Interactive Assessments of CT (IACT): Digital Interactive Logic Puzzles to Assess Computational Thinking in Grades 3–8

Elizabeth Rowe [1] , Jodi Asbell-Clarke.[1] , Mia Almeda[1], Santiago Gasca[1], Teon Edwards[1], Erin Bardar[1], Valerie Shute[2], and Matthew Ventura [3]

[1]*EdGE at TERC, Cambridge, MA, USA*
[2] *Educational Psychology and Learning Systems, Florida State University, Tallahassee, FL, USA*
[3] *Efficacy and Learning Research, Pearson, Boston, MA, USA*

DOI: 10.21585/ijcses.v5i1.149

Abstract

The Inclusive Assessment of Computational Thinking (CT) designed for accessibility and learner variability was studied in over 50 classes in US schools (grades 3-8). The validation studies of IACT sampled thousands of students to establish IACT's construct and concurrent validity as well as test-retest reliability. IACT items for each CT practice were correlated to examine construct validity. The CT pre-measures were correlated with post-measures to examine test-retest reliability. The CT post-measures were correlated with external measures to examine concurrent validity. IACT studies showed moderate evidence of test-retest reliability and concurrent validity and low to moderate evidence of construct validity for an aggregated measure of CT, but weaker validity and reliability evidence for individual CT practices. These findings were similar for students with and without IEPs or 504s. IACT is the first CT tool for grades 3-8 that has been validated in a large-scale study among students with and without IEPs or 504s. While improvements are needed for stronger validity, it is a promising start.

Keywords: computational thinking, assessment, game-based learning, neurodiversity

1. Introduction

Computational Thinking (CT) has been attracting increased attention over the past decade in K–12 education, prompting a call for new models of pedagogy, instruction, and assessment (Shute, Sun, & Asbell-Clarke, 2017; CSTA, 2017; National Academy of Sciences, 2010). The explosion of CT in education today parallels the turn towards scientific inquiry in science education in the 1990s (AAAS, 1993; Duschl, 1990: NRC, 1996). In 1996, the National Research Council issued new science education standards focusing on inquiry practices (NRC, 1996), yet even in 2018 there were few widely accepted tools for assessing students' scientific inquiry in classroom settings (Kruit, Oostdam, van den Berg, & Schuitema, 2018).

Developing learning assessments for any new focus of education is particularly challenging. In most educational research, new assessment methods are validated using existing "standard" measures of learning in the same content area. With an emerging field such as CT, no such standard measures exist. The few items that are in development and validation in today's research rely heavily on text and coding, which may preclude the measurement of CT for a broad range of learners. It is not only the novelty of the field that challenges the development of assessment in CT, it is also the nature of CT itself. Like scientific inquiry, CT is a thinking process. Measuring thinking processes is more nuanced than assessing whether

or not a learner can solve a math problem or define a science term. Measuring learners' abilities to plan, design, and solve complex problems require methods for making thinking visible, which is not done by a typical school test (Ritchhart, Church, & Morrison, 2011). Even when CT is applied in a natural setting, such as in a coding environment, the final product may not reveal the CT practices as much as the thinking processes involved in designing code (Grover & Basu, 2017).

Addressing these issues for assessing CT may be particularly important to broadening participation in Computer Science and other Science, Technology, Engineering, and Mathematics (STEM) fields. Learning assessments often include irrelevant barriers (e.g., reading or coding prerequisites) that may mask conceptual understanding for some learners (Haladyna & Downing, 2004). Many learners who struggle academically because of neurodiverse conditions may have 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). Recognizing and nurturing these talents may be crucial for developing our future workforce (Martinuzzi & Krumay, 2013). In fact, many large IT companies, including Microsoft and Google, have programs specifically designed to recruit neurodiverse individuals (Wang, 2014). To capture this valuable expertise without the extraneous barriers that limit many neurodiverse learners' participation, a new form of learning assessment for CT is required.

This paper reports on the exploration of assessment items intended to measure CT within a game-based learning research study. Interactive Assessments of CT (IACT), designed for upper elementary- and middle-school students (grades 3–8), is a set of interactive, online, logic puzzles that were created to measure four fundamental CT practices: Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm Design. The IACT assessment items were originally designed to be used as pre/post measures of CT practices in a study of the logic puzzle game *Zoombinis* (Asbell-Clarke, Rowe, Almeda, Edwards, Bardar, Gasca, Baker, & Scruggs, 2020). They were intended to identify evidence of CT Practices that: a) are apparent during the *process* of solving a task, as opposed to a final product; and b) are independent of a specific application and thus transferable or generalizable to other tasks. They were also designed to use as little text, specific coding notation, or other features that might impede some learners and/or mask their ability to solve CT problems. Because the study specifically included learners who have Individual Education Plans (IEPs) related to academic struggles, the assessments needed to avoid extraneous factors that can impede some learners, such as the need to read and interpret complex word problems and excessive time pressure. The final constraint placed on the assessments was that they needed to be completed by all students within one class period (40-50 minutes, depending on the district). Students with IEPs were allowed up to 50 percent more time to complete the IACT assessments if necessary. In this paper, we report the findings from validation studies of IACT using two samples of elementary- and middle-school students, each with thousands of students, to understand IACT's construct and concurrent validity as well as test-retest reliability.

2. Background

The online logic puzzles that make up IACT were designed to serve as external pre/post assessments in a national, game-based learning study of over 50 upper elementary- and middle-school classes during a study of implicit CT practices demonstrated in *Zoombinis* gameplay in the 2017-18 school year (Asbell-Clarke, et al., 2020). Unfortunately, no validated instrument was available to measure CT practices at these grade levels when we started the study, so we designed two sets of IACT logic puzzles, one version for upper elementary and one version for middle school, each version with two comparable forms. During the *Zoombinis* study, we collected the pre/post IACT data along with teacher ratings of their students' CT at the end of the study. We used the teacher ratings to try to establish the concurrent validity of the IACT items. Because this method was not as rigorous as we would have liked, we extended the study to a second district-wide sample of over 3,000 students in grades 2–8 in a mid-sized Northeastern public school district. During the 2017-18 and 2018-19 school years, we were able to collect IACT data in May/June of each school year as well as corresponding items from another external instrument (Bebras) for the students in grades 5–8 in 2018-19. This paper reports on the findings of both of these samples for the validation studies of IACT.

2.1 Background on the Measurement of Computational Thinking

CT is a way of thinking used to design systematic and replicable ways to solve problems, emphasizing Abstraction and Algorithmic Thinking (Shute, Sun, & Asbell-Clarke, 2017; Wing, 2006). Rooted in ideas from research for the LOGO environment (Papert, 1980, 1991), CT includes practices and understandings dealing 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). Barr and Stephenson (2011) suggest that, in K–12, CT involves problem-solving skills and particular dispositions, such as confidence and persistence, when confronting particular problems. 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).

In designing a middle-school curriculum called Foundations for Advancing Computational Thinking (FACT), Grover, Cooper and Pea (2014) used pedagogical strategies to support transfer from block-based to text-based programming, along with formative and summative assessments (including quizzes and tests as well as open-ended programming assignments) related to the acquisition of computational thinking skills. Their findings show that students ages 11–14 using the FACT curriculum experience improved algorithmic learning, understanding of computing, and transfer of skills from the introductory programming environment, *Scratch*, to a text-based programming context. Building on this research, Lundh, Grover, Jackiw, and Basu (2018) suggest a framing of Variables, Expressions, Loops, and Algorithms (VELA) to prepare young learners for CT.

Many of the CT assessments developed to date are strongly tied to computer-science frameworks and rely on the construction or analysis of coding artifacts (Tang, Yin, Lin, Hadad, & Zhai, 2020). 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. 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).

Recent initiatives to integrate CT with STEM require assessments that are more decontextualized or domain-general (Tang, et al., 2020; Karalar, & Alpaslan. 2021). The Computational Thinking test (CTt) (González, 2015) and Bebras Tasks (Dagienė & Futschek, 2008; Dagienė, Stupurienė, & Vinikienė, 2016) are two such instruments that have shown promise in assessing core CT constructs for middle-grades students (Wiebe et al., 2019). The CTt is an online, 28-item, multiple choice instrument shown to be valid and reliable with middle-school students in Spain (Román-González, Moreno-León, & Robles, 2017). Although designed for students with no programming experience, some items on the CTt have block-based, programming-like elements in them. However, research studies have not shown this 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, which originated as a set of short competition tasks through which students in grades 5–12 apply CT to solve "real life" problems, have recently been looked at as assessment tools because their items map well to CT constructs (Barendsen et. al., 2015; Dagienė, Stupurienė, & Vinikien, 2016; Izu, Mirolo, Settle, Mannila, & Stupurienė, 2017). Like the CTt, Bebras Tasks do not rely on prior knowledge of an application or programming language, which makes them well-suited for use as a pre-assessment tool. 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) for Bebras Tasks to stand alone as a standard assessment for CT in K–12 education. However, Wiebe and colleagues (2019) explored a promising hybrid assessment that includes items from

both the CTt and Bebras as a "lean" assessment of current, generally recognized core CT skills. The Bebras items were most closely related to the intended constructs, grade band, and the nature of the logic puzzles that are the focus of this study, so we used selected Bebras items as external measures for evidence of concurrent validity for the logic puzzles with the second sample. The Bebras items are, however, more dependent on text than IACT and may present difficulties for students with certain IEPs.

2.2 Description of IACT Items

To measure foundational CT in grades 3–8, we developed a set of interactive logic puzzles focusing on four fundamental CT practices: Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm Design. We chose to design a set of online puzzles because we were working with classes already using a web-based game, and the delivery and data collection for the assessment items could be integrated with the delivery of the game.

Knowing that these CT practices are rarely mutually exclusive within a set of activities, we identified a set of puzzles that might emphasize one practice over the others even if all practices were part of the activity. We designed the puzzles with minimal text and minimal prerequisite experience with coding or other specific activities. We drew inspiration from puzzle formats often used in psychological assessments, avoiding text and context-dependent scenarios. While these assessments are used in a variety of contexts involving executive functioning and reasoning; the overlap with CT practices is intriguing and merits study.

