Assessing Young Children's Computational Thinking Abilities

A thesis submitted by

Emily Relkin

In partial fulfillment of the requirements for the degree of

Master of Arts

In

Child Study and Human Development

TUFTS UNIVERSITY

Thesis Advisor

Dr. Marina Umaschi Bers
Eliot-Pearson Department of Child Study & Human Development
Tufts University

Committee Members

Dr. Martha Pott
Eliot-Pearson Department of Child Study & Human Development
Tufts University

Dr. Brian Gravel
Department of Education
Tufts University

Abstract

Computational thinking (CT) is a category of thought processes that allows framing and solving problems with computers, robots and other computational devices. The development of CT abilities through computational learning is increasingly vital to success in our technologically-oriented society. A major barrier to computational learning in schools is the current lack of validated instruments for assessing CT abilities in young children. We have developed a new CT assessment instrument called "TACTIC-KIBO" (Tufts Assessment of Computational Thinking In Children - KIBO version). This instrument uses the KIBO programmable robot to evaluate the CT abilities of kindergarten and early elementary school children in classroom or research settings. TACTIC-KIBO rates overall CT ability into four levels by evaluating performance in seven CT categories that are based upon Bers' Developmentally Appropriate Seven Powerful Ideas of Computational Thinking (Bers 2018). In a pilot study, fifteen 5-7-year olds with various levels of past exposure to the KIBO robot were assessed with this new instrument. The children were also videotaped as they engaged in structured interactive play sessions with the KIBO robot to allow their CT skills to be rated independently by researchers with expertise in assessing CT. Results show a high correlation ($r=0.895$, $p <0.001$) between the ratings made with TACTIC-KIBO and expert assessments. TACTIC-KIBO could be easily administered and scored for children with a wide range of CT abilities in an average of 16 minutes per child. Expert ratings of interactive play sessions took longer to perform and were subject to inter-rater variability. Detailed analysis of participants' responses reveals potential areas in which TACTIC-KIBO can be improved in the future. Results of this pilot study are encouraging and provide support for the further development of TACTIC-KIBO as a practical assessment tool for educators and researchers to use when measuring CT abilities in young children.

Keywords: Computational Thinking, developmentally appropriate robotics, assessment, coding, programming, early childhood, educational technology

Acknowledgements

I am extremely fortunate to have had incredible support from my mentors, peers and family as I pursued a Master's degree. I would like to start by expressing my great appreciation of my thesis advisor and mentor, Dr. Marina Umaschi Bers, who has inspired me to pursue a Ph.D. in the field of new technologies for young children at Eliot Pearson. I am honored to have so much support from a mentor as brilliant as Dr. Bers and look forward to learning much, much more from her in the years to come.

I would also like to thank the two other wonderful members of my thesis committee. Dr. Martha Pott, my academic advisor at Eliot Pearson, is one of the most involved and caring professors that I have ever had the privilege to know. The time I have spent with Dr. Pott has been truly invaluable - I would not have found my way without her! Dr. Brian Gravel, a professor of Education at Tufts, is extraordinarily knowledgeable and I cannot thank him enough for his time, input, and extensive support he gave me. My thesis committee, through their wisdom and kindness, made this process truly meaningful and actually enjoyable for me!

I would like to thank several members of the DevTech Research Group at Tufts. Dr. Amanda Sullivan, my first role model in the lab, took me under her wing and first introduced me to the KIBO Robot. I want to thank Dr Sullivan for teaching me more than I ever thought I would learn and for being an amazing professor. Dr Sullivan's dedication and willingness to help has been an inspiration to me. I have also cherished the feedback and advice of Amanda Strawhacker. It is not often that you find someone as sharp and caring as Amanda, and she has also been a role model for me. I am grateful to Amanda for all the time and energy she put into helping me with this thesis, from coding to interviews to general advice and more. Another member of DevTech, Melissa Viezel, has been a true friend to me. Her organization and empathy have made her an amazing lab manager. Another rising star at DevTech, Claudia Mihm, has always impressed me with her work and how much she contributes to our lab. I cannot wait to see what incredible work Claudia is going to accomplish. The pilot study that is the centerpiece of my thesis would not be possible without Amanda Sullivan, Amanda Strawhacker, Melissa Vizel, and Claudia Mihm having tirelessly coded many hours of videos. I would also like to acknowledge Madhu Govind, who I am so glad to have met this year. Madhu is so efficient, supportive, and empathetic to work with and is talented in so many ways. Madhu has contributed so much the DevTech Lab and I am thankful that we have the opportunity to work together and to have become close friends. I also thank Miki Vizner for laying the foundation for this work with his Masters thesis "Big Robots for Little Kids".

I greatly appreciate Meera Menon's assistance and would like to thank her for her expert advice on statistical analyses. A special thanks to the schools, teachers, parents, and children who contributed their time and effort to participate in the pilot study. In particular, I would like to thank Hanna Gebretensae and Lynne May Lim at the Eliot Pearson Children's School in Medford, MA. I would also like to thank Ting-Ting Han and Michelle Kith at the Winter Hill Community Innovation School in Somerville MA. I appreciate the willingness and flexibility of these schools and teachers in accommodating this research.

On a personal note, I would like to thank my friends and family. Thank you to Myles Dworkin and Zoey Relkin, two of my most precious and loyal friends. Thank you to my brother, Paul Relkin, a software engineer, who introduced me to LEGO Mindstorms at a very young age. Thank you particularly to both of my parents Dr. Norman and Felicia

Relkin for their assistance and encouragement with this thesis and support in all of my endeavors. I am very lucky to have two incredible people as parents and I owe this thesis to them!

Table of Contents

	Abstract	2
	Acknowledgements	3
	Introduction	8
	A brief history of computational thinking	10
	Computational thinking and coding in children	12
	Coding as the new literacy	14
	Past attempts at developing CT assessment tools	16
	Why teach Computational Thinking to young children?	21
	Design considerations for a CT robotics assessment tool	22
	The KIBO Robotics platform	25
	Materials and Methods	27
	Overview	27
	Participants	28
	Informed Consent/Assent	29
	Survey data	31
	Testing environment	31
	Robotic activities	31
	CT assessment protocol	32
	Scoring systems	34
	Collection of expert ratings for validity/reliability determination	34
	Statistics	35
	Inter-rater reliability measurements	35
	CT score depiction	36
	Results	36
	Demographics	36
	Administration time	38
	Primary study outcome	39
	Correlations with demographic variables	40
	Inter-rater reliability	43
	Individual participant results	44
	CT category correlations of TACTIC-KIBO vs IPS	48
	Discussion	50
	CT ratings based on Interactive Play Sessions	60
	What does TACTIC teach us about CT measurement?	62
	Study limitations	65
	Scoring	67
	Future directions	70
	References	76
	Appendices	82
1.	Individual participants – full data set	82
2.	TACTIC-KIBO CT Assessment Tool	97
3.	Interactive Play Session Protocol	101
4.	Interactive Play Session Rating System	103

Tables

Table 1: Powerful ideas of Computational Thinking (based on Bers, 2018)	15
Table 2: Block sequences and commands for all levels	32
Table 3: Primary TACTIC-KIBO scoring system	35
Table 4: Demographics summary	37
Table 5: Demographics by each participant	38
Table 6: Correlation of TACTIC-KIBO and IPS-KIBO with demographic variables	41
Table 7: Cohen's Kappa for primary rater vs all others	43
Table 8: Correlation matrix for CT categories	49

Figures

Figure 1: Process of creating the assessment tool	22
Figure 2: 2aPhotograph of full KIBO robotics kit. 2bKIBO robot sensors	27
Figure 3: Pilot study flow chart	30
Figure 4: The general layout of the testing environment	31
Figure 5: The set of robotic activities used in the TACTIC-KIBO assessment	33
Figure 6: Administration time	39
Figure 7: Primary study outcome	40
Figure 8: Percentage of children responding correctly	42
Figure 9: IPS-KIBO interrater reliability	44
Figure 10: CT scores sorted from lowest to highest	54
Figure 11: Sample Computational Learning curriculum with evaluable learning outcome suitable for assessment with TACTIC-KIBO	75

Introduction

In the course of a child's development, many different patterns of thinking emerge as the brain matures and new life experiences are acquired. One class of thought processes that has taken on particular importance since the advent of computer technology is called Computational Thinking (CT). CT can be defined as the set of thought processes that allows framing and solving problems using computers, robots and other computational devices (Sullivan, Bers, & Mihm, 2017). Mastery of CT requires the acquisition of a broad set of abilities that includes processes of pattern recognition, conceptualization, planning and problem solving. Developing CT abilities is not only valuable for computer programming but helpful in a variety of other contexts. The availability of computers and robots as well as the increasing technological orientation of our society has increased the importance of CT as well as the need for CT educational programs (Brennan & Resnick, 2012). Hemmendinger, (2010) stated that the goal of CT should not be to have everyone thinking like a computer scientist but to have them apply their knowledge to solve problems and think of other questions in all disciplines. Coding could be considered a new form of literacy. It allows people to express themselves in unique ways as well as gain power and understand the world around them (Bers, 2018; Vee, 2013; Faucault, Martin, Gutman, & Hutton, 1988).

The process of acquiring CT has been referred to as Computational Learning (CL) (Werner, Denner, & Campe, 2014). Past studies have shown that children as young as four years old can begin to acquire CL skills (Bers, 2017; Bers, 2008; Sullivan, Bers, & Mihm, 2017; Leidl, Bers, & Mihm, 2017). There are now many educational initiatives designed to bring STEM subjects into the curriculum and some of these foster CL beginning as early as at kindergarten and preschool levels (Werner, Denner, & Campe, 2014, Lee et al., 2011). While schools have traditionally taught problem solving through projects such as building block construction and puzzles, some educators have begun to use computer programs and robotics platforms for this purpose in recent years. For example, the ICODE (Internet Community of Design Engineers) project involves a competition to encourage middle school and high school students to design their own robots and learn about abstraction, automation, and debugging (Lee et al., 2011).

Despite the proliferation of programming languages, educational apps and robotics platforms designed to teach coding and promote CL, there are currently no formal assessment tools specifically designed to measure CT and CL in young children. The lack of CT assessment tools creates challenges for educators, researchers and other specialists in the robotics field who are working with young children. For example, Lee et al., 2011 provided examples of how children in K-12 could learn to use abstraction, automation, and analysis through engagement with technologies such as robotics. These authors argue "Because CT is not evaluated by standardized testing, it is difficult in the current educational climate for teachers to teach CT concepts directly… the field requires systematic assessment procedures…" (Lee et al., 2011, p. 36).

Without a basis for measuring baseline abilities and gauging progress, early elementary level teachers are limited in their ability to assess their students, to document and communicate progress in school and to assess the effectiveness of their curricula. Robotics designers lack the assessment tools needed to develop better robots for children and to measure the effectiveness of their creations. Researchers also need these tools for characterizing participants in studies and assessing study outcomes.

As technology has become more and more pervasive in our lives, it is vital that children acquire competency in CT. However, in the US, most states do not have adequate (or in some cases, any) computer science education standards. Some states focus only on computer programming language competency and do not emphasize or assess computational thinking (Wilson, Sudol, Stephenson, & Stehlik, 2010). A well-regarded and validated instrument for assessing computational thinking abilities could serve to increase the focus on CT and promote standards for assessing CL in schools.

In this thesis, I will:

1. Describe the conceptual foundations for creation of a CT assessment instrument
2. Introduce the KIBO robotic platform that is used in conjunction with the assessment
3. Present TACTIC-KIBO, a prototype CT assessment instrument
4. Report on the conduct of an IRB-approved pilot study examining whether TACTIC-KIBO is age-appropriate for kindergarten and first grade students and a practical instrument for measuring CT abilities

A Brief History of Computational Thinking

Although the phrase "Computational Thinking" was not commonly used until the early 2000s, the term actually dates back many years. CT originally was used in a mathematics context to describe the types of thought process required to project and quantify future needs. For example, it was said that computational thinking was required to estimate future tax payments or calculate projections of future travel expenses (Prakken, 1942; The Mathematics Teacher, 1943).

In 1962, a pioneer in computer science named Alan Perlis was among the first to acknowledge the importance of the type of thought processes required by emerging information technologies and to propose that everyone getting an education should learn computer programming. He considered being able to program the path to comprehending the "theory of computation" (Grover & Pea, 2013). Two decades later, Seymour Papert used the term "computational thinking" in his book *Mindstorms: Children, Computers, and Powerful Ideas,* and described the relevance of CT to various aspects of everyday life (Papert, 1980). Papert later explored this concept in greater depth and concluded that the goal of CT is to make mathematical ideas more accessible and more powerful. He explained that CT gives an individual the power to create knowledge and helps increase thinking abilities (Papert, 1996).

In 2006, Jeanette Wing published an article about CT in the *Communications of the Association for Computing Machinery* which helped to popularize the concept (Wing, 2006). Wing defined CT as solving problems, designing systems and understanding human behavior by drawing upon concepts of Computer Science. In 2008, Wing expanded on her work by stating that CT is a vital and universal skill that should be taught to everyone in early childhood (Wing, 2008). Wing explained that CT stems from mathematics and both CT and math involve problem solving (Wing, 2006; Wing, 2008). According to Wing, CT can also give people the opportunity to succeed in professions relating to the humanities and the arts, teaching researchers about "data mining and data federation to discover new trends, patterns and links in our understanding and appreciation of humankind" (Wing, 2008).

