Programming Microworlds for Elementary School Mathematics

What we've been learning

E Paul Goldenberg, Education Development Center (EDC), epgoldenberg@edc.org
June Mark, Education Development Center (EDC), dspencer@edc.org
Deborah Spencer, Education Development Center (EDC), dspencer@edc.org
Kate Coleman, Education Development Center (EDC), kcoleman@edc.org
Kathryn Chiappinelli, Education Development Center (EDC), kchiappinelli@edc.org
Kristen Reed, Education Development Center (EDC), kreed@edc.org
Zak Kolar, Education Development Center (EDC), zkolar@edc.org

Abstract

This paper shares what we've learned from 6+ years of work in 7 schools (3 districts) with ~700 children using programming microworlds (MWs) for mathematics learning in regular elementary school math classes. We introduced one cohort (~100 children) to our first MWs in grade 2 (age 7) and followed them through grade 5, each year introducing new MWs designed around the mathematics of that grade. At every grade, we also introduced grade-level MWs to classes that had *not* used any MWs in earlier grades, thus testing each MW with both our long-term group and with novices to learn what adjustments might improve accessibility. Our research methodology was direct observation by at least two of our staff along with the teacher(s) in every class, cognitive interviews with selected students at the top, middle, and bottom of typical class performance, and teacher interviews. Our prior MW papers describe our underlying idea as using computer programming as a *language* to help young children express and explore their mathematical ideas, a supplement to natural language and conventional mathematical notation. Here, we focus less on the language aspect and more on the *construction*: *showing* what you mean, building small programs, and seeing the resulting *actions*. We also share our *un*answered questions and how our thinking and MW design—*our* construction—evolved based on observations.

Introduction

This paper shares what we've learned in 6+ years of work in 7 schools (3 districts) with ~700 children using programming microworlds (MWs) for learning in regular elementary school math classes. We introduced one cohort (~100 children) to our first MWs in grade 2 (age 7) and followed them through grade 5, each year introducing new MWs designed around the math of that grade. At each grade, we also introduced our MWs to classes that had not used our MWs, testing with both novices and our long-term to adjust as needed for accessibility. Our methodology was direct observation by two staff in every class, cognitive interviews with selected students at the top, middle, and bottom of typical class performance, and teacher interviews. Prior papers describe our idea as using programming as a language for young children to express and explore mathematical ideas, supplementing natural language (imprecise and ambiguous) and conventional mathematical notation (precise but so concise as to be unforgiving) [3, 4]. Here, we focus not on language but on *construction*.

Our MWs are not special coding pullouts or enrichment, not virtual manipulatives, and not tutoring apps where initiative and evaluation rests in the computer and children merely respond. Using subsets of a powerful language (Snap!) children *program* to explore math *in their class*. The subsets vary, exposing both imperative (sequencing commands, kids' common "coding" experience) and functional programming (constructing expressions by composing functions), list manipulation, filtering a set of numbers, and even using predicates (expressions that evaluate inputs as True or False). Limiting the tools affords access; variety of tools gives rich exposure to both mathematics and programming.

Constructivism and Constructionism as a Foundation

Constructionism is more than academic dressing for the oft-quoted "I hear and I forget; I see and I know; I do and I understand" (incorrectly attributed to Confucius and likely a version of a deeper saying of Xunzi). It's also neither Piaget ("knowledge is derived from action" [6], often taken more extremely as if to say only from action) nor Bruner (learning's trajectory is "Enactive, Iconic, Symbolic" in that order), but it accords with them. Doing does not have to be "tactile" or "concrete." Colors are neither, yet children learn color names (abstractions about abstractions!) early. What makes doing powerful is that what you do, you also see and often describe to others [1]. Doing helps learning by offering children more channels to incorporate information: enactive, iconic and symbolic all at once. More access routes afford more access.

Construct*ion*ism is a constructivist pedagogy, not an epistemology. Artifacts we've constructed can support our other efforts at communication. Communicating ideas helps us more fully develop them as well as share them with others. And a construction leaves traces that an action alone doesn't—it is less ephemeral—and so it is reviewable and rediscussable.

Communication is fragile even when people share common language and experiences. Learners experience teachers' words in personal ways: learners do the interpretation and incorporation; only they build their ideas. As an epistemology, constructivism seems the only choice. It can influence pedagogy: teachers who believe they can't "impart" ideas are more likely (if given a chance) to set up classrooms with a richer soup of experience than verbal explanation and colorful posters.

