*Title Page with Author details

Assessing Implicit Computational Thinking in *Zoombinis* Puzzle Gameplay

Elizabeth Rowe[1*], Ma. Victoria Almeda[1], Jodi Asbell-Clarke[1], Richard Scruggs[2], Ryan Baker[2], Erin Bardar[1], Santiago Gasca[1]

[1] Educational Gaming Environments Group (EdGE), TERC, USA
[2] University of Pennsylvania, USA

*Corresponding author. Educational Gaming Environments (EdGE) Group, TERC, Cambridge, MA 02140, USA.

E-mail addresses: elizabeth_rowe@terc.edu(E. Rowe), mia_almeda@terc.edu (M. Almeda), jodi_asbell-clarke@terc (J. Asbell-Clarke), rscr@gse.upenn.edu (R. Scruggs) ryanshaunbaker@gmail.com (R. Baker), erin_bardar@terc.edu (E. Bardar) santiago_gasca@terc.edu (S. Gasca).

1 INTRODUCTION

Computational Thinking (CT) is an emerging field in K–12 education that focuses on problem-solving practices related to computer-driven systems. CT practices include problem decomposition, pattern recognition, abstraction, and algorithm design (Authors, 2017d). The rapid rise of interest in CT in K-12 education is calling for new models of pedagogy, instruction, and assessment (National Academy of Sciences, 2010).

While an important application of these practices is in the development of computer programs (coding), CT can be applied to a broader range of problems that do not involve coding. CT is a way of thinking, or set of habits of mind, that offer a specific approach to problem solving. Computational thinkers see patterns in types of problems, and are able to abstract those patterns to generalized groups so that they can re-use and modify algorithms from previous problems. CT practices are used in everyday activities involving structure, sequencing, and ordered procedures such as recipes for cooking, assembly instructions, or even daily routines to structure the school day.

As educators become interested in the teaching and learning of CT, the challenge of designing and validating learning assessments for CT arises. Measuring CT requires measuring learners' abilities to plan, design, and solve complex problems, which is not done by a typical school test (Ritchhart, Church, & Morrison, 2011). Even when CT is assessed in a natural setting, such as in a when artifacts from coding activity are reviewed, the learning assessment may not reveal the CT practices as a problem-solving approach as much as a review of how the student crafted the code (Grover & Basu, 2017).

The development of novel forms of assessment of CT may be particularly important to broadening participation in Computer Science and other Science, Technology, Engineering, and

Mathematics (STEM) fields. Many learners who are considered "learning disabled" have demonstrated particular areas of strength in tasks related to CT, such as pattern recognition and systematic thinking (Baron-Cohen, Ashwin, Ashwin, Tavassoli & Chakrabarti, 2009; Dawson, Soulières, Gernsbacher, & Mottron, 2007; O'Leary, Rusch, & Guastello, 1991). Many IT companies such as Microsoft and Google recognize and nurture the unique cognitive assets of neurodiverse learners, knowing they may be vital to our future workforce (Martinuzzi & Krumay, 2013; Wang, 2014). Unfortunately, however, current educational assessments often include irrelevant barriers (e.g., reading or coding prerequisites) that may mask conceptual understanding for some learners (Haladyna & Downing, 2004).

A promising avenue for building more inclusive assessments of CT practices may lie in digital games. There is increasing evidence that games can play a significant role in promoting STEM learning for children and young adults (Steinkuehler & Duncan, 2008; Authors, 2015; Squire, 2011). Digital games can also be used as powerful tools for providing formative, stealth assessments, that are able to measure implicit learning in a natural and engaging environment for learners (Shute, Ventura, & Kim, 2013; Authors, 2015).

This paper describes an emergent approach to developing game-based assessments of students' computational thinking in the puzzle game, *Zoombinis*. Using the digital log data generated through gameplay, researchers designed automated detectors of CT practices in gameplay based on theoretical and empirical grounding. The process of designing the detectors and results on their confidence of predicting those CT practices in *Zoombinis* gameplay are reported on here.

2. BACKGROUND ON COMPUTATIONAL THINKING ASSESSMENTS

CT is the way of thinking used to design systematic and replicable ways to solve problems, and includes specific problem-solving practices such as Problem Decomposition, Pattern Recognition, Abstraction, and Algorithmic Thinking (Authors, 2017; CSTA, 2017; Wing, 2006). These CT practices rely upon and may build facility with logic, representations, and sequential thinking, as well as broader ways of thinking such as tolerance for ambiguity, persistence in problem solving, and abstraction across applications (Allan et al., 2010; Barr & Stephenson, 2011; Brennan & Resnick, 2012; Grover & Pea, 2013; Weintrop et al., 2016). CT is also seen to be related to creativity and innovation (Mishra, Yadav, & the Deep-Play Research Group, 2013; Repenning et al., 2015) as well as integrating into many STEM areas (Barr & Stephenson, 2011; Sengupta, Kinnebrew, Basu, Biswas, & Clark, 2013; Weintrop et al., 2016). Furthermore, CT may foster particular dispositions in K-12 education, such as confidence and persistence, when confronting particular problems (Barr and Stephenson, 2011).

The development of CT learning assessments in K-12 is still a relatively young endeavor. Many current assessments used in K-12 are strongly tied to computer-science frameworks as opposed to focusing on CT and most assessments analyze products from, or ask questions about, the construction or analysis of coding artifacts. These include assessments such as the Fairy Assessment (Werner, Denner, Campe, & Kawamoto, 2012), Dr. Scratch (Moreno-León & Robles, 2015), Ninja Code Village (Ota, Morimoto, & Kato, 2016), REACT (Real Time Evaluation and Assessment of Computational Thinking) (Koh, Basawapatna, Nickerson, & Repenning, 2014), CodeMaster (von Wangenheim, et al., 2018) and tools developed by Grover, Cooper, and Pea (2014), which are all designed for specific programming environments like Alice, Scratch, AgentSheets, App Inventor, Snap!, or Blockly. The reliance on coding may prevent these assessments from measuring CT with learners who do not have sufficient coding

experience. As such, these tools may not be well-suited for use as pre-assessments or for use with interventions that are not primarily focused on coding (Wiebe, London, Aksit, Mott, Boyer, & Lester, 2019).

A few recent assessment endeavors have removed some of the reliance on coding. The Computational Thinking test (CTt) (González, 2015) is an online, 28-item, multiple choice instrument that shows promise in assessing core CT constructs for middle-grades students through situational questions rather than coding (Wiebe et al., 2019). The CTt was shown to be valid and reliable with middle-school students in Spain (Román-González, Moreno-León, & Robles, 2017). Some items on the CTt have block-based, programming-like elements in them, but this has shown not to be problematic for students who reported having little or no prior programming experience (Wiebe et al., 2019). This result is supported by Weintrop, Killen, Munzar, and Franke (2019), who found that students perform better on questions presented in block-based form compared to text-based questions.

Bebras Tasks (Dagienė & Futschek, 2008; Dagienė, Stupurienė, & Vinikienė, 2016) originated as a set of short competition tasks through which students in grades 5–12 apply CT to solve "real life" problems. The Bebras tasks have recently been studied as assessment tools of CT practices (Barendsen et. al., 2015; Dagienė, Stupurienė, & Vinikien, 2016; Izu, Mirolo, Settle, Mannila, & Stupurienė, 2017). Like the CTt, many Bebras Tasks do not rely on prior knowledge of an application or programming language, but the psychometric properties of Bebras Tasks have not been fully demonstrated and some tasks may be considered too peripheral to core CT skills (Román-González, Moreno-León, & Robles, 2017). Wiebe and colleagues (2019) explored a hybrid of CTt and Bebras as a "lean" assessment of CT practices, and found

promising results. The Bebras items are, however, dependent on textual questions and scenarios which may present a barrier for neurodiverse learners.

3. AN EMERGENT APPROACH TO GAME-BASED ASSESSMENT OF IMPLICIT LEARNING

In an effort to build learning assessment of CT practices that are inclusive for a wide range of learners, we look to a model of *implicit learning*—foundational knowledge that may be demonstrated through everyday activities but may not be expressed explicitly by the learner on a test or in schoolwork (Authors, 2015). There is ample research showing that traditional IQ tests and academic exams do not measure all of the cognitive abilities required in many everyday activities (Sternberg, 1996), and a large body of previous literature illustrates the implicit mathematical abilities in studies of gamblers at the race track (Ceci & Liker, 1986); street children using early algebra skills in their vending of fruit and snacks (Nunes, Schliemann, & Carraher, 1993) and housewives calculating "best buys" at the supermarket (Lave, Murtaugh, & de la Roche, 1984). Learners may demonstrate implicit knowledge through behaviors in everyday activities, such as games, that they are not yet able to express formally (Polanyi, 1966; Ginsburg Lee, & Boyd, 2008).

Learning assessments used in more current educational research typically attempt to measure explicit knowledge, and are often laden with terminology and formalisms that may present barriers to learners' expression of their underlying knowledge (Arena & Schwartz, 2013). The assessment of implicit knowledge proves inherent difficulty for researchers because it is, by definition, unexpressed and thus cannot be measured through traditional pen and paper tests or possibly even clinical interviews (Reber, 1989). Self-contained or decontextualized tests do not call upon this type of previous knowledge or experience of learners to support new learning

(Arena and Schwartz, 2013). Well-designed games provide an opportunity to support and measure implicit learning.

Game-based learning assessments (GBLA) show promise to assess implicit knowledge by avoiding jargon within test items, construct-irrelevant material, and test anxiety, all of which can make traditional assessments less effective at assessing student competency (Authors, 2015, 2017c; Shute, 2011). GBLA research is often grounded in an Evidence-Centered Design (ECD) framework (Mislevy, Steinberg, & Almond, 2003). ECD seeks to establish a logically coherent, evidence-based argument between three important models: a **competency model**, which involves variables about targeted cognitive constructs; a **task model**, which includes activities that support students' demonstration of these cognitive constructs; and an **evidence model**, which provides the rationale and specifications of how to identify and evaluate targeted cognitive constructs (Grover et al., 2017).

Researchers have developed stealth assessments guided by the ECD framework using educational data mining techniques to discern evidence of learning from the vast amount of click data generated by online science games and virtual environments such as *Progenitor X* (Halverson, Wills & Owens, 2012), *EcoMUVE* (Baker & Clarke-Midura, 2013), *Physics Playground* (Shute et al., 2013, 2016), *INQ-ITS* (Li, Gobert, & Dickler, 2017; Li et al., 2018), *Shadowspect* (Kim & Rosenheck, 2018), *Earthquake Build* (Lee, 2016), and *Surge* (Clark, Nelson, Chang, D'Angelo, Slack, & Martinez-Garza, 2011). Within ECD, measures of learning must be considered and designed along with the game mechanics. While ECD approaches have been applied to the assessment of CT and CS (SRI International, 2013; Tissenbaum et al., 2018), no one to the best of our knowledge has studied a puzzle game as an assessment of CT.