Theoretical Framing of Computational Thinking used in IACT

IACT was designed to measure the CT practices evident within the learning game *Zoombinis*. While not intended to include all potential facets of CT, IACT is grounded in emergent theoretical literature that is helping define the evolving constructs of CT in the educational field. The term CT was introduced by Jeanette Wing (2006) to describe the thought processes involved in formulating problems so that the solutions are represented in a form that can be effectively carried out by an information-processing agent (Cuny, Snyder, & Wing, 2010). The role of CT in K–12 education has been described as laying "the conceptual foundation required to solve problems effectively and efficiently (i.e., algorithmically, with or without the assistance of computers) with solutions that are reusable in different contexts" (Shute, Sun, & Asbell-Clarke, 2017). While many CT practices were discussed in Seymour Papert's research on procedural thinking in the early programming environment for children called LOGO (Papert, 1980; Papert & Harel, 1991), today CT is thought to encompasses much more than programming. There is also evidence that these CT practices may support a variety of other cognitive and non-cognitive activities, especially for learning in STEM subjects (e.g., Barr & Stephenson, 2011; Sneider, Stephenson, Schafer, & Flick, 2014).

Domain-general CT is often operationalized as a set of practices that include: problem decomposition, abstraction, algorithmic thinking, conditional logic, recursive thinking, and debugging (CSTA, Shute et al., 2017; Tang, et al., 2020). For the development of IACT, we focus on the CT practices that were most closely related conceptually to the puzzles in *Zoombinis* gameplay. We selected four fundamental CT practices outlined by CSTA (2017) and Shute, Sun, & Asbell-Clarke (2017):

- *Problem Decomposition* is reducing the complexity of a problem by breaking it into smaller, more manageable parts.
- *Pattern Recognition* is seeing trends and groupings in a collection of objects, tasks, or information.
- *Abstraction* is generalizing from observed patterns and making general rules or classifications about objects, tasks, or information by discerning relevant from irrelevant information.
- *Algorithm Design* is establishing reusable procedures that solve sets of problems.

While not an exclusive definition of CT, a focus on these practices lays a strong foundation for CT (CSTA, 2017). While CT can include many other practices such as modelling, debugging, and data visualization, this study focuses on these four CT practices because they are highly related to *Zoombinis* gameplay and they show promise of generalization to problem-solving in a variety of disciplines. When educating young learners in upper elementary and middle school, it may be important to ensure these

broadly applicable practices have a solid foundation and upon which more nuanced facets of CT can be built.

2.3 Design of the IACT Items

The IACT items were designed for use in a game-based learning study where students may not have had any previous exposure to computer science or coding. While not designed as clinical assessments of executive functioning, the IACT items drew from models from similar psychological assessments that were designed for a broad range of neurodiverse learners to ensure maximum accessibility.

The authors worked with a game-based learning assessment company to design the IACT logic puzzles. Two sets of IACT items containing similar logic puzzle items were designed, one for upper elementary- and one for middle-school learners. The item sets were conceptually and structurally the same for both grade bands, but differed in terms of difficulty (e.g., based on the number of variables to consider in a pattern and size of the array for Abstraction problems). The item sets were distributed across two comparable forms for each grade band, a pre-test and a post-test, that were balanced and could serve as external pre/post measures of gains in our game-based learning studies. All items went through a minimum of two rounds of iteration and testing with think-aloud interviews with 8-10 students per round to test that the wording was eliciting the CT practices of interest.

The four fundamental CT practices that were evident in the *Zoombinis* gameplay (excerpted from Asbell-Clarke, et al., 2020), and thus formed the constructs measured with IACT are:

- ***Problem Decomposition***: When approaching a complex problem, learners may need to simplify the problem—decomposing it into manageable parts and then tackling one part at a time. This is comparable to the practice of isolating variables in a science experiment or to factoring equations into terms in mathematics. Everyday examples of problem decomposition include taking the steps to bake a cake (choosing a recipe, gathering ingredients, mixing batter, baking, and frosting), or when planning a party (dealing with the guest list, then the menu, and then the music). When confronted with a multi-faceted puzzle (for example, sorting objects by both shape and colour), players often need to consider one part of the puzzle at a time (shape) and then consider the other (colour).
- ***Pattern Recognition***: Pattern Recognition is required for all kinds of sorting and classification tasks as well as seeing trends in data and other forms of information. In science, learners look for patterns in the characteristics of animals to classify them into species, and patterns in the motions of planets to understand the laws of the universe. In math, learners use patterns to understand numbers and units, to collect like terms in an equation, and to generalize problems into classes of problems. Pattern Recognition is the precursor for abstraction, which is at the heart of CT.
- ***Abstraction***: Abstraction is the ability to rise above the details and see the rules that can be applied generally to other situations. When learners can look across multiple instantiations of a phenomenon and draw the common characteristics or patterns that can be abstracted, they are able to design generalized solutions to problems. For example, scientists are able to generalize the laws of gravity from the vast amount of observed evidence, and patterns within the evidence, that enable an abstracted claim about how the universe works. Similarly, mathematicians create systems of numbers and representations to exploit the inherent patterns of quantity in our world. The goal of abstraction is to design replicable systems of solutions that help us effectively and efficiently meet new challenges.
- ***Algorithm Design***: Once abstracted, a set of rules can be operationalized through an algorithm. An algorithm is a sequence of instructions or steps required to accomplish a task. Everyday examples of algorithms include recipes in a cookbook, and consistent daily routines used to accomplish everyday tasks. Scientists design algorithms for replicable experimentation and for automated procedures required in large-scale data collection and analysis. Algorithms are used constantly in math ranging from standard processes of multiplication and division, all the way to abstract computer modelling of sophisticated phenomenon.

The example IACT items described in this section are from the elementary school test. A set of items for each CT practice was preceded by a warm-up item to familiarize students with the mechanics of the item. Descriptions and illustrations of the test items are outlined in Table 1 and described in more detail below.

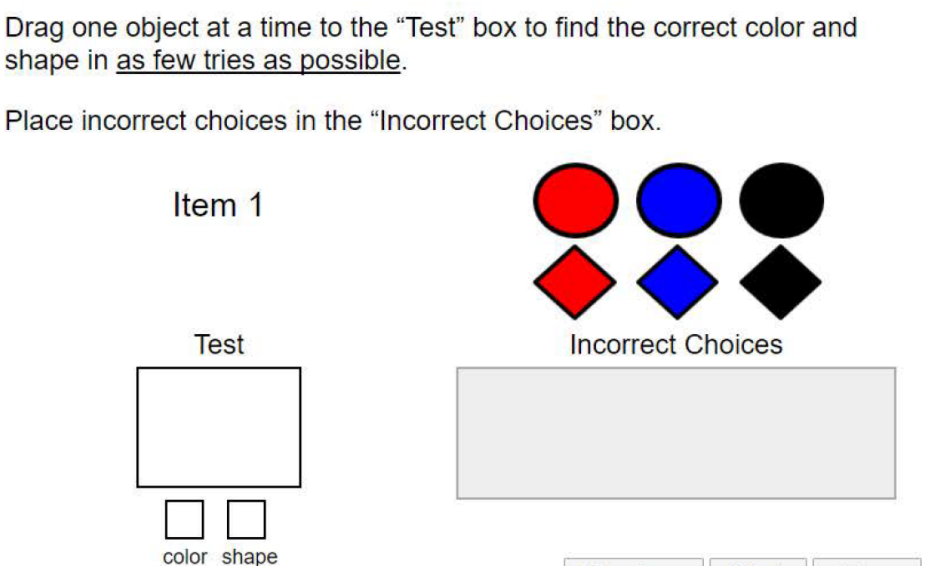
Table 1: Operationalization of CT Practices in IACT Assessment Items

CT Practice	Puzzle Type	Task	Measure
Problem Decomposition	Mastermind	Identify combination of colour and shape of item through testing	Efficiency: ratio of moves to required moves
Pattern Recognition	Raven's Progressive Matrix	Select one piece that best completes a pattern	Number correct
Abstraction	Sudoku	Fill grid spaces with coloured shapes according to a general rule	Percentage correct
Algorithm Design	Maze solving	Design a sequence of moves to complete the maze	Efficiency: ratio of moves to required moves

Problem Decomposition

Items related to problem decomposition involve a series of progressively harder puzzles that are similar to the game *Mastermind*. Students use feedback from the item to figure out which values (combination of colour and/or shape and/or pattern) solve the puzzle, The puzzle mechanic requires the student to drag objects to the test box to get feedback (i.e., correct or incorrect) with regard to colour and shape in as few moves as possible. Figure 1 shows an easy problem. The correct answer for this item is "red diamond." If the student first placed the red circle in the test box, a green check will appear for "colour" and a red X would appear for shape. This tells the student to continue using a red object but not a circle. This leaves only the red diamond option, which is correct. The number of moves to solve the problem reflects the efficiency of problem decomposition skill.

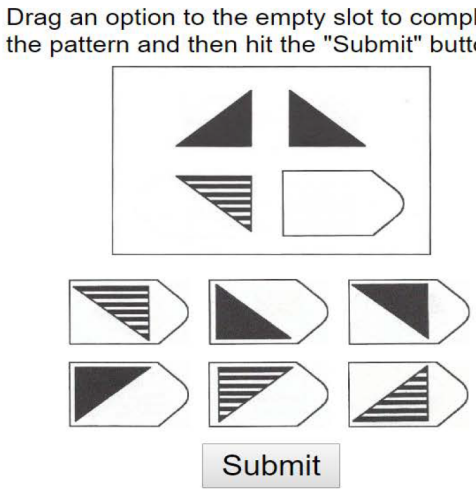
Figure 1. Example logic puzzle item targeting Problem Decomposition



Pattern Recognition

Raven's Progressive Matrices (RPM) (Raven, 1981) were used to assess Pattern Recognition. The RPM items serve as a baseline of learners' ability to infer and apply different patterns in increasingly complex situations. RPM were designed to measure abstract reasoning involving patterns, and Raven (2000) pointed out that the RPM focuses on two components of general cognitive ability—making sense out of apparent chaos and generating a high-level schema to handle complexity. In the fairly easy item shown in Figure 2, the student needs to recognize that the top two black triangles are mirror images of each other, thus the bottom two should also be mirror images. Option 5 (middle item in the bottom row) is the correct response.