On Being Usable and Useful in Regular Mathematics Classes

Because our idea was to support mathematics learning, we designed our MWs to be usable and useful *in* regular math classes, not as a supplement, pullout, enrichment, or "fun" day. That imposes constraints. Not only must MWs support the mathematics the teacher expects to teach, but teachers must recognize that at first sight or they will skip it. And it must take no more than a page of introductory reading or a video of no more than three minutes to convince a teacher with no computer background that she can easily introduce and manage it in a classroom of 20 children or so. Finally, it must be plausible that using these MWs doesn't steal time from an already overcrowded curriculum; whatever time this departure from the textbook takes, it is fully repaid in results—perhaps an easier pace through the textbook or even pages that can be skipped. More simply, to be adopted, using the MWs must entail essentially zero extra effort or time.

Our Microworlds and Introducing Them to Children (Ages 7 to 11)

Our MWs all contain three basic elements (Fig 1b): a palette containing blocks children need for their programs and buttons for choosing among explorations; space for the children's programs, a stage on which some visual representation of their mathematical actions can appear; and buttons for specialized actions like **undo** or **reset** or choosing among sets of puzzles.

Introduction and explanation time is overhead. Our research shows that most 7-year-olds who have had no prior experience programming can, in their regular mathematics class and with no more than a ten-minute introduction, mostly active, work independently and successfully using programming blocks in Snap! to express and explore and do the essential elements of the mathematics they would otherwise be doing on paper after teacher explanation with examples.

We've learned a lot about how to keep this introduction short and exploratory, not just another teacher-talk explanation. With 7-year-olds sitting on the rug, the teacher asks what the children see (Fig. 1a) and what they think should be done.

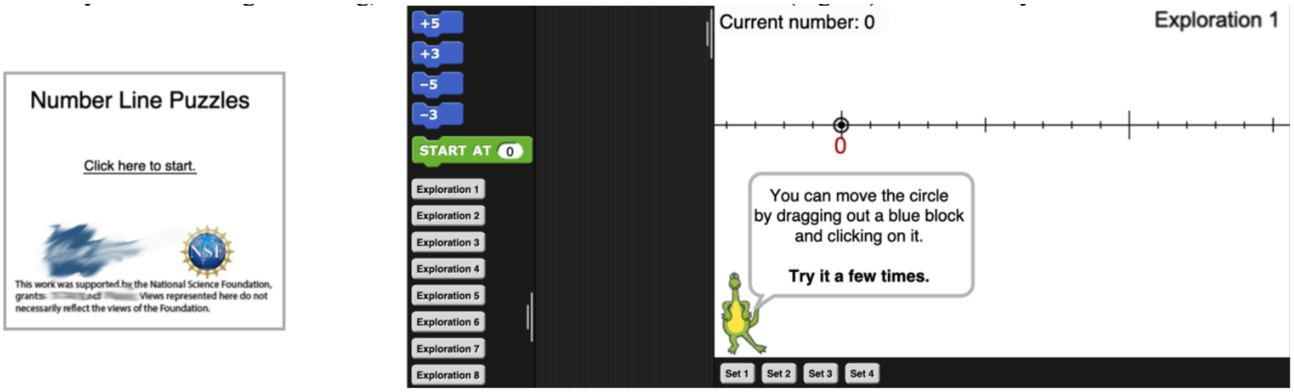


Figure 1: The integer number line MW [described in 7]. (a) Landing screen for integer number line puzzles. (b) The first exploration.

Teacher: What does it say I should do?

Child: Click here to start.

Teacher: Who would like to show how?

In 2017, few kids knew what "click here" meant. Now, no demonstration is needed. In this MW they next see (Fig. 1b).

Teacher: What is Dino telling us?

A child might read it out loud. For those who might need help with reading, clicking on Dino or Dino's words will read them out loud. Few 7-year-olds, even now, know what "drag out a blue block and click on it" means, so the teacher shows.

Teacher: OK, I'll drag out a blue block. (Drags out +5.) Hmm… nothing happened!

If no child tells the teacher, the teacher says "Oh, I didn't click on it!" and invites a child to show how. Clicking `+5` draws an arc that "adds 5," moves the circle to the sum, and labels the new number (Fig. 2). Children can't yet be expected to conclude that it *adds* 5. Perhaps it just *goes to* 5. Children infer meanings from context—in this case, more experiments.

Teacher: Who'd like to drag out a different blue block and try it out?