REDACTED seeks to remain as open to emergent evidence of implicit learning in games while still pursuing the logical coherence of the ECD framework. We have used our methods in digital games using an emergent method of stealth assessment with a more naturalistic and bottom-up approach by designating our task model as the predefined activities that elicit implicit learning within the game activities (Authors, 2017a, 2019). We then observe strategies that players demonstrate as they play the game and we identify those that are consistent with learning constructs of interest, such as CT. This approach for building evidence of learning is a modification of most applications of the ECD framework, where explicit learning outcomes are defined in advance and assessment tasks stem from those outcomes (Mislevy & Hartel, 2006).

To study implicit CT in *Zoombinis*, we leverage methods from the field of educational data mining, which offers unique opportunities for providing scalable, replicable measures of implicit learning in games (Authors, 2015, 2017a; Baker & Clarke-Midura, 2013; Martin, Petrick, Forsgren, Aghababyan, Janisiewicz, & Baker, 2015; Shute et al., 2010; Hicks et al., 2016; Li et al., 2018). This paper describes the use educational data mining techniques to build automated detectors of CT practices using game log data in a popular CT learning gamed called *Zoombinis* as evidence of their implicit learning.

The central questions addressed by this research are: 1.) What indicators of implicit CT can be reliably predicted by automated detectors in *Zoombinis*? 2.) How do in-game measures of implicit CT in *Zoombinis* relate to external measures of CT? (I.e., are these valid assessments?)

4 A DESCRIPTION OF THE GAME *ZOOMBINIS*

Zoombinis (Author, 2015) is an award-winning learning game that was designed in the 1990s and re-released for current platforms. Players guide Zoombini characters on a journey through a series of challenging logic puzzles, leading them to safety in Zoombiniville. The game

consists of a suite of 12 puzzles for learners ages 8 and above with four difficulty levels per puzzle. *Zoombinis* puzzles were designed to develop mathematics concepts essential for computer programming and data analysis, such as sets, logical relationships, dimensions, mappings, sorting, comparing, and algorithms. Players can play in practice mode, where they can select physical characteristics for Zoombinis, or journey mode, where the game randomly generates characteristics for a group of Zoombinis which the player shepherds through several puzzles.

In this paper, we focus on three *Zoombinis* puzzles, *Pizza Pass*, *Mudball Wall*, and *Allergic Cliffs* (see Figure 1).

4.1 *Pizza Pass*

In *Pizza Pass*, the Zoombinis' path is blocked by one or more trolls that demand a meal (pizza or pizza and a sundae) with a specific set of toppings. However, the trolls only say whether (a) they want more toppings, (b) do not like at least one of the toppings, or (c) the meal is perfect. If there is more than one troll, each troll must receive its particular meal preference**.**

4.2 *Mudball Wall*

A large wall split into grid-squares blocks the Zoombinis' progress. Three Zoombinis line up on planks at the bottom of the screen, waiting to be launched over the wall. Each grid-square of the wall contains 0–3 dots, indicating how many Zoombinis will be launched over the wall when the player fires a mudball onto that grid-square. A machine allows players to choose the shape and color of the next mudball to fire. The shape and color of the mudball determine the landing position of the mudball on the wall. There is a limited amount of mud, and only those Zoombinis who make it over the wall by the time the mud runs out are safe.

4.3 *Allergic Cliffs*

The Zoombinis must cross two bridges spanning a chasm. Each bridge is supported by a cliff face that is allergic to one or more Zoombini trait. Players choose which of the two bridges each Zoombini should cross. Each Zoombini that causes a cliff face to sneeze is knocked back along the bridge to the starting side, and one of the six pegs holding both bridges up is dislodged. When all six pegs are gone, both bridges collapse, stranding the remaining Zoombinis. All Zoombinis that have safely crossed a bridge will move on.

5 IMPLICIT COMPUTATIONAL THINKING IN *ZOOMBINIS*

The learning mechanics embedded in *Zoombinis* puzzles align with contemporary constructs of CT as outlined by CSTA (2017) and related research (Wing, 2006; Brennan & Resnick, 2012). CT is increasingly important for developing 21st century skills, and also may provide unique opportunities to support inclusive STEM learning.

5.1 CT Practices and Progression

Drawing from several definitions of CT emerging in the field (Weintrop et al., 2016; Grover, 2017; Grover & Basu, 2017; Wing, 2011; Barr & Stephenson, 2011; Authors, 2017d), we defined a learning progression of CT that is consistent with the literature and is also aligned with the practices observed in *Zoombinis* gameplay (Figure 2). While shown linearly, this progression is highly iterative with many embedded small loops among phases repeating as new problems and contexts are encountered. We hypothesize that the following CT practices will be evident in *Zoombinis* gameplay:

1. **Problem Decomposition:** the reduction of the complexity of a problem by breaking it into smaller, more manageable parts.

2. **Pattern Recognition**: the ability to see trends and groupings in a collection of objects, tasks, or information.
3. **Abstraction**: the ability to make generalizations from observed patterns to make general rules or classifications about the objects, tasks, or information.
4. **Algorithm Design**: the creation of a replicable solution to a set of problems.

5.2 Phases of CT Problem Solving in *Zoombinis*

For each of the three puzzles studied, we identified the common strategies or methods players used to solve the problems in *Zoombinis* gameplay and associated each with one or more CT practices. These include:

1. **Trial and Error**: Player demonstrates no evidence of testing hypotheses in an ordered or intentional way. The player's actions are independent of prior actions or game feedback, and do not build from feedback in a productive way.
2. **Systematic Testing**: The player shows evidence of testing hypotheses about an underlying rule. They use an ordered, planned method with the end goal of finding a working solution to implement. During testing, the learner's next action depends on the result of their previous action and game feedback.
3. **Systematic Testing with a Partial Solution**: The player has solved one dimension of the puzzle and is testing hypotheses about a second dimension to find the complete solution.
4. **Implementing a Full Solution**: The player completes the puzzle once a working solution for all dimensions of the puzzle has been found.

We labeled these four mutually exclusive phases of CT within the gameplay of two puzzles: *Pizza Pass* and *Mudball Wall*.

In some cases, we had to further specify the practice. For example, in the *Mudball Wall* puzzle, we distinguished **explicit** and **implicit** problem decomposition. As players chose the color and shape of the mudballs they launched to solve the puzzle, some players held one attribute (e.g., shape) constant while trying all values of the other attribute (e.g., color). We labeled this as explicit problem decomposition because the player was making the problem simpler by making the entire pattern visible. Another strategy was to choose mudballs in color/shape combinations that revealed information about both dimensions simultaneously (row AND column) to more quickly establish the overall grid rules (e.g., player tried all combinations of values in sequence: red circle, blue star, yellow square, green diamond, pink triangle). We called this implicit problem decomposition because the player was implicitly decomposing both attributes at the same time but not making the entire pattern visible.

5.3 *Zoombinis* Puzzle Strategies

Through our analysis of videos and playback from each of the three puzzles, we identified and operationalized specific strategies players commonly used that are consistent with CT practices (described in more detail in Authors 2018, 2019). In *Zoombinis*, strategies are implicit algorithms or repeatable solutions we observed players taking to solve puzzles that could be replicated with programming. While strategies present in each round of play were labeled as described below, they were not considered implicit algorithm designs unless players repeated them across multiple rounds of play.

5.3.1 *Pizza Pass*. We found three strategies frequently used in *Pizza Pass* gameplay.

1. **One at a time strategy**: Player tries one topping at a time and, after trying all toppings, combines only those the troll likes.
2. **Additive Strategy**: Player tries one new topping at a time and on subsequent deliveries, retains only the toppings the troll likes.
3. **Winnowing strategy**: Player tries all toppings at once, then removes one at a time on subsequent deliveries.

5.3.2 *Mudball Wall*. For *Mudball Wall*, we identified five common strategies:

1. **Color or Shape Constant:** Player holds one attribute (color or shape) constant, while systematically testing values of the other attribute to establish the rule of a row or column. Player recognizes that sufficient information about the rule of the row/column is revealed after placement of four mudballs in that row/column.
2. **2D Pattern Completer**: Special case of Color or Shape Constant strategy in which a player completes an ENTIRE row AND an ENTIRE column to establish the full grid pattern before moving on to implementation.
3. **Maximizing Dots:** Player appears to actively target dots on the grid using information available to them from previous moves.
4. **All Combinations:** Player tries all shape/color pairs, changing both attributes between moves so as not to repeat a shape or color. The resulting diagonal pattern provides the full set of evidence needed to complete the puzzle.
5. **Alternating Color and Shape:** Player systematically alternates between holding color (e.g., red) and shape (e.g., circle) constant to establish the rule of a row or column.

5.3.3 *Allergic Cliffs*. We also identified three common strategies in *Allergic Cliffs* gameplay:

1. **Nothing in Common:** Player chooses two (or more) Zoombinis in a row that differ on all attribute values. This strategy applies to systematic testing early in the round when the player appears to be trying to establish bridge/cliff rules.
2. **Hold Attribute Constant:** Player chooses four or more Zoombinis in a row that have different values of the same attribute (e.g., noses) to test all values.
3. **Hold Value Constant:** Player chooses three or more Zoombinis in a row that have at least one attribute value in common (e.g., red noses or shaggy hair). This strategy applies to systematic testing early in the round plus systematic testing with partial solution.

5.4 *Zoombinis* Gameplay Efficacy Related to CT

In addition to CT practices and strategies, we also labeled other characteristics of gameplay efficacy for all puzzles that may be related to CT and provide an overall sense of understanding of the game demonstrated by the player. These include:

1. **Gameplay Efficiency:** indicates how well the learner appeared to understand the game mechanic and applied an effective strategy across an entire round of play. Researchers selected from three values (1=Not at all Efficient; 2=Somewhat Efficient; 3=Highly Efficient).
2. **Learning Game Mechanic:** moves that indicate a lack of understanding of the game mechanic (e.g., repeating identical mudballs, pizzas, or Zoombinis)
3. **Acting Inconsistent with Evidence:** moves that contradict evidence available from prior moves, assuming the player understands the game mechanic.

