Young Children, Mathematics, and Coding: A Low Floor, High Ceiling, Wide Walls Environment

George Gadanidis
Western University, Canada

EXECUTIVE SUMMARY

In recent years, there have been renewed calls for young children to learn to code, using computer programming environments that offer low floor, high ceiling, wide walls coding experiences. That is, students engage with coding with minimal pre-requisite knowledge, have opportunities to explore more complex coding concepts and problems, and can pursue many different interests and for a wide audience. This chapter considers how a low floor, high ceiling, wide walls learning environment may be used to couple coding with mathematics, so as to provide young children both a meaningful context for coding and a rich mathematics learning experience. Using cases from classroom-based research and math and coding apps currently under development, the discussion is organized around two questions: (1) How might we design low floor, high ceiling, wide walls mathematics experiences for young children? and (2) How might coding be used to model mathematics concepts and relationships?

INTRODUCTION

In his book *Mindstorms*, Seymour Papert (1980) proposed Logo as both a coding and a mathematics learning environment:

The idea of "talking mathematics" to a computer can be generalized to a view of learning mathematics in "Mathland"; that is to say, in a context which is to learning mathematics what living in France is to learning French. (p.6)

Papert also saw Logo as a low floor, high ceiling learning environment, where students engage both with coding and mathematics with minimal prerequisite knowledge, yet with opportunities to explore concepts and relationships well beyond the curriculum of their grade level. Although Logo offered a lot of potential, and it was implemented as a mathematics and coding experience in a number of classrooms, it is fair to say that Logo did not gain widespread acceptance, and did not infiltrate mandated curriculum documents.

In recent years, there have been renewed calls for young children to learn to code. As in the initial stages of Logo, some of these calls are coming from research institutions, such as MIT and Carnegie Mellon, which have developed their own low floor, high ceiling programming languages (*Scratch* and *Alice*, respectively). In addition, there have been calls for coding skills from leading technology personalities, such as Bill Gates of Microsoft and Dick Costello of Twitter, as well as from non-profit coding advocacy organizations, such as code.org.

One key difference, compared to Logo's history, has been the serious consideration given by those in education policy and decision-making positions to include coding skills in mandated curriculum documents. For example, starting in the Fall of 2014, the new UK national curriculum mandates that children at all grades will learn to code. The new curriculum "replaces the old ICT programme of study, which focused on computer literacy, with more up–to–date content teaching children how to code, create programmes and understand how a computer works" (UK Government News Release, 4 February 2014). Below are some of the coding expectations for Key Stage 1 (grades 1-2) and Key Stage 2 (grades 3-6) from the new UK curriculum (Berry, 2013, p.6):

- **Key Stage 1:**
  - Understand what algorithms are
  - Create and debug simple programs
  - Use logical reasoning to predict the behaviour of simple programs

Related Content