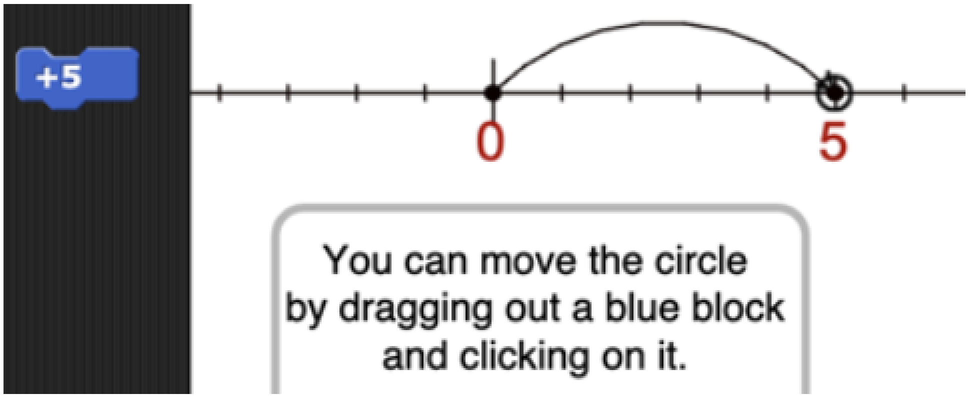


Figure 2: Estela, recently from Guatemala, did not know the + or – symbols or operations. Her Spanish-fluent teacher explained but Estela had no experience to give meaning to the explanation. Playing let her see the effect and build a basis for understanding.

After a few tries, the teacher drags out the green block and asks the class what it says and what they think might happen if they put a different number in place of the 0. In a class of 20, there's always at least one ready to demonstrate.

Teacher: *When you are done with a puzzle and want a new one, click the next button, here.*

The teacher demonstrates by clicking Exploration 2.

Teacher:OK! *Now you know everything you need. You can work on your own.*

Introductions can be brief and interactive for all ages, all programming, and all mathematics—angles, arrays, coordinates, fractions, decimals, attributes of number sets, prime factorization. Children can be independent researchers quickly.

Five design principles we began with

Design principle 1: Treating children like mathematicians—experience before formality. Our EDC mathematics group sees similarities in the way mathematicians work and children learn. Mathematicians often approach new problems by tinkering. Their tinkering isn't random, of course—they know enough to have ideas about what might be productive—but experimenting and looking for pattern and structure is a common start. Then they look for ways to generalize. *Then* formalize results and reduce them to neat presentations that skip over the messy exploratory period and dead ends and make the path look straightforward. Children learning on their own also start with play and build experience from which they generalize. Instruction that skips the development of familiarity with the territory and jumps straight to formality leaves many learners without the experiences on which that formality can be based. Instead, they may simply get rules. Even if the rules are "explained," those explanations need a firm base to rest on or they, themselves, are just more rules. Because learners are ultimately the constructors of the knowledge/understanding, they need a basis in experience.

Design principle 2: Mathematics with "legs." To be accepted by teachers, MWs must immediately be recognized as on topic. But age/grade-level-appropriate content can be taught in ways that serve only the immediate goal or in ways that foreshadow and support future ideas. For example, without using any terminology or formal notation of algebra, arithmetic, can be taught in ways that prepare students for algebra. *Math Workshop* [8] has many elegant examples. Rather than introducing notation just as "this is how it's done," it introduced the idea that the notation meant something and that children could read it *without instruction*, from context, the way they learn *all* language. Where it was relevant, it used the name "pattern indicator" for algebraic (or algebra-like) notation but *nowhere* was there blather about "variables" or about symbols or letters "standing for numbers," or "rules." It was just there and not distracting. (See [2] for detailed description.)

Our MWs use the same principle: restricting focus to grade-level, the *ways* children explore, learn to experiment and search for abstractions foreshadow mathematical ideas through and beyond high school and keep the activity interesting for a wide range of students. We introduce elements that *can* be ignored and are not distracting (like the pattern indicators, or numbers to the left of 0 on the number line as in Figures 1b and 2), but interestingly are generally *not* ignored by children.

Design principle 3: Leaving room for happy accidents. To achieve our goal—use in "ordinary" classrooms with diverse children—our environments must feel accessible to teachers: low overhead, promise of immediate relevance and "pay back" for time invested, and no surprise content that might make a teacher feel obligated to explain things that she is not ready for (or feels her kids are not ready for). Even if the MWs taught no math "content" other than what is called for in class, the fact that children are learning it through active experimentation and evaluating their own results would feel like an achievement to us. But we're greedy. We want more. We want children's exploratory steps and slips and accidents to build ideas on the side or to confirm (or disconfirm) ideas they already have and to allow children to move beyond the experiments we scaffold and explore ideas beyond those we imagine: a mathematical lab (maker space?) that attracts children who were not curious and serves those who are.

Not starting a number line with 0 even for 7-year-olds is part of that. They don't have to pay attention to it, but they can. Over time, we also learned that children need latitude where they type in numbers. A fractions MW doesn't expect children to type decimals. A few do! Children try all kinds of crazy experiments we didn't anticipate and so didn't design for—biggest number is a favorite. So we redesigned. Experiments that have meaning should be honored. In a fractions MW about eighths, a child who types 0.375 should get the right result. We've never seen 7-year-olds use start at to start at 1.5 or 1½, but if they did, the MW should not break or scold them. Only when we cannot act "sensibly" (e.g., for sheer lack of space) does the MW say "I can't do that." The message is that the *machine* can't do it. Perhaps the child can! Children also add extraneous spaces and make other typos. We have redesigned so that if we can infer the intent in otherwise coherent and correctly communicated ideas, we handle them properly even if the form is not yet conventional.