6 METHODS

This paper reports the process of building automated detectors of implicit CT demonstrated within data logs from a wide variety of *Zoombinis* gameplay. The process of building automated detectors, successfully applied to two physics games (Authors, 2017a), includes six steps:

1. Define constructs (see 4.1)
2. Hand-label *Zoombinis* gameplay
3. Synchronize labels to gameplay process data
4. Distill gameplay process data into features
5. Build automated detectors of players' CT practices
6. Validate the detectors as formative assessments of implicit CT practices.

Previous work provides a more detailed description of the hand-labeling process mentioned in step 2 (Authors, 2017a, 2017b). In particular, more information about the discussions between researchers during labeling of *Mudball Wall* can be found in Authors (2019). In this paper, we briefly describe the reliability of hand labeling and the categories of features engineered to the models (steps 3-4) (Authors, 2019). This paper's discussion focuses primarily on steps 5 and 6 and describes the process involved in building detectors of CT using a sample of upper elementary- and middle-school students. Additionally, we summarize our findings from prior research on one puzzle, *Mudball Wall* (Authors, 2019), and present new findings on two puzzles, *Pizza Pass* and *Allergic Cliffs*. As part of step 5, we present evidence from goodness metrics to assess the quality of the automated detectors across all three puzzles. For step 6, we discuss the correlational results between our detectors and external CT assessments.

6.1 Data Collection & Sample

For the hand labeling and modeling of CT, participants were drawn from a sample of 194 students participating in one-hour *Zoombinis* playtesting sessions that included multiple *Zoombinis* puzzles, such as *Pizza Pass, Mudball Wall*, and *Allergic Cliffs*. Participants were recruited from local schools, clubs, and after-school programs. Most playtesting sessions were small groups of 4–6 students playing on individual computers in their classroom, after-school program, or at REDACTED. These small groups were encouraged to talk to each other throughout the session, just as they would in classrooms. A few students played by themselves and were asked to think aloud as they solved puzzles in the presence of a researcher. This diversity of approaches to data collection was intended to mirror the range of conditions of classroom implementation in which the detectors will likely be used.

A minimum sample of 70 students across grades 3–8 evenly divided by gender and grade level was sought (see Table 1 for details). To ensure a high likelihood of finding a range of CT practices, players with higher success rates (i.e., 100% of Zoombinis through a puzzle multiple times) in these three puzzles were oversampled compared to players with lower success rates. Oversampling was done to improve the likelihood we would have enough examples to represent a wide range of CT behaviors in the hand-labeled sample to build reliable, generalizable detectors. These success rates were not shared with the researchers who were hand-labeling the puzzle gameplay.

A broader sample was used for validating our detectors as formative assessments. Thirty-six teachers of 54 3rd–8th grade classes who participated in our national *Zoombinis* implementation study were used to compare our CT detectors to external measures of CT. Only students who completed all post-assessment items belonging to the 4 facets were included. It was possible for participants to have played some but not all 3 puzzles resulting in missing detector

values for the puzzles they did not play. For this reason, the total number of students varied by puzzle. Between 741 and 918 students had rounds with 3 or more moves in *Pizza Pass*, *Mudball Wall*, or *Allergic Cliffs*.

6.2 Hand-label *Zoombinis* gameplay

Based on informal observations of children playing *Zoombinis* as well as our own experience playing the game, the initial labeling system for each puzzle included preliminary labeling schemas for the CT practices defined in 4.1, phases of problem solving, puzzle strategies, and gameplay characteristics. Through an iterative process, a team of four researchers watched playback independently, discussed their labeling, and refined label definitions of implicit CT until they agreed that the labeling system was an exhaustive representation of all the emergent gameplay behaviors previously seen (Authors, 2017b). Researchers repeated this process until Cohen's kappas exceeded 0.70 for a set of 10 players. Once we achieved reliability, one researcher (not a labeler) assigned a set of 10 players for labeling with a second researcher pre-designated as the Primary labeler and the other researcher designated the Reliability Check for the labeling of each player. The two labelers were Primary labelers for half of the players. Only data from the Primary labeler was used to create the detectors.

For the final labeling of gameplay reported here, researchers watched an entire round once all the way through, looking for evidence of one or more specific strategies in the gameplay. Researchers then either re-watched the round at reduced speed or step through event-by-event to determine which labels to apply for each category. Researchers used the categories of labels to select the most appropriate set of labels for each event and followed the rules for each category. For all puzzles, labeling was started at Round 2 based on the assumption that players are learning how to play in Round 1. All rounds with less than three events were

excluded due to too little information. The remaining rounds were labeled using the guidelines laid out in the labeling manuals for each game.

A non-labeling researcher checked the reliability on a weekly basis for careless errors and labeling drift. When found, the labelers were first asked to independently double-check their labeling against the labeling manuals and make any corrections. If the reliability was still below 0.70, the researchers met to review specific cases to determine whether they were aligned with the labeling manual. In cases where the disagreements were due to different conceptual understandings of the player behaviors and both aligned with the labeling manual, those disagreements were retained. This process was repeated for each puzzle, taking approximately 3–4 months to complete per puzzle. The puzzles are discussed in their order of completion—*Pizza Pass*, *Mudball Wall*, and *Allergic Cliffs*.

Figures 3–5 provide simplified labeled data extracted from REDACTED playback tool for *Pizza Pass*, *Mudball Wall*, and *Allergic Cliffs*. The panels in Figure 3 illustrates a player's CT progression through Level 1 of *Pizza Pass*. In this example, the player decomposed the problem of finding the perfect pizza to serve the troll by systematically testing one topping at a time. With each pizza served, the troll provided audio and visual feedback, indicating whether or not he liked the topping. Once the player had tried all possible single toppings, the *Problem Decomposition* label was turned off because at this point, the player had all the necessary information to assemble the final successful pizza. As shown in the final panel of Figure 3, the *Pattern Recognition* and *Abstraction* labels were applied along with *Implementing a Full Solution* to reflect the fact that the player kept only toppings the troll had previously approved and combined them into one pizza.

Figure 4 shows an example of labeling *Mudball Wall*. After the first mudball was launched, the player began systematically testing mudballs by holding the mudball (circle) constant while changing only the color between launches. This revealed colors were assigned to rows. Once the rule for rows was established, the player began testing the other dimension (column) by holding color (blue) constant and changing the shape of the mudball between launches. Since they had one dimension of the solution, the researcher switched the Phase label from *Systematic Testing* to *Systematic Testing with a Partial Solution*. While establishing the underlying rules for both dimensions, the *Explicit Problem Decomposition* label was applied. The *Pattern Recognition* label was turned on during *Systematic Testing* to reflect the player's understanding that for this particular puzzle, circles go in the last column and blue mudballs go in the top row. Once the full set of rules for the grid were revealed (rows correlate to color and columns to shape) and the player began applying those rules by targeting cells with the *Maximizing Dots* strategy, the *Abstraction* and *Implementing Full Solution* labels were turned on.

In this labeling example for *Allergic Cliffs*, the player began by systematically testing the Zoombinis by placing 3 Zoombinis in a row with the same hair value (ponytail) on the top bridge. Panel 3 of Figure 5 shows that the third Zoombini was rejected from the top bridge and then placed successfully across the bottom bridge. At this point, a partial solution is in place. The player has enough information to know that ponytail hair and pink noses will successfully cross both bridges and the top bridge accepts more than one value for feet (*Pattern Recognition* and *Abstraction*). By placing a Zoombini with sunglasses eyes on the bottom bridge, the player reveals that the bridges select on feet, with the bottom bridge accepting only spinner feet. Once the player begins placing only spinner feet on the bottom bridge and everything else on the top bridge, the *Implementing Full Solution* label is turned on. As shown in Panel 5 of Figure 5, the

player makes a mistake late in the round by placing a Zoombini with the wrong feet on the bottom bridge. Since it is a single mistake, the play is still considered *Highly Efficient* for the round.

6.2.1 Evidence Model of CT Practices Grounded in Hand Labeling

Through the hand-labeling process, our evidence model (Figure 6) links the specific actions that a player tries in *Zoombinis* with the competency variables of interest—**Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm Design**. Our task model relates to the three puzzles and possible behaviors or CT indicators that the player demonstrates in these puzzles (Authors, 2017d). For instance, when a player tries one topping at time on a pizza in *Pizza Pass*, that player is likely demonstrating Problem Decomposition. In *Mudball Wall*, consider a player who understands the rule of one dimension by holding the color constant and changing the shape in between mudball launches; this player is likely demonstrating Pattern Recognition. In *Allergic Cliffs*, when a player consistently applies the *Hold Attribute or Hold Value Constant* strategy across two rounds, this could indicate evidence for Algorithm Design. By building detectors that identify when a player is demonstrating these relevant indicators, we can use these detectors to provide evidence for implicit CT learning in *Pizza Pass, Mudball Wall,* and *Allergic Cliffs*.

6.2.2 Inter-rater Reliability of Hand Labeling

Labeling for all puzzles was done originally at the event level. Because some of the labels relied on three or more events happening in a row, essentially using the future to predict the present, the labels were aggregated to the round level.

To achieve inter-rater reliability, two researchers independently labeled all rounds of Level 1 play and tested their levels of agreement using Cohen's kappa (Cohen, 1960). Cohen's kappa provides chance-corrected agreement indices (i.e., take into account the possibility of chance agreement) and a range between -1 and 1, with a value of 1 signifying perfect agreement and values of 0 or below indicating no agreement above chance. These kappa values are reported for all labels except gameplay efficiency. *Gameplay Efficiency* is an ordinal rating and we reported a Cronbach's alpha (Cronbach, 1951) to reflect the internal consistency between the ratings of the two researchers. Inter-rater reliability was calculated at the round level for all CT constructs across all puzzles.

Table 2 shows interrater reliability results of phases of problems solving, CT practices, and gameplay efficacy related to CT for *Pizza Pass, Mudball Wall,* and *Allergic Cliffs*. Researchers generally achieved acceptable interrater reliability across all three puzzles with kappa values of 0.70 or more, except for a few labels in *Pizza Pass* (i.e., *Learning Game Mechanic*), *Mudball Wall* (Syst*ematic Testing* and *Implicit Problem Decomposition*) and *Allergic Cliffs* (*Systematic Testing with Partial Solution* and *Acting Consistent with Evidence*). Cronbach's alpha for *Gameplay Efficiency* was relatively high across all puzzles, with values from 0.96 to 0.98.