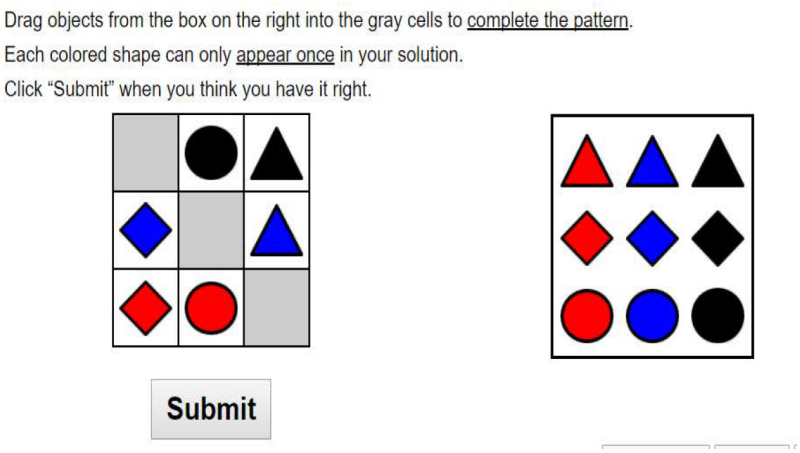
Figure 2. Example logic puzzle item targeting Pattern Recognition



Abstraction

To assess Abstraction, pattern-matching puzzles were used. These puzzles require students to identify an underlying general rule associated with the patterns of objects and complete the puzzle by applying that rule. As shown in Figure 3, students drag objects from an inventory on the right into the grey cells on the left to complete the pattern, and thus applying the inferred rule. Each coloured shape can only appear once in the solution. In the example shown, the underlying pattern is relatively easy to discern—rows are the same colour, and columns are the same shape. Thus, the correct answer would be black square (upper left), blue circle (middle), and red triangle (lower right). In later tasks, the patterns are more complex and have more cells, thus generating more complex rules.

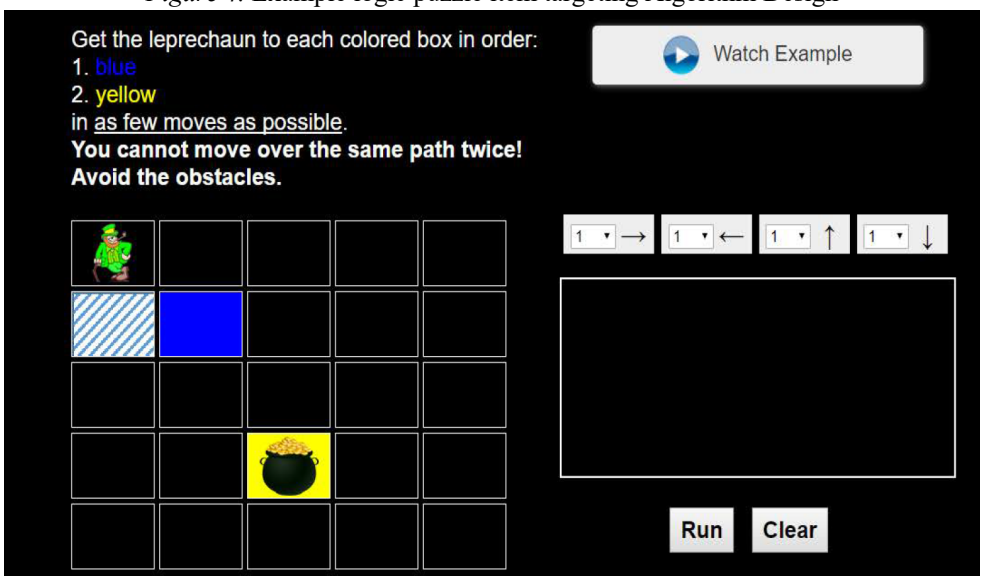
Figure 3. Example logic puzzle item targeting Abstraction



Algorithm Design

To assess Algorithm Design, the puzzles require a student to set up a sequence of arrows that will guide a character along a path in a maze that follows specified criteria. As shown in Figure 4, the sequencing task requires the student to insert directional arrows, along with the number of iterations needed to guide a leprechaun to a pot of gold in the fewest steps possible, while avoiding certain obstacles. The fewest steps possible in this case is five (one step to the right, one step down, one step to the right, and two steps down).

Figure 4. Example logic puzzle item targeting Algorithm Design



These items were designed to measure individual CT practices as well as being aggregated for an overall CT measure.

The remainder of this paper discusses evidence for the validity and reliability of the IACT logic puzzles as measures of CT practices.

3. Research Questions & Hypotheses

The central question guiding this research is the extent to which the interactive logic puzzles provide a valid and reliable assessment of upper elementary- and middle-school learners' CT practices. To study this question, the results from students' performance on the puzzle tasks were examined using four types of analyses to test specific types of validity and reliability evidence with the following hypotheses about the results of each of these analyses:

- ***Study 1***: Correlations among each set of IACT items (associated with each CT practice) were examined to establish evidence of construct validity—that all measures are distinct facets of the same broader CT construct.

  ***Hypothesis 1*:** The CT practice measures will be moderately correlated with one another. Since the CT measures examine practices that are all facets of the same broader CT construct, we hypothesize that they will generally be aligned and share variance along the dimension of CT.
  ***Study 2*:** The CT pre-measures were correlated with post-measures to establish evidence of IACT test-retest reliability.
  ***Hypothesis 2*:** The pre-measures will be moderately to highly correlated with the post-measures, suggesting the versions of the logic puzzles are measuring the same construct at each time. The two sets of items designed were assumed to be equivalent and randomly assigned between the pre- and post-measures. It is intended that the variation between pre-and post-measures be attributable only to changes in the learners' practices rather than differences in the test questions themselves.

- ***Study 3*:** The CT post-measures were correlated with external measures (i.e., teacher ratings of their students' CT practices for one sample and students' scores on Bebras items for another sample) to examine concurrent validity.
  Hypothesis 3: We hypothesize that the CT measures will be moderately correlated with teacher ratings of each CT skills and student scores of Bebras items, at the aggregate level and possibly at the level of each individual CT practice. The latter is questionable because of the amount of overlap among the individual practices.
- ***Study 4*: The validity and reliability analyses in studies 1-3 will be repeated separately for students with and without IEPs or 504 plan.**
  Hypothesis 4: There will be no significant differences in the validity and reliability of IACT scores by IEP/504 status of the students.

3.1 Methods

The validation studies for IACT took place within other research studies. The first sample was collected during a national study of classes in grades 3-8 using the CT learning game, *Zoombinis*. This sample is referred to as the *Zoombinis* sample. The second sample was collected during a longitudinal study of the development of CT in grades 3–8 as part of our Research-Practice Partnership (RPP) with a mid-size suburban district outside a major Northeastern U.S. city. This sample is referred to as the RPP sample.

Zoombinis *Sample*

During the 2017-18 academic year, 146 teachers from 37 states and 6 countries applied to participate in the *Zoombinis* classroom implementation study. To participate, teachers needed to meet the following criteria:

- They are an elementary- or middle-school educator (grades 3–8) in the U.S.
- They teach at least one class that supports CT through logic, coding, or preparation for coding (e.g., math, science, computer science, tech ed., etc.).
- Their students have access to Internet-enabled computers to take the pre- and post- assessments required for the study.
- They complete a teacher agreement outlining the study requirements.
- They obtain administrative approval to participate in the study.

Forty-one teachers met all of these criteria and were accepted into the study. These teachers taught a total of 146 classes (10 charter, 21 private, 115 public). They were allowed to contribute a maximum of 3 unique classes in the study. To qualify as a unique class, the class must have either covered different subject areas or different grade levels of the same subject area. If teachers used *Zoombinis* in duplicate sections of these classes, their students took the pre- and post-assessments, but the teachers did not complete the other study data collection requirements. These classes are considered "non-study" classes to avoid oversampling in the research studies, but they were retained for the validity and reliability analyses reported here. Fifty-seven of these classes were labelled as study classes (6 charter, 4 private, 47 public) and 91 were non-study classes (4 charter, 17 private, 70 public).

The initial *Zoombinis* student sample consisted of 3,234 elementary- and middle-school students across 146 classes from charter, private, and public schools (see Table 2). Of these, 2,456 students completed the pre-assessment and 1,828 completed the post-assessment. A total of 1,498 students completed both measures, belonging to 101 classes across 37 teachers. From the subset of 1,828 who completed the post-assessments, 851 students did not have a complete set of teacher ratings of their CT practices. These students were also excluded from the concurrent validity analyses, resulting in a sample size of 977 students with post-assessment and teacher ratings data.

RPP Sample.

Our second sample was collected from a small, suburban public-school district in the Northeastern U.S. as part of an RPP that has the mission to promote the infusion of CT into existing STEM curricula. As part of the longitudinal study on the impact of the RPP on students' CT practices in 2017 - 2020, logic puzzles are administered to all students in grades 2–8 at the end of each school year. In

grades 6–8, science teachers administered the logic puzzles during class time. In grades 2–5, technology teachers administered the logic puzzles during technology class time. All logic puzzles were completed in May-June of 2018 and 2019, because they are rising 6th graders 5th grade students took the middle-school forms of the test. For the RPP middle-school sample (grades 5–8), we also added five Bebras items to the pre- and post- assessments to provide evidence of concurrent validity.

The initial RPP student sample was comprised of 3,402 elementary- and middle-school students across the district in grades 2–8 (see Table 2) for 2017-18 and 3,697 students for 2018-19. Students who "never joined," "withdrew" from the study, or did not have a complete set of pre-test and post-test scores were excluded from further analyses. The reduced samples across 6 elementary and 2 middle schools consisted of 3,066 students for 2017-18 and 2,909 students for 2018-19. A set of 1,414 students had complete pre-test and post-test scores during the first and second years of data collection---23% of these students had IEPs/504s (*N*=337), slightly above the national percentage of students with learning disabilities (Digest for Education Statistics, 2016; Horowitz, Rawe, Whittaker, 2017).

Table 2. Description of student samples

	Zoombinis			**RPP**		
	Pre-Assessment	Post Assessment	Both	2017-18	2018-19	Both
Total sample of students	3,234	3,126		3,402	3,697	
Never joined or withdrew from the study	262	154		146	300	
Incomplete Assessment Data	533	1155		178	476	
Final sample of students	2,439	1,817	1,435	3,078	2,921	2,301
Analyses	**Construct Validity**	**Construct Validity**	**Test-Retest**	**Construct Validity**	**Construct Validity**	**Test-Retest**

Data Cleaning

There were 2,788 *Zoombinis* students who either had pre-test or post-assessments. Two outliers from the performance on Algorithm Design (mean number of moves) were dropped. 18 students were excluded because they had completed an incorrect assessment for their grade level. This resulted in a total of 2,768 *Zoombinis* students. There were 3,666 RPP students who either had pre- or post-tests. Most of the students who had data from one year only were aging in (2nd grade in year 2) or aging out (8th grade in Year 1) of the sample. The final sample of RPP students with data from both years was 1,414. Details of data cleaning to arrive at the final sample of students who completed pre- and post-assessments can be found in Appendix A.