Design principle 4: Students judge their own work. Designing so children can *easily* tell if they've done what they intended serves fidelity to mathematics as well as personal autonomy and confidence. Assessing a solution's correctness must take less work than *finding* the solution and not require redoing the problem or undoing the solution. Solutions should be *actions* children take, *things they build*, programs.

Design principle 5: Cognitive load of non-mathematical elements must be low. Non-mathematical elements add load that competes with the mathematics. Presentation must be clear (unambiguous, not distracting, readable by young children). Word problems have their roles but add challenge that is not always mathematical and should be used thoughtfully. More generally, reading and writing add cognitive load for children, especially the youngest. Mathematical notation and visual representations are concise and precise but readable only after mastering the conventions. Its concision makes it highly demanding and unforgiving. Copying problems (from a book or app) to paper before solving them is also time off task. Drawing is also slow, and both children and teachers vary in the level of precision and neatness they tolerate. Our MWs also require a language—block-based programming—for children to express and explore their mathematical ideas but its overhead seems very low, and learning the language is interactive the way children learn their natural language.

What We Learned From Three of Our Many Types of Microworlds

Four number line microworlds. No puzzles use numbers to the left of 0 in the 7-year-olds' number line (Figs 1, 2). But nearly all children land there by accident or on purpose. Many recognize those numbers (older siblings?) and call "Oooh! Negative numbers!!" Even the ones who don't recognize them aren't put off. The notation looks like subtraction, so children know how to get back to "normal" numbers and often know which number they'll reach. No teaching needed. For some, this is intriguing and accessible enough to attract further play without luring the teacher into unexpected territory.

That early MW foreshadows the idea of linear combinations (of 3 and 5). A similar MW for older children uses ±9 and ±15 to create surprise around the study of common factors which children have practiced *finding*, but with little sense of the mathematical implications and no opportunity to explore or generalize. We haven't yet been able to try MWs that let children *pick* pairs of numbers and explore how their linear combinations do or do not limit what numbers they can reach.

Our first cohort of 7-year-olds asked for capabilities we hadn't offered. After figuring out how to go from 0 to 1 (e.g., using +3, +3, −5) they wanted to create a single *block* that did that, like the blocks we provided. We unhid that option. But many wanted more—*our* blue blocks drew a single arc but theirs still drew three—so we created **combine steps** to take a script that the children wrote and perform the desired arithmetic before drawing an arc for the result. Children, themselves, took the lead in requesting more and better abstractions. We've now incorporated these ideas in some of our puzzles.

Fraction and decimal number line MWs for older children are crafted to look and feel like the integer MW to show that arithmetic with ± and ± or ±0.3 and ±0.5 behaves like arithmetic with ±3 and ±5. Experiments manipulating eighths and seeing results give children experience that makes feel natural, just like three goats plus three goats. This helps dispel the add-everything-in-sight idea that the written notation lures them into, making so many write despite lessons, until they've built their own body of experience and derived a logic from it—another data point supporting experience before formality. Formalisms don't *have* meaning until they are seen as shorthand for things one already understands. Seeing the current number described as when the number is marked as on the number line gives early experience with equivalence. Before trying it in classrooms, we were unsure whether this might confuse or distract children. It doesn't.

Two map and coordinate microworlds. Our Map MW lets 6- and 7-year-olds experiment with shape, direction and distance and with using lengths and distances to solve other problems. The MW shows a map personalized with familiar names and places (Fig. 3a), a Smiley icon, and four blue blocks to move Smiley *north*, *east*, *south*, or *west*. Teachers often re-teach these terms before using the MW, but kids seem learn mostly by *using* them. If a block moves Smiley in an unwanted direction, **undo** lets children fix it. This MW also has a block that lets kids rename buildings; and a **repeat** block that lets them make long moves more easily. (Young children don't care about efficiency, as we discuss later, but they do appreciate having fewer blocks to drag out.) For this MW, we use a slightly longer on-the-rug intro to maps. Long strips of paper represent roads to which kids assign names, and cards mark the locations of buildings the kids can name. Few 7-year-olds know names of local roads other than their own and one or two others, but accurate layout doesn't matter. The key idea is that assigning names to roads lets them *use* those names to specify locations of buildings and directions to them.