Strategies varied per puzzle (see 4.2). Players could have more than one strategy label per round. Table 3 shows the reliability of hand labeling of the strategies (implicit algorithms) players demonstrated in each of the puzzles. All labels had kappas exceeding 0.70. Two features of *Allergic Cliffs* gameplay made labeling particularly challenging compared to the other two puzzles. First, the transitions in *Pizza Pass* and *Mudball Wall* are single, discrete events (pizza delivery and throwing a mudball) that can only be completed one at a time. In *Allergic Cliffs*,

while only one Zoombini moves across a bridge at a time, more than one Zoombini can be placed in the queue to go over a bridge, making it difficult to discern which event is tied to which Zoombini. Second, in *Pizza Pass* and *Mudball Wall*, feedback remains visible beyond when it was provided by the game. Pizzas remain sorted by whether or not they were liked by the troll and mudball shape and color remain on the spot they hit. In a sense, patterns in these puzzles are in a "glass box." In *Allergic Cliffs*, however, the only feedback that remains visible on the screen is which Zoombinis made it over which bridge, not which ones were rejected by which bridge. Thus, patterns in finding solutions in *Allergic Cliffs* are invisible to researchers in a "black box," making it more difficult to reliably label CT behaviors in this puzzle.

Labeling was originally done at the event level and the kappas were not as high for the *Allergic Cliffs* strategies. For this reason, one set of labels was combined to create a single detector. During the labeling process, it became clear that it was too difficult to distinguish *Hold Attribute Constant* and *Hold Value Constant* by individual moves, so they were combined to create a merged *Hold Attribute or Value Constant* label.

6.3 Distill Gameplay Process Data into Features

The design and implementation of the hand-labeling scheme informed the feature engineering process. For each puzzle, we worked with domain experts to construct a list of 90 hypothetical features which we believed might be indicative of players' gameplay strategies. Using raw log data, we computed values for the approximately 75 features that were feasible given the available data. While iterating through this feature-building process, we created additional features based on insights that arose. The resulting feature set consisted of between 74 and 113 features, depending on the puzzle. Of these, 41 features were common across puzzles. This included the minimum, maximum, average, and standard deviation of a feature's values

over the course of the round. Feature values were aggregated at the round level when detecting gameplay efficiency across all puzzles and when detecting strategies in *Allergic Cliffs*. These round-level features were added to the log data, which were then used for the modeling process. All these features represent potentially meaningful evidence of players' use of CT practices in *Zoombinis*. Table 4 shows selected feature categories, examples, and rationale for each puzzle. The full list of features used in each puzzle can be found in Appendices A–C.

6.4 Build Automated Detectors of Players' CT Practices

We used RapidMiner 5.3 to build separate detectors for each hand label of implicit CT in *Pizza Pass, Mudball Wall*, and *Allergic Cliffs*. For each label of CT, we attempted to fit the detectors using four common classification algorithms previously used in detecting affect and engagement in computer-based learning environments: W-J48, W-JRip, linear regression with a step function, and Naive Bayes. These classification algorithms allow us to predict whether or not a student is demonstrating a given dimension of CT, in the form of a label. The goal of building these detectors is to replace the hand labeling of CT labels with an automated model that can be applied directly to data.

Detectors were evaluated using 4-fold student-level, batch cross-validation, in which models are repeatedly trained on three groups of students and tested on the 4th group. Cross-validation processes are important in order to select algorithms which are not over-fit to particular sets of training data, as cross-validation estimates the degree to which the model applies to unseen data. In particular, student-level batch, cross-validation avoids over-fitting to the behavior of individual students (i.e., avoids building a model tuned to specific students' idiosyncrasies).

We used AUC ROC (Area Under the Receiver Operating Characteristic curve) computed using the A' approach (Hanley & McNeil, 1982) as the primary goodness metric to evaluate model performance, followed by Cohen's kappa. In cases where the models had high kappa but low AUC values, we used both metrics to select the best performing model of CT behavior. Cohen's kappa assesses the degree to which our models are better than chance at predicting CT labels. A kappa of 0 indicates that the model performs at chance, and a kappa of 1 indicates that the model performs perfectly. AUC is the probability that if the detector is comparing two students, one labeled as demonstrating implicit CT and the other labeled as not demonstrating CT, the detector will correctly determine which student demonstrated CT. A model with AUC of 0.5 performs at chance, and a model with AUC of 1.0 performs perfectly. Metrics for all labels were calculated with one data point per round of gameplay.

6.5 Validate the Detectors as Formative Assessments of Implicit CT

We applied the detectors of CT to our broader sample of 1000+ upper elementary- and middle-school students from the *Zoombinis* implementation study. These detectors not only produce an inference of whether implicit CT is present or absent, but also produce a confidence in that inference. For example, if the *Highly Efficient Gameplay* detector has a confidence of 80% for a round, this indicates that there is an 80% probability that the student was being highly efficient in that round. As in Baker (2015), we average detector confidence values for each student across all rounds. Hypothetically, if the *Highly Efficient Gameplay* detector indicated that a student had completed five rounds with confidence values of 72%, 68%, 95%, 40%, and 80%, the average confidence for demonstrating highly efficient gameplay is 71%. This represents the most likely estimate for how often this hypothetical student was demonstrating *Highly Efficient Gameplay* in 72% of his or her actions, as it retains all information available and avoids treating

a student who is repeatedly 95% likely to be demonstrating a dimension of CT the same as a student who is repeatedly 51% likely to be demonstrating that dimension of CT.

These students also completed an 18-item CT assessment using digital interactive logic puzzles aligned with foundational CT constructs—Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm (see Table 5). Example items can be found in Authors (2017d). Aggregated CT assessments were found to have low to moderate validity and reliability (Authors, under review).

Three types of metrics were used for this study (mean number of correct responses for Pattern Recognition, mean percentage of spaces completed completely for Abstraction, and mean efficiency of responses for Problem Decomposition and Algorithm Design). scores were converted to Z scores for each facet to aid in interpretation. As assessment forms were more difficult in middle school than elementary, these Z scores were calculated within each grade band to take into account differences of difficulty between forms. For the results reported here, the Z-scores for Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm Design were calculated for each facet and then averaged to create an aggregated CT score.

Having the average predicted probabilities of each implicit CT construct and the aggregated CT score per student, we computed Pearson correlations between students' in-game measures and standardized post-CT assessment scores. We calculated the corresponding p-values for the correlation and used the Benjamini-Hochberg (Benjamini & Hochberg, 1995) method to adjust the alpha values for multiple comparisons.

7 RESULTS

Through the process of hand labeling, identification of salient CT practices, and building automated detectors of those CT practices, we were able to design implicit CT assessments for Zoombinis. We used external post-assessments of CT with digital, iterative, logic puzzles (Authors, in preparation) to determine the validity of those detectors.

7.1 Measures of Implicit CT from Zoombinis Gameplay

We evaluated the degree to which our models can accurately infer the absence or presence of each hand-applied CT label. Tables 6–8 shows the performance of all of the CT models in *Pizza Pass*, *Mudball Wall*, and *Allergic Cliffs*.

In *Pizza Pass*, the best-performing algorithms had lower reliability than human labelers for detecting phases of problem-solving and CT practices. In particular, kappas for the rare *Winnowing* strategy detector (only 10 labeled occurrences) had a relatively poor kappa value of 0.14 compared to its reliability of 0.75 for hand-labeling. This same detector had the worst AUC, only 0.63. The rest of the detectors yielded much higher AUC, with values between 0.77 and 0.92. In particular, the *One at a Time* strategy achieved the highest AUC value of 0.92, indicating that this model can correctly distinguish between the absence and presence of this strategy 92% of the time.

In *Mudball Wall*, there was a range of kappa values obtained for different constructs. At one extreme, the *Color Shape Constant* and *Alternating Color and Shape* strategies obtained the lowest performance, with kappa of 0.10 and 0.14 respectively and AUC ROC of 0.60 and 0.63. Most students frequently applied *Maximizing Dots*, where they actively targeted dots on the grid based on previous information from previous moves, demonstrating pattern recognition. This strategy achieved a much higher kappa value of 0.68, indicating that the model is 68% better

than chance at identifying this strategy. At the other extreme of kappa values, *Learning Game Mechanics* achieved excellent kappa performance, with a value of 0.87. In terms of AUC performance, these *Mudball Wall* CT detectors achieved a wide range of AUC values, from 0.60 to 0.93. Except for *Trial and Error* (AUC = 0.78), the phases of problem solving detectors achieved high AUC values of 0.86 to 0.88, meaning that these models can correctly distinguish between the absence and presence of these CT constructs 86% to 88% of the time.

In *Allergic Cliffs*, we found greater discrepancies in reliability between our automated detectors and human labelers. In particular, detectors of the *Nothing in Common* strategy, *Hold Attribute or Hold Value Constant* strategy, and *Learning Game Mechanics* performed at chance, with kappa values of zero. In particular, the J48 model for the *Hold Attribute or Hold Value Constant* strategy consisted of only one leaf and was not helpful in distinguishing the absence or presence of this label. Few too examples of the absence of this label may have made it difficult for the algorithm to predict cases in which the player did not demonstrate the *Hold Attribute or Hold Value Constant* strategy. This label was therefore dropped from further analysis. *Systematic Testing* and *Problem Decomposition* yielded negative kappa values, indicating that these models are worse than chance at detecting these CT constructs. In general, these findings may be in part due to the challenges of the patterns being tested by students in *Allergic Cliffs* did not remain visible. Researchers also found it more challenging to reliably label patterns of implicit CT. In comparison, detectors of *Pattern Recognition* and *Abstraction* both achieved a kappa value of 0.59, indicating that these models are 59% better than chance at identifying these CT practices. When taking AUC performance into account, all the detectors except the *Nothing in Common* strategy were better than chance at inferring the absence of presence of implicit CT learning. In

particular, detectors of *Implementing Full Solution*, *Gameplay Efficiency*, and *Learning Game Mechanics* yielded the highest performance, with AUC values of 0.86 to 0.95.

Model performance for some detectors resulted in a combination of relatively high AUC and comparably low kappa values, including *Winnowing* in *Pizza Pass*; *Alternating Color and Shape*, *Color or Shape Constant*, *Systematic Testing*, and *Implicit Problem Decomposition* in *Mudball Wall*; *Systematic Testing*, *Implementing Full Solution*, *Nothing in Common*, and *Hold Attribute Constant or Hold Value Constant* in *Allergic Cliffs*. These findings suggest that these models can distinguish these CT constructs in general but are relatively poor at determining a cut-off for ambiguous cases.

However, the relatively high AUC values suggest that these detectors are of sufficient quality for validating these models against post-CT assessments, as they are reliable when probability estimates are taken into account. Except for *Nothing in Common* and *Hold Attribute Value or Hold Value Constant*, all detectors of strategy are at a level of quality where they can be used to estimate when players are repeating the same strategies across puzzle rounds, as this consistent behavior may be indicative of applying implicit algorithms.