3.2 Data Collection

In each study, a variety of data was collected: student pre-post assessments, teacher logs of their CT instructional practices, and teacher interviews. In addition, as external measures of CT, teacher ratings of their students' CT practices were collected in the *Zoombinis* sample, and student scores on Bebras items were used for middle-school students (grades 5–8) in the RPP sample. The pre-post assessments and, when available, teacher CT ratings or student scores on Bebras items were used for the validation study of IACT assessments reported on in this paper.

Assessment Data Collection

In the IACT pre and post-tests, there were 3-6 items of increasing difficulty levels for each of the 4 practices of CT. All items had time limits for completion—2 minutes for the easier items and 5 minutes

for the more difficult items. Teachers agreed to allot 30–45 minutes of class time for the administration of the pre-assessment and again for the post-assessment. Assessments were designed to take 30 minutes to allow students 150 percent of that to complete the assessment.

For the *Zoombinis* sample, teachers decided when to administer the pre-post assessments based on when *Zoombinis* best fit into their CT instruction. Teachers asked their students to complete the pre-assessments before they started playing *Zoombinis*. When teachers completed their CT instruction, they administered the post-assessment.

For the RPP sample, the IACT items were administered near the end of each school year through the district in grades 2–8. The data from Spring 2017 is used as the Time 1 measure for this study and data from Spring 2018 is used as the Time 2 measure.

All IACT data were collected through our team's game data architecture, *Data Arcade*. *Data Arcade* facilitated the collection of all pre-assessment data and unlocked the game once the pre-assessment had been completed. Teachers created non-identifying usernames for their students in *Data Arcade*. Only teachers knew the real identities of students in their classes. *Data Arcade* then assigned a unique password and UserID number to each student. Teachers, in turn, shared the usernames and passwords with the students. This UserID was used to link assessment, game, CT rating, and Bebras data.

IACT Scoring

Pre-post assessment scale scores were calculated for the IACT logic puzzle items as the means of items per category: Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm Design. This was calculated slightly differently for each set of tasks. The Problem Decomposition tasks provided feedback after each move, and the number of moves was unlimited, so the scores relied on the mean efficiency a student used to solve the puzzles. Efficiency is defined as the number of moves the student took divided by the minimum number of moves needed to solve the puzzle. In cases where the player lucked into getting a solution in less than the minimum number of moves, their efficiency was given a value of 1. The Pattern Recognition tasks simply had the student choose a response, so the scoring used the mean number of correct responses. Because the Abstraction puzzles allowed for individual array spaces to be counted as incorrect or correct, the mean percentage of spaces with correct responses was used. The Algorithm Design puzzles allowed for testing so the mean efficiency (# moves / minimum # moves needed) was also used in scoring. Because the first items for each category were used for practice, these items were dropped from mean calculations. Table 3 describes the calculations of the scale scores for each CT practice.

The middle school form was designed to be more difficult than the elementary form. To account for this, scores were standardized by form (elementary vs. middle school). The standardized scores for the CT practices were examined individually and in aggregate (Table 3). An aggregate measure of CT was calculated by first standardizing the means of each item type to produce a Z-score for each CT practice. The final Z-scores were averaged to create the aggregate CT measure used in this study. The units are the number of standard deviations from the mean Z-score of the four CT practices.

Table 3. Scoring of assessment items for each CT practice

CT Practice	Number of items	Measure used for scoring
Problem Decomposition	4	Mean efficiency (# moves/min # moves)
Pattern Recognition	5	Mean number of correct responses
Abstraction	6	Mean percentage of array spaces completed correctly
Algorithm Design	3	Mean efficiency (# moves/min # moves)
Aggregated CT	18	Average of Z-scores of 4 CT practice measures above

If a student had 0 number of moves on one of the Problem Decomposition or Algorithm Design items, this indicated that the student timed out of solving the puzzle. These timed-out instances were excluded from the computation of the mean efficiency in Problem Decomposition and Algorithm Design. Appendix B summarize the number of students who had complete, timed out, and missing pre-post assessments (mean efficiency) for Problem Decomposition and Algorithm Design, respectively.

Teacher CT Ratings Sheets

A CT rating sheet was designed and reviewed by 3-4 teachers before its use in the full study. Teachers in the study were given a brief description of the instrument by a research team member, along with its purpose and how to use it. After they administered the post-assessments, Zoombinis teachers were asked to rate each of their students based on the 4 CT practices in their students' work. This was an attempt to have an external measure of CT that still did not rely on text responses or coding. Sample behaviours were provided with a rubric so that teachers had a shared definition of each CT practice (See Table 4).

Table 4. Teacher ratings of their students' CT practices: Definitions and sample behaviours

CT Practice Definition	Sample Behaviours
Problem Decomposition: Breaking a problem into smaller, more manageable parts	1. When faced with a complex task, breaks it into smaller, simpler tasks.
	2. Considers one variable at a time when thinking about cause and effect.
Pattern Recognition: Identifying patterns, trends, or similarities between things	1. Identifies similarities and differences in sets of objects.
	2. Applies a pattern to predict an outcome.
Abstraction: Removing specific differences/details to make a generalized solution that will work for multiple problems	1. Identifies general rules to explain trends and patterns.
	2. Identifies common strategies that can apply to many problems
Algorithm Design: Creating an ordered series of instructions for solving a problem or performing a task	1. Writes or describes exact set of instructions for a complex task
	2. Recognizes the importance of the order of events in solving a problem.

Teachers received a Google Sheet with the *Data Arcade* usernames of students in each of their classes. Next to each student's username was a dropdown 5-point rating scale: Great Extent (5), Large Extent (4), Moderate Extent (3), Slight Extent (2), Not at All (1). For each student, teachers were asked to select one rating for each CT practice based on the extent to which they had seen those behaviours in their students' work and overall classroom practices.

Bebras Items

Because the CT Rating sheet depended on teacher's ratings without substantive preparation or guidance, we also wanted to use a set of relatively established external items to compare results with the IACT items. We selected the Bebras tasks because they were closest to our needs, but we still had reservations that they would adequately measure CT practices among neurodiverse learners. We selected five items from Bebras that aligned with the four CT practices of our study. The first item was a maze task where students were to sequence a series of arrows to send a robot through a maze. This is analogous to the IACT items for Algorithm Design. The second item was a pattern matching game where shapes were combined to make another shape, analogous to the Raven's Progressive Matrices we used in IACT to measure Pattern Recognition. The third and fourth items were Problem Decomposition items analogous

to the IACT Problem Decomposition item that mimicked the game *Mastermind*, but with considerably more text. The fifth Bebras item required students to generalize a rule to break a code, similar to the IACT Abstraction items. The five Bebras items used in this study are provided in Appendix C.

IEP/504 Plan Status

The RPP school district provided IEP/504 plan status for all students in their district each year of the study. Only students with IEP or 504 status of 'Active' in a specific school year were categorized as having an IEP/504 plan.

4. Results

To study the construct validity and reliability of the IACT items, addressing the first and second hypotheses, a series of correlational analyses were conducted using Pearson correlations with the following items: 1) pre-test measures, 2) post-test measures, and 3) test-retest of the same CT practice. To address the third hypothesis, namely concurrent validity, Pearson correlations were computed between post-test measures and teacher ratings of their students' CT for the *Zoombinis* sample, and between post-test measures and Bebras items for the RPP sample. All of these analyses were completed separately for students with and without IEP/504 plans to address the fourth hypothesis. Across all findings, mean moves was expected to be negatively correlated with mean number of correct responses and mean percentage of correct responses, as higher mean moves suggested less efficient solutions in the pre-test and post-test measures.

Construct Validity

Tables 5 and 6 display the correlations among the standardized IACT measures for the *Zoombinis* and RPP samples. In both samples, there are low to moderate correlations among the measures for both the pre- and the post-assessments (see Appendix D for the complete list of correlations). This supports the first hypothesis that the measures of the CT practices are similar yet distinct. The intercorrelation is highest between Pattern Recognition and Abstraction, which points to the strong dependence of these practices. Abstraction can be thought of as the generalization of observed patterns into a rule or category, so it is reasonable that students who are strong in Abstraction would also be strong in Pattern Recognition.

Table 5. Pearson intercorrelations of pre-assessment measures for the *Zoombinis* and RPP samples

Correlations between CT Practice	**CT Practice**	***Zoombinis* sample (*N*= 2206-2314)**	**RPP sample (*N*= 2732-2937)**	**Average across samples**
Problem Decomposition Avg Efficiency)	Pattern Recognition (Correct)	0.18	0.12	0.15
	Abstraction (Percent Correct Spaces)	0.23	0.16	0.20
	Algorithm Design (Avg Efficiency)	0.27	0.14	0.21
Pattern Recognition (Correct)	Abstraction (Percent Correct)	0.32	0.30	0.31
	Algorithm Design (Avg Efficiency)	0.24	0.21	0.23

Abstraction (Percent Correct Spaces)	Algorithm Design (Avg Efficiency)	0.26	0.26	0.26

Note: Significant at an alpha level of 0.0001.

Table 6. Pearson intercorrelations of post-assessment measures for *Zoombinis* and RPP samples

Correlations between CT Practices	**CT Practice**	***Zoombinis* sample (*N*= 1600-1773)**	**RPP sample (*N*= 2315-2623)**	**Average across samples**
Problem Decomposition Avg Efficiency)	Pattern Recognition (Correct)	0.19	0.12	0.16
	Abstraction (Percent Correct)	0.24	0.20	0.22
	Algorithm Design (Avg Efficiency)	0.22	0.14	0.18
Pattern Recognition (% Correct)	Abstraction (Percent Correct)	0.35	0.31	0.33
	Algorithm Design (Avg Efficiency)	0.23	0.23	0.23
Abstraction (Percent Correct Spaces)	Algorithm Design (Avg Efficiency)	0.24	0.23	0.24

Note: Significant at an alpha level of 0.0001.

Test-Retest Reliability

Table 7 displays the correlations for test-retest reliability among the standardized CT measures. Correlation coefficients may be higher for the *Zoombinis* sample than the RPP sample because all students in *Zoombinis* classrooms experienced some degree of CT intervention whereas this was true for less than a third of the RPP sample. There were varied results for the measures of individual CT practices, with acceptable test-reliability for the aggregated CT measure but below what was expected for the individual practices across both samples. The Pattern Recognition items were Raven's Progressive Matrices drawn from a public sample on the Internet. While there are no published test-retest results for these particular sets of items, this research typically has test-retest reliability of between 0.70 and 0.85 (e.g., Abdel-Khalek, 2005; Raven, Raven, & Court, 2000).