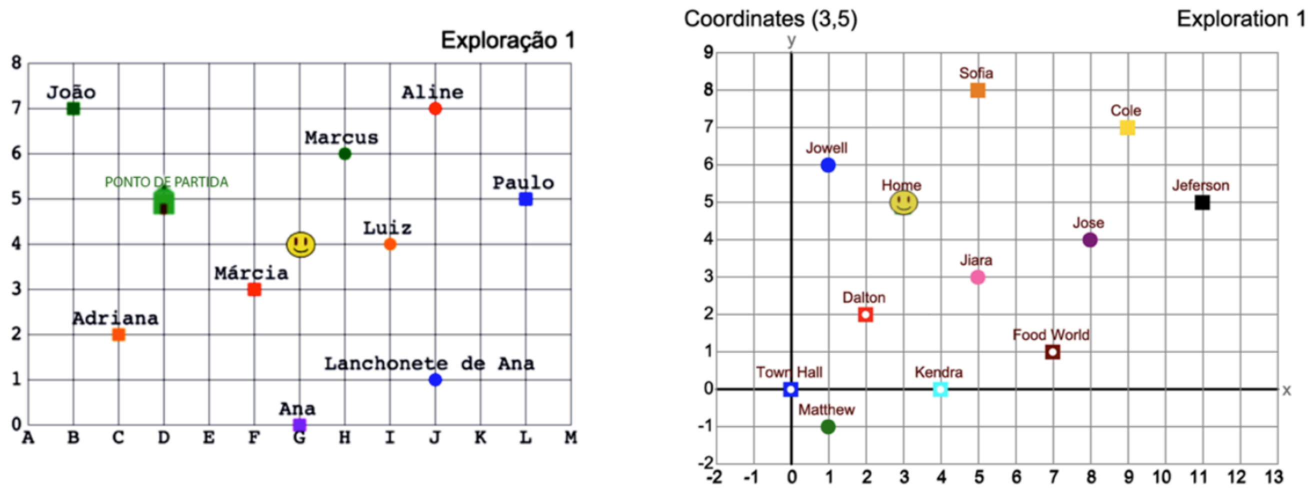


Figure 3: (a) Map of a village (Brazilian version) laid out on a grid showing a luncheonette, a starting point, and homes of 8 kids; (b) The stage image for the Coordinates MW for 9 and 10 year olds.

Children have heard phrases like "the corner of X and Y" or "on X between A and B" so, even though few have used such language and none have heard of coordinates, they *all* (!) readily figure out where `Start at: Letter L Number 5` will place Smiley. Mathematical "legs." And no explanation needed. And if a puzzle says "Start at Adriana," children figure out how to specify her address. Naming objects to help us refer to and manipulate them is a major idea in mathematics.

When coordinates formally enter the curriculum for 10-year-olds, our Coordinates MW lets children *use* them to do things they want, and build experiences that also foreshadow future ideas. The stage layout (Fig. 3b) is like Map, except that it uses numbers for both coordinates, has two darkened lines (axes), and label the top with Smiley's current location.

Several purple blocks let children build, remove, or rename buildings by specifying their coordinates, enticing most into far more practice with coordinates than our puzzles ask for. And asking children to build a building exactly halfway between Cole and Sofia or between Jiara and Jose reminds them that they now *do* know how to specify such numbers. As in the Map MW, we provide blocks for "walking" along streets in the four compass directions. This Coordinates MW adds a new idea: two kinds of direct flight. Starting at Dalton, both 1FLY to point (5, 3)` or `FLT east(x) 3 north(y) 1` will fly Smiley to Jiara. But only the second then flies to a new place—same *direction*, not same end point! Slope won't appear until algebra, and vectors might not appear at all, but the idea of specifying a *direction* to fly comes easily to 10-year-olds. As always, we don't force attention at this age to any sector but the first—some teachers would feel that out of place—nor to any of the formalities of what will later be seen as vector addition, but the experiential seeds are planted.

Properties-of-integers microworlds—visual and active. The currently developed microworld gives children blocks that let them filter integers 0 through 99 by multiples (Fig. 4a), non-multiples, prime, composite, square, >, <, and more, and see the resulting set. They also have a block (Fig. 4b) that lets them combine properties. In both cases, the resulting set is also displayed in bold black on a 0-to-99 chart (Fig. 4c). Children may specify properties not only of the integer but of its digits or their sum or product (e.g., green in Fig. 4b).