The rest of our detectors, especially *Implementing Full Solution*, *Pattern Recognition*, *Abstraction*, and *Highly Efficient Gameplay*, showed acceptably high AUC across all puzzles. For example, the *Highly Efficient Gameplay* detector achieved a range of AUC values from 0.84 to 0.91, indicating that these models can correctly distinguish between a highly efficient student and less efficient student 84%–91% of the time.

7.1 Validate the Detectors as Formative Assessments of Implicit CT

In Table 9, we summarize the Pearson correlations between each detector related to phases of problem solving, CT practices, and gameplay efficacy for *Pizza Pass, Mudball Wall,* and *Allergic Cliffs*.

Except for *Systematic Testing with Partial Solution* in *Pizza Pass*, all phases of problem solving were significantly associated with students' post-assessment scores. Higher incidence of *Systematic Testing* and *Implementing Full Solution* were associated with higher CT scores while higher incidence of *Trial and Error* was correlated with lower scores.

Similarly, most of the CT practices detectors were significantly associated with students' post-assessment scores, except for *Explicit Problem Decomposition* in *Mudball Wall*. Players with higher incidence of *Problem Decomposition*, *Pattern Recognition*, and *Abstraction* had higher CT scores.

Except for *Learning Game Mechanic* in *Pizza Pass and Gameplay Efficiency in Allergic Cliffs*, detectors related to gameplay efficacy were significantly associated with students' post-assessment scores. Demonstrating *Highly Efficient Gameplay* was significantly related to better performance while demonstrating more *Learning Game Mechanic* and *Acting Inconsistent with the Evidence* were significantly associated with worse performance.

As shown in Table 10, correlations for player's strategies or implicit algorithms were not as strong as seen in phases of problem solving, computational thinking practices, and gameplay efficacy. For *Pizza Pass*, the *Additive* strategy was weakly associated with external CT assessment scores. In comparison, the *One at a Time* strategy and *Winnowing* strategy achieved stronger correlations. Higher probabilities of *One at a Time* were associated with higher post-assessment scores while higher probabilities of *Winnowing* were associated with lower post-assessment scores.

For *Mudball Wall*, *Maximizing Dots* achieved the strongest correlation with more of this strategy being associated with better post-intervention CT performance. *Alternating Color and Shape* was significantly associated with post-test scores, with higher probabilities of this construct being correlated with higher scores. *Holding Color or Shape Constant*, *2D Pattern Completer*, and *Try All Combinations of Color and Shape* were weakly associated with external CT assessment scores.

Due to the lack of accuracy in predicting the absence and presence of the *Hold Attribute Constant or Hold Value Constant* strategy, this detector was dropped from correlational analysis. *Nothing in Common* was significantly and negatively associated with post-test scores.

8 DISCUSSION

The assessment of CT learning presents several challenges to researchers including the novelty of CT as a field, and thus a dearth of established definitions and measures; as well as the requirement to measure the ways of thinking of CT within a process rather than a resulting artifact or test question (Grover & Basu, 2017).

Many current CT assessments are laden with symbolic notation or decontextualized scenarios that present extraneous barriers for some learners. In an effort to mitigate these barriers, our research team is exploring methods to measure implicit learning— learning that can be demonstrated through behaviors and activity within the learning activity itself (Authors, 2015). Digital games provide unique affordances for implicit learning assessment because they can motivate players to persist in complex problem-solving (Shute, Ventura, & Ke, 2015; Steinkuehler & Duncan, 2008; Qian & Clark, 2016); and they provide digital log data that can analyzed in a replicable and scalable manner (Plass, Mayer, & Homer, 2020).

We developed our emergent methodology through the study of implicit science learning in two digital games (Authors, 2017, 2019). In this paper, we apply this methodology to use gameplay log data and build detectors of implicit CT, grounded on hand labels of recorded gameplay process data. We were able to detect students' implicit learning of CT practices in different puzzles of *Zoombinis,* achieving good agreement with hand labels of 70+ students per puzzle. We present evidence from AUC, our primary goodness metric, to assess the quality of the automated detectors across all three puzzles. Except for the *Nothing in Common* strategy *in Allergic Cliffs,* a construct for which detection proved infeasible within our approach, it was possible to construct a range of detectors across all three puzzles. These detectors varied in quality, from the low 0.6s, a level of quality similar to detectors predicting affect and engagement in online learning environments (Pardos et al., 2013) to detectors in the mid 0.9s, a level quality higher than the detectors seen in many medical applications (Revell et al., 2013). Even detectors at the lower end of our performance range have proven able to predict long-term student outcomes (Pardos et al., 2013; San Pedro et al., 2013), which suggest that these detectors can be used for formative assessments of students' implicit CT and for informing teachers which students need learning support.

Findings from our correlational analyses of a sample of 797-980 (depending on puzzle) students in grades 3-8 also show evidence that our CT detectors, especially those related to phases of problem solving, CT practices, and gameplay efficacy, are valid assessments of students' implicit CT learning in that they achieve convergent validity with external CT assessments. These findings suggest that it is possible to deploy these in-game assessments at scale and in real time to reveal students' implicit knowledge to educators and designers.

We can leverage these models to help teachers using *Zoombinis* determine when players are demonstrating gameplay behaviors consistent with CT practices. Specifically, these models can reliably identify which players are consistently using the same strategy across rounds, as evidence of learners designing implicit algorithms to solve problem-solving scenarios. We have previously used confidence levels from valid strategy detectors (e.g., *One at a Time* strategy in *Pizza Pass* and *Maximizing Dots* in *Mudball Wall*) to compute the proportions of actions in which a player applied a strategy across two rounds of the same puzzle. For instance, our detectors can indicate that a player applied a *One at a Time* strategy 70% of the time in all his actions in any two consecutive rounds of gameplay in *Pizza Pass*, suggesting consistent use of the strategy and application of an algorithm solution across rounds. Similarly, game designers could use this methodology to create their own CT detectors to predict implicit learning and generate real-time adaptations in digital games that support players' zone of proximal development (Vygotzky, 1978). These are potentially useful future directions for integrating data-driven in-game implicit learning assessments into the classroom.

In sum, this unique approach provides an opportunity to provide scalable and replicable measures of implicit learning in learning environments such as games that can be used as formative real-time implicit learning assessments during instruction. In building and validating three game-based assessments, two physics games (Authors, 2017) and *Zoombinis*, we emphasized the importance of examining and understanding learners' processes of solving problems in authentic situations, rather than in more traditional and constrained testing contexts. Because *Zoombinis* puzzles involve the same implicit CT practices sometimes needed to solve real-world problems, our detectors are grounded on learners' emergent CT behaviors as they encounter authentic problem-solving scenarios within the game. Our approach also shows

promise in alleviating some of the constraints imposed by an ECD model of assessments. When game-based learning assessments primarily rely on designers' a priori learning trajectories, these may not entirely capture learners' spontaneous and emergent strategies that are likely indicative of their cognitive strengths and implicit knowledge of a salient phenomenon. These in-game assessments also have the potential to mitigate the barriers of current traditional assessments by assessing learning through puzzles that do not rely on textual or verbal representations. We hope that this work contributes to a new area of research that highlights the need for CT assessments based on educational data mining techniques to reach, better understand, and assess the implicit STEM knowledge of young learners.

9 CONCLUSION

Game-based implicit learning assessments may provide a new genre of formative assessment that can reveal what learners know implicitly, not just what they can say on a test or class assignment. This form of assessment in an interest-driven environment such as a game has the potential to engage a variety of learners because it examines their learning in an environment where they are typically motivated and interested in finding solutions. This may be particularly important for learners who are often disengaged in school content. Implicit learning assessments that use educational data mining techniques grounded in hand labeling of play behaviors provide an opportunity to look "under the hood" at what learners can demonstrate through behaviors which may reveal cognitive strengths that go unrecognized when relying on traditional academic tests (Nguyen, Garner, & Sheridan, 2018). Designing and validating formative game-based assessments to reach a broad range of learners at a scale and in real-time is just a beginning and is a rich area for future research.

REFERENCES

Allan, W., Coulter, B., Denner, J., Erickson, J., Lee, I., Malyn-Smith, J., & Martin, F. (2010). Computational Thinking for Youth. ITEST Small Working Group on Computational Thinking.

Arena, D.A., Schwartz, D.L. (2013). Experience and explanation: using videogames to prepare students for formal instruction in statistics. *Journal of Science Education and Technology*, 1–11.

Authors (under review). Digital Interactive Logic Puzzles for Formative Assessment of Computational Thinking in Grades 3-8.

Authors (2019, October). Modeling Implicit Computational Thinking in *Zoombinis* Mudball Wall. Paper presented at the Technology, Mind, and Society Conference (TMS), Washington, DC.

Authors (2019). Advancing Research in Game-Based Learning Assessment: Tools and Methods for Measuring Implicit Learning. In E. Kennedy & Y. Qian (Eds.), *Advancing Educational Research with Emerging Technology* (pp. 99–123), Hershey, PA: IGI Global.

Authors (2018, April). Labeling Implicit Computational Thinking in Pizza Pass Gameplay. Late-breaking work presented at the ACM SIGCHI Conference on Human Factors in Computing Systems (CHI 18), Montreal. https://doi.org/10.1145/3170427.3188541

Authors (2017a, August). Assessing implicit computational thinking in Zoombinis gameplay. In *Proceedings of the 12th International Conference on the Foundations of Digital Games* (p. 45). ACM.

Authors (2017b, October). Assessing Implicit Computational Thinking in Zoombinis Gameplay: Pizza Pass, Fleens & Bubblewonder Abyss. In *Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play* (pp. 195–200). ACM.

Authors (2017c). Assessing implicit science learning in digital games. *Computers in Human Behavior*, *76*, 617–630.

Authors (2017d). Demystifying computational thinking. *Educational Research Review*, *22*, 142-158.

Authors (2015). Serious games analytics to measure implicit science learning. In *Serious Games Analytics* (pp. 343–360). Springer, Cham.

Authors (2015). Zoombinis. Game (Android, IOS, MacOS, Windows, Web). (5 March 2019).

Baker, R. S. (2015). Big data and education (2nd ed.). New York, NY: Teachers College. Columbia University. Retrieved from http://www.columbia.edu/~rsb2162/bigdataeducation.html.

Baker, R. S., & Clarke-Midura, J. (2013, June). Predicting successful inquiry learning in a virtual performance assessment for science. In *International conference on user modeling, adaptation, and personalization* (pp. 203–214). Springer, Berlin, Heidelberg.

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society: series B (Methodological)*, *57*(1), 289–300.