Table 7. Results for test-retest reliability

	***Zoombinis* sample (*N*= 1330-1434)**	**RPP sample (*N*=1955-2299)**	**Average across samples**
Correlation coefficients for test-retest reliability	Pearson *r*		
Problem Decomposition (Avg Efficiency)	0.26	0.23	0.25
Pattern Recognition (% Correct)	0.21	0.14	0.18

Abstraction (% Correct Spaces)	0.38	0.41	0.40
Algorithm Design (Avg Efficiency)	0.27	0.18	0.23
Aggregated CT	0.55	0.43	0.49

Note: Significant at an alpha level of 0.0001.

While results for the measures of individual CT practices are considerably lower than what is indicated in prior literature, the finding for the aggregated CT measure in the *Zoombinis* sample indicate strong test-retest reliability. The aggregated CT measure is more stable across time as compared to measures of individual CT practices.

Concurrent Validity

Standardized measures of individual CT practices from IACT did not strongly correlate with the individual practices measured by the teacher ratings and scores on Bebras items (see Appendix E for results from the *Zoombinis* sample and the RPP samples, respectively). The correlations between the IACT measures and the external measures were no higher for corresponding practices than they were for non-corresponding practices. Neither teacher ratings nor student performance on comparable Bebras items were able to distinguish well between the individual practices of CT. This may likely be due to the highly overlapping nature of the CT practices discussed earlier. In the *Zoombinis* sample, the correlations between teacher CT ratings were moderately high, ranging from 0.70 to 0.78 for *Zoombinis* (see Appendix F), suggesting that these teachers were not distinguishing between CT practices when rating their students. There was some distinction between practices when using the Bebras items, however, suggesting that it may have been a limitation of the teacher rating sheet in supporting teachers' distinction of the individual CT practices.

As seen in Table 8 while the individual practices were not correlated with the external measures, the aggregated measure of CT was moderately correlated with the teacher CT ratings for the *Zoombinis* sample , $r\,(941) = 0.29$, $p < 0.0001$ and with students' Bebras scores for the RPP sample, $r\,(1408) = 0.40$, $p < 0.0001$. In particular, the IACT aggregated measure of CT was positively associated with an aggregated CT measure using five Bebras items, providing some evidence that these measures assess the same construct. In other words, students who performed better in all four CT practices as measured by IACT were also more likely to answer a higher percentage of the Bebras items correctly. While the correlations were moderate between aggregated measures of CT, the hypothesized moderate relationship between IACT and Bebras items at the individual CT practice level was not found. Those results can be found in Appendix E.

Table 8. Results for concurrent validity

Correlations between External CT measures	***Zoombinis Teacher Ratings* (*N*=941)**	**RPP sample (*N*=1408)**	**Average across samples**
Aggregated CT	0.29	0.40	0.35

Comparison of Reliability and Validity by students with and without IEP/504s

All three analyses above included students with and without IEP/504s. In this section those analyses are repeated separately for students with and without IEP/504s and compared using a Fisher's Z-transformation in order to test the significance of differences between correlations from both groups.

Construct Validity

Tables 9 and 10 display the correlations among the standardized pre and post IACT measures for students with and without IEP/504s in the RPP sample. In both samples, there are low to moderate correlations among the measures for both the pre- and the post-assessments. After transforming these correlations to Fisher's Z scores, there were no significant differences in the construct validity by IEP/504

status of the students. This supports the fourth hypothesis that the measures of the CT practices are similar yet distinct regardless of student IEP/504 status.

Table 9. Pearson intercorrelations of pre-assessment measures for students with and without IEP'/504s in the RPP samples

Correlations between CT Practice	**CT Practice**	***Students with IEP/504s* (*N*= 584-658)**	***Students without IEP/504s* (*N*= 2230-2279)**
Problem Decomposition Avg Efficiency)	Pattern Recognition (Correct)	0.15	0.08
	Abstraction (Percent Correct Spaces)	0.16	0.12
	Algorithm Design (Avg Efficiency)	0.17	0.10
Pattern Recognition (Correct)	Abstraction (Percent Correct)	0.29	0.28
	Algorithm Design (Avg Efficiency)	0.16	0.21
Abstraction (Percent Correct Spaces)	Algorithm Design (Avg Efficiency)	0.27	0.23

Note: No differences between correlations were significant at an alpha level of 0.05.

Table 10. Pearson intercorrelations of post-assessment measures for students with and without IEP'/504s in the RPP samples

Correlations between CT Practice	**CT Practice**	***Students with IEP/504s* (*N*= 462-573)**	***Students without IEP/504s* (*N*= 1853-2049)**
Problem Decomposition Avg Efficiency)	Pattern Recognition (Correct)	0.10	0.09
	Abstraction (Percent Correct Spaces)	0.21	0.16
	Algorithm Design (Avg Efficiency)	0.19	0.10
	Abstraction (Percent Correct)	0.32	0.28

Pattern Recognition (Correct)	Algorithm Design (Avg Efficiency)	0.29	0.20
Abstraction (Percent Correct Spaces)	Algorithm Design (Avg Efficiency)	0.30	0.19

Note: No differences between correlations were significant at an alpha level of 0.05

Test-Retest Reliability

Table 11 displays the correlations for test-retest reliability of the standardized CT measures among students with and without IEP/504 plans. There was no significant difference in test-retest reliability across these groups.

Table 11. Results for test-retest reliability by student IEP/504 status

Pearson r for test-retest reliability	***Students with IEP/504s* (*N*= 337)**	***Students without IEP/504s* (*N*=1077)**
Aggregated CT	0.41	0.37

Note: No differences between correlations were significant at an alpha level of 0.05.

Concurrent Validity

The aggregated measure of CT was moderately correlated with students' Bebras scores for the students with IEP/504 plans, $r\,(275) = 0.34$, $p < 0.0001$, and students without IEP/504 plans, $r\,(1004) = 0.41$, $p < 0.0001$. These correlations were not statistically different providing some evidence that the concurrent validity of the IACT does not differ by student IEP/504 status.

Table 12. Results for concurrent validity by student IEP/504 status

Correlations with IACT and Bebras measures	***Students with IEP/504s* (*N*=275)**	***Students without IEP/504s* (*N*=1004)**
Aggregated CT	0.34	0.41

Note: No differences between correlations were significant at an alpha level of 0.05.

5. Discussion

Validated measures of CT practices were needed to conduct research on game-based learning with the CT learning game, *Zoombinis.* The target audience included learners with IEPs who may have difficulty with textual assessment items and/or have no pre-existing knowledge of any type of coding language (including block-style introductory coding). Because of this we designed the IACT items based on upon similar models from psychological assessments that are typically used with a neurodiverse audience. In one case, for Pattern Recognition, we used an instrument drawn directly from clinical usage, the Raven's Progressive Matrices (Raven, 2000). For the other CT practices, we modified common interactive logic puzzles that used little to no text and required no previous coding experience. These items were designed for use in the *Zoombinis* study, and while they are not exactly aligned with the CT practices themselves, may provide a model for how more generalizable puzzles can be used to assess CT practices with young and neurodiverse learners and outside coding and computer science examples. The IACT items are intended

to measure the four fundamental CT practices—Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm Design—that were most evident in students' *Zoombinis* gameplay.

To explore the validity of these items, data were collected from two samples, along with external measures. The first sample included over 2500 elementary- and middle-school students who took part in a game-based learning study for the game *Zoombinis.* For this sample, we collected IACT data as well as teachers' CT ratings of their students on the same CT practices as IACT. We found with this sample that IACT items showed promise to measure CT, but we lacked a solid external measure for validation. Thus, we extended the study to include a second sample from a district-wide study where assessments were administered at the end of two different school years. For this sample, we collected IACT data as well as teachers' CT ratings of their students on the same CT practices as IACT, and we added Bebras items that aligned with the CT practices that were also collected from middle-school students.

The first hypothesis we studied was that the IACT demonstrated construct validity and thus could independently measure the four practices of CT. This hypothesis was confirmed. The items for each of the CT practices showed distinct results.

The second hypothesis we studied was that the IACT demonstrated a reliable measure over time. The test-retest reliability results between the pre- and post-tests for the individual CT practices were not strong enough to make a clear argument that learners perform consistently on these items for individual practices over time. Findings related to the aggregated measure of CT, however, indicated moderate test-retest reliability suggesting that this is a more consistent measure to use than items related to individual CT practices. This finding suggests that using an aggregated measure of CT can be appropriate for examining change in students' overall CT practices between two different points in time.

In confirmation of the third hypothesis, the aggregated CT assessment showed moderate evidence of concurrent validity. Our research correlated the IACT items to other external measures of these CT practices—a teacher CT rating sheet and Bebras items for the middle-school students in the RPP sample. Learners' overall performance aggregated across the four CT practices correlated with the teacher CT ratings of their students and the students' Bebras scores enough to make an argument for concurrent validity of the IACT items as an overall measure of CT. More refinement is needed for the IACT measures before they could serve as distinct assessments of individual CT practices, and it may be that these practices have too much overlap for distinction.

The final hypothesis, that the reliability and validity of IACT would not different by student IEP/504 status, was confirmed across all three analyses. This supports our decision to design IACT items that were interactive (instead of multiple choice) and relied on limited text.

6. Conclusions

These findings suggest that IACT shows promise to contribute to the field of CT assessment but needs refinement to reach strong validity. In this current research, we have demonstrated moderate test-retest reliability and concurrent validity, and low to moderate construct validity for an aggregated measure of CT. IACT may be able to be further refined to distinguish and assess individual CT practices with future research.

As the field of CT education rapidly moves forward, it is important to establish a body of learning assessments that adequately measure students' practices associated with CT. In particular, it is important that these assessments are designed to capture the strengths and weaknesses demonstrated by a broad range of learners, including learners who may struggle with textual assessments and who have no pre-existing coding experience. This suggests the need for CT assessments that can measure practices without relying on text or coding. The IACT logic puzzles represent an important first step in this endeavor.

Not only are these items among the first with validation studies using a large number of learners, but they also have the unique strength of being designed with accessibility and learner variability in mind. The assessments extract information about students' CT practices in Problem Decomposition, Pattern Recognition, Abstraction, and Algorithm Design through students' activity in a set of logic puzzles as

opposed to coding tasks or written questions. This work contributes not only to the field of measurement of CT, but also to the important task of finding inclusive ways to assess learning.