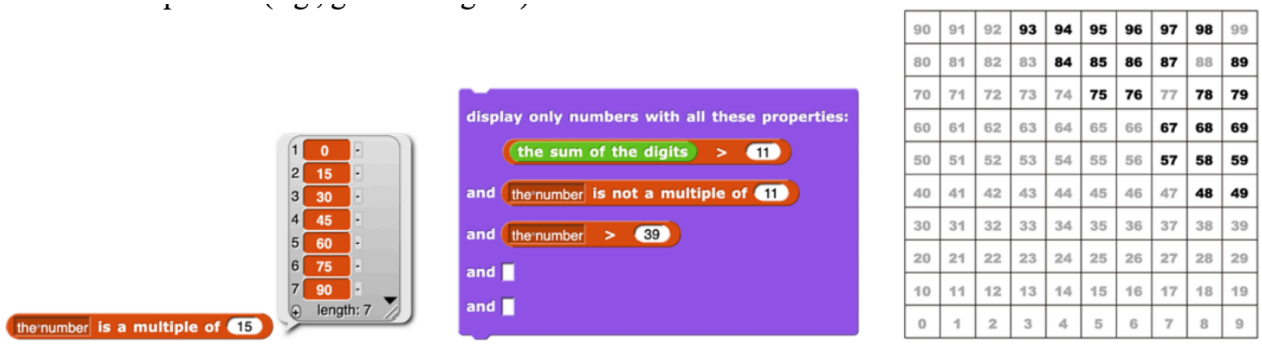


Figure 4: A number set defined by a single property (a). Specifying two properties (b) displays the set shown in 4c.

After children explore the tools, we present the chart with some cells highlighted (Fig. 5). To darken the numbers in those cells, children seek properties of the set. Early puzzles require only a single constraint. Later, easily recognized sets are filtered with a second constraint, e.g., odd and greater than 60 or, e.g., Fig. 5b. And later, challenges like 5c.

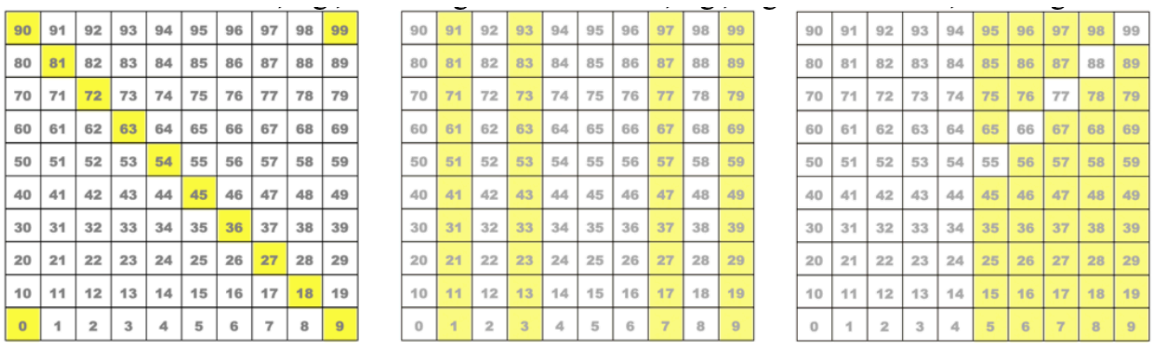


Figure 5: What properties describe these sets of highlighted numbers?

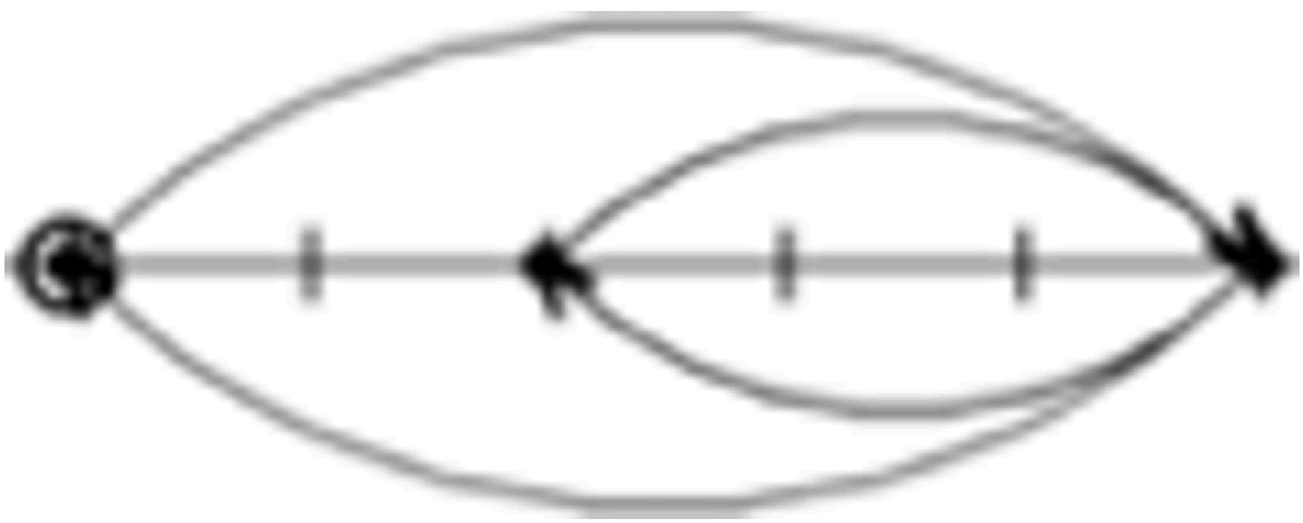
Target topics include multiples, common multiples, or multiples of *n* but not *m*. The exploring is what gives this MW "legs." Sifting through ways to abstract from examples (in this case, numbers) is core to mathematical thinking. We designed this as an advanced MW, but with appropriate puzzles, we now see that it has far earlier entry points. A puzzle that highlights only 5, 15, 25, 35, 45, 55, 65, 75, 85, 95 tempts kids at first to think "easy, multiples of 5" and be surprised that's not enough and be surprised yet again when they see the varied solutions they and their classmates find.

