informatics, just to mention a few. Earlier studies indicate that students from different universities taking computer programming courses in Tanzania, experience challenges in attaining expected learning outcomes (Duveskog et al., 2003; Apiola et al., 2011; Oroma et al., 2012). Consequently, the majority of students fail to apply knowledge, skills and competencies acquired from the course in real life settings. This shows that the majority of students who have studied computer programming courses in Tanzania lack the required skills and competences. Most of computer programming courses are taught using conventional approaches such as a mixture of lectures, reading and practical sessions (Austin, 1987; Gray et al., 1998). These conventional approaches do not seem to produce competent graduates. Most of the approaches do not encourage interactions between teachers and students, especially in large classes. In principle, teachers have to be knowledgeable and creative enough to adopt teaching styles that would enable learners achieve intended learning outcomes (ILOs) (Darling-Hammond, 2000). Thus, teaching should aim to expose learners to various real life problems in computer programming in order to impart skills, knowledge and competency as per ILOs. Regardless of how teaching is done, literature indicates that teachers should provide classroom practices that lead to significantly facilitation of students to learn (Wenglinsky, 2002). The most important skills that need to be developed among the learners in computer programming include in problem solving and analytical skills (Riley, 1981). However, according to Riley (1981), many students entering higher education have problem-solving skills that are “sadly inadequate”. Problem solving and analytical skills cause difficulty in computer programming (Bonar and Soloway, 1989; Linn and Clancy, 1992). A study by Rist (1996) found that main source of difficulty in computer programming does not seem to be only on the syntax or understanding of concepts, but also how the program is planned, analyzed and designed. According to Gal-Ezer (1996), though a lecture contains an introduction to algorithmic in his or her mind, emphasize should be on practice. This in turn will help the learners to work with any computer programming language. McGill and Violet (1997) stated that most introduction computer programming courses and their respective text books only give emphasis on lower level knowledge and practical knowledge. This means the emphasis is on “know that” and “know how” that are related to computer programming concepts and syntax. This is among the reasons why many students fail to understand and explain semantic actions in a computer program (Henderson, 1986; Tan et al., 2009).

Another problem affecting learners in computer programming courses is how the subject is taught. In most cases, it is taught using conventional approaches (Gray et al., 1998). According to Gage and Berliner (1992) blending lecturing is appropriate in order students attain ILOs. This argument is contrary to what Dalton and Goodrum (1991) argue that computer programming and problem solving strategy instruction are inseparable; when used together may give good means of teaching and manageable problem solving skills. According to Maheshwari (1997), a course in computer programming should be systematically designed in such a way that it provides direct instruction activities; gives out feedback and brings about hands-on practical experiences.