Limitations

There are several limitations to this study. Foremost is the lack of established external measures to which the validity of the IACT assessments can be compared. CT is an emergent field in K-12 education, and there are few assessment instruments for this age group and/or for learners with neurodiversity. IACT was designed for research in a game-based learning study that included neurodiverse students who may have not had previous experience with CT or coding as a target audience. The specific context of the research study also meant that the IACT items focus on four fundamental concepts of CT and do not attempt to define CT nor encompass all practices that could be included in CT. This assessment was designed to be administered in one class period, limiting the number of items for each CT construct. This likely played a role in the lower than typical correlations. Third, item type is confounded with CT construct (i.e., all items for a specific CT construct have the same unique item format), making a factor analysis of all CT items not meaningful (i.e., if items clustered by construct it could also be due to having a similar item type). Finally, multilevel analyses were not used in the reported study of the IACT items. While the data we used for these analyses has a nested structure, we did not have sufficient sample size at each level to adjust these correlations for this nestedness (e.g., students nested in courses nested in teachers). Future validation of IACT would need to account for variation among classes and teachers.

References

Abdel-Khalek, A. M. (2005). Reliability and factorial validity of the standard progressive matrices among Kuwaiti children ages 8 to 15 years. *Perceptual and Motor Skills, 101*(2), 409–412.

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.

American Association for the Advancement of Science. (1993). Benchmarks for science literacy. New York: Oxford University Press.

Asbell-Clarke, J., Rowe, E., Almeda, V., Edwards, T., Bardar, E., Gasca, S., Baker, R.S., & Scruggs, R. (2020). The Development of Students' Computational Thinking Practices in Elementary- and Middle-School Classes using the Learning Game, *Zoombinis. Computers in Human Behavior*, https://doi.org/10.1016/j.chb.2020.106587

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? *ACM Inroads, 2*(1), 48–54.

Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Paper presented at the Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada.

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.

Dewey, J. (1938). *Logic, the theory of inquiry*. New York: Holt Pub.

Duschl, R. A. (1990). *Restructuring science education: The importance of theories and their development*. Teachers College Press.

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., 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.

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

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

Horowitz, S. H., Rawe, J., & Whittaker, M. C. (2017). The State of Learning Disabilities: Understanding the 1 in 5. New York: National Center for Learning Disabilities.

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.

Karalar, H., & Alpaslan, M. M. (2021). Assessment of Eighth Grade Students' Domain-General Computational Thinking Skills. *International Journal of Computer Science Education in Schools*, *5*(1), 35 - 47. https://doi.org/10.21585/ijcses.v5i1.126

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.

Kruit, P. M., Oostdam, R. J., van den Berg, E., & Schuitema, J. A. (2018). Assessing students' ability in performing scientific inquiry: instruments for measuring science skills in primary education. *Research in Science & Technological Education*, 1–27.

Lundh, P., Grover, S., Jackiw, N., & Basu, S. (2018). Concepts Before Coding: Instructional Support for Introductory Programming Concepts in Middle School Computer Science. Annual Meeting of the American Education Research Association.

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.

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.

National Research Council (1996). National Science Education Standards. Washington, DC: The National Academies Press. p. 23. doi:10.17226/4962.

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.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.

Papert, S. (1991). Situating constructionism. In I. Harel & S. Papert (Eds.), *Constructionism* (pp. 1-11). Norwood, NJ: Ablex.

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.

Raven, J.C., 2000. The Raven's progressive matrices: Change and stability over culture and time. *Cognitive Psychology, 41(1)*, 1–48.

Raven, J., Raven, J. C., & Court, J. H. (2000). *Manual for Raven's progressive matrices and vocabulary scales. Section 3: The standard progressive matrices*. Oxford, UK: Oxford Psychologists Press; San Antonio, TX: The Psychological Corporation.

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.

Román-González, M., Moreno-León, J., & Robles, G. (2017, July). Complementary tools for computational thinking assessment. In Proceedings of International Conference on Computational Thinking Education (CTE 2017), S. C Kong, J Sheldon, and K. Y Li (Eds.). The Education University of Hong Kong (pp. 154–159).

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., Sun, C., & Asbell-Clarke, J. (2017). Demystifying computational thinking. *Educational Research Review, 22,* 142–158.

Tang, X., Yin, Y., Lin, Q., Hadad, R., & Zhai, X. (2020). Assessing computational thinking: A systematic review of empirical studies. *Computers and Education*, *148*, [103798]. https://doi.org/10.1016/j.compedu.2019.103798

U.S. Department of Education, Office of Special Education Programs, Individuals with Disabilities Education Act (IDEA) database, retrieved July 26, 2016, from https://www2.ed.gov/programs/osepidea/618-data/state-level-data-files/index.html#bcc. See Digest of Education Statistics 2016, table 204.30.

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.

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

Appendices

Appendix A. Details of Data Cleaning

In the *Zoombinis* study, there were instances when teachers administered the pre-test later than expected. Specifically, students took the pre-test at a time when the post-test was supposed to be administered. In these cases, students' pre-test scores were treated as responses to the post-test measures. Students in these cases had missing pre-test scores. This rule was applied to 12 percent of our student sample (n = 329). The pattern of results did not vary when these students were removed, so they were retained for the analyses presented in this paper. It was possible for participants to complete some but not all of the items, so the total number of responses varied. Between 23 and 169 students had missing pre-test measures, depending on the CT practice. As many as 2,416 students (Study class students = 999, non-study class students = 1,417) had completed all pre-test measures belonging to at least one CT practice (Table 2) and were included in the reliability analyses. Of these 2,416 students, 1,523 students were from grades 3–5 (733 females, 790 males) and 893 students were from grades 6–8 (422 females, 468 males, 3 other).

For the RPP sample, between 22 and 255 students had missing pre-test measures. From a maximum number of 3,056 students with pre-test measures, 857 students were from grades 2–4 (429 females, 428 males) and 2,199 were from grades 5–8 (1,119 females, 1079 males, 1 other).

Table 1. Number of students who took the pre-assessment

	Zoombinis	**RPP**
Final Sample of Students	2,439	3,078
Students with Missing Pre-Assessment Scores*	23–169	22–255
Number of Students with Pre-Assessment Scores*	2,270–2,416	2,823–3,056

*Varies by CT Practice

In the *Zoombinis* sample, between 44 and 188 students had missing post-test measures, depending on the CT practice item set. A maximum of 1,773 students (study class students = 1016, non-study class students = 757) completed all post-assessment measures belonging to at least one CT practice (Table 2). Of these 1,773 students with complete post-assessment data in one CT practice, 1,174 students were from grades 3–5 (566 females, 608 males) and 599 students were from grades 6–8 (273 females, 323 males, 3 other). There were 1281 students who completed all pre-test and post-test measures across all CT practices (Study class students = 702, non-study class students = 579).

For the RPP sample, between 32 and 367 students had missing post-test measures. As many as 2,889 students completed all post-assessments belong to at least one CT practice (Table 3). From the 2,889 students with complete post-assessment data in one CT Practice, 1,147 students were from grades 2–4 (575 females, 572 males) and 1,742 students were from grades 5–8 (891 females, 850 males, 1 other).

Table 2. Number of students who took the post-assessment

	Zoombinis	**RPP**
Final Sample of Students	1,817	2,921
Students with Missing Post-Assessment Scores*	44–188	32–367
Number of Students with Post-Assessment Scores*	1,629–1,773	2,554–2,889

*Varies by CT Practice

Appendix B: Timed-out items for Problem Decomposition and Algorithm Design

Table 1: Number of students with pre-assessment and post-assessment items for Problem Decomposition (average efficiency)

	Zoombinis **(pre-assessment)**	***Zoombinis*** **(post-assessment)**	**RPP 2017-18**	**RPP 2018-19**
Students with Complete Items for Problem Decomposition (average efficiency)	2,416	1,773	3,056	2,889
Students Who Timed Out of Assessment Items for Problem Decomposition (average efficiency)	106	51	180	142
Students Who Timed Out of All Assessment Items for Problem Decomposition (average efficiency)	23	44	22	32

Appendix 2. Number of students with pre-assessment items and post-assessments items for Algorithm Design (mean number of optimal moves)

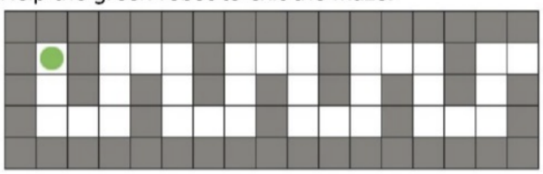
	Zoombinis **(pre-assessment)**	***Zoombinis*** **(post-assessment)**	**RPP 2017-18**	**RPP 2018-19**
Students with Complete Assessment Items for Algorithm (average efficiency)	2,270	1,691	2,823	2,607
Students Who Timed Out of Pre-Assessment Items for Algorithm Design (average efficiency)	793	458	1,228	1,070
Students Who Timed Out of Post-Assessment Items for Algorithm Design (average efficiency)	169	41	255	314

Appendix C. Bebras Items

We selected 5 Bebras items that corresponded to our 4 types of logic puzzles. Items 3 and 4 are most similar to our Problem Decomposition items. Items 1, 2, and 5, and 1 were most similar to our Algorithm Design, Pattern Recognition, and Abstraction items, respectively.

Bebras Item 1

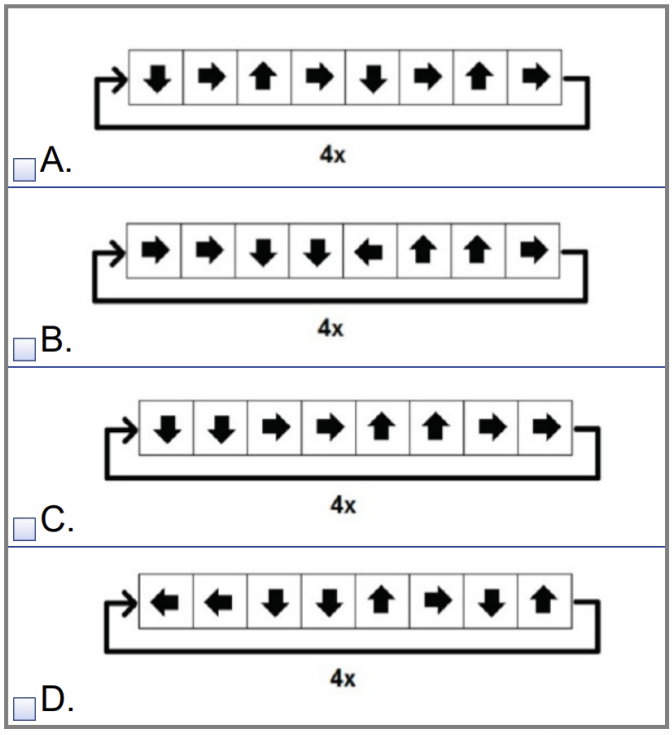
Help the green robot to exit the maze.