Barendsen, E., Mannila, L., Demo, B., Grgurina, N., Izu, C., Mirolo, C., ... & Stupurienė, G. (2015, July). Concepts in K-9 computer science education. In Proceedings of the 2015 ITiCSE on Working Group Reports (pp. 85–116). ACM.

Baron-Cohen, S., Ashwin, E., Ashwin, C., Tavassoli, T., & Chakrabarti, B. (2009). Talent in autism: hyper-systemizing, hyper-attention to detail and sensory hypersensitivity. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *364*(1522), 1377–1383.

Barr, V., & Stephenson, C. (2011). Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?. *Inroads*, *2*(1), 48–54.

Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada* (Vol. 1, p. 25).

Clark, D. B., Nelson, B., Chang, H., D'Angelo, C. M., Slack, K., & Martinez-Garza, M. (2011). Exploring Newtonian mechanics in a conceptually-integrated digital game: Comparison of learning and affective outcomes for students in Taiwan and the United States. *Computers and Education, 57*(3), 2178–2195.

Ceci, S.J., & Liker, J.K. (1986). A day at the races: A study of IQ, expertise, and cognitive complexity. *Journal of Experimental Psychology 115*(3), pp. 255–266.

Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and psychological measurement*, *20*(1), 37–46.

Cronbach, L. J. (1951). Coefficient alpha and the internal structure of tests. *Psychometrika*, *16*(3), 297–334.

Computer Science Teachers Association. (2017). *CSTA K-12 Computer Science Standards*. Retrieved from https://www.csteachers.org/page/standards.

Dagienė, V., & Futschek, G. (2008, July). Bebras international contest on informatics and computer literacy: Criteria for good tasks. In International conference on informatics in secondary schools-evolution and perspectives (pp. 19–30). Springer, Berlin, Heidelberg.

Dagienė, V., Stupurienė, G., & Vinikienė, L. (2016, June). Promoting inclusive informatics education through the Bebras challenge to all K-12 students. In Proceedings of the 17th International Conference on Computer Systems and Technologies 2016 (pp. 407–414). ACM.

Dawson, M., Soulières, I., Ann Gernsbacher, M., & Mottron, L. (2007). The level and nature of autistic intelligence. *Psychological Science*, 18(8), 657–662.

Ginsburg, H. P., Lee, J. S., & Boyd, J. S. (2008). Mathematics Education for Young Children: What It Is and How to Promote It. Social Policy Report. Volume 22, Number 1. Society for Research in Child Development.

González, M. R. (2015). Computational thinking test: Design guidelines and content validation. In Proceedings of EDULEARN15 conference (pp. 2436–2444).

Grover, S., & Basu, S. (2017, March). Measuring student learning in introductory block-based programming: Examining misconceptions of loops, variables, and Boolean logic. In *Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education* (pp. 267–272). ACM.

Grover, S., Basu, S., Bienkowski, M., Eagle, M., Diana, N., & Stamper, J. (2017). A framework for using hypothesis-driven approaches to support data-driven learning analytics in measuring computational thinking in block-based programming environments. *ACM Transactions on Computing Education (TOCE)*, *17*(3), 14.

Grover, S. (2017). Assessing algorithmic and computational thinking in K-12: Lessons from a middle school classroom. In *Emerging research, practice, and policy on computational thinking* (pp. 269–288). Springer, Cham.

Grover, S., & Pea, R. (2013). Computational Thinking in K–12 A Review of the State of the Field. *Educational Researcher*, *42*(1), 38–43.

Grover, S., Cooper, S., & Pea, R. (2014, June). Assessing computational learning in K-12. In Proceedings of the 2014 conference on Innovation & technology in computer science education (pp. 57–62). ACM.

Haladyna, T. M., & Downing, S. M. (2004). Construct- irrelevant variance in high- stakes testing. *Educational Measurement: Issues and Practice*, *23*(1), 17–27.

Halverson, R., Wills, N., & Owen, E. (2012). CyberSTEM: Game-based learning telemetry model for assessment. Presentation at 8th Annual GLS, Madison, WI.

Hanley, J. A., & McNeil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, *143*(1), 29–36.

Hicks, D., Eagle, M., Rowe, E., Asbell-Clarke, J., Edwards, T., & Barnes, T. (2016, April). Using game analytics to evaluate puzzle design and level progression in a serious game. In *Proceedings of the Sixth International Conference on Learning Analytics & Knowledge* (pp. 440–448). ACM.

Izu, C., Mirolo, C., Settle, A., Mannila, L., & Stupurienė, G. (2017). Exploring Bebras Tasks Content and Performance: A Multinational Study. *Informatics in Education*, *16*(1), 39–59. https://files.eric.ed.gov/fulltext/EJ1140704.pdf.

Kim, Y. J., Almond, R. G., & Shute, V. J. (2016). Applying evidence-centered design for the development of game-based assessments in physics playground. *International Journal of Testing*, *16*(2), 142–163.

Kim, Y. J., & Rosenheck, L. (2018). *A playful assessment approach to research instrument development*. Paper presented at the Thirteenth International Conference of the Learning Sciences, London, UK.

Koh, K. H., Basawapatna, A., Nickerson, H., & Repenning, A. (2014, July). Real time assessment of computational thinking. In *IEEE Symposium on Visual Languages and Human-Centric Computing* (pp. 49–52). IEEE.

Lave, J., Murtaugh, M., & de la Roche, O. (1984). The dialectic of arithmetic in grocery shopping. In B. Rogoff & J. Lave (Eds.), *Everyday Cognition: Its development in social context* (pp. 76–94). Cambridge, Mass: Harvard University Press.

Lee, S. (2016). Effects of Representation Format in Problem Representation on Qualitative Understanding and Quantitative Proficiency in a Learning Game Context. Unpublished Doctoral Dissertation, Florida State University.

Li, H., Gobert, J., & Dickler, R. (2017). Automated Assessment for Scientific Explanations in On-Line Science Inquiry. *Proceedings of the International Conference on Educational Data Mining*.

Li, H., Gobert, J., Graesser, A., & Dickler, R. (2018). Advanced Educational Technology for Science Inquiry Assessment. *Policy Insights from the Behavioral and Brain Sciences*, *5*(2), 171-178.

Martin, T., Petrick Smith, C., Forsgren, N., Aghababyan, A., Janisiewicz, P., & Baker, S. (2015). Learning fractions by splitting: Using learning analytics to illuminate the development of mathematical understanding. *Journal of the Learning Sciences*, *24*(4), 593–637.

Martinuzzi, A., & Krumay, B. (2013). The good, the bad, and the successful–how corporate social responsibility leads to competitive advantage and organizational transformation. *Journal of Change Management*, *13*(4), 424–443.

Mishra, P., Yadav, A., & Deep-Play Research Group. (2013). Rethinking technology & creativity in the 21st century. *TechTrends*, *57*(3), 10–14.

Mislevy, R. J., & Haertel, G. (2006). *Implications of evidence-centered design for educational testing*. Menlo Park, CA: SRI International.

Mislevy, R. J., Steinberg, L. S., & Almond, R. G. (2003). On the structure of educational assessment. Measurement: Interdisciplinary Research and Perspective, 1(1) 3–62.

Moreno-León, J., & Robles, G. (2015, November). Dr. Scratch: a Web Tool to Automatically Evaluate Scratch Projects. In WiPSCE (pp. 132-133). https://www.researchgate.net/profile/Jesus_Moreno-Leon/publication/284181364_Dr_Scratch_a_Web_Tool_to_Automatically_Evaluate_Scratch_Projects/links/564eccb508aefe619b0ff212.pdf.

National Academy of Sciences on Computational Thinking (2010). Report of a Workshop on The Scope and Nature of Computational Thinking. National Academies Press.

Nguyen, A., Gardner, L. A., & Sheridan, D. (2018). A framework for applying learning analytics in serious games for people with intellectual disabilities. *British Journal of Educational Technology*, *49*(4), 673-689.

Nunes, T., Schliemann, A.D., & Carraher, D.W. (1993). *Mathematics in the Streets and in Schools*. Cambridge, U.K: Cambridge University Press.

O'Leary, U. M., Rusch, K. M., & Guastello, S. J. (1991). Estimating age- stratified WAIS- R IQS from scores on the Raven's standard progressive matrices. *Journal of Clinical Psychology*, 47(2), 277–284.

Ota, G., Morimoto, Y., & Kato, H. (2016, September). Ninja code village for scratch: Function samples/function analyser and automatic assessment of computational thinking concepts. In 2016 *IEEE Symposium on Visual Languages and Human-Centric Computing* (VL/HCC) (pp. 238–239). IEEE.

Pardos, Z. A., Baker, R. S., San Pedro, M. O., Gowda, S. M., & Gowda, S. M. (2013, April). Affective states and state tests: Investigating how affect throughout the school year predicts end of year learning outcomes. In *Proceedings of the third international conference on learning analytics and knowledge* (pp. 117-124).

Plass, J., Mayer, R. E., & Homer, B. (2020). Handbook of game-based learning. Cambridge, MA: MIT Press.

Polyani, M. (1966). The logic of tacit inference. *Philosophy, 41*, 1–18.

Qian, M., & Clark, K. R. (2016). Game-based Learning and 21st century skills: A review of recent research. *Computers in Human Behavior, 63,* 50–58.

Raven, J.C. (1981). *Manual for Raven's progressive matrices and vocabulary scales. Research supplement No.1: The 1979 british standardisation of the standard progressive Matrices and mill hill vocabulary scales, together with comparative data from earlier studies in the UK, US, Canada, Germany and Ireland.* San Antonio, TX: Harcourt Assessment.

Reber, A. S. (1989). Implicit learning and tacit knowledge. *Journal of Experimental Psychology: General*, *118*(3), 219.

Revell, A. D., Wang, D., Wood, R., Morrow, C., Tempelman, H., Hamers, R. L., ... & DeWolf, F. (2013). Computational models can predict response to HIV therapy without a genotype and may reduce treatment failure in different resource-limited settings. *Journal of Antimicrobial Chemotherapy*, *68*(6), 1406–1414.

Ritchhart, R., Church, M., & Morrison, K. (2011). *Making thinking routines visible: How to promote engagement, understanding, and independence for all learners*. San Francisco, CA: Jossey-Bass.

San Pedro, M.O.Z., Baker, R.S.J.d., Bowers, A.J., Heffernan, N.T. (2013) Predicting College Enrollment from Student Interaction with an Intelligent Tutoring System in Middle School. *Proceedings of the 6th International Conference on Educational Data Mining*, 177-184.

Sengupta, P., Kinnebrew, J. S., Basu, S., Biswas, G., & Clark, D. (2013). Integrating computational thinking with K-12 science education using agent-based computation: A theoretical framework. *Education and Information Technologies*, *18*(2), 351–380.