Design changes derived from our work with the children

We structure puzzles carefully to build on each other. Some kids use that order, some skip around. Originally, we built no paper component for our MWs but, inspired by children's happy chirp "I did it!" when they finished a good puzzle—in all grades in all schools as if it were built in—we added a sheet with three columns: puzzle name, "I did it!", and "I showed someone" for kids to track their work. Some didn't care. But some liked the paper element, familiar but non-intrusive.

We knew at the start that accessibility required brief, clear language, voice alternatives to text, and non-English versions (all MWs are now in multiple languages, and we add others as soon as we get translations suited for kids in school). Over time, we discovered other needs. Our Angles MW for grade 4 required an **undo** button. When we saw how the 9-year-olds used it for systematic trial and debugging, we started introducing it in earlier MWs where the need hadn't been so obvious.

Many of our ideas about what makes a good puzzle were on target, but some weren't. We asked 7-year-olds to analyze number line scripts to remove unneeded steps (e.g., cutting two steps from the script +3,+3,+5,−3 still adds 8). They can, but few find it interesting. They love getting a desired result but don't care to get it more efficiently. In fact, they prefer complexity, the bigger the better. In trying, say, to move from 0 to 4, many wandered all over, ignoring loops in their path—loops that signify jumps that can be removed. Nearly all 7-year-olds love the task of visiting every marked number on the line. Some rework the pattern of jumps to look pretty. This esthetic pull began influencing our design of all MWs.



Kids didn't care about loops, but we did, to foreshadow arithmetic inverses. Children's responses in the Angles MW nudged us toward visual puzzles even in contexts like the number line. The need to limit words also nudged us that way. We created visual puzzles like Figure 6 which *require* concision and focus on inverse steps. Children did like these. Whether they also hatched an inkling of arithmetic inverses is not yet clear. But their ease in solving the puzzles showed their actions to be systematic. Visually presented puzzles became the core of the Number-Properties MWs (e.g., Fig. 5).

Some elements that we added "just for fun" turned out to increase "on-task" play even as children strayed from our puzzles. We first built `put name on building at (0,0)` into Coordinates to allow personalization, but children loved adding friends' names, leading to spontaneous use of coordinates beyond our "assigned" tasks. Similarly, color choice in some worlds was originally just "decorative" until we saw it leading to more spontaneous mathematical actions than we could include. Playful or esthetic elements, well-chosen and designed, can increase on-task behavior rather than distract from it.

Recognizing when a solution to one problem can be applied to another is key to both mathematical and computational thinking, so we built puzzles that asked kids to solve a puzzle and *keep* the script, either to solve the puzzle a different way or to use (with or without modification) in another puzzle. No interest! Kids often didn't even notice the call to save the script. If they did save it, they built a new one for the new puzzle anyway. Classroom discussion *does* serve this goal—kids love to show their work—but that must be left to good teaching, not MW design. Just having the puzzle isn't enough.

Another puzzle type that failed aimed at proof. On the integer number line, we asked, "Is it possible to get from 0 to 1 using only two blue blocks?" The most concise way uses three: +3, +3, −5. Most children decided that they couldn't do it with two but then asked for help because they "can't do it." A few took the question literally as we intended and concluded "no" without giving reasons. Some did give valid proofs-by-exhaustion, but stunningly remained unconvinced until one of us agreed—the opposite of our aim to make sure they can always evaluate their own results. The problem was not that seven-year-olds can't reason about or understand proof. The problem was that our question was not interpreted as intended. Young kids expect, perhaps from experience, that tasks they're given *can* be done. This had two consequences. One, of course, was many defeats, kids who just felt they're "not good at math." Another was to reinterpret our question: "I did it! I used only two blue blocks, +3 and −5, but I used one of them twice!" or "I used repeat! It isn't blue" At age 7, it seems that many kids have the *logic* for (informal) proof but don't understand requests for proof. Even at 11, many children reinterpret "Is it possible?" to mean "Can *you* do it?" When we've modeled tackling impossible tasks and giving (informal) proof, children have felt permitted to declare a task unsolvable, but we can't lightly hand that kind of teaching off to others.