The arrows below represent the instructions that the green robot can follow.

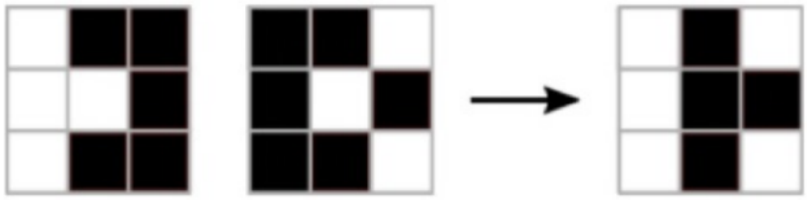
Choose the correct set of instructions that will take the green robot to the exit. The robot will repeat these instructions 4 times.

· **Select the correct answer** *



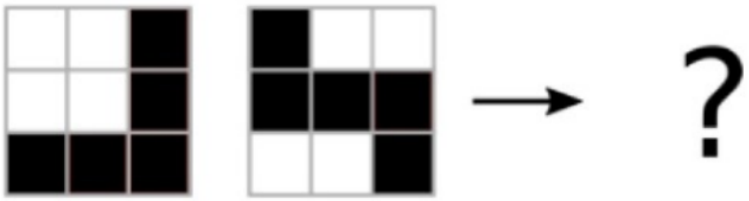
Bebras Item 2

Combining Card A and Card B, you get Card C:



Card A Card B Card C

Question:
How many black cells will Card F have after combining Card D and Card E?



- **Select the correct answer** *

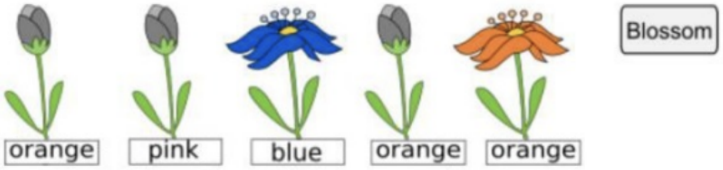
- [ ] 3
- [ ] 4
- [ ] 5
- [ ] 6

Bebras Item 3

Jane is playing a computer game.
First, the computer secretly chooses colors for five buds. The available colors for each flower are blue, orange, and pink. Jane has to guess which flower has which color. She makes her first five guesses and presses the Blossom button.

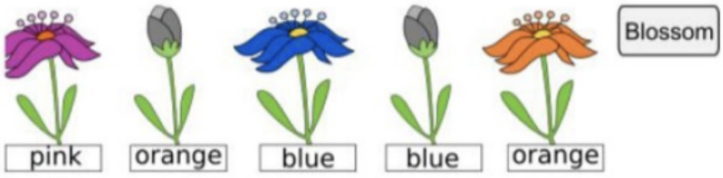
The buds, whose colors she guessed correctly, break into flowers. The others remain as buds.

Jane's first go:



Jane then has another go at guessing and presses the Blossom button again.

Jane's second go:



Question:
What colors did the computer choose for the flowers?

- **Select the correct answer** *

☐ blue pink blue orange orange
☐ pink blue blue blue orange
☐ pink blue blue pink orange
☐ pink pink blue pink orange

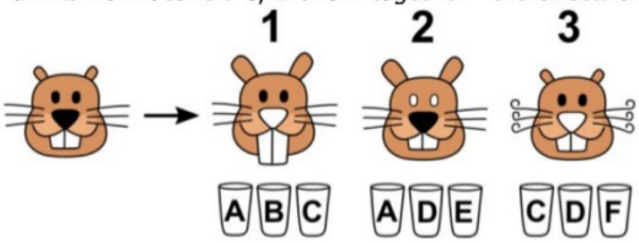
Bebras Item 4:

Betaro Beaver has discovered five new magic potions:
one makes ears longer
another makes teeth longer
another makes whiskers curly
another turns the nose white
the last one turns eyes white.

Betaro put each magic potion into a separate beaker. He put pure water into another beaker, so there are six beakers in total. The beakers are labeled A to F. The problem is, he forgot to record which beaker contains which magic potion!

To find out which potion is in each beaker, Betaro set up the following experiments:

Expt 1: A beaver drinks from beakers A, B and C together - the effects are shown in Figure 1.
Expt 2: A beaver drinks from beakers A, D and E together - the effects are shown in Figure 2 .
Expt 3: A beaver drinks from beakers C, D and F together - the effects are shown in Figure 3.



Question:
Which beaker contains pure water?

- **Select the correct answer** *

- [ ] A
- [ ] B
- [ ] C
- [ ] D
- [ ] E
- [ ] F

Bebras Item 5

Agents Boris and Bertha communicate using secret messages.
Boris wants to send Bertha the secret message:
MEETBILLYBEAVERAT6

He writes each character in a 4 column grid from left to right and row by row starting from the top. He puts an X in any unused spaces. The result is shown below.

Then he creates the secret message by reading the characters from top to bottom and column by column starting from the left:
MBYVTEIBE6ELERXTLAAX

Bertha then uses the same method to reply to Boris. The secret message she sends him is:
OIERKLTEILH!WBEX

Question:
What message does Bertha send back?

Select the correct answer *

- ☐ OKWHERETOMEET!
- ☐ OKIWILLBETHERE!
- ☐ WILLYOUBETHERETOO?
- ☐ OKIWILLMEETHIM!

Appendix D. Construct Validity

For the Problem Decomposition and Algorithm Design items, we tried several means of scoring those items. These included the *total number of moves* the player used to solve the problem, *percentage of items solved correctly*, the *efficiency* of the number of moves players used relative to the maximum number of moves needed to find a solution, and the *percentage of items solved optimally*. Because students could make more than one attempt to solve each Algorithm Design item, the *total number of runs* they made was also examined.

Percent Correct: Whether or not each item was answered correctly, regardless of the number of moves, was recorded. It was possible for students to time out of each item without answering correctly. The percentage of items answered correctly was calculated.

Efficiency: For Problem Decomposition items, a maximum of 3 moves was needed to solve elementary problems while 4 moves were needed to solve the middle-school problems. The efficiency with which each item was solved was calculated by dividing this maximum number by the actual number of moves taken. In cases where players were able to solve the problem with fewer than the maximum number of moves, their efficiency was capped at 100 percent. With Algorithm Design items, the minimum number of moves needed to find a solution depended on the number of submissions players made. This minimum number of moves was recorded for each item.
Optimal Solutions: Optimal solutions for Problem Decomposition items were those solved with 100 percent efficiency. For Algorithm Design items, a solution was considered optimal if it was solved with 100 percent efficiency with one submission.

Tables 1 and 2 report the correlations between all of the scoring approaches taken for the Problem Decomposition and Algorithm Design items with each other and with the Pattern Recognition and Abstraction item percent correct scores.

Table 1. Pearson intercorrelations of pre-assessment measures for the *Zoombinis* and RPP samples

Correlations between CT Practices	**CT Practice**	***Zoombinis* sample (*N*=2206-2416)**	**RPP sample (*N*=2772-3056)**	**Average across samples**
Problem Decomposition (# Moves)	Problem Decomposition (Percent Optimal)	-0.80	-0.77	-0.79
	Problem Decomposition (Percent Correct)	-0.18	-0.25	-0.22
	Problem Decomposition (Avg. Efficiency)	-0.84	-0.79	-0.81
	Pattern Recognition (Correct)	-0.15	-0.12	-0.13
	Abstraction (Percent Correct Spaces)	-0.21	-0.11	-0.16
	Algorithm Design (# Moves)	-0.06	-0.04	-0.05
	Algorithm Design (Percent Correct)	-0.23	-0.07	-0.15
	Algorithm Design (# Runs)	0.21	0.07	0.14
	Algorithm Design (Avg. Efficiency)	-0.23	-0.08	-0.15
	Algorithm Design (Percent Optimal)	-0.19	-0.04	-0.12

Problem Decomposition (Percent Optimal)	Problem Decomposition (Percent Correct)	0.24	0.33	0.29
	Problem Decomposition (Avg. Efficiency)	0.89	0.87	0.88
	Pattern Recognition (Correct)	0.18	0.13	0.15
	Abstraction (Percent Correct Spaces)	0.22	0.16	0.19
	Algorithm Design (# Moves)	0.08	0.06	0.07
	Algorithm Design (Percent Correct)	0.22	0.10	0.16
	Algorithm Design (# Runs)	-0.21	-0.08	-0.14
	Algorithm Design (Avg. Efficiency)	0.23	0.11	0.17
	Algorithm Design (Percent Optimal)	0.20	0.07	0.14
Problem Decomposition (Percent Correct)	Problem Decomposition (Avg. Efficiency)	0.54	0.65	0.60
	Pattern Recognition (Correct)	0.07	0.04	0.06
	Abstraction (Percent Correct Spaces)	0.10	0.09	0.09
	Algorithm Design (# Moves)	0.11	0.09	0.10
	Algorithm Design (Percent Correct)	0.14	0.13	0.13
	Algorithm Design (# Runs)	-0.07	*0.01*	-0.03
	Algorithm Design (Avg. Efficiency)	0.14	0.13	0.13
	Algorithm Design (Percent Optimal)	0.06	*-0.01*	0.03

