Chapter 2
Teaching How to Think Like a Programmer: Emerging Insights

Filiz Kalelioğlu
Baskent University, Turkey

Yasemin Gülbahar
Ankara University, Turkey

Dilek Doğan
Ankara University, Turkey

ABSTRACT

This chapter aims to provide a general description of the preferred pedagogical approaches for the delivery and practice of computer science education based on a review of the literature. Pedagogical approaches mainly used in the teaching of computer science are unplugged activities, robotics programming, block-based or initial programming environments and cross-curricular activities. The preference of these pedagogical approaches varies according to the learners' age and level. Whilst all of these approaches can be used for all ages, some are aimed more at the beginner level than others. The benefits of using each of these approaches will be discussed in this chapter by way of considering educational tips.

A CONCEPTUAL FRAMEWORK FOR COMPUTING

Rapid developments in technology and ease of technology access have provided a level of transformation in education. As a result, digital competencies and skills are now expected of students, as emphasised by the International Society for Technology in Education (ISTE, 2016). ISTE underlines the importance of regular use of certain digital tools in teaching practices which assist students' 21st century knowledge and skills. These standards are as follows:

DOI: 10.4018/978-1-5225-3200-2.ch002

- **Empowered Learner:** Students use technology to take an active role in choosing, achieving and demonstrating competency in their learning process, informed by learning sciences.
- **Digital Citizenship:** Students understand human, cultural and societal issues related to technology and practice legal and ethical behaviour.
- **Knowledge Constructor:** Students organise a variety of resources via digital tools to construct knowledge, produce creative artefacts and make meaningful learning experiences for themselves and others.
- **Innovative Designer:** Students use technological tools for design process to identify and solve problems by creating new, useful and imaginative solutions.
- **Computational Thinker:** Students apply digital tools to develop and employ strategies for problem solving.
- **Creative Communicator:** Students use digital platforms and tools to create original materials or select appropriate platforms for communication.
- **Global Collaborator:** Students use collaborative digital technologies to connect with other learners from a variety of backgrounds and cultures, explore local and global issues and contribute to project teams (ISTE, 2016).

In order to equip students with these competencies and skills, some educational reform must be made and national policies and directions must be changed for future needs. For this purpose, some countries have already started to integrate computing as well as ICT usage into their curricula.

Wing (2008) stated that computing is related to scientific questing, technological innovation and societal demands, and defined computing as the automation of our abstractions that give the audacity and ability to scale. Negroponte (1996) said that computing is no longer about computers, it is about living. According to Jones (2014), computing is a balanced combination of computer science (CS), information technology (IT) and digital literacy. Computer science is one part of computing and includes computation and information (see Figure 1). CS is a practical subject which relates to how computers and computer systems work, and how they are designed and programmed. In addition, IT is concerned with computer systems' purposeful application to solve real-world problems.

Computing is a summary of Science, Technology, Engineering, and Mathematics (Jones, 2014).

- Like mathematics, it has its own theoretical foundations and mathematical underpinnings, and involves the application of logic and reasoning.
- Like science, it embraces measurement and experiment.
- Like engineering, it involves the design, construction, and testing of purposeful artefacts.
- It requires understanding, appreciation, and application of a wide range of technologies.

In addition, Wing (2008) emphasised that the teaching of computing principles is important for computational thinking, rather than just computer programming skills. Computational thinking (CT) is defined by Wing (2006) a such that "it involves solving problems, designing systems and understanding human behaviour, by drawing on the concepts fundamental to computer science" (p. 33). Also it is defined by Computing at Schools - Barefoot (2014) as looking at a problem in a way that a computer can help people to solve it and CT is not thinking about computers or thinking like computers. Computers don't think for themselves. In addition, CT is not programming or a way to solve a problem quickly, or a solution. It does not give a solution and state what a good solution is. In 2011, to move the discus-