We used to have puzzles that involved writing or drawing as part of the response: e.g., "what numbers can you label using only the +3 and −3 blocks" or "find a path…and draw it on your page." We abandoned those for puzzles whose solution was only the script the child builds, not something to put on paper, scrapping some puzzle types we'd like to have had. "Compare these two scripts" can prompt valuable thinking but is better in class discussion than in puzzle design.

Important Unanswered Questions

A major unknown at this point is how to support teachers. We've learned how to design for face validity and appeal to teachers but haven't yet learned how to optimize *supports* to let teachers pick up these MWs on their own and use them independently and effectively. Guides? Professional development? Video? Supplementary materials for children? This is essential for the ultimate success—and ultimate test—of the use of MWs and exploratory learning.

Assessment is another issue. We see intense engagement of all (!) students and convincing examples of mathematical thinking in the MWs. It is important both to quantify that and to be able to make claims about school performance. The latter requires assessing with measures schools use. The results shouldn't be expected to match. Children behave differently in different media. We chose our medium, MWs, precisely because it does change accessibility, especially for those who otherwise perform poorly and seem disengaged. But if we haven't helped kids do better in school (i.e., on school tests), we may not have succeeded. We don't know. Our work wasn't funded for comparison or efficacy studies. That is another step.

Weirdly, we can't say *why* our children are so engaged. Of course, our conjectures are the ones that please us: freedom to experiment without risk, the excited feeling of one's mind at work, and success. "I did it!" But they're conjectures. We don't *know*. We do know that any observer can see that the engagement is real; satisfying but not enough. How much is Hawthorne effect? Some, perhaps, but not all; puzzles varied in their ability to generate thought and interest. More to learn.

Though we *design* for a specific grade, it puzzles us to see almost identical variety in engagement, rate and success both a grade earlier and a grade later. The MWs seem equally useful as introduction, investigation, and review. Because nobody engages when things are opaquely beyond them or boringly beneath them, we assume that "learning trajectories" are not nearly as pure as reported. Kids *experiment* in the MWs, so their experience varies with their knowledge and understanding. The comparable performance may mean that kids get *different* things at these different times. This, like the other areas, needs research. We've offered exemplars, ideas, observations, and questions. We've also documented technical details [5]—the programming behind MWs—so others can use our MWs or build their own for classrooms or research.

ACKNOWLEDGMENTS

This work was funded by the National Science Foundation grants Nos. 1741792 and 1934161.

References

1. E. Paul Goldenberg. 1979. Special Technology for Special Children. Baltimore: University Park Press.
2. E. Paul Goldenberg. 2019. Problem posing and creativity in elementary-school mathematics. Constructivist Foundations 14(3), 601–613. https://constructivist.info/14/3/319.goldenberg.pdf
3. E. Paul Goldenberg and Cynthia J. Carter. 29 May 2021. Programming as a language for young children to express and explore mathematics in school. Brit. J. Ed. Tech. https://bera-journals.onlinelibrary.wiley.com/doi/epdf/10.1111/bjet.13080
4. E. Paul Goldenberg, Cynthia J. Carter, June Mark, Kristen Reed, Deborah Spencer, and Kate Coleman. 2021. Programming as language and manipulative for second grade mathematics. Digital Experiences in Math. Educ. 7, 48-65 https://doi.org/10.1007/s40751-020-00083-3
5. Zak Kolar, Hannah Moser, and E. Paul Goldenberg. 2023. Building a Microworld in Snap! https://snap.berkeley.edu/static/doc/building-a-Snap-MW
6. June Mark, J., Paul Goldenberg, Jane Kang, Mary Fries, and Tracy Cordner. 2014. Transition to Algebra. Portsmouth, NH: Heinemann
7. Jean Piaget. 1971. Science of Education and the Psychology of the Child. NY: Viking Press.
8. Deborah Spencer, June Mark, Kristen Reed, Paul Goldenberg, Kate Coleman, Kathryn Chiappinelli, and Zachary Kolar. Using Programming to Express Mathematical Ideas. MTLT 116(5): 322-329 https://doi.org/10.5951/MTLT.2022.0354
9. https://solveme.edc.org/. https://SolveMe.edc.org
10. Robert Wirtz., Morton Botel, Max Beberman and W. W. Sawyer. 1964. Math Workshop. Encyclopaedia Britannica Press, Chicago IL.