CT exercised in the context of using computers and other technological devices may be a good way to make learning mathematics and science easier and more interesting (diSessa, 2000). Many computer science initiatives such as "Bootstrap", have taken CT and integrated it into teaching mathematical concepts such as algebra, physics and data science by having students design their own video games (K-12 Computer Science Framework Steering Committee, 2016).The relationship between mathematical learning and CT is still under study and both similarities and divergences in the trajectory of each has been noted (Rich, Pokimica, Wherfel, Strickland, & Moran, 2017).

Despite all of the history and the known benefits of computational learning, CT is conceptually still in its infancy. There are many different ideas of what teaching CT should involve and no agreed upon curriculum structure (Lockwood & Mooney, 2017). Even the definition of CT is highly debated. For example, some researchers question whether CT is separable from technologies or whether it is inherently different from Computer Science (Denning, 2009). Some believe that it can help us think about the world around us (Wing, 2006; Wing 2008; Papert, 1996; Hemmendinger, 2010) while others believe that its purpose is to help us problem solve when coding and computing (Guzidal, 2008; ISTE and CSTA, 2011). The concepts and categorizations currently available are still relatively crude and likely to evolve over time. In this context, the development and validation of a CT assessment tool could help advance this field and provide much needed empirical data to inspire future conceptual frameworks.

Computational Thinking and Coding in Children

Jean Piaget is one of the most influential figures in cognitive psychology, epistemology and developmental psychology. He is best known for his descriptions of children's cognitive development. Piaget defined major stages such as the sensorimotor, pre-operational, concrete operational, and formal operational stages. Piaget's work led to a paradigm shift in education away from the belief that children were less intelligent versions of adults or "empty vessels" that needed to be filled with knowledge. Instead, Piaget fostered the more enlightened view that children think differently than adults and that they need the opportunity to construct their own knowledge. Piaget believed that being able to understand how children acquire knowledge helps adults understand knowledge itself. "Constructivism" is a term coined by Piaget to reflect that children build (construct) their own knowledge through experience rather than being taught by teachers (Piaget, 1968).

Seymour Papert, a mathematician and student of Piaget, is a seminal figure in computer science for children. Papert further refined Piaget's insights and applied them to the emerging field of children's computer programming (Papert,1993; Papert & Harel, 1991). Papert coined the term "constructionism" to express the belief that people learn and understand the world by constructing/programming objects and materials such as computers and other technological tools (Papert & Harel, 1991). Papert added to Piaget's constructivist theory by taking the environment, artifacts, and individual decisions into account.

Papert and colleagues created the computer programming language "LOGO" specifically for use by children. LOGO was designed with a "low floor, high ceiling" model which meant that it was simple enough for a beginner to use but also could become challenging for more advanced users (Grover & Pea, 2013). Papert and colleges were also responsible for the development of some of the first simple programmable robots for children, such as the LOGO turtle. This later led to the creation of the LEGO Mindstorms robotics platform, a groundbreaking commercial and academic success that helped to bring programming and robotics into the school curriculum on a much larger scale (Slotnick, 2017).

Coding as the New Literacy

One of Seymour Papert's powerful ideas was to use CT to express and understand ideas and concepts. One of Seymour's students, Dr. Marina Umaschi Bers, wrote in the book *Coding as a Playground,* that robots such as LOGO that allow children to think like a computer also allow children to express themselves in a way that can be understood by others. She writes that literacy as well as coding allow people to express and represent their ideas through speaking, reading, and writing (Bers, 2018).

At least one pilot study has found a possible relationship between language processing and coding in the brain using fMRI brain scans. Researchers concluded that in early childhood, learning to program may have ties to learning language. However, this study was done in adult software developers and may have limited applicability to coding and CT abilities in young children (Siegmund et al., 2014). Additional studies are needed using a sample of elementary school-aged children to better understand the functional neuroanatomy associated with CT abilities.

Vee (2013) states that writing and programming have many commonalities such as "historical trajectory, social shaping, affordances for communication, and connections to civic discourse" (Vee, 2013 pp. 43). Vee argues that since programming may soon be considered a new literacy, educators should begin to think about computer programming differently and understand the field of Computer Science more deeply.

Table 1
Table 1: Powerful Ideas of Computational Thinking (based on Bers, 2018)

Powerful Idea	Definition	Example
Algorithms	Sequencing/order, logical organization	Child understands that KIBO blocks must be scanned in a specific order
Modularity	Breaking up larger task into smaller parts, instructions	Child use repeat blocks in order to accomplish a goal rather than scanning a large number of blocks
Control Structures	Recognizing patterns and repetition, cause and effect	Child recognizes that they must use a begin and an end when making a program and are able to use if blocks and repeat blocks
Representation	symbolic representation, models	Child sees the difference between the blue motion blocks and the Orange sound blocks
Hardware/Software	Smart objects are not magical, objects are human engineered	Child describes what the function of KIBOs electronics do. Child understands that you must give the robot a program in order for it to work
Design Process	Problem solving, perseverance, editing/ revision	Child has the capability to plan and test an idea in order to improve a project
Debugging	Identifying problems, problem solving, perseverance	Child identifies a bug in either hardware or software and is able to fix the problem

While curriculum for teaching LOGO and robotics with platforms such as Mindstorms and LEGO WeDo are available, early developers generally treated the creation of programs as an end in itself and did not explicitly provide teachers with a framework for assessing the CT skills. Based on a framework and resources for educators (Google for Education, 2010), as well prior research on Scratch, ScratchJr, and KIBO, Marina Bers proposed the "Seven Powerful Ideas" of early childhood computational thinking as a developmentally appropriate construct (Bers 2018). These are further described in Table 1.

The Seven Powerful Ideas provide a developmentally appropriate framework for assessing the acquisition of Computational Thinking skills by young children. As such, these concepts were chosen as the categorical framework for the new CT assessment tool that is the focus of this thesis.

Past Attempts at Developing CT Assessment Tools

There have been relatively few past studies in which computational thinking skills were assessed in either children or adults. Werner, Denner, & Campe (2014) examined children ages 11 and 12 as they created their own computer games using the program Alice 2.2 or the story-telling Alice programming environment. The outcomes of this assessment were "programming constructs." By their formulation, the most elementary level of programming is one in which the child uses simple computer program building blocks. Next, is "patterns" in which a child puts together multiple programming concepts that extend the program but may not necessary code for a complete set of actions. Lastly, the researchers looked at "game mechanics" which they viewed as the highest level of coding. This is when children combine patterns and programming concepts to create an actual game. The program is interactive in that it provides step-by-step programming guidance at first and then gives children the ability to create their own program without assistance. In this study, multiple researchers coded the first 10% of projects and then met to resolve any discrepancies. The investigators used Cohen's k test to establish inter-rater reliability. A problem confronted by these researchers was students sometimes seemed to generate code without fully understanding their own work. Another limitation of this study was that the assessment was very lengthy and the platform was not suitable for use in younger children.

Some past studies that attempted to assess proficiency in computational thinking skills used informal methods of collecting data such as interviews and free-form observations. Wang, Wang, & Liu, (2014) assessed computational thinking ideas in children. These researchers observed children using a tangible programming tool for children ages 5-9. After the children completed two tasks the researchers interviewed them and asked questions such as, "What other things do you think the blocks can do?" and "Have you ever noticed sensors or something like that in your life?" This was said to provide the researchers with a broad qualitative assessment of four Computational Thinking ideas including abstraction (when a programmer conceptualizes in order to solve a problem), automation (when a computer is instructed to execute repetitive tasks), decomposition and analysis (stripping down a problem into its bare essentials, a reflective practice which validates that the abstractions are right), and creativity (freedom in creation). However, this approach did not provide a basis for categorizing children's abilities in terms of stages of CT competency.

Other relevant work on the assessment of CT in young children using robots was carried out by Mioduser, Levy, & Talis in Israel. In their initial study, they tested a group of six kindergarten students using a robotic platform that employed on-screen programming (Mioduser, Levy, & Talis, 2009). This study focused on learning and changes in thinking rather than programming outcomes. They observed the children in five 30-40 minute sessions over the span of one week. The research used a computerized control environment with a LEGO robot and modifiable landscapes. There were two parts to this study: description and construction. The tasks for this study ranged from low difficulty to high difficulty. All of these tasks and descriptions were videotaped and coded. Children's responses were coded as "spontaneous" or "supported" (Mioduser, et al., 2009). Supported answers involved probing from a researcher in order to help bring the child to what was referred to as their "Zone of Proximal Development." This refers to the use of social interaction as a facilitation to allow children to participate in experiences that might otherwise be beyond their reach.

Vygotsky's cognitive theories emphasize that children's social interactions with others is vital for development. Vygotsky defined the term, "Zone of Proximal Development" (ZPD) as the potential for performance at a higher level of ability attainable through the guidance from others (Vygotsky, 1978). He pointed out the value of ZPD in predicting development by saying, "what lies in the zone of proximal development at one stage is realized and moves to the level of actual development at a second. In other words, what the child is able to do in collaboration today he will be able to do independently tomorrow" (Vygotsky, 1987, pp. 21). In regard to assessment involving a child's development, Vygotsky believed that children should be assessed in a way that is not static, but which is dynamic and interactive. This type of assessment as now called DA (Dynamic Assessment). Vygotsky stressed the importance of testing children outside of expectations for their mental age in order to bring them out of their typical comfort zone. (Vygotsky,1934/1998, p. 203, Vygotsky, 1978).

Based on their observations in this study, Mioduser and colleagues proposed three constructs relevant to assessment of CT using robots. The first construct is the *episode*, which is the observation of a specific action or sequence of actions taken by the robot that leads to the formation of a literal mental representation. This can be conceived of as a memory of the events that can be played back in the child's mind or recounted to others. The second construct is the *script*, which is the child's recognition of repetitive actions as a pattern. The third construct is *rules,* which are sequences and repetitions of actions from which the child forms an abstract representation that may not be directly linked to the temporal sequence of specific observed events.

Mioduser & Levy (2010) further studied the effect of acquiring programming skills on the explanations that children give when they observe the behavior of robots. They hypothesized that as children gained experience in programming robots, they would apply different descriptions to the robotic behaviors they observed. For example, a novice may be more likely to view a robot's actions as magical or to attribute personal agency to the robot, whereas a child experienced in programming might be more prone to provide a mechanical explanation for the same behaviors.

As in the previous study, a small group of kindergarten students were followed, but over a longer period of 6 weeks with weekly sessions that involved construction of robots as well as solicitation of explanations about the robot's behaviors. In this study, construction tasks were designed to increase in complexity as the child learned. In addition, the researchers did not help the child beyond simple prompting. An example of a prompt they used was "Why do you think that happened?" The researchers recorded the amount of prompting the children required in each session. Finally, in order to establish interrater reliability, the researchers had three independent raters code 20% of the session videos and established 97% accordance, then one graduate student rated the rest of the videos (Mioduser, & Levy, 2010).

While the results generally supported their hypothesis that acquisition of coding skills altered the explanations children gave of a robot's actions, Mioduser, & Levy, (2010) observed that the nature of the explanations given by children changed as a function of the complexity of the robot's behavior. As hypothesized, when the robot's behaviors were relatively simple, the children with less experience programming tended to ascribe human or animal-like motivations to the robot while those who were more experienced gave more mechanistic explanations. However, when children observed more complex programs, they all tended to give a combination of both types of explanations, which the investigators referred to as a "bridging" explanation.

The use of children's explanations of robotic behaviors as a measure of CT has some benefits but also has certain limitations. The K-12 Computer Science Framework suggests that assessments of programing should involve not only the student's ability to write a program but also their ability to explain the significance of the program (K-12 Computer Science Framework Steering Committee, 2016). Braitenberg, (1984) argued that it is more difficult to understand a robot's behavior through observing a program than it is to understand the behavior by creating a program. Young children's expressive language skills may also limit the nature of the explanations they provide for the behaviors they observe. The investigators themselves critiqued their own study by stating, "While deepening how we understand young children's evolving knowledge of autonomous artificial behaviors, it is limited in its small sample and disconnect from classroom situations" (Mioduser, et al., 2009 p.19). They also observed that combining the task of robot construction and explanations in the same subjects introduced a potential confound and recommended that future studies examine these separately.

