Constructionism 2018
Constructionism, computational thinking and educational innovation
Vilnius, Lithuania, August 21 to 25

Brian Harvey

University of California, Berkeley
USA

May I Teach an Algorithm?

After all these years, I'm still not sure what Constructionism entails. Here's an example taken from our work on the Beauty and Joy of Computing curriculum: We have a yearlong Tic-Tac-Toe project. Early in the year, students draw the board and use mouse clicks on the board to let two human players alternate

moves. A month later, they start building data structures that will let the program analyze the board. At that time, they check whether either player has won, or whether the game is tied. Two months after that, they return to the project, letting the program be one of the players, and determining the program's move with rules such as "if I can win on this move, do it." A Tic-Tac-Toe program is an obvious project, one of the things many learners do spontaneously. But I had an ulterior motive in the design of the project: I want students to practice using higher order functions (MAP, KEEP, COMBINE). The data structures in the program are built to accommodate that. The question is, because I have this motive and I impose this design, am I being hopelessly instructionist? Is it /my/ project rather than the kid's project? How much flexibility is required for the kid to "own" the project. And, is the kid owning the project what makes it Constructionist?

About speaker

Brian Harvey is a Teaching Professor Emeritus in Computer Science at the University of California, Berkeley. He was formerly a high school computer science teacher, and has volunteered teaching in grades 1-8. He wrote the three-volume 'Computer Science Logo Style' for teenagers, and co-wrote (with Matthew Wright) 'Simply Scheme', a textbook for undergraduates. He is co-developer (with Jens Mönig) of the Snap! visual programming language, an extension of Scratch with first class procedures and first class lists, and co-developer (with Daniel Garcia) of 'The Beauty and Joy of Computing', a CS breadth course for non-computer scientists that uses Snap! He is currently part of a four-year effort to bring BJC to 100 New York City high school teachers, while working with colleagues at Education Development Center on a complete revision of the curriculum to make it more suitable for high school teachers and students.