



Contact: Elizabeth Shea, sheapr@cox.net
703.472.3921

CODE MASTER™ BY THINK FUN

A logic game that gives kids the “coding gene”



Available exclusively at Target in July 2015
Visit: www.ThinkFun.com/CodeMaster
Ages 8+, MSRP: \$19.99

In 2014, ThinkFun brought Kickstarter sensation Robot Turtles to market and introduced the world to the only screen-free game for teaching coding skills to preschoolers. In July of 2015, ThinkFun will once again make its imprint on the code literacy movement with a refreshing new approach to teaching programming concepts.

Introducing Code Master

Designed by computer science educator Mark Engelberg, Code Master is a programming logic game that engages kids ages 8 and up in a Minecraft®-style¹ fantasy to harvest power crystals and continue to the destination portal to a new world – and a new challenge. Unlike other apps, games, and robots that teach coding, Code

Master teaches the mental model of how to “think like a computer,” often a predictor of success in advanced computer science. We like to say it gives kids “the coding gene.”

In Code Master, play is screen-free so the player is fully responsible for taking on the role of the computer. Across a series of 60 fun puzzles of advancing difficulty, players direct an Avatar, use a Guide Scroll to write their program and tap into their logic skills to illuminate their unique path.

Playful and intuitive, Code Master takes you on an entertaining and cleverly-disguised tour of the core control constructs of programming. Colors and maps are used instead of digits and diagrams. The Avatar's world is a finite state machine, and the Map is a state transition diagram. The Guide Scroll is a flow diagram for the program controller. State machines and program controllers may be advanced undergraduate concepts, but kids playing Code Master experience them first-hand and develop an intuitive understanding for these computational models.

¹ Minecraft® is a registered trademark of Mojang Synergies AB.

Code Master™ Backgrounder, page 2

The academic basis of Code Master's design ensures that each puzzle you solve strengthens your ability to think like a computer and visualize the outcome of your coded instructions. In fact, it's the only game to place strong emphasis on clearly visualizing how a program will play out. Code Master does this by turning each programming task into a fun logic puzzle, where the set of available instructions and the overall shape of the program are constrained.

Code Master puzzle challenges start out easy and are accessible to beginning players. As the game progresses, the concepts get harder but the challenges stay fun and players learn without noticing that they are doing so.

The case for the coding gene

Research suggests that one of the key predictors of success in computer science classes is the ability to "think like a computer." This is an ability to mentally step through code, keeping track of state changes, seeing in your mind's eye how a sequence of statements will unfold, understanding what conditional branches will be triggered and knowing when loops will continue or exit. Donald Knuth, a famous computer scientist, once estimated that only 2% of the population had this ability. This leads naturally to the question: why are there so few people with this ability? Is there a "coding gene" or can this ability be taught? At ThinkFun, we believe it *can* be taught – and even better, it can be *taught through play*.

Inventor Mark Engelberg's inspiration

Code Master inventor Mark Engelberg is a game and puzzle designer with background as a computer science educator and Virtual Reality pioneer at NASA. He's been in the ThinkFun family for years, having designed its logic game Chocolate Fix and contributed thousands of challenges to its iconic strategy game Rush Hour. Mark believes that computer science can and should be taught at an early age and put these ideas into practice with his own son, who is now one of the youngest professional software engineers in the world.

In Code Master, Mark's approach boils down to three steps:

1. We **teach** a mental model of how to think like a computer.
2. We **introduce** specific control flow constructs.
3. We **engage** through a fantasy puzzle based on state machines and program controllers.

For a look at what inspired Mark to invent Code Master, see his statement below and [check out his video](#) on ThinkFun.com.

A couple years ago, when my son was taking an Automata class, I saw him working those problems and was reminded how fun it can be to search the maze of pathways for a valid sequence of transitions that get you from the start to the end. Usually there are many such sequences, but seeing this years later, as a puzzle inventor, it occurred to me that it would be a fun puzzle to try to look for a sequence of transitions that satisfied certain criteria. For example, in this diagram, find a string that this machine accepts that uses two 0's, two 1's, and one 2.

It occurred to me that this kind of puzzle could be made accessible, even to young kids, by using colors for the transitions rather than digits. This picture -- the kind of thing you'd see in an automata textbook -- is precisely the level 12 puzzle in Code Master.

Code Master™ Backgrounder, page 3

The crystals in Code Master were inspired by the next level in the automata hierarchy, pushdown automata, which add an element of memory. The way that the crystals are put on the avatar's staff is a surprisingly good metaphor for the way information is added to a pushdown automata's stack.

If I had stopped there, Code Master would already be rich with possibilities (and maybe someday we'll do a version with just that level of complexity in the form of Code Master Jr.), but to make the "full-strength" version,

I needed to move up to the top level of the automata hierarchy. At the top level, there are many, many computational models to choose from that are all equivalent to one another. I drew inspiration from the computational model that you learn when studying other advanced undergraduate CS material such as assembly language or compilers. Thus, the guide scroll was born, a simplified program controller, complete with an instruction pointer and the ability to do conditional branching and loops, which closely corresponds to the way computers actually execute programs. Take a look at: https://mitpress.mit.edu/sicp/full-text/book/book-Z-H-31.html#%_sec_5.1.1, specifically Figures 5.2 and the second part of Figure 5.5. This is what the guide scroll emulates.

One thing I find fascinating is to read forums where people argue over which is the best programming language for children to learn first. There's always a pretty vocal contingent of people wistfully remembering the days where people learned with BASIC. On the one hand, we have far better languages now that allow kids to do much more sophisticated things with less effort. But I understand why a lot of people my age recall their initial exposure to programming through BASIC so fondly -- BASIC transparently exposed the (simplistic) program execution model. Many modern programming languages, even those targeted towards kids and students, have forgotten how valuable that can be.

As I explained above, these days, students often don't receive clear instruction about the nature of program execution until their third year of college. I think that's crazy, and Code Master puts a stake in the ground and asserts that you can learn this material intuitively, up front, through play, and become better prepared to pursue programming. Just from playing Code Master, you're not going to be able to go out and write the world's next great program. But I do think that if you play Code Master, then when you get into a programming class, you're going to be one of the students that just "gets it", and if we can deliver on that, that's a real achievement to be proud of.

###

About ThinkFun, Inc.

Now celebrating its 30th year, ThinkFun is the world's leader in addictively fun games that ignite 21st century thinking skills and give children an early advantage. Its broad range of games, brainteasers and puzzles are breaking new ground in homes and on the web. ThinkFun stimulates crucial brain development in logic, visual/spatial, math, language, focus and reasoning, preparing kids for a world where math, engineering, coding and creativity are of increasing importance. www.ThinkFun.com.

Editor's Note: Visit www.ThinkFun.com/MediaCenter for our latest news.