Vizner, Bers, Scarlett, & Gravel, (2017) proposed a developmental model of programming with the KIBO robot based on observations of children with different skill levels and exposures to the platform. This model consists of four stages of programming and provided some parameters for assessing children's coding proficiency. The stages are:

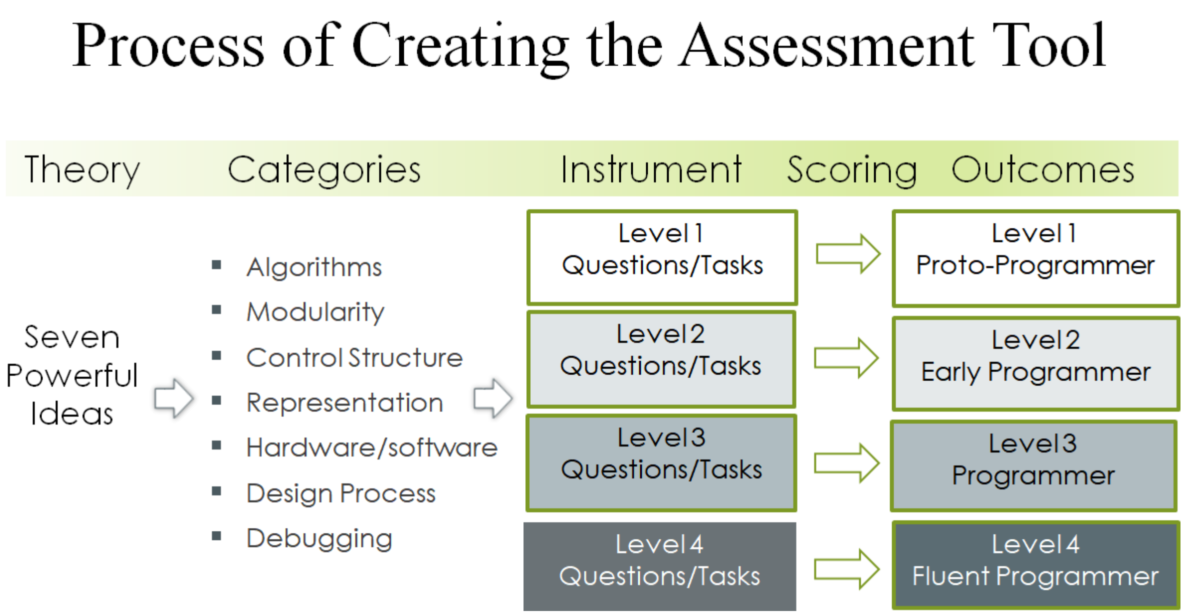
Proto-Programming, Early-Programming, Programming, and Fluent-Programming. A Proto-Programmer understands that KIBO and the blocks are a toy and has the fine and gross motor skills to play with it but does not understand that the blocks are more than just blocks. This child does not create their own code and may press the "On" button repeatedly without first programming the robot. An Early Programmer is capable of creating programs with the Begin and End blocks. The child may try to use as many blocks as possible and may scan blocks that are not part of a meaningful program sequence. The next stage, Programmer, is defined by using 3-6 instructions without using complex blocks such as Repeat. This child may debug a program using trial and error but needs assistance from others when creating programs that are complex. Vizner called the highest level, "Fluent Programmer. A Fluent Programmer may solve 6+ instructional tasks and create a program with 6+ blocks that is syntactically correct. This child may debug by checking their work and revising.

This developmental model of programming lays out an important framework for the present study. However, the classifications suggested by Vizner and colleagues were not implemented as an actual assessment instrument that can be readily used in a classroom setting. The present study further adapts and operationalizes Vizner's developmental model as part of an assessment tool employing a programmable robot to probe CT categories derived from Bers' Seven Powerful Ideas. (Figure 1)

Why teach Computational Thinking to young children?

As Resnick (2007) points out, in today's society it is vital for children to have the ability to think creatively. Past studies have shown that by the time children are in kindergarten they are already affected by gender stereotypes and are beginning to apply these stereotypes to themselves and to others (Sullivan & Bers, 2016). In this study both boys and girls performed equally well on simple programming tasks with a KIBO prototype which was novel to them and perhaps not yet stereotyped. However, girls performed significantly lower on complex programming tasks and these children also indicated in a pretest that LEGOS were more for boys than they were for girls. In order to reach these children before gender stereotypes have been engrained, it seems worthwhile to focus CT and CL education and assessment on children 4-6 years of age.

Figure 1



Additionally, studies have shown that giving young children computer science opportunities such as programming and learning CT skills is one of the best means for making gaps in development and achievement smaller and aiding children's futures and the economy (K-12 Computer Science Framework Steering Committee, 2016).

Design Considerations for a CT Robotics Assessment Tool

Based on the work in this area carried out by past investigators, various considerations for the design of a CT assessment tool for young children can be identified:

1. **Age appropriateness:** In order to optimally impact the acquisition of computational thinking abilities, educational interventions need to target late preschool and early elementary school levels when children are in the process of developing representational and abstract thinking and have sufficient linguistic skills to acquire proficiency in programming simple robots. An assessment tool for this age group must use age-appropriate language and tasks to assure that language abilities and factors such as manual dexterity and attention span and the use of jargon (Sattler, 2014 p.176) are not the limiting factors in the measurements.
2. **Authentic interaction:** To place young children at ease and avoid stress that can be associated with formal testing, a CT assessment tool should be structured in as authentic a way as possible, ideally capturing the dynamics of play and familiar teacher-child, parent-child and/or peer to peer interactions. Authentic assessment should be something that is worthwhile for the child (Dewey, 1938). Shaffer & Resnick (1999) describe authenticity in assessment as being relevant to the learning process. They also argue that computers and new technologies can lead to different types of authentic assessment and learning if used in the right ways. The assessment should not be intrusive or disruptive and should include context specific prompts (Ming, Ming, & Bumbacher, 2014).
3. **Ease of administration:** Past researchers studied children's computational thinking over multiple sessions spanning days to weeks and carried out assessments requiring three or more hours of time per child (Mioduser & Levy, 2010; Wener, Denner, & Campe, 2014; Mioduser, et al., 2009). A practical assessment tool for teachers must be administered in less time and permit serial re-assessments. Many educational assessments currently are too lengthy and or complex for teachers to give effectively. Past research has found that teachers

continue to feel unprepared to conduct high level assessments and many have a low assessment literacy despite educational efforts (DeLuca, & Bellara, 2013). The National Research Council indicates that it is difficult to assess CT and recommends that teachers must be extremely skilled in coding through professional development in order to properly assess children (National Council, 2011). Ideally, teachers who are not particularly skilled at coding and assessments should be able to administer CT assessments.

4. **Time constraints:** The duration of testing sessions should be kept relatively short in light of the limited attention spans of children in this age group and the limitations on time available to teachers for individualized assessments (Moyer & Gilmer, 1953). To be of practical value for pre-school and early elementary school use, the instrument should require no longer than 15-30 minutes for a single assessment.
5. **Sensitivity:** Following the "low floor, high ceiling" model (Papert, 1980), an ideal assessment tool should be equally useful for assessing complete novices as well as relative experts (Sattler, 2014).
6. **Scoring:** To create an assessment tool that does not require an expert to administer or score, the ratings system employed should use simple outcome categories and/or numeric scores that are straightforward to calculate (Koretz, McCaffrey, Klein, Bell, & Stecher, 1992).
7. **Communication of results:** While numeric scores are suitable for rating the level of CT proficiency, descriptive names (e.g. "Early Programmer") based on the score equivalency may better convey the meaning to teachers, parents and children. Sattler, Dumont, Coalson, (2016) say that it is important to present results in a fashion that is easy to understand to novice laypersons but can still give quality technical data.

To create an assessment tool that meets the criteria above, several of the approaches taken by past investigators can be adapted to create a novel assessment instrument. Our initial implementation draws on the KIBO robot but this approach may be applicable to other robots and programming platforms. To permit standardization and increase sensitivity, it may be best to present children with specific structured scenarios rather than to simply observe free play. Structured scenarios can permit assessments to be carried out in children who have only rudimentary understanding of the robot's operation and limited manual dexterity. To capture the full range of CT abilities from novice to fluent programmer, the scenarios can be presented in gradually increasing complexity, from simplest to highest. The assessment itself can include a combination of questions and simple tasks to help create a balance between the need for well-developed language skills and nonverbal communication. Scoring and categorizations can be flexible so that missing one question or failing to complete one task does not disqualify the child from achieving a rating appropriate to their skill level. With these considerations in mind, the assessment tool must be adapted to the particular form and protocols of the KIBO Robotics platform, which is described below.

The KIBO Robotics Platform

The KIBO robotic platform is used throughout the world as a means of teaching coding skills to young children. KIBO was originally developed under a grant from the National Science Foundation (NSF Grant No. DRL-1118897) as a developmentally appropriate tool for teaching the basics of computer programming to neurotypical children ages 4-7 (Sullivan, Elkin, & Bers, 2015).

The KIBO robotics kit (Fig 2a) is an exemplary robotics platform designed with a "low floor and high ceiling" (Papert & Harel, 1991). To use the KIBO robot, children must align and scan the barcodes on programming blocks representing steps in a program that guides the robot's actions (See Figure 2a). The child then presses a green flashing button to make the robot perform the program. Examples of programing blocks are "Spin," "Sing," "Turn Right," and "Shake." Every program must have a "Begin" and "End" block. The programs that KIBO performs can be as simple as a three-block program and as complex as a program featuring conditional blocks, repeat blocks and nested statements. The more complex blocks (repeats and ifs) require "parameter" stickers which includes "if near" "repeat until dark." By virtue of resembling building blocks, programming blocks are in a format that is familiar to young children and arguably more manageable than computer screens that have only two dimensions.

The KIBO robot also has four openings on its upper surface for sensors that can detect light, sound and proximity as well as modules that can flash a light or record/play sounds. In addition, KIBO contains attachable art platforms that allow children to use various materials to decorate their robot, and an attachable expression module that allows children to attach writings or drawing using a dry erase board. There are also attachable building platforms compatible with LEGO building blocks. KIBO has been shown to help children learn technological literacy as well as other curriculum such as math, science, art, and language. (Sullivan, et al., 2015)

Figure 2

2A. Photograph of full KIBO robotics kit. 2B. KIBO robot sensors
(Courtesy of Kinderlab Robotics)

A.

B.


KIBO robot was selected as the robotics platform for this study for several reasons. It is a time-tested, award-winning robotics platform designed for preschool to elementary school age children. It has an existing user base that spans multiple continents and languages, making it available to a large number of children and teachers. Its use of tangible programming blocks is not only advantageous for children but can help to create a more user-friendly environment for teachers or other evaluators to interact with the children they are assessing. KIBO uses programming principles that are analogous to those used in other robotic platforms for young children as well as platforms developed for older children and adults. This should make it easier to adapt the new assessment instrument to other platforms in the future.

Materials and Methods

Overview

To help characterize the ease of assessment using TACTIC-KIBO as well as age-appropriateness and face validity, a pilot study was carried out with fifteen children. kindergarten and early elementary school children ages five, six, and seven years with past exposure to the KIBO robot were recruited and assessed by a single primary rater using this new instrument. The children were videotaped during the assessment and as they engaged in structured interactive play sessions (IPS) with the KIBO to allow their CT skills to be assessed by expert raters and reviewed by the children's teachers. The Interactive Play Session (IPS) was created as a means of measuring the validity of TACTIC-KIBO based on its correlation to expert assessments. The IPS sessions had three parts, the first being a confrontational naming game used to test the child's knowledge of KIBO hardware. The second part was a free-play construction session in which the child was encouraged to program a project of their own choice. Finally, the IPS included a construction challenge in which the child was asked to augment a program using higher level skills. The challenge concept was designed to bring the child to their Zone of Proximal Development, as described by Vygotsky.

A simple curriculum was created to guide the IPS with the goal of collecting sufficient information to permit an assessment of computational thinking ability based on review of the IPS by independent expert raters. A scoring sheet for the IPS was developed to help standardize scoring by the outside raters. The experts rated 25% of the students using the same four-level classification system used for the CT assessment tool but based exclusively on the behaviors observed during the interactive play sessions. The expert ratings of the interactive play sessions and the ratings based on the CT assessment tool were used to establish the inter-rater reliability of the primary examiner's assessments to minimize the effects of bias in ratings.(See Fig 3).