Shute, V. J. (2011). Stealth assessment in computer-based games to support learning. *Computer games and instruction*, *55*(2), 503-524.

Shute, V. J., Masduki, I., Donmez, O., Dennen, V. P., Kim, Y. J., Jeong, A. C., & Wang, C. Y. (2010). Modeling, assessing, and supporting key competencies within game environments. In *Computer-based Diagnostics and Systematic Analysis of Knowledge* (pp. 281–309). Springer, Boston, MA.

Shute, V. J., Ventura, M., & Ke, F. (2015). The power of play: The effects of Portal 2 and Lumosity on cognitive and noncognitive skills. *Computers & Education*, *80*, 58–67.

Shute, V. J., Wang, L., Greiff, S., Zhao, W., & Moore, G. (2016). Measuring problem solving skills via stealth assessment in an engaging video game. *Computers in Human Behavior*, *63*, 106-117.

Squire, K. (2011). Video games and learning. *Teaching and participatory culture in the digital age. New York, NY: Teachers College Print*.

SRI International (2013). Exploring CS Curricular Mapping. Retrieved from http://pact.sri.com/?page_id=1380

Steinkuehler, C., & Duncan, S. (2008). Scientific habits of mind in virtual worlds. *Journal of Science Education and Technology*, *17*(6), 530–543.

Sternberg, R. J. (1996). *Successful intelligence: How practical and creative intelligence determine success in life* (pp. 191–192). New York, NY: Simon & Schuster.

Tissenbaum, M., Sheldon, J., Sherman, M. A., Abelson, H., Weintrop, D., Jona, K., Horn, M., Wilensky, U., Basu, S., Rutstein, D., Snow, E., Shear, L., Grover, S., Lee, I., Klopfer, E., Jayathirtha, G., Shaw, M., kafai, y., Mustafaraj, E., Temple, W., Shapiro, R. B., Lui, D., & Sorensen, C. (2018). The State of the Field in Computational Thinking Assessment. In Kay, J. and Luckin, R. (Eds.) Rethinking Learning in the Digital Age: Making the Learning Sciences Count, 13th International Conference of the Learning Sciences (ICLS) 2018, Volume 2. London, UK: International Society of the Learning Sciences.

von Wangenheim, C. G., Hauck, J. C., Demetrio, M. F., Pelle, R., da Cruz Alves, N., Barbosa, H., & Azevedo, L. F. (2018). CodeMaster--Automatic Assessment and Grading of App Inventor and Snap! Programs. *Informatics in Education, 17*(1), 117–150. https://files.eric.ed.gov/fulltext/EJ1177148.pdf.

Vygotsky, L. S. (1978). Mind in society. Cambridge, MA: Harvard University Press.

Wang, S. How Autism Can Help You Land a Job. *The Wall Street Journal*, March 27, 2014.

Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, *25*(1), 127–147.

Weintrop, D., Killen, H., Munzar, T., & Franke, B. (2019, February). Block-based Comprehension: Exploring and Explaining Student Outcomes from a Read-only Block-based Exam. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 1218–1224). ACM.

Werner, L., Denner, J., Campe, S., & Kawamoto, D. C. (2012, February). The fairy performance assessment: measuring computational thinking in middle school. In Proceedings of the 43rd ACM technical symposium on Computer Science Education (pp. 215–220). ACM.

https://www.cs.auckland.ac.nz/courses/compsci747s2c/lectures/wernerFairyComputationalThinkingAssessment.pdf

Wiebe, E., London, J., Aksit, O., Mott, B. W., Boyer, K. E., & Lester, J. C. (2019, February). Development of a Lean Computational Thinking Abilities Assessment for Middle Grades Students. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 456–461). ACM. https://dl.acm.org/citation.cfm?id=3287390

Wing, J. M. (2011). Research notebook: Computational thinking—What and why. *The Link Magazine*, 20–23.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33–35.

ACKNOWLEDGMENTS

We are grateful for NSF grant #1502882 and the study participants. We are thankful for the many contributions of our research group, the EdGE group at TERC, without whom the study could not have been conducted.

Table 1: Number of students labeled by puzzle, grade level, and gender.

Grades	**Gender**	***Pizza Pass***	***Mudball Wall***	***Allergic Cliffs***
3rd–5th Grade	Females	17	20	18
	Males	18	17	18
6th–8th Grade	Females	17	16	18
	Males	19	21	18
Total	**Females**	**34**	**36**	**36**
	Males	**37**	**38**	**36**

Table 2: Reliability of hand labeling of *Pizza Pass*, *Mudball Wall*, and *Allergic Cliffs* Phases of Problem Solving, CT practices, and Gameplay Efficacy

Label	**Kappas**		
	Pizza Pass	***Mudball Wall***	***Allergic Cliffs***
Phases of Problem Solving			
Trial and Error	0.88	0.83	0.94
Systematic Testing	0.85	0.61	0.93
Systematic Testing w/Partial Solution	0.81	0.87	0.65
Implementing w/Full Solution	0.77	0.95	0.89
Computational Thinking Practices			
Problem Decomposition	0.82	n/a	0.93
Explicit Problem Decomposition	n/a	0.78	n/a
Implicit Problem Decomposition	n/a	0.68	n/a
Pattern Recognition	0.82	0.84	0.75
Abstraction	0.74	0.93	0.75
Gameplay Efficacy			
Gameplay Efficiency*	0.96	0.98	0.99
Learning Game Mechanic	0.66	0.85	0.83
Acting Inconsistent with Evidence	0.81	0.77	0.69

Note: *Gameplay Efficiency is an ordinal label (3 values). The values reported in this table are Cronbach's alphas, not Cohen's kappas.

Table 3: Reliability of hand labeling of *Pizza Pass*, *Mudball Wall*, and *Allergic Cliffs* strategies (implicit algorithms)

Pizza Pass		***Mudball Wall***		***Allergic Cliffs***	
Strategy	**Kappa**	**Strategy**	**Kappa**	**Strategy**	**Kappa**
One at a Time	0.79	Color or Shape Constant	0.75	Nothing in Common	0.93
Additive	0.84	2D Pattern Completer	0.83	Hold Attribute Constant	0.89
Winnowing	0.75	Maximizing Dots	0.78	Hold Value Constant	0.78
		Try All Combinations of Color and Shape	0.82		
		Alternating Color and Shape	0.87		

Table 4: Puzzle Feature Categories, Examples, and Rationale

Category	**Example Feature**	**Feature Rationale**
Pizza Pass, Mudball Wall, and *Allergic Cliffs*		
Overall Gameplay: These features describe general aspects of a student's play such as the outcome of each round (all Zoombinis through, Some Zoombinis through, no Zoombinis through, Quit)	Number of Zoombinis ejected from the puzzle	Zoombinis being ejected from the puzzle is a significant sign play is not going well.
Topping Futzes (*Pizza Pass* only): Switches toppings before the pizza is delivered to the troll.	Number of futzes since last pizza	More futzing may indicate that the player is unsure about their next move.
Mudball Duplicates (*Mudball Wall* only): These features capture the extent to which learners throw similar mudballs at the wall	Current consecutive number of duplicate mudballs	Duplicate mudballs are wasted resources because no new information is gained. This suggests players may not understand the game mechanic.
Zoombini Timing (*Allergic Cliffs* only): These features describe the speed with which learners select Zoombinis to cross the bridges.	Time since the last unsuccessful Zoombini.	More time since the last unsuccessful Zoombini may indicate if players have figured out the underlying rule

Note: In *Pizza Pass*, players can add and remove toppings from a pizza as often as they like before delivering that pizza to a troll; we use the term "futz" for a player modifying a pizza in this way. This is contrasted to the term "change," which we only use to describe modifications that affect the final, delivered pizza.

Table 5: Description of Post-CT Assessments

CT Practice	Description of puzzle	Number of items	Metric used for scoring
Problem Decomposition	Puzzles that require finding the correct answer by decomposing the problem space.	4	Mean efficiency (# moves/optimal number of moves)
Pattern Recognition	Raven's Progressive Matrices (RPM; Raven, 1981)	5	Mean percentage of correct responses
Abstraction	Puzzles that require identification of an underlying rule and generalization of the rule to the rest of the pattern.	6	Mean percentage of spaces completed correctly
Algorithm Design	Puzzles that require sequencing of arrows that guide a character along a path in a maze with obstacles.	3	Mean efficiency (# moves/optimal # moves)

Table 6: Kappa and AUC values for best performing models for *Pizza Pass*

	W-J48		**W-JRip**		**Step Reg**		**Naïve Bayes**	
Label	**Kappa**	**AUC**	**Kappa**	**AUC**	**Kappa**	**AUC**	**Kappa**	**AUC**
Phases of Problem Solving								
Trial and Error	0.61	0.82	0.64*	0.84*	0.50	0.81	0.59	0.81
Systematic Testing	0.43	0.69	0.49*	0.77*	0.37	0.71	0.25	0.68
Systematic Testing with Partial Solution	0.53	0.78	0.50*	0.79*	0.25	0.69	0.48	0.74
Implementing Full Solution	0.65	0.78	0.59	0.70	0.32*	0.85*	0.02	0.54
Computational Thinking Practices								
Problem Decomposition	0.36	0.79	0.52*	0.81*	0.38	0.71	0.56	0.79
Pattern Recognition	0.59	0.75	0.62*	0.78*	0.34	0.71	0.02	0.54
Abstraction	0.65*	0.78*	0.59	0.70	0.32	0.85	0.02	0.54
Strategy (Implicit Algorithms)								
One at a Time	0.72	0.84	0.83*	0.92*	0.52	0.91	0.09	0.62
Additive Strategy	0.52	0.78	0.63*	0.82*	0.31	0.67	0.46	0.74
Winnowing	-0.01	0.49	0.12	0.61	0.14*	0.63*	0.02	0.53

Gameplay Efficacy								
Highly Efficient Gameplay	0.80	0.89	0.73*	0.91*	0.52	0.83	0.70	0.85
Learning Game Mechanics	0.33	0.57	0.40	0.70	0.37	0.76	0.49*	0.82*
Acting Inconsistent with Evidence	0.68	0.80	0.70*	0.83*	0.40	0.76	0.40	0.82

Note: * best-performing models (shaded)