Problem Decomposition (Avg. Efficiency)	Pattern Recognition (Correct)	0.18	0.12	0.15
	Abstraction (Percent Correct Spaces)	0.23	0.16	0.20
	Algorithm Design (# Moves)	0.10	0.08	0.09
	Algorithm Design (Percent Correct)	0.26	0.13	0.20
	Algorithm Design (# Runs)	-0.22	-0.07	-0.15
	Algorithm Design (Avg. Efficiency)	0.27	0.14	0.21
	Algorithm Design (Percent Optimal)	0.21	*0.05*	0.13
Pattern Recognition (Correct)	Abstraction (Percent Correct Spaces)	0.32	0.30	0.31
	Algorithm Design (# Moves)	0.11	0.15	0.13
	Algorithm Design (Percent Correct)	0.28	0.24	0.26
	Algorithm Design (# Runs)	-0.26	-0.21	-0.23
	Algorithm Design (Avg. Efficiency)	0.24	0.21	0.23
	Algorithm Design (Percent Optimal)	0.25	0.10	0.18
Abstraction (Percent Correct Spaces)	Algorithm Design (# Moves)	0.09	0.12	0.11
	Algorithm Design (Percent Correct)	0.28	0.25	0.27
	Algorithm Design (# Runs)	-0.26	-0.24	-0.25
	Algorithm Design (Avg. Efficiency)	0.26	0.26	0.26
	Algorithm Design (Percent Optimal)	0.24	0.16	0.20

Algorithm Design (# Moves)	Algorithm Design (Percent Correct)	0.34	0.45	0.40
	Algorithm Design (# Runs)	-0.12	-0.11	-0.11
	Algorithm Design (Avg. Efficiency)	0.11	0.35	0.23
	Algorithm Design (Percent Optimal)	-0.08	0.07	-0.01
Algorithm Design (Percent Correct)	Algorithm Design (# Runs)	-0.41	-0.29	-0.35
	Algorithm Design (Avg. Efficiency)	0.96	0.98	0.97
	Algorithm Design (Percent Optimal)	0.43	0.32	0.37
Algorithm Design (Runs)	Algorithm Design (Avg. Efficiency)	-0.40	-0.30	-0.35
	Algorithm Design (Percent Optimal)	-0.64	-0.51	-0.57
Algorithm Design (Avg. Efficiency)	Algorithm Design (Percent Optimal)	0.50	0.36	0.43

Note: Significant at an alpha level of 0.05 except if in italics.

Table 2. Pearson intercorrelations of post-assessment measures for *Zoombinis* and RPP samples

Correlations between CT Practices	**CT Practice**	***Zoombinis* sample (*N*=1600-1774)**	**RPP sample (*N*=2315-2889)**	**Average across samples**
Problem Decomposition (# Moves)	Problem Decomposition (Percent Optimal)	-0.76	-0.65	-0.71
	Problem Decomposition (Percent Correct)	-0.11	-0.12	-0.12
	Problem Decomposition (Avg. Efficiency)	-0.83	-0.67	-0.75
	Pattern Recognition (Correct)	-0.14	-0.11	-0.13

	Abstraction (Percent Correct Spaces)	-0.19	-0.13	-0.16
	Algorithm Design (# Moves)	-0.02	-0.05	-0.04
	Algorithm Design (Percent Correct)	-0.22	-0.09	-0.16
	Algorithm Design (# Runs)	0.23	0.13	0.18
	Algorithm Design (Avg. Efficiency)	-0.20	-0.09	-0.15
	Algorithm Design (Percent Optimal)	-0.17	-0.06	-0.12
Problem Decomposition (Percent Optimal)	Problem Decomposition (Percent Correct)	0.23	0.25	0.24
	Problem Decomposition (Avg. Efficiency)	0.90	0.72	0.81
	Pattern Recognition (Correct)	0.20	0.13	0.17
	Abstraction (Percent Correct Spaces)	0.24	0.12	0.18
	Algorithm Design (# Moves)	*-0.02*	0.08	0.03
	Algorithm Design (Percent Correct)	0.21	0.11	0.16
	Algorithm Design (# Runs)	-0.25	-0.08	-0.17
	Algorithm Design (Avg. Efficiency)	0.21	0.11	0.16
	Algorithm Design (Percent Optimal)	0.21	0.09	0.15
Problem Decomposition (Percent Correct)	Problem Decomposition (Avg. Efficiency)	0.48	0.70	0.59
	Pattern Recognition (Correct)	0.08	*0.05*	0.07

	Abstraction (Percent Correct Spaces)	0.09	0.12	0.11
	Algorithm Design (# Moves)	*0.00*	0.05	0.03
	Algorithm Design (Percent Correct)	0.09	0.07	0.08
	Algorithm Design (# Runs)	-0.06	*0.03*	-0.02
	Algorithm Design (Avg. Efficiency)	0.09	0.06	0.08
	Algorithm Design (Percent Optimal)	0.06	*0.03*	0.05
Problem Decomposition (Avg. Efficiency)	Pattern Recognition (Correct)	0.19	0.12	0.16
	Abstraction (Percent Correct Spaces)	0.24	0.20	0.22
	Algorithm Design (# Moves)	*-0.01*	0.10	0.05
	Algorithm Design (Percent Correct)	0.23	0.14	0.19
	Algorithm Design (# Runs)	-0.25	-0.14	-0.20
	Algorithm Design (Avg. Efficiency)	0.22	0.14	0.18
	Algorithm Design (Percent Optimal)	0.21	0.09	0.15
Pattern Recognition (Correct)	Abstraction (Percent Correct Spaces)	0.35	0.31	0.33
	Algorithm Design (# Moves)	0.08	0.15	0.12
	Algorithm Design (Percent Correct)	0.36	0.30	0.33
	Algorithm Design (# Runs)	0.27	0.26	0.27
	Algorithm Design (Avg. Efficiency)	0.24	0.24	0.24

	Algorithm Design (Percent Optimal)	0.20	0.17	0.19
Abstraction (Percent Correct Spaces)	Algorithm Design (# Moves)	0.08	0.13	0.11
	Algorithm Design (Percent Correct)	0.28	0.23	0.26
	Algorithm Design (# Runs)	-0.23	-0.20	-0.22
	Algorithm Design (Avg. Efficiency)	0.24	0.23	0.24
	Algorithm Design (Percent Optimal)	0.22	0.21	0.22
Algorithm Design (# Moves)	Algorithm Design (Percent Correct)	0.33	0.47	0.40
	Algorithm Design (# Runs)	-0.09	-0.09	-0.09
	Algorithm Design (Avg. Efficiency)	0.09	0.37	0.23
	Algorithm Design (Percent Optimal)	-0.12	-0.08	-0.10
Algorithm Design (Percent Correct)	Algorithm Design (# Runs)	-0.35	-0.32	-0.34
	Algorithm Design (Avg. Efficiency)	0.95	0.98	0.97
	Algorithm Design (Percent Optimal)	0.38	0.42	0.40
Algorithm Design (Runs)	Algorithm Design (Avg. Efficiency)	-0.35	-0.33	-0.34
	Algorithm Design (Percent Optimal)	-0.66	-0.57	-0.62
Algorithm Design (Avg. Efficiency)	Algorithm Design (Percent Optimal)	0.47	0.47	0.47

Note: Significant at an alpha level of 0.05 except if in italics.

Appendix E. Concurrent Validity

Table 1. Correlations between post-test measures and teacher ratings of their students' CT skills

CT Practice	Problem Decomposition	Pattern Recognition	Abstraction	Algorithm Design
Problem Decomposition (Moves)	-0.13	-0.11	-0.07	-0.10
Problem Decomposition (Percent Optimal)	0.16	0.15	0.10	0.15
Problem Decomposition (Percent Correct)	0.07	0.08	*0.04*	*0.04*
Problem Decomposition (Avg. Efficiency)	0.17	0.15	0.10	0.14
Pattern Recognition (Percent Correct)	0.28	0.23	0.19	0.23
Abstraction (Percent Correct Spaces)	0.23	0.15	0.13	0.16
Algorithm Design (#Moves)	*-0.03*	-0.08	*-0.05*	*-0.04*
Algorithm Design (Percent Correct)	0.15	0.09	0.08	0.12
Algorithm Design (# Runs)	-0.16	-0.14	-0.09	-0.13
Algorithm Design (Avg. Efficiency)	0.18	0.14	0.13	0.17
Algorithm Design (% Optimal)	0.23	0.21	0.18	0.22

Note: Significant at an alpha level of 0.05 except if in italics. *N*=892–944 depending on the measures.

As shown in Table 2, IACT logic puzzle items for Algorithm Design (Moves) were weakly correlated with Bebras items in an unexpected positive direction. As Algorithm Design (Moves) is more indicative of persistence than correctness, it is possible that students who persisted and had a great number of moves in IACT also had lower scores in each of the five Bebras items.

Table 2. Correlations between post-test measures and students' scores on Bebras items

CT Practice	Problem Decomposition (Item 3)	Problem Decomposition (Item 4)	Pattern Recognition (Item 2)	Abstraction (Item 5)	Algorithm Design (Item 1)
Problem Decomposition (Moves)	-0.08	-0.09	*-0.04*	-0.11	-0.10
Problem Decomposition (Percent Optimal)	0.07	0.13	*0.04*	0.10	0.07
Problem Decomposition (Percent Correct)	*0.00*	*0.05*	*-0.01*	*0.01*	0.07
Problem Decomposition (Avg. Efficiency)	0.05	0.13	*0.04*	0.10	0.12
Pattern Recognition (Percent Correct)	0.18	0.15	0.10	0.13	0.17
Abstraction (Percent Correct Spaces)	0.23	0.22	0.12	0.22	0.28

Algorithm Design (#Moves)	*0.05*	0.08	0.06	0.09	0.07
Algorithm Design (Percent Correct)	0.11	0.13	0.09	0.09	0.24
Algorithm Design (# Runs)	-0.10	-0.09	-0.07	-0.13	-0.14
Algorithm Design (Avg. Efficiency)	0.11	0.12	0.08	0.09	0.24
Algorithm Design (% Optimal)	0.11	0.08	0.07	0.11	0.13

Note: Significant at an alpha level of 0.05 except if in italics. *N*=1,243–1,399.

Appendix F. Concurrent validity of external CT measures

Table 1. Correlations between teacher ratings of their students' CT skills

	Pattern Recognition	**Abstraction**	**Algorithm Design**
Problem Decomposition	0.78	0.75	0.75
Pattern Recognition		0.77	0.70
Abstraction			0.70

Note: Significant at an alpha level of 0.0001; *N*=1,091.

Table 2. Correlations between Bebras items

CT Practice	**Problem Decomposition (Item 4)**	**Pattern Recognition (Item 2)**	**Abstraction (Item 5)**	**Algorithm Design (Item 1)**
Problem Decomposition (Item 3)	0.15	0.11	0.12	0.13
Problem Decomposition (Item 4)		0.12	0.13	0.09
Pattern Recognition (Item 2)			0.06	0.07
Abstraction (Item 5)				0.09

Note: Significant at an alpha level of 0.05; *N*=1,355-1,387.