Participants

15 school children were recruited for this study. The study was carried out during school hours or during after school hours in classrooms at three locations: 1. The Eliot-

Pearson Children's School; 2. The Winter Hill Community Innovation School; 3. Tufts Bioengineering Camp. The inclusion and exclusion criteria were as follows:

Inclusion Criteria

1. Typically developing
2. Ages 5-7 inclusive
3. Any gender
4. English speaking/ understanding
5. Parental informed consent and child assenting to participation

Exclusion Criteria

1. No parent/guardian available to consent
2. Inability to comply with testing

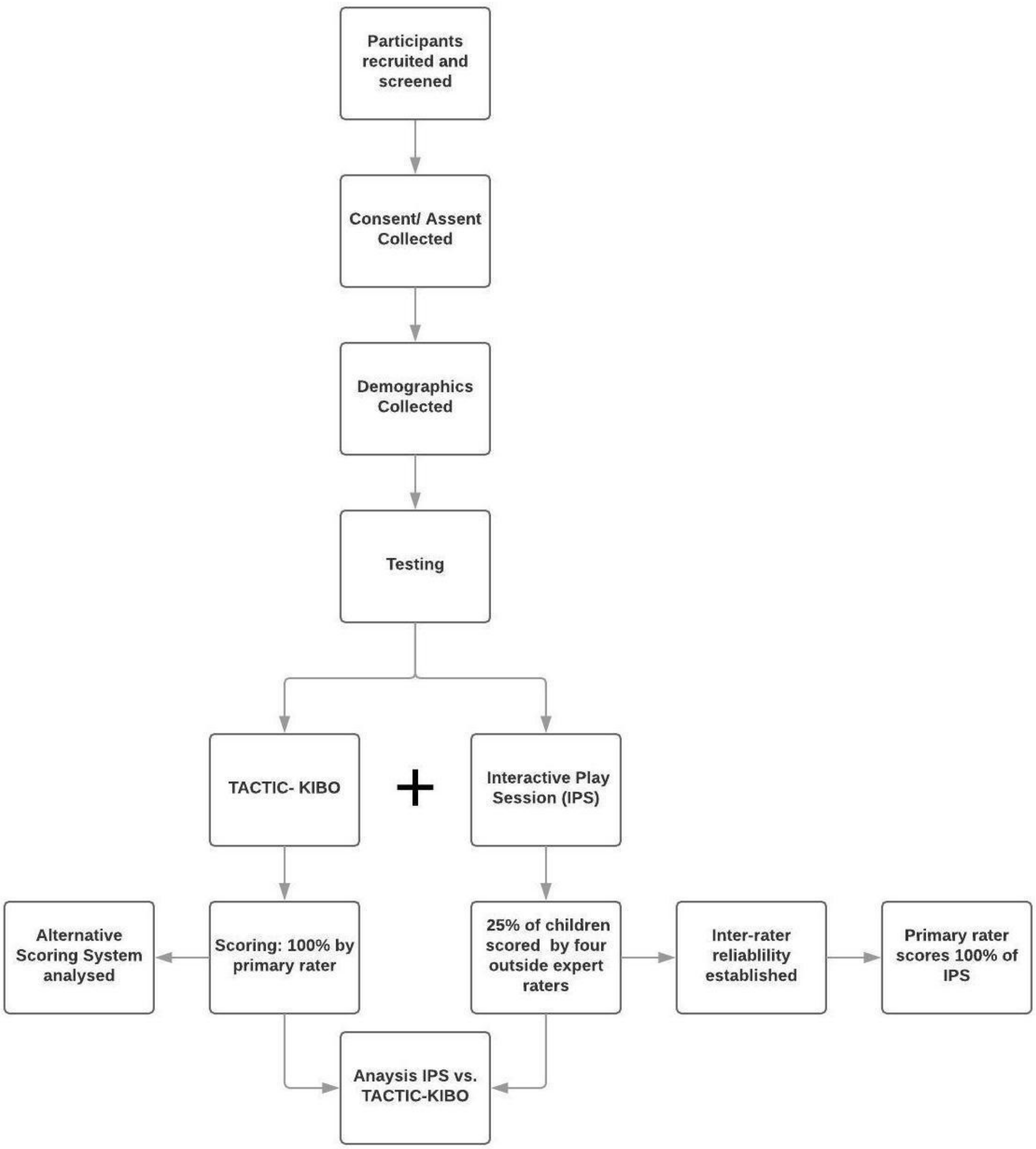
Informed Consent/Assent

This study is approved under Tufts University Social, Behavioral, & Educational IRB Protocol #105050. Informed consent was obtained by the investigator with the assistance of teachers who identified candidates based on inclusion and exclusion criteria.

Written consent was obtained from the parent/legal guardians and assent was obtained from the children verbally or by signature if the child was at least seven years old. Parents were also asked to give consent for their child to be videotaped for research purposes. Children or their parents could choose to discontinue participation at any point during this research without penalty.

Figure 3

Pilot Study Flow Chart



Survey Data

Demographic data (date of birth, gender, hours of experience with KIBO, experience with other robotics platforms, experience with programmable robots) were collected from the parent/legal guardian after the consent was signed.

Testing environment

Only the participating child and the investigator were present during the testing sessions. All sessions were video recorded from two angles. The general layout of the testing environment is shown in Figure 4.

Robotic Activities

A complete set of questions and tasks for the proposed CT assessment tool can be found in Appendix 2. The CT assessment tool consists of KIBO Robot programs of escalating complexity which serve as the framework for asking questions and posing tasks for the child to complete. Figure 5 and Table 2 show the block sequences and commands for all levels. Appendix 2 contains the questions and tasks associated with each level of activity.

Figure 4

Physical arrangement of child, examiner and KIBO robotics kit during assessment sessions

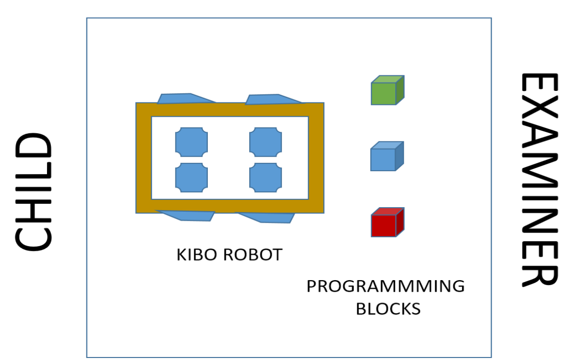


Table 2

Activity 1	A program that causes KIBO to move a short distance after the ON button is pressed. (3 blocks)
Activity 2	The same as level 1 but with a light that comes on after KIBO moves forward. (4 blocks + light)
Activity 3	Adds a repeat x 2 loop that doubles the distance the KIBO moves and doubles the number of times the light turns on. (6 blocks + 1 light + 1 parameter sticker)
Activity 4	Adds a distance sensor and a conditional statement that causes the KIBO to move and flash a light only when it's near a wall or other obstacle, but not when far from the wall. (6 blocks, 1 light, 1 parameter sticker, 1 light sensor)
Activity 5	Nests the repeat loop and conditional statement of levels 3 and 4 to cause it to travel forward if near and turn light on if far (7 blocks, 1 light, 2 parameter sticker, 1 light sensor)

CT Assessment Protocol

A consent/demographic survey from the parent/legal guardian of the child was signed and returned before the assessment and interactive play sessions. Prior to the child entering the room, the KIBO robot and blocks were prepared for the first level and the additional blocks, stickers, sensors and modules needed for the remaining levels were placed in a box kept next to the play area.

Figure 5

The set of robotic activities used in the TACTIC-KIBO assessment

Action

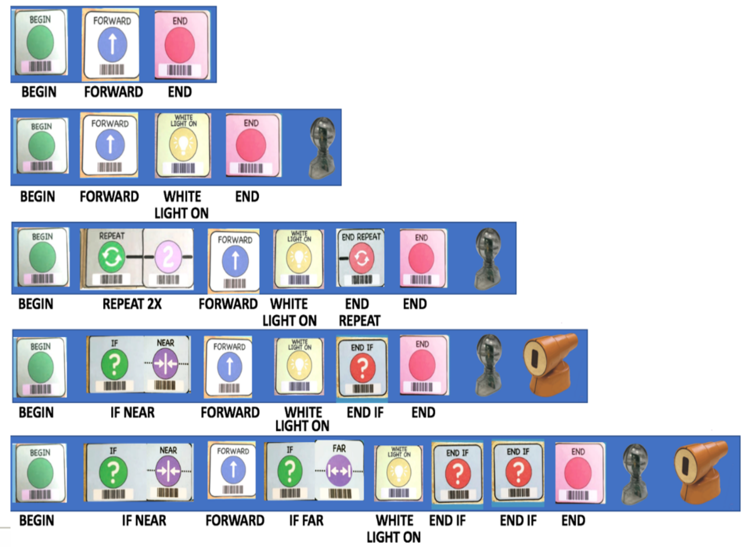
Level 1. KIBO moves forward

Level 2. KIBO moves and flashes light

Level 3. KIBO moves twice as far, flashes twice

Level 4. KIBO moves forward and flashes light only if near wall

KIBO moves forward if near wall, flashes light if far from wall



The assessment began with the examiner introducing themselves and showing the child to the play area containing the KIBO kit. The child was asked whether they have played with KIBO before and whether they remember what the KIBO is called. These questions are primarily designed to place the child at ease. Although these responses were recorded, they are not used in scoring. The child was then told about the planned assessment and asked to give their verbal assent to participate. Only assenting children were tested.

The examiner programmed the KIBO in front of the child using the specified block sequence and, when needed, added sensors and other add-ons. For each question or task, the child was allowed to observe the robot's behavior once and could ask to see it repeated no more than three times. The examiner then proceeded through the series of questions and tasks for each level as listed in Appendix 2. This continued until the child reached a level in which they could not give satisfactory responses for at least two questions or tasks, or until all four levels had been completed. When the child was unable to answer a question or perform a task, the examiner could repeat the question, define words and give general encouragement but could not provide any explanations, major assistance or actual answers. At least every 15 minutes, the child was offered a break for 5 minutes during which they could go to the bathroom, color or just rest. After completion of the TACTIC-KIBO, children were given a 5-minute break and then asked to participate in the Interactive Play Session (IPS)

Scoring Systems

The four levels of activity associated with the TACTIC-KIBO map to the four levels of computational thinking that are the designated outcome measures. Children who were unable complete at least 3 questions/tasks through level 2 were classified as Proto-Programmers. Children who did not successfully complete 4 or more questions in level 2 were termed "Early-Programmers" Children who completed 4 or more questions in level 2 but not in level 3 but were "programmers". Lastly, children who answered 4 or more questions correctly on all levels were categorized as "fluent programmers". The complete set of classifications is shown in Table 3. In addition to these classifications, each response was scored either as 1 point for satisfactory or 0 for unsatisfactory. The total numeric score reflects the total number of satisfactory responses.

Collection of Expert Ratings for Validity/Reliability Determination

To provide a measure of the accuracy of the new CT assessment instrument, an Interactive Play Session (IPS) was carried out with each child. A sample script for the interactive play session is provided in Appendix 3. For IPS sessions, the complete KIBO kit including the robot, blocks, repeat stickers and sensors were made available to the child. Their play sessions were video recorded for later analysis. The author administered and rated all of the children based on the IPS. In addition, 25% of videos of the IPS were also rated by researchers with extensive experience working with KIBO at the DevTech lab. The children's teachers also reviewed videos of the IPS and commented on the child's performance relative to the classroom behaviors but did not formally rate CT abilities.

Table 3

Primary TACTIC-KIBO Scoring system

# Satisfactory Responses	Outcome Category
Up to level 2 >2	Proto Programmer
Level 2 > 2 and Level 3 < 3	Early Programmer
Level 3 > 2 and Level 4 < 3	Programmer
Level 4 > 2	Fluent Programmer

Statistics: The primary outcome of the study was the correlation between the TACTIC-KIBO ratings and IPS ratings by the primary rater. The Pearson correlation coefficients and significance of the correlations were calculated in SPSS version 24 (IBM Corp). The correlation between demographic variables (such as age, gender, past exposure to KIBO and past exposure to other robotics platforms) and outcomes of TACTIC-KIBO were also examined. Additional correlations were carried out for each of the seven categories of CT contrasting TACTIC-KIBO scores with ratings based on the IPS.

Inter-rater reliability measurements: To assess rater bias, 4 of the 14 children (~20%) who completed the IPS were chosen at random and videos of their IPS sessions were reviewed and scored by four DevTech researchers who were blinded to the results of the

TACTIC evaluations. Inter-rater reliability was calculated by determining Cohen's Kappa between the primary examiner vs the four other examiners. The median scores of all examiners were also calculated and compared to those of the primary rater. Raters were provided with a scoring sheet but evaluated videos independently and without consulting or discussing differences.

CT Score Depiction

To efficiently communicate the full set of results in all 7 CT categories, radial plots were prepared in Excel. These graphs include each of the 7 categories as an axis on a seven spoke plot. Scores within each category are noted as points on each axis, with lower score towards the middle of the plot and higher scores towards the edges. Individual category scores were not validated or intended for measurement uses but are included to provide pilot data concerning the applicability of Bers' Seven Powerful Ideas to the CT assessment.

Results

Demographics

In this pilot study, 10 out of 15 children met all inclusion and exclusion criteria. Five children were included who deviated from the original inclusion/exclusion criteria. These deviations were not outside the scope of the populations approved for study under the IRB protocol. During review of video recordings by one of the teachers, it was revealed was one participating child (#10) was diagnosed with high functioning Autism Spectrum Disorder. In addition, it was revealed that English was not the first language of three of the participants (#13,#14,#15) . Finally, it was disclosed that one child (#4) had challenges with respect to hand-eye coordination.

Although the original criteria for inclusion in this study involved typically developing, primary English-speaking children, others were included to obtain a more representative sample. According to the US Centers for Disease Control and Prevention, one in seven children in America have been diagnosed with a mental, behavioral, or developmental disorder (Centers for Disease Control and Prevention, 2012). In 2013, the Census Bureau showed that 21% of American children speak a language other than English at home (Ryan, 2013). Therefore, including diverse and non-native English-speaking participants was considered useful for obtaining a more representative study sample.

Table 4

Demographics Summary

	Overall *n*	Overall %
Grade		
Pre-kindergarten	1	6.7
Kindergarten	10	66.7
First Grade	4	26.6
Age		
5 years	11	73.3
6 years	1	6.6
7 years	3	20
Prior Experience		
KIBO	6	66.7
ScratchJr	6	66.7

**N=15 Prior experience was parent report*

Table 5
Demographics by each participant

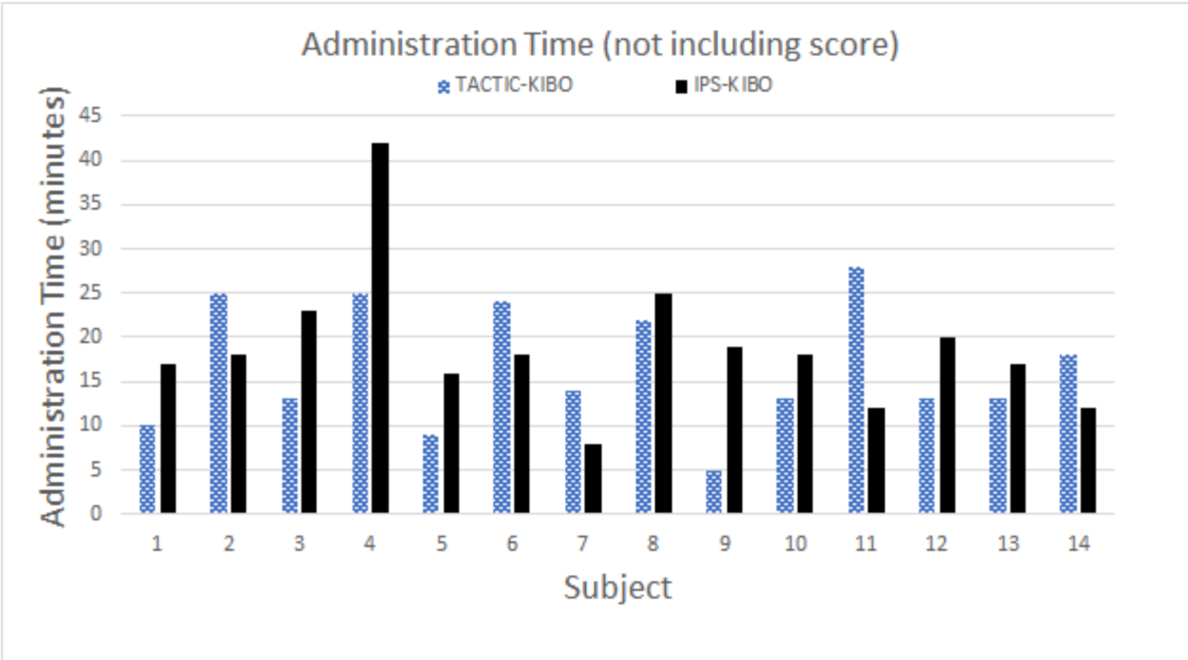
Child Number	Age	Gender	Venue	Grade	Prior KIBO	Prior ScratchJr	Prior Lego WeDo
1	5	female	E.P	K	yes	no	no
2	5	female	E.P	K	yes	yes	maybe
3	5	male	E.P	K	n/a	n/a	n/a
4	5	male	E.P	K	no	yes	no
5	5	male	E.P	K	yes	yes	no
6	5	female	E.P	Pre-K	yes	no	no
7	5	male	T.C	K	yes	yes	no
8	5	male	T.C	K	no	no	no
9	5	male	T.C.	K	no	yes	no
10	5	female	T.C.	K	n/a	n/a	n/a
11	5	female	E.P.	K	yes	yes	no
12	7	female	W.H.	1	n/a	n/a	n/a
13	6	female	W.H.	1	n/a	n/a	n/a
14	7	male	W.H.	1	n/a	n/a	n/a
15	7	female	W.H.	1	n/a	n/a	n/a

Abbreviations: Pre-K = Pre-Kindergarten; K= Kindergarten; EP=Elliot-Pearson Children's School; T.C.= Tufts Bioengineering Camp W.H=Winter Hill Community Innovation School

Administration time

Figure 6 shows the time for administration of TACTIC-KIBO versus the IPS for each participant. The mean time for TACTIC was 16 minutes versus 19 minutes for the IPS. While this difference was small, TACTIC-KIBO was scored during its administration, while IPS took an addition 15 -30 minutes per subject to review performance and establish ratings.

Figure 6



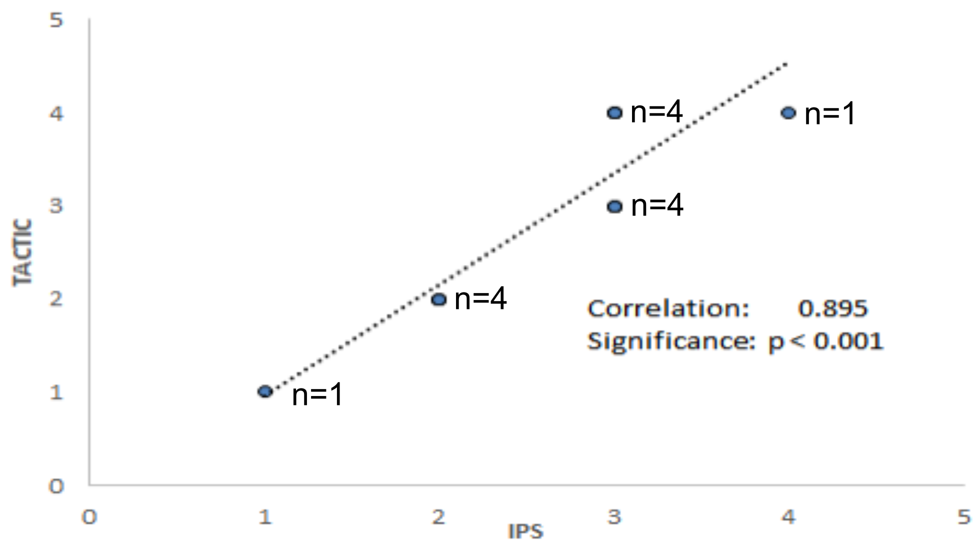
Primary Study Outcome:

The primary outcome measure of this study was the correlation between the TACTIC-KIBO ratings and IPS ratings by the primary examiner. Results are shown graphically in Figure 7.

14 out of 15 children completed TACTIC-KIBO as well as IPS. One child completed TACTIC but did not assent to the IPS. Among the 14 children who completed both assessments, there was a highly significant correlation between the total TACTIC scores and the expert rating of the IPS ($r= 0.895$, $p< 0.001$) Discrepant ratings occurred exclusively among four children rated as Level 4 Fluent Programmers by TACTIC who were judged to be level 3 by IPS.

Figure 7

Scatter plot showing correlation between primary outcome, total TACTIC-KIBO score and expert rating of KIBO Interactive Play Session (IPS-KIBO). N=14



Correlations with demographic variables

Since CT abilities develop with age and can be acquired through learning, it was predicted that variables such as age, grade and past experience with KIBO would interact with the assessed level of CT abilities. However, none of the demographics variables correlated with TACTIC-KIBO or IPS-KIBO. TACTIC-KIBO and IPS ratings did correlate significantly with one another, as previously noted.

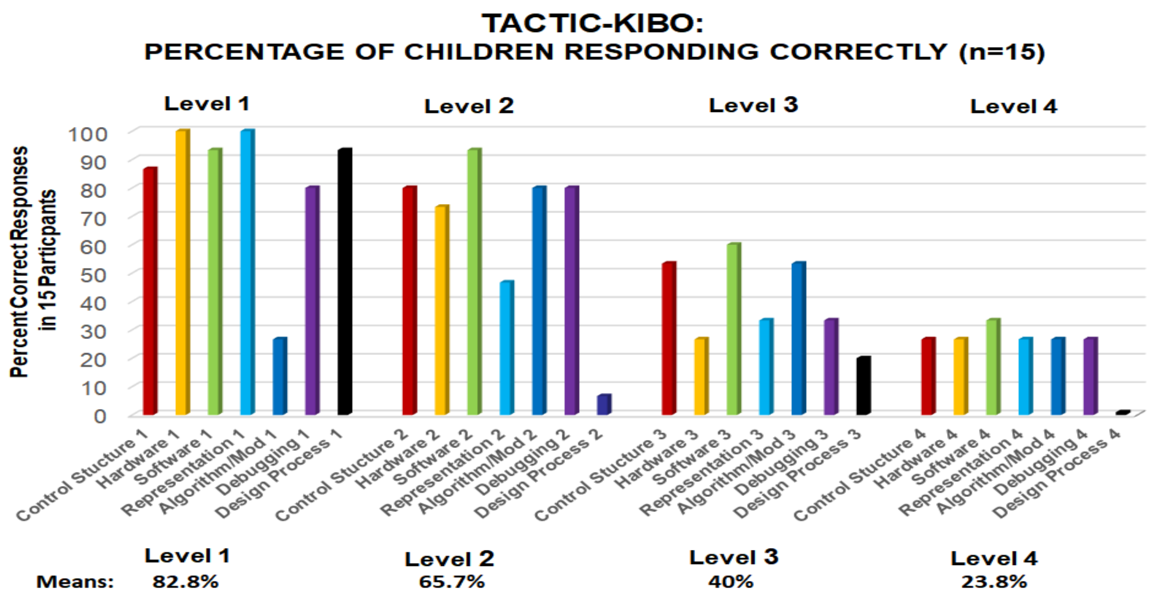
Table 6
Correlation of TACTIC-KIBO and IPS-KIBO with Demographic variables

		TACTIC Total	IPS Total
Age	Pearson R	0.135	0.181
	p value	0.322	0.268
Gender	Pearson R	0.365	0.431
	P value	0.100	0.062
Venue	Pearson R	0.076	0.154
	p value	0.397	0.300
Grade	Pearson R	0.162	0.191
	p value	0.290	0.256
Prior KIBO Experience	Pearson R	-0.186	-0.306
	P value	0.262	0.143
Prior ScratchJr Experience	Pearson R	0.014	-0.038
	P value	0.481	0.448
Prior LegoWeDo Experience	Pearson R	-0.086	-0.172
	P value	0.385	0.278
TACTIC-KIBO Total	Pearson R	1	.895*
	P value		0.000
IPS Total	Pearson R	.895*	1
	P value	0.000	

*p <. 001

To better understand how the individual questions and tasks with TACTIC-KIBO performed at each level, the total number of correct responses per question in all 15 subjects were calculated and plotted. (Figure 8)

Figure 8



Each bar in Figure 8 reflects the percentage of correct responders among the 15 participants for one question in each category on each level. Categories not queried or answered incorrectly were counted as incorrect responses.

It can be readily appreciated from Figure 7 that certain questions evoked more incorrect responses at a given level than others. For example, across all questions on level one, the average number of correct responders was 12 out of 15 participants. However, the Level 1 Algorithms and Modularity question had only 4 correct responders out of 15. When the responses to this question were examined in greater detail, an unexpected pattern was observed. Virtually every one of the children who scored high on their Total TACTIC-KIBO score responded incorrectly to the Level 1 Algorithms and Modularity question, while the four correct responders had TACTIC-KIBO total scores or 1 or 2. Possible explanations for this unexpected pattern of responses are provided in the Discussion section.

Other questions that resulted in a greater than expected percentage of incorrect responders included the Design Process probes on Levels 2 and 4.

Inter-Rater Reliability

Because a single examiner administered and scored all of the TACTIC-KIBO and IPS-KIBO assessments in this study, it was necessary to test for rater bias. To do so, four outside raters were recruited who had expertise in teaching and assessing CT abilities. These raters were shown video recordings of the IPS of four of the participants. The IPS total ratings of the primary rater (dark blue) and the four other raters are shown in Figure 8.

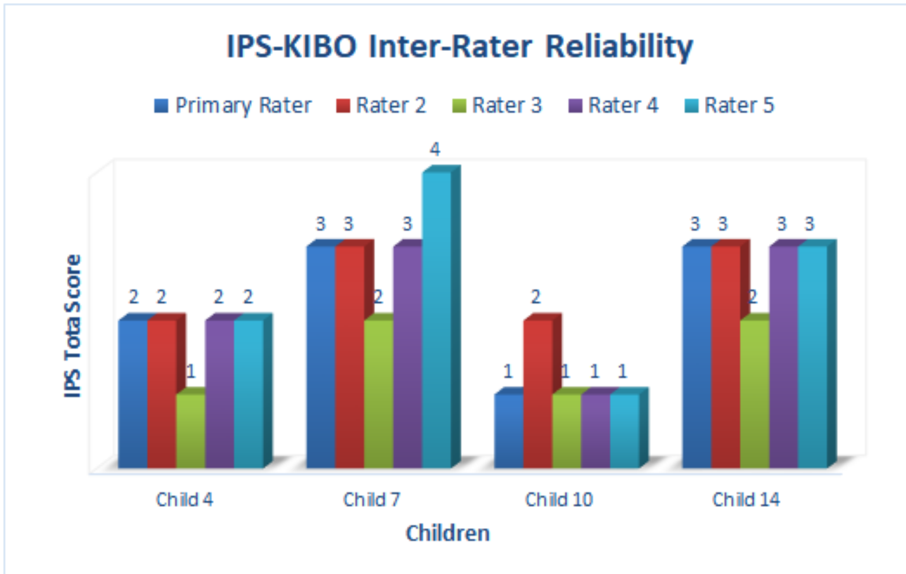
In Figure 9, the total IPS scores for 4 children are shown for the primary rater (dark blue) and four other expert raters. The primary rater's evaluation exactly matched the median and mode for all raters in each case, for a Cohen's kappa of 1.0 relative to the median. Cohen's kappa for the primary rater compared to each of the individual raters is found in the table below

Table 7

Cohen's Kappa for Primary rater vs all others

	Primary vs Rater 2	Primary vs Rater 3	Primary vs Rater 4	Primary vs Rater 5
Cohen's Kappa	**0.6 (moderate)**	**0 (none)**	**1.0 (strong)**	**0.667 (moderate)**

Figure 9



Some variability in the IPS ratings is evident across examiners. However, the median scores across all raters for each of the children matched that of the primary examiner exactly. There was no indication of bias in ratings by the primary examiner based on these inter-rater reliability measurements.

Individual Participant Results

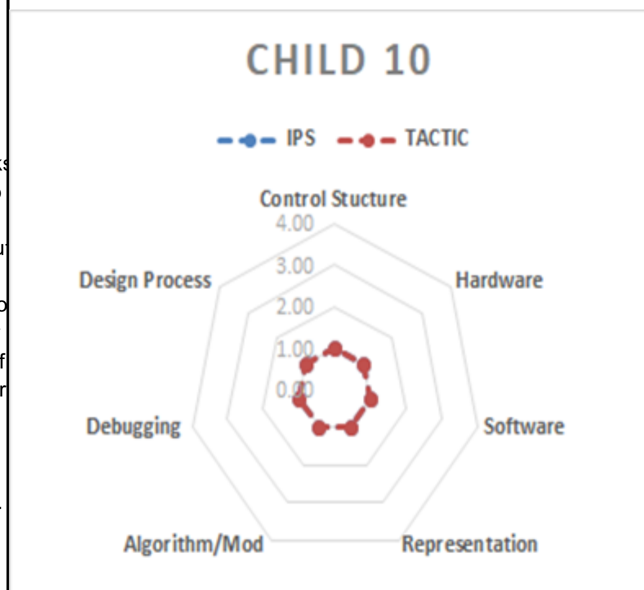
TACTIC-KIBO was designed to measure overall CT abilities in young children by assessing performance in the seven domains derived from Bers' Seven Powerful Ideas of Computational Thinking. All results presented thus far have pertained to the Total TACTIC-KIBO and TOTAL IPS Scores as measures of overall CT ability. Individual categories of CT are shown in the form of radar plots below. Each radar plot shows the highest attained level for each category on TACTIC and the examiner's ratings for each category on the IPS. Four cases are shown below. The complete data set for this study is found in Appendix 1.

Level 1 Example

Child 10

Age	Gender	TACTIC-KIBO DURATION	IPS DURATION	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC -KIBO SCORE	IPS SCORE
5	Female	5m11s	18m49s	no	no	1	1

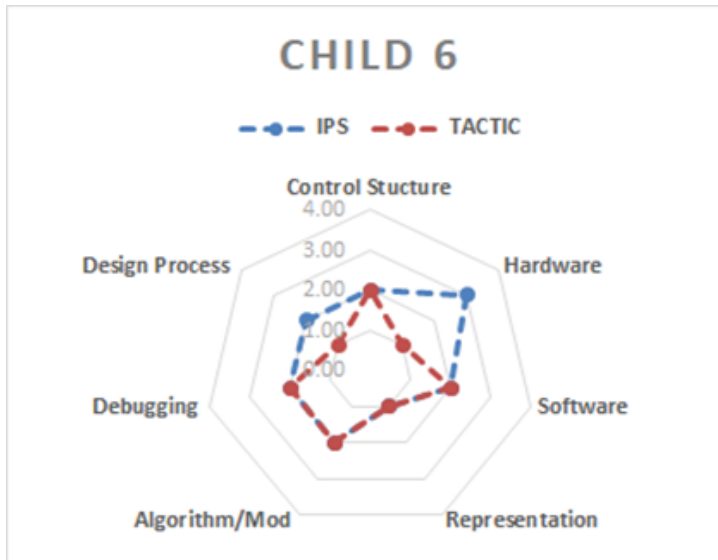
Examiners Comments: Child does not know how to use the robot. Child always approaches KIBO backwards, starting with end and going to begin. Child thinks robot worked because we only scanned "two blocks" . Knows that green button turns robot on. Thinks that the bad sound is the good sound. Thinks light bulb works automatically. Says "the batteries are trying to see something from the KIBO, they are trying to see ou of the little glass pieces" points to the two bumps near the batteries that the screws come out of. Has the moto skills to scan but does not know which block to scan or what to scan. Tries to debug by putting sensors in all of the robots ports. Does not know what any of the sensor do. Child is dragging robot around. Child cannot be pushed to Zone of proximal dev. Child scans all of parameter cards at once in a line. "the light Is sticking out because of wires". Child begins scanning her shoe.



Teacher Comments Says that child is on the Autism Spectrum (Not disclosed prior to testing). She says that her mom reports she sometimes has outbursts, recedes, and has trouble following instructions but that she did not observe this happening in the camp. However, teachers says that they could tell with their research testing that she was very regimented and often repeated activities. She has a better than expected reading abilities for someone her age. Teacher brings up that the child scans from end to begin. "That's something that I have seen before, when they are just developing KIBO awareness... I thought it was reall interesting because she had such high reading ability so you would think she would scan from left to right… I think some children just work backwards"(Strawhacker, Personal Conversation 2018).

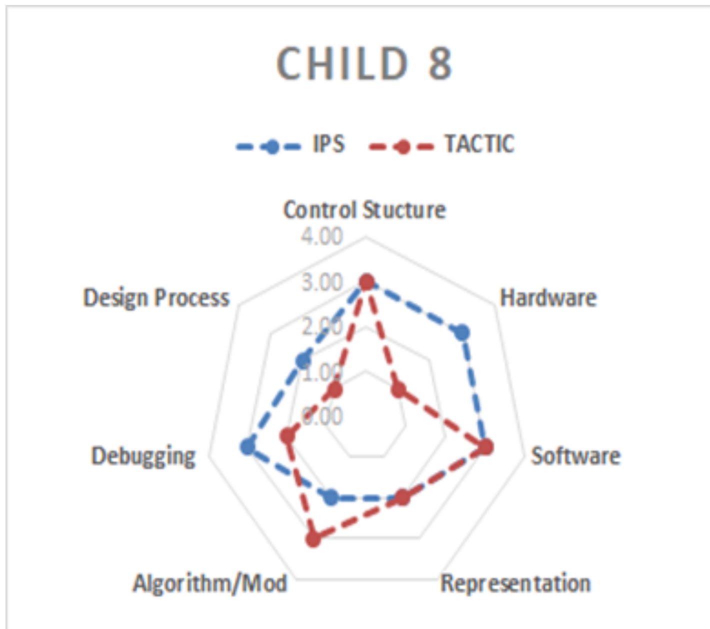
Level 2 Example

<table>
<tr><th colspan="8">Child 6</th></tr>
<tr><td>Age</td><td>Gender</td><td>TACTIC-KIBO DURATION</td><td>IPS Duratio</td><td>PRIOR KIBO</td><td>PRIOR SCRATCH.JR</td><td>TACTIC-KIBO SCORE</td><td>IPS SCORE</td></tr>
<tr><td>5</td><td>Female</td><td>9m21s</td><td>16m06s</td><td>yes</td><td>yes</td><td>2</td><td>2</td></tr>
<tr><td colspan="4">Examiner's Comments: When child comes into room she immediately starts t play with sensors saying that she is "putting on everything she can" Seems excited about the robot. She is able to debug simple programs. She did not know exactly how KIBO works in that she often forgets begin and end and sometimes forgets which button to press in order for i to go. During IPS, she says her idea is to put every single block together to make the "biggest program"

Teacher Comments: Child is very advanced for her age. She is young five years old and is often able to understand concepts such as reading, writing, art that her peers do not yet understand. Child is bilingual.</td><td colspan="4">
</td></tr>
</table>

Level 3 Example

<table>
<tr><th colspan="8">Child 8</th></tr>
<tr><th>Age</th><th>Gender</th><th>TACTIC-KIBO DURATION</th><th>IPS Duration</th><th>PRIOR KIBO</th><th>PRIOR SCRATCH.JR</th><th>TACTIC -KIBO SCORE</th><th>IPS SCORE</th></tr>
<tr><td>5</td><td>Male</td><td>13m40s</td><td>8m26s</td><td>no</td><td>no</td><td>3</td><td>3</td></tr>
<tr><td colspan="4">Examiner's Comments: Child comes up with unique and creative ways to solve the problem that aren't always right. Child does not know how to use end repeats but has no trouble finding repeat and correct number parameter cards. Child is not sure how to use If blocks and confuses if and repeat parameters. Child only wants to try out one program in IPS(which does not work because it has an improperly placed "if" in it) child debugs this by taking out if block altogether.

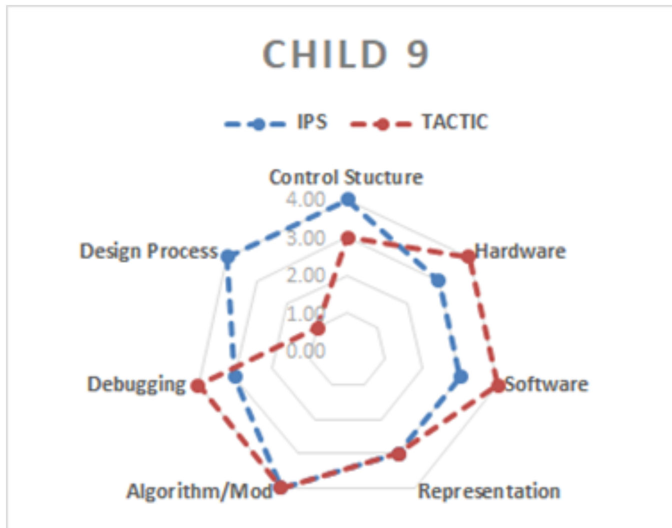
Teacher comments: Teacher says that he is really tactile with tools he is exploring. He often picks an idea and doesn't know how test it so he just forgets about the idea. He is creative.</td><td colspan="4">
</td></tr>
</table>

Level 4 Example

Child 9							
Age	**Gender**	**TACTIC -KIBO DURATION**	**IPS Duratio**	**PRIOR KIBO**	**PRIOR SCRATCH.JR**	**TACTIC-KIBO SCORE**	**IPS SCORE**
5	**Male**	**22m**	**24m45s**	**no**	**yes**	**4**	**4**

Examiner comments: Child performs modularity by picking up groups of blocks at once and placing them into his program. At first child does not understand that you need an end repeat and an end if after implementing repeat/ blocks. Child then confuses end if and end repeat. However, after watching me correct the program in TACTIC, Child learns and applies this knowledge to IPS. Child knows all blocks asked except he thinks that the ear is for smelling and is not sure what the wait for clap block does. Child comes up with a very comple program for IPS with a nested loop.

Teacher Comments: Says that the child is quiet observes and really understands concepts. He is extremely intelligent and was often explaining concepts to other students in the camp. Might speak a different language at home



CT Category Correlations of TACTIC KIBO vs IPS

Although TACTIC-KIBO was designed for measuring overall CT abilities rather than performance in all of the seven CT categories, it was of interest to examine whether any categories correlated on the IPS verses TACTIC. A surprising degree of correlation was observed with the exception of the Design Process category, which was identified in other analyses as one in which the TACTIC-KIBO questions did not perform optimally.

Table 8

Correlation Matrix for CT Categories

		TACTIC-KIBO Control Structure	TACTIC-KIBO Software	TACTIC-KIBO Hardware	TACTIC-KIBO Representation	TACTIC-KIBO Algorithms	TACTIC-KIBO Debugging	TACTIC-KIBO Design Process
IPS Control Structure	Rho	.493*	.522*	.569*	.304	.612*	.603*	.263
	p	.037	.028	.017	.145	.010	.011	.182
IPS Software	Rho	.774**	.791**	.699**	.621**	.702**	.782**	.496*
	p	.001	.000	.003	.009	.003	.000	.036
IPS Hardware	Rho	.471*	.559*	.483*	.351	.532*	.428	.162
	p	.044	.019	.040	.109	.025	.063	.291
IPS Representation	Rho	.425	.587*	.779**	.619**	.430	.357	.281
	p	.065	.014	.001	.009	.062	.105	.165
IPS Algorithms	Rho	.634**	.727**	.762**	.548*	.664**	.807**	.355
	p	.007	.002	.001	.021	.005	.000	.106
IPS Debugging	Rho	.729**	.693**	.474*	.374	.707**	.661**	.265
	p	.002	.003	.043	.094	.002	.005	.180
IPS Design Process	Rho	.559*	.625**	.597*	.392	.576*	.703**	.177
	p	.019	.008	.012	.083	.016	.003	.272

Table 8 shows the correlation matrix for categories on TACTIC-KIBO vs IPS-KIBO. All rho values are Spearman correlation coefficients. P values calculated for Spearman rho. Red highlighted Rhos are highly significant ($p<0.01$). Yellow highlight values are significant ($p<0.5$). This table shows multiple correlations between the majority of the CT categories of the IPS and TACTIC-KIBO. The TACTIC-KIBO Design Process category has only one moderate correlation to IPS software. Based upon other analyses (See Figure 8) it seems likely that the lack of correlation for KIBO Design Process is due to issues with the questions/tasks of that category.