Table 7: Kappa and AUC values for *Mudball Wall*

	W-J48		W-JRip		Step Reg		Naïve Bayes	
Label	**Kappa**	**AUC**	**Kappa**	**AUC**	**Kappa**	**AUC**	**Kappa**	**AUC**
Phases of Problem Solving								
Trial and Error	0.38	0.73	0.44	0.67	0.40	0.76	0.35*	0.78*
Systematic Testing	0.16	0.45	0.34	0.52	0.16*	0.86*	0.03	0.56
Systematic Testing with Partial Solution	0.40	0.70	0.60	0.78	0.50*	0.88*	0.54	0.83
Implementing Full Solution	0.61	0.86	0.60	0.78	0.60*	0.88*	0.54	0.83
Computational Thinking Practices								
Problem Decomposition								
Explicit Problem Decomposition	0.25	0.67	0.31	0.66	0.23	0.64	0.21*	0.69*
Implicit Problem Decomposition	0.09	0.56	0.05	0.53	0.16	0.64	0.18*	0.68*
Pattern Recognition	0.53	0.71	0.56	0.75	0.55*	0.83*	0.36	0.78
Abstraction	0.60	0.83	0.60	0.82	0.61*	0.86*	0.53	0.81
Strategy (Algorithms)								
Color or Shape Constant	0.16	0.60	0.16	0.54	0.14	0.57	0.14*	0.63*

2D Pattern Completer	0.19	0.68	0.06	0.43	0.15	0.67	0.20*	0.73*
Maximizing Dots	0.71	0.85	0.80	0.87	0.68*	0.89*	0.57	0.85
Try All Combinations of Color and Shape	0.24	0.52	0.21	0.59	0.30*	0.73*	0.06	0.57
Alternating Color and Shape	0.13	0.53	0.15	0.53	0.15	0.59	0.10*	0.60*
Gameplay Efficacy								
Highly Efficient Gameplay	0.62	0.79	0.60	0.84	0.54	0.83	0.63*	0.84*
Learning Game Mechanics	0.83	0.91	0.87*	0.93*	0.76	0.92	0.69	0.87
Acting Inconsistent with Evidence	0.64*	0.85*	0.69	0.82	0.55	0.81	0.50	0.81

Note: * best-performing models (shaded)

Table 8: Kappa and AUC values for *Allergic Cliffs*

	W-J48		W-JRip		Step Reg		Naïve Bayes	
Label	**Kappa**	**AUC**	**Kappa**	**AUC**	**Kappa**	**AUC**	**Kappa**	**AUC**
Phases of Problem Solving								
Trial and Error	0.18	0.58	0.15	0.60	0.33*	0.73*	0.20	0.73
Systematic Testing	0.00	0.51	-0.03	0.43	-0.03*	0.62*	-0.05	0.47
Systematic Testing with Partial Solution	0.34*	0.68*	0.28	0.60	0.16	0.62	0.25	0.67
Implementing Full Solution	0.63	0.78	0.67*	0.86*	0.48	0.82	0.25	0.68
Computational Thinking Practices								
Problem Decomposition	0.00	0.51	-0.03	0.43	-0.03*	0.62*	-0.05	0.47
Pattern Recognition	0.59*	0.81*	0.50	0.79	0.44	0.79	0.32	0.72
Abstraction	0.59*	0.81*	0.50	0.79	0.44	0.79	0.32	0.72
Strategy (Algorithms)								
Nothing in Common	0.00*	0.53*	-0.03	0.52	-0.07	0.52	0.02	0.56
Hold Attribute Constant or Hold Value Constant	0.06*	0.64*	0.06	0.52	-0.07	0.52	0.02	0.56

Gameplay Efficacy								
Highly Efficient Gameplay	0.62	0.85	0.63	0.82	0.61*	0.86*	0.55	0.78
Learning Game Mechanics	0.01	0.75	0.01*	0.95*	0.01	0.92	0.00	0.61
Acting Inconsistent with Evidence	0.60	0.75	0.57*	0.78*	0.54	0.77	0.40	0.75

Note: * best-performing models (shaded)

Table 9: Correlations between *Pizza Pass*, *Mudball Wall,* and *Allergic Cliffs* Phases of Problem Solving, Computational Thinking practices, and Gameplay Efficacy and Post Assessment Scores

Detectors	**Correlations with Post Assessment Scores**		
	***Pizza Pass* (N=990)**	***Mudball Wall* (N=797)**	***Allergic Cliffs* (N=989)**
Phases of Problem Solving			
Trial and Error	-0.18**	-0.16**	-0.09**
Systematic Testing	0.12**	0.14**	0.18**
Systematic Testing w/Partial Solution	0.08	0.17**	0.17**
Implementing w/Full Solution	0.20**	0.22**	0.24**
Computational Thinking Practices			
Problem Decomposition	0.23**	n/a	0.18**
Explicit Problem Decomposition	n/a	0.02	n/a
Implicit Problem Decomposition	n/a	0.22**	n/a
Pattern Recognition	0.23**	0.12**	0.14**
Abstraction	0.20**	0.21**	0.14**
Gameplay Efficacy			
Gameplay Efficiency	0.18**	0.24**	0.05
Learning Game Mechanic	-0.06	-0.25**	-0.20**
Acting Inconsistent with the Evidence	-0.24**	-0.21**	-0.11**

**Significant after Benjamini-Hochberg correction

Table 10: Correlations between *Pizza Pass*, *Mudball Wall*, and *Allergic Cliffs* strategy detectors (implicit algorithms) and Post Assessment Scores

Correlations with Post-Assessment Scores					
Pizza Pass		***Mudball Wall***		***Allergic Cliffs***	
Strategy	**r**	**Strategy**	**r**	**Strategy**	**r**
One at a Time	0.13**	Color or Shape Constant	0.02	Nothing in Common	-0.07**
Additive	0.02	2D Pattern Completer	0.02	Hold Attribute or Value Constant	n/a
Winnowing	-0.20**	Maximizing Dots	0.25**		
		Try All Combinations of Color and Shape	0.07		
		Alternating Color and Shape	0.13**		

**Significant after Benjamini-Hochberg correction

Figures 1-6

Figure 1: 4 *Zoombinis* Screenshots. (1) *Puzzle Map* with labels (top left); (2) *Allergic Cliffs* (top right); (3) *Pizza Pass* (bottom left); and (4) *Mudball Wall* (bottom right)

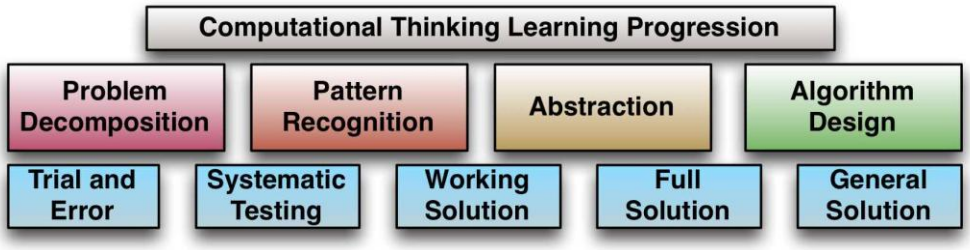

Figure 2. Computational Thinking Learning Progression

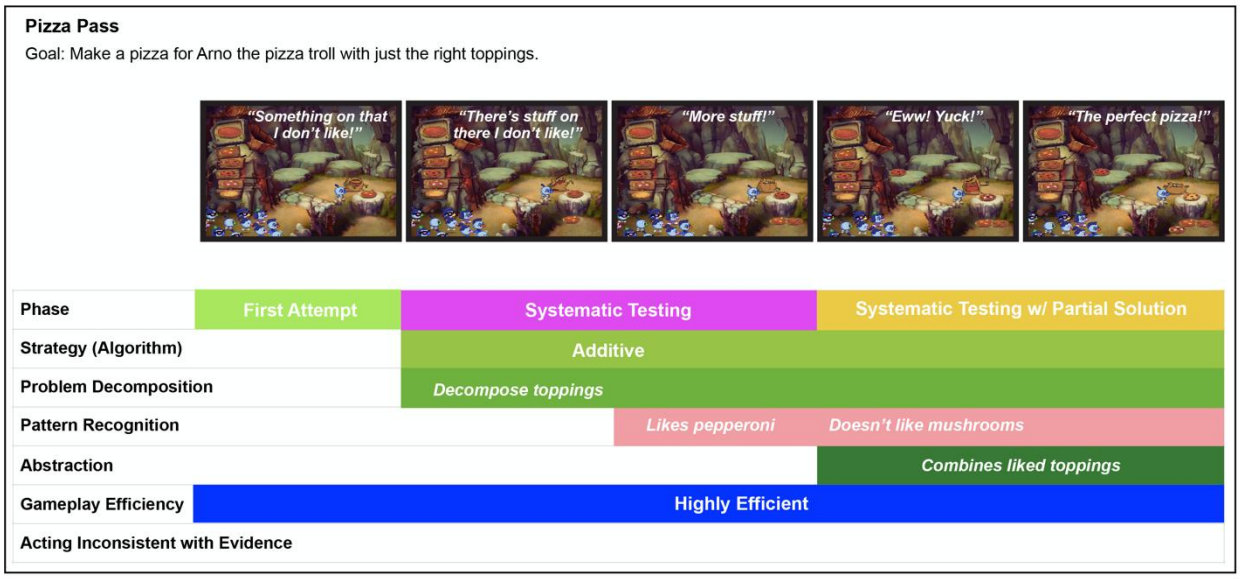

Figure 3: Sample annotated *Pizza Pass* labeling

Mudball Wall

Choose mudballs that hit the bricks with dots. There's a pattern you must discover…

Phase	First Attempt	Systematic Testing		ST w/ Partial Solution	Implementing Solution
Strategy (Algorithm)		Shape or Color Constant			Maximizing Dots
Problem Decomposition		Explicit Decomposition	*Decompose rows*	*Decompose columns*	
Pattern Recognition				*Circles in last column*	*Blue in top row*
Abstraction				*Rows are colors; columns are shapes*	
Gameplay Efficiency	Highly Efficient				
Acting Inconsistent with Evidence					

Figure 4: Sample annotated *Mudball Wall* labeling

Allergic Cliffs

Goal: Get your Zoombinis across one of two bridges by figuring out what attribute value(s) trigger an allergic reaction.

Phase	First Attempt	Systematic Testing	ST w/ Partial Solution	Implementing Full Solution
Strategy (Algorithm)	Hold Value Constant			
Problem Decomposition		*Testing ponytail hair*	*Testing sunglasses*	
Pattern Recognition			*Ponytail, pink noses go over both bridges*	*Only spinner feet go over bottom bridge*
Abstraction			*Bridges select on eyes or feet*	*Bridges select on feet*
Gameplay Efficiency	Highly Efficient			
Acting Inconsistent with Evidence				*Feet don't fit rule*

Figure 5: Sample annotated *Allergic Cliffs* labeling

Figure 6. Competency, Task, and Evidence Models for CT in *Zoombinis*