Chapter V
Algorithm Education Using Structured Hypermedia

Tomasz Müldner
Acadia University, Canada

Elhadi Shakshuki
Acadia University, Canada

Andreas Kerren
Växjö University, Sweden

ABSTRACT

Understanding of algorithms is one of the most challenging aspects of the study of computer science. Over two decades of research has been devoted to improving techniques to learn and teach algorithms. In this work, we present a new approach for explaining algorithms that aims to overcome various pedagogical limitations of the current visualization systems. The main idea is that, at any given time, a learner is able to focus on a single problem. This problem can be explained, studied, understood, and tested before the learner moves on to study another problem. The structured hypermedia algorithm explanation (SHALEX) system is the system we designed and implemented to explain algorithms at various levels of abstraction. In this system, each abstraction is focused on a single operation from the algorithm using various media, including text and an associated visualization. The explanations are designed to help the user to understand basic properties of the operation represented by this abstraction, for example its invariants. SHALEX allows the user to traverse the graph-based algorithm model, using a top-down (from primitive operations to general operations) approach, a bottom-up approach, or a mix of these two approaches. Since the system is implemented using a client-server architecture, it can be used both through distance education and in the classroom setting. To aid and monitor the learner, we also developed an agent in SHALEX that provides help and monitors the completion rate.
INTRODUCTION

In the ninth century, Mukhammad ibn Musa Al-Khorezmi, the chief mathematician in the academy of sciences in Baghdad, helped establish the Indian numbering system, using decimal notation and the zero. (Al-Khorezmi apparently came from the oasis of Khorazem, at the southern end of the Aral Sea, in what is now Uzbekistan.) When his work reached Europe, the Europeans misspelled the author’s name and called its use “algorism” or “algorithm.”

Nowadays, an algorithm means a specific set of instructions for carrying out a procedure or solving a problem, and algorithms play an intrinsic role in computer science. Because of their importance, researchers have been trying to find the best way to teach and learn algorithms. One of the best known approaches has been to use visualization; especially its understanding as “the power or process of forming a mental image of vision of something not actually present to the sight” (Petre et al., 1998a). The following describes a typical approach used for algorithm visualization:

1. Take the description of the algorithm (usually this description comes in the form of the code in a specific programming language, such as C).
2. Represent data graphically in the code using bars, points, and so forth.
3. Represent the flow of control using animation.
4. Illustrate the animated algorithm and hope that the learner will understand the algorithm.

Step 2 is often automatically generated from the source code, possibly with the help of specifying “interesting events” (Brown & Sedgewick, 1998).

Although, there have been several attempts by many researchers describing the use of animation to software explanation, we will not review these usages. Interested readers are referred to the following two best-known anthologies: Gloor (1992) and Gloor (1998). There are various problems with the above approach. First, providing a graphical representation of an algorithm is just another way to show the code of the algorithm—instead of using a textual programming language, we use a graphical language. Executing the visualization, we simulate the code written in a textual language, using a graphical language and the representation is typically at the low level of abstraction that shows the low level steps.

The work presented herein describes an alternative and systematic procedure to explain algorithms that can be used in various settings, including continuing education. Our approach is based on results of evaluations of existing algorithm visualizations, some findings from cognitive psychology and constructivism theory regarding the active use of algorithm explanation systems, and experience from software engineering and verification regarding the use of multiple levels of abstraction and properties, such as invariants, to explain algorithms and justify their correctness. Based on the experience with design patterns (Gamma et al., 1995), we believe that algorithm explanation should not necessarily be prepared by experts, who have intimate knowledge of the algorithm and the best way of coding it.

In this chapter, we describe a system called structured hypermedia algorithm explanation (SHALEX). This system provides several novel features, such as reflection of the high-level structure of an algorithm rather than low-level steps, and support for programming the algorithm in any procedural programming language. We use a graph-based algorithm abstraction model, in which algorithms may be studied top-down, bottom-up, or using a mix of the two. In this model,
Related Content

Transitioning to E-Learning: Teaching the Teachers
www.igi-global.com/chapter/transiting-learning-teaching-teachers/6243?camid=4v1a

Adapting E-Learning Across Cultural Boundaries: A Framework for Quality Learning, Pedagogy, and Interaction
Catherine McLoughlin (2007). Globalized E-Learning Cultural Challenges (pp. 223-238).
www.igi-global.com/chapter/adapting-learning-across-cultural-boundaries/19303?camid=4v1a

Video Game Genre Affordances for Physics Education
Kostas Anagnostou and Anastasia Pappa (2011). International Journal of Game-Based Learning (pp. 59-74).
www.igi-global.com/article/video-game-genre-affordances-physics/50557?camid=4v1a

Establishing Identification in Virtual Science Museums: Creating Connections and Community
www.igi-global.com/article/establishing-identification-virtual-science-museums/9076?camid=4v1a