Teaching Advanced Programming Concepts in Introductory Computing Courses: A Constructivism Based Approach

Author:

Kleanthis Thramboulidis, Electrical & Computer Engineering, University of Patras, Patras 26500, Greece, thrambo@ee.upatras.gr

Abstract

Teaching object-oriented programming in introductory computer courses is still an area not well understood by instructors and teachers. A new approach, quite different from the one used to teach the procedural paradigm, is required. We have developed and used for several years a teaching approach that is greatly influenced by constructivism, which stresses the importance of prior knowledge on top of which new knowledge is built. A real-life system was adopted, to exploit the prior knowledge that students have from every-day life. This perspective guided us in making a shift in focus from the algorithm-centered view to the software-engineering-centered view and more precisely to a design-first approach. We also recognized the need imposed by the complexity and the event driven nature of today's applications, for improved techniques and mechanisms concerning exception handling, garbage collection and concurrency. In this paper we describe the constructivism-based approach that we use to teach exception handling and concurrency in introductory computer courses as well as our experience from using this approach. The first results of this approach are very promising. We found that students' conceptions evolved during the course to the point that they were able to confront by the end of the course the requirements imposed by today's demanding applications concerning the issues of exception handling and concurrency. Students found the course extremely challenging and the pass-fail ratio was improved considerably.

Index Terms

Constructivism in education, teaching exception handling, teaching concurrency, semaphore.