Discussion

In this pilot study, TACTIC-KIBO showed considerable promise as a means of assessing computational thinking abilities in young children. Using pre-specified criteria for administration and scoring, TACTIC-KIBO identified four levels of CT abilities ranging from novice to fluent in kindergarten and first grade children. TACTIC-KIBO scores correlated significantly with expert assessments based on observation of KIBO interactive play sessions. According to past research and recommendations, administration time and ease of scoring were suitable for use in classroom and research settings (Moyer & Gilmer, 1953; Ruff & Lawson ,1990).

Among its strengths, TACTIC-KIBO was engaging and enjoyable for the majority (14/15) of children tested. This included students who were not yet fully literate and in whom English was a second language. TACTIC-KIBO duration proved suitable for 5-7-year old children despite the relatively short attention spans of children in this age group. It was easy to administer in a school setting and could be completed in an average time of 16 minutes.

All participants completed the TACTIC-KIBO testing, but one child was not cooperative with IPS testing. That child had been taken away from a group of friends during recess to be tested and may have felt resentment towards the testing as a consequence.

Scoring of TACTIC-KIBO was relatively straightforward, although judging answers as either "right" vs "wrong" created some challenges especially when responses seemed partially correct. Refinements to the scoring criteria may help to simplify this aspect of scoring in future versions of TACTIC-KIBO. In contrast to the rating of the IPS which took about 30 minutes per subject (in addition to the average play session time of 19 minutes), TACTIC-KIBO scoring took less than a minute to complete and was carried out during the assessment itself.

TACTIC-KIBO classified four children as "Fluent Programmers" (Level 4) who were rated as "Programmers" (Level 3) based on observation of the IPS. While the source of this discrepancy cannot be fully identified based on the available data, there are reasons to question the accuracy of the IPS in these cases. The IPS is not a true gold standard for measuring CT abilities. An examiner's ability to rate a child's CT abilities based on review of the IPS depends on the skills the child happened to display (or not display) during the play sessions. Children were allowed to play freely during IPS with only general direction and encouragement from the examiner. As a consequence, some children who are fluent programmers may not have engaged in level 4 activities during the play sessions. In contrast, TACTIC-KIBO is a standardized test that probes all four levels of CT abilities in children who perform sufficiently well on the first three levels. TACTIC KIBO may outperform the IPS in assessing level 4 abilities as a consequence.

Past studies have shown that less structured testing environments like the IPS can lead to underperformance in children's coding exercises. For example, a past study looking at the acquisition of debugging skills examined why failures in debugging occurred in young children. Researchers reported that environmental factors such as fatigue, interruptions and other external (or internal) distractions can have an adverse effect on the child's debugging performance (Selby, Dorling, & Woollard, 2014). Since TACTIC-KIBO was implemented before the IPS, it is possible that children were either fatigued or distracted by the time that the IPS was conducted and did not perform at their best ability due to this type of environment. Additionally, although the IPS is designed to get children to their highest level of CT ability it is possible that children simply did not want to perform in this way due to the structure of this play session. A more standardized and consistent assessment such as TACTIC might be more helpful for measuring CT abilities in young children.

The numerical scores shown in Figure 10 provide further evidence that TACTIC-KIBO may have correctly identified the set of fluent programmers among the study group. Inspection of the data reveals that as subjects are grouped by TACTIC or IPS at levels 1, 2 and 3, there is a jump in the number of correct responses. There is a numerical transition of this kind at the point of the first of the 5 subjects scored by TACTIC as level 4. That subject scored 20 correct responses compared to only 16 in the subject before them, who was ranked by both tools as a level 3. Although this evidence is somewhat circumstantial, it suggests that TACTIC may be more sensitive to distinguishing level 3 from level 4 CT abilities than IPS.

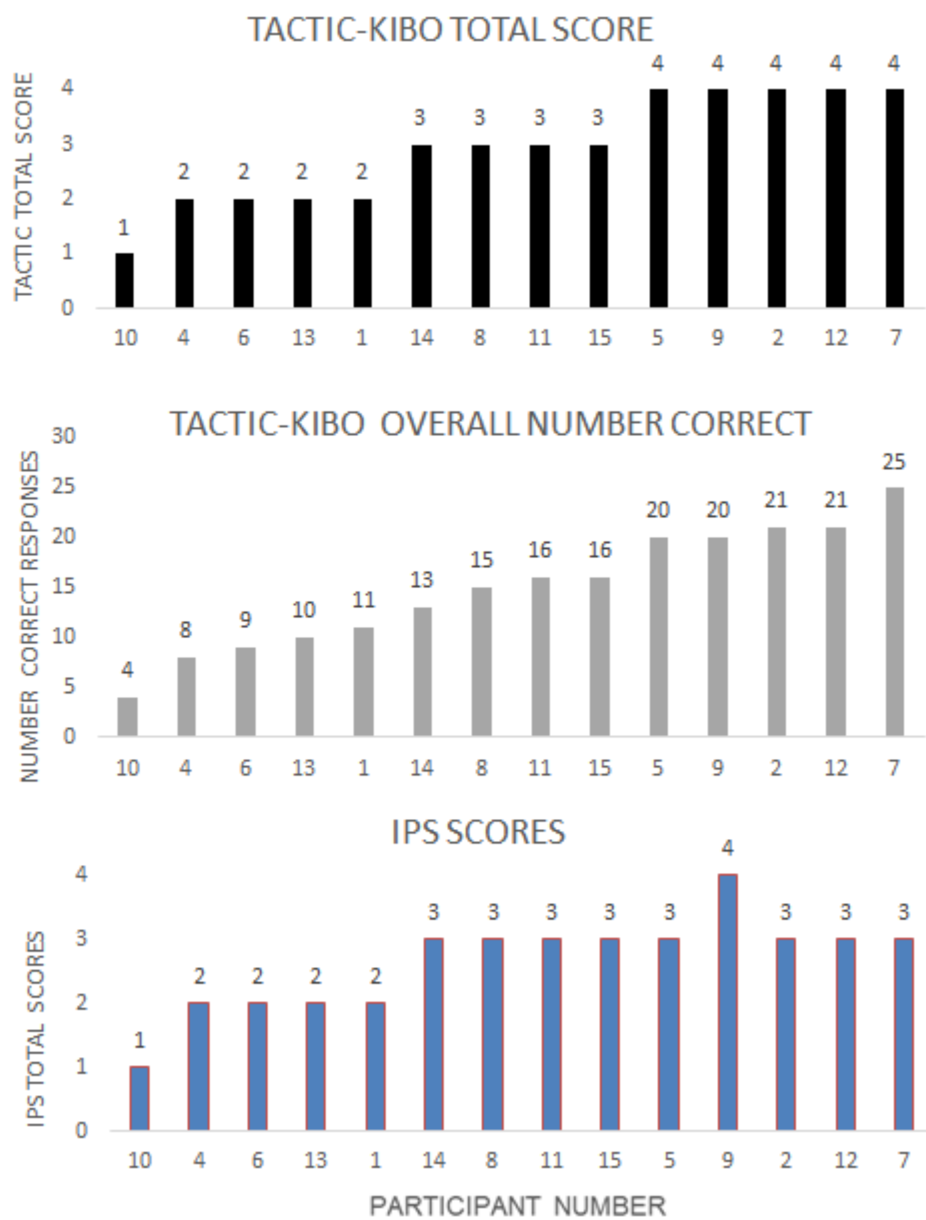
One possible way to address this question in future studies would be to allow experts to rate children first based on an actual interview and observation period rather than a pre-recorded video, so the degree of certainty about their abilities is higher. TACTIC-KIBO assessment would then be carried out blind to the expert's rating. This may provide a greater level of certainty about the child's abilities than the current protocol afforded.

Another possible interpretation of the observed discrepancy between TACTIC-KIBO and IPS is that level 3 and level 4 are not truly distinct as a result of conceptual issues with the 4 level rating system used in this study. However, there are indications that Levels 3 and 4 are separable from the numeric analysis of TACTIC-KIBO's performance. The mean number of correct responses was distinct for each of the four levels (Figure 8). Only one child was rated as level four on IPS, but this indicates that raters were able to identify features that distinguished programmers from fluent programmers in their assessments.

None of the children in this pilot study in either kindergarten or first grade were able to answer Design Process level 4 correctly which involved using the conditional "if" blocks in a nested statement. It was suggested that in the future I should leave out conditional blocks until level four to make this level more challenging. Further evidence addressing this question can be obtained by using TACTIC-KIBO in children of higher grades (e.g.: 2-4) . Typically, when learning CT, it is recommended that conditionals are taught last. Rich, et al., (2017) propose that learning the sequencing aspect of Computational Thinking should follow this trajectory: first, one should start with learning that order matters and how specificity is important when communicating directions; next, these concepts should be taught in relation to computer programming; and, lastly, after these skills have been mastered, students learn more complex patterns of sequencing such as

Figure 10

Figure 10 shows the CT ratings of 14 participants sorted from lowest to highest. Top graph shows Tactic Score based on standard scoring. The middle graph depicts numeric scoring of total questions correct on TACTIC Bottom graph shows IPS scores.



conditionals and loops. However, there is currently a paucity of information about what age conditional programming instructions should be introduced to children.

Children ages 5-6 may have a limited ability to grasp the programming of conditionals (e.g.: if-then) which are closely associated with level 4 CT abilities. According to Barrouillet and Lecas (1999), younger children have a very limited interpretation of basic conditionals. Empirical studies suggest understanding of conditionals develops gradually and progressively between ages 6 and 12 (Janveau-Brennan, & Markovits, (1999). Additionally, Lynne May Lim, a very experienced kindergarten teacher at the Eliot Pearson Children's School and Amanda Strawhacker, an expert at KIBO robotics shared their thoughts.

> *"I think at this point in their development the ifs are hard because it is much more abstract and is not as concrete. Even with first graders (it is difficult) because I have seen them use it too…. I would expect most kindergarteners to be at level 3…" (Lynne May Lim personal communication, 2018).*

> *"Personally, I don't often include it (if blocks) in curricula unless I know we have a lot of time to master the other basics of KIBO because it is definitely the hardest concept... You have to have the working memory to hold something in your mind and synthesize to compare it so something else. I have seen first graders do it after they master all other concepts but I think really by second grade they can get it pretty easily" (Amanda Strawhacker, personal communication, 2018).*

Children ages five through seven go through a pivotal change in cognitive development from Piaget's preoperational stage to the concrete operational stage. Theorists such as Piaget have shown age has an effect on the child's cognitive abilities. Piaget believed that children ages 4-6 are in a pre-operational stage and still in the process of learning how their environment and objects work. At the same time they are egocentric and struggle to see the world from other perspectives. Children in this age range may exhibit magical thinking. Once a child moves to the concrete operational stage at age 7, they begin to think logically, their problem-solving skills develop, and they begin to become able to see things from other perspectives (Piaget, 1959).

That being said, past studies have shown that age did not correlate with performance on conditional and repeat programs with KIBO in children (Elkin, Sullivan, & Bers, M.U. (2016). Children aged 5 were able to understand concepts of representation and abstraction that Piaget would not predict happening until age 7 or older (Strawhacker, Sullivan, & Bers, 2013). We did not see clear age differences in this study. However, other variables that influence performance may not have been sufficiently well-controlled to permit accurate measurement of age-related effects.

In this pilot study, the TACTIC-KIBO administration and scoring systems were designed to measure overall CT abilities. This approach may not optimally assess the highest level of performance in the individual categories related to the Seven Powerful Ideas. A caveat in the interpretation of the data on mean number of correct responses per level is that the method of administration used in this pilot allowed children to advance to the next level of questions only if they successfully responded in three or more categories on the preceding level. The assumption in the analysis of correct responses is that children would not be able to respond correctly on higher levels if they did not meet the three-correct response criterion on the lowest levels. It is possible, even likely, that this approach underestimated the total number of correct responses children may have provided in a given category. Analysis of individual responses within categories suggest this may be the case as evidenced by the fact that some children gave incorrect responses on lower levels and nevertheless responded correctly on higher levels.

Expert ratings based on the Interactive Play Sessions may provide more accurate ratings of performance within each of the categories, although the IPS was not specifically designed to assess every category with equal sensitivity. Future versions of TACTIC-KIBO could be created with expanded questions and task sets to better assess CT abilities within specific categories.

As stated above, one of the limitations of TACTIC-KIBO in the present study is that it likely underestimates some of the individual categories of CT ability owing to the method of administration employed. One potential way to address this in future versions of TACTIC would be to use some of the administration strategies of intelligence tests such as the RIAS-2 (Reynolds Intellectual Assessment Scales Second Edition). In the RAIS-2 assessment, the examiner starts with questions based upon the examinee's age (the higher the age the harder the questions). Examiners also must establish "basal levels" and implement "reverse rules". Establishing a basal level means that a person taking the assessment must answer two consecutive questions correctly before moving on to harder questions. If a basal level is not obtained then the examiner must use the reverse rule meaning that they administer the test in reverse order until a basal level of two questions is answered correctly. Once the basal level is established the examiner must return to the point in which they originally began the test and ask questions until the examinee gets a prespecified number of questions incorrect (Reynolds & Kamphaus, 2015). This way of scoring and administration is much more complex and requires a more skilled and trained examiner. As a consequence, this may be best implemented on an electronic platform such as a tablet that can provide real time feedback and prompts to the examiner. Starting at a higher level for some children based on age and skill level could save time and increase efficiency of CT testing, but was beyond the scope of the present study.

Despite this limitation, there was consistency in the rate of correct responses to tasks across categories within each level, with a few notable exceptions. One notable and very interesting exception was the Level 1 Algorithms and Modularity question. After being shown the execution by KIBO of a three-step program consisting of the blocks "BEGIN" "FORWARD" and "END," children were asked to predict what would happen if the "END" block was removed. In a remarkably consistent manner, children who scored highest on TACTIC-KIBO overall incorrectly responded that the robot would keep moving forever or that it would perform the same program as BEGIN-FORWARD-END. In contrast, most children who scored lower on TACTIC-KIBO overall responded correctly, stating that the robot would not do anything if the "END" block was removed.

This unexpected response to the Level 1 Algorithms and Modularity probe may be a consequence of a peculiarity of KIBO programming that diverges from rules in programming other platforms, such as the screen-based app, Scratch Jr. In KIBO, all programs must begin with a "BEGIN" block and stop with a "END" block. This is analogous to higher level programming languages such as JAVA where programming statements need to end with a semicolon in order to be recognized as valid syntax. However, other computer programming languages such as Python forego the use of punctuation at the end of statements. The Level 1 TACTIC-KIBO Algorithm/Modularity question was based on the expectation that children at that level would understand the "BEGIN" and "END" rule that is a basic part of KIBO coding. In carrying out programming tasks independently on the IPS, children that scored higher on TACTIC-KIBO did successfully remember to end their programs with an "END" block. However, it was surprising to observe that the children with the highest rated CT abilities did not correctly predict the consequences of taking away an "END" block from a working program during TACTIC-KIBO testing. Their responses seem to suggest a local and concrete interpretation based on the semantic meaning of the "END" block function rather than a more global, syntax-based, rule-based interpretation. The higher performing children seemed to interpret each block literally and serially, in this case seeing the robot as starting the program as a consequence of the "BEGIN" block, moving forward as commanded by the "FORWARD" block and then continuing indefinitely because there was no command to "END." Even more surprising, children who were rated lower on the total TACTIC-KIBO score gave the expected correct response that the robot would do nothing if the "END" block was removed. Another possible factor contributing to this unexpected result could be the order in which the questions were administered. The question that is placed immediately before is Level 1 Representation which asks "What does the red (END) block do?" This question may have primed higher performing children into thinking of the END block in a more local/ concrete way. Considered in context, the END block can be thought of as stopping the robot's previous action of moving forward. In any case, the Level 1 Algorithm and Modularity probe failed to perform as expected and will need to be modified in future versions of TACTIC-KIBO. Possible alternatives are to change the order of questions or to ask the child to predict the consequence of removing the "BEGIN" block. This interesting occurrence involving the level 1 modularity category points to the need for further research on the relationship between computational syntax, computational semantics, and overall computational thinking.

Another category of TACTIC-KIBO probes that appeared to have performed suboptimal were those designed to measure Design Processes. Design Processes Levels 2, 3 and 4 were virtually indistinguishable by TACTIC-KIBO, as seen on the bar graph of pooled results from 15 children (Figure 8). In the interactive play session, participants exercised design process skills, as evidenced by their planning and testing various coding ideas and projects. In TACTIC-KIBO, the design process probes involved asking the child to plan modifications to the code that would lead to a specific change in the robot's behavior when running a given program. The difficulty that most children had with this class of task may indicate that it is harder at this age to modify someone else's program than it is to modify their own code from scratch. To improve the performance on these questions, it may be necessary to give multiple choices rather than ask the child to suggest a solution themselves. Scoring may be made more liberal by allowing responses that indicate the right blocks or sensors to be added without having to specify their exact place in the modified program.

CT Ratings Based on Interactive Play Sessions

The primary purpose of the Interactive Play Sessions in this study was to serve as a yardstick for measuring the performance of the TACTIC-KIBO instrument and help establish validity. Inter-rater reliability was conducted for the Interactive Play Session by having four additional expert raters evaluate videos of four of the interactive play sessions each.

A reasonably positive correlation was observed between TACTIC-KIBO and IPS expert ratings, except for three children who were misclassified as Level 4 instead of Level 3. Inter-rater reliability was adequate for the primary rater (E.R.) relative to three of the four expert raters. One expert's ratings failed to correlate with any of the other raters on the overall CT score or in the seven individual category ratings.

The IPS ratings also provide a window into the children's abilities within each of the CT categories based upon Bers' Seven Powerful Ideas. Inter-rater reliability for the individual category rating was lower than for total CT level score but still adequate relative to the primary rater and three of the four expert raters. These results suggest that there were not major biases in the ratings performed by the primary rater. They also illustrate that even among experienced raters, evaluating CT abilities based on observation of interactive play is subject to variability. Although this study did not directly evaluate variability in TACTIC-KIBO ratings, it should be subject to less variability than the IPS owing to its more structured design and objective scoring system. Future studies should have multiple raters score the same TACTIC-KIBO performances to see if there is inter-rater reliability of this assessment.

Adult assistance provided during the IPS was designed to bring children to their Zone of Proximal Development. However, raters may have taken note of when assistance was required and may have downgraded their CT ratings, particularly in candidates for level 4, as a consequence. The IPS required more improvisation and expertise by the examiner and was more time-consuming than TACTIC-KIBO to administer and score. All IPS sessions were administered by the same rater (E.R.). In scoring the IPS for this pilot study, no special training could be given to the outside examiners and no other interventions were carried out to obtain mor uniform ratings across examiners. In future studies, training videos can be provided and a certification process carried out to assure more uniform ratings. As with TACTIC-KIBO, IPS is subject to variability in performance owing to fatigue, shifting attention and other factors.

What does TACTIC-KIBO teach us about CT and its measurement?

TACTIC-KIBO was designed based on the principles in Bers' Seven Powerful Ideas and Vizner's Developmental Model of Programming. Total TACTIC-KIBO scores aligned quite well with those of expert raters. This result can be considered a partial validation of the principles embodied in Bers' and Vizner's works. If the Seven Powerful Ideas of Computational Thinking were not valid dimensions of computational thinking abilities, it is unlikely that TACTIC-KIBO would have succeeded to the extent that it did. Additional studies will be needed to fully validate this conceptual framework.

Another important aspect of the theory behind TACTIC-KIBO relates to the concept put forth in Mioduser & Levy (2010) that children's interpretations and descriptions of robotic behavior provide a window into their computational thinking abilities. The experience of carrying out this pilot study has strongly supported that idea and established that it is a viable approach to measuring CT abilities in young children. Nevertheless, practical considerations mandate that children of this age have some hands-on play to be fully engaged, and not just passively view and describe what an adult shows them the robot can do. So, while TACTIC-KIBO uses observation and description to probe CT abilities as part of a structured experience, it has some elements in common with the more free-form interactive play session.

It is unclear from the present study whether a four-level rating system for CT abilities in young children is optimal. Differentiation between level 3 and level 4 was problematic in some cases and somewhat variable between TACTIC-KIBO and IPS. With more stringent scoring of TACTIC-KIBO avoiding adult assistance, it seems possible that a three level scoring system might have sufficed. However, abilities consistent with level 4 are likely to be more common in children in higher grades and in those with more extensive experience with coding. It therefore seems justified to continue using the four level scoring system going forward. While the available data does not permit us to fully validate Vizner's Developmental model, the four level hierarchy did provide a useful construct for rating CT abilities using TACTIC-KIBO and also contributed significantly to the rating system used for the IPS.

While TACTIC-KIBO provided a good measure of overall CT abilities, it was not administered in a way that allowed accurate assessment of CT abilities in all seven categories of Bers' Seven Powerful Ideas. Each level of TACTIC-KIBO has only one question or task for each category. This limitation was necessary to keep the administration short and appropriate for kindergarten age children. In addition, the administration stopped at whatever level less than four responses were correct. This approach also expedited testing but did not give children the opportunity to show higher level abilities in specific categories even if their overall TACTIC-KIBO score was lower. In this regard, the IPS may have been a better instrument for assessing performance in a given CT category since it allowed the child to demonstrate their highest level of ability in different categories. For example, a child with level 2 CT abilities could score at level 4 on a category such as hardware. Although that scenario could not occur with TACTIC-KIBO for the reasons stated above, it is one possible outcome of the IPS. It is possible that TACTIC-KIBO could be revised in ways that enhance its ability to assess CT within the seven categories. For example, versions of TACTIC-KIBO that focus on specific categories could be devised that might sacrifice obtaining an overall score for greater sensitivity in measuring abilities in just hardware, software, or one of the other categories.

Although the focus of this study was the creation and initial validation of an instrument to assess CT in young children, the testing process provided an opportunity to observe Bers' Seven Powerful Ideas in action. It is encouraging that the Seven Powerful Ideas provided such a good framework for the categories of CT assessment in young children. We observed that certain categories lent themselves to all four levels of assessment better than others. All four levels of ability were readily distinguished for categories such as hardware, representation and control structure. Categories such as algorithms /modularity and design structure proved more difficult to distinguish across levels. This difficulty may reflect the properties of the respective categories or an inherent order in the development of abilities in these categories. Concepts such as hardware that involve concrete artifacts may be the most easily understood by young children at all levels. More abstract concepts such as design processes are less readily probed in children, particularly in children with lower levels of CT abilities.

Although the small data set collected in this pilot did not allow extensive statistical analyses, there was significant correlation among certain categories in the IPS ratings. To assess the underlying factors in the assessment of CT, a larger data set (n>100) would lend itself to techniques such as Factor Analysis. Such techniques may allow distilling a subset of questions from the TACTIC-KIBO instrument that best predict overall CT abilities, allowing a potential reduction in the testing time required.

Study Limitations

Limitations to this study include the sample size, biases in subject selection, biases in test administration and ratings, context, and issues relating to the lack of availability of a true gold standard for measuring CT in young children.

Fifteen children consented to testing in this pilot study. This number was sufficient to gain some initial insights into TACTIC-KIBO's performance, including the extent to which the instrument is practical to administer and suitable for assessing CT abilities in young children. However, fifteen participants is too small a sample to be considered representative of all kindergarten- and first grade children. There was some bias in the selection of participants to the extent that their teachers selected children they believed were most likely to enjoy testing. The behavior of this select group of children is not likely to reflect the full spectrum of behaviors found in all kindergarten classrooms. Students 5-7 years old were tested in this study. However, there were insufficient numbers of students above age 5 for stratified analyses, the results for all ages were pooled together in this pilot study. Future studies will stratify or otherwise control for age as it is likely to be an important covariate in assessments of CT abilities.

Most of the participants had some past experience with KIBO, providing only a limited opportunity to determine how children inexperienced with programmable robots performed on TACTIC-KIBO. In the future, it will be important to include children with no past coding experience as well as a larger sample of Early Programmers, Programmers and Fluent Programmers to more fully evaluate TACTIC-KIBO's characteristics as an assessment tool.

Each child's TACTIC-KIBO and IPS assessments were carried out in a single session. Mood, attention and other variables can affect the moment-to-moment performance of young children. Testing on a single day may be influenced by these variables and may not capture the child's peak abilities. To help address this limitation, teachers were shown video recordings of the IPS sessions and were asked to evaluate whether the child's performance during testing was consistent with their usual classroom behavior. In all cases, the teachers stated that the child's behavior during testing was consistent with their expectations. In the future, repeat testing on more than one day (perhaps using alternate forms of TACTIC-KIBO) could help to establish the consistency of ratings of CT abilities

Another potential source of error was differences on the testing environment and the time of day the testing took place. The participants were recruited at three different sites: a public school (Winter Hill Community), lab school (Eliot Pearson), and mixed venue (Bioengineering Break Camp). In all three venues, testing took place in an otherwise empty classroom or hallway with minimal traffic and noise. However, there were subtle differences in lighting, noise levels and other aspects of the testing environments that may have presented differing levels of distraction for the participants. Since TACTIC-KIBO is designed to be used in real world classroom setting, variation in the testing environment can be expected. Likewise, testing in schools can be expected to take place at different times of day. It seems unlikely that differences in the testing environments or testing time significantly had a major influence on the outcomes of this pilot study.

All TACTIC-KIBO and IPS sessions were administered by one examiner (ER). This facilitated completion of the study and eliminated potential variance that would have been introduced by having multiple examiners. However, having a single examiner administering both TACTIC-KIBO and the IPS created the potential for bias. To address this, four expert raters were asked to score the IPS of four of the participants based on review of the video recordings. Using a standard measure of inter-rater reliability (Cohen's kappa), the primary rater's assessments were found to have moderate to strong consistency with those of three out of the four raters. This suggests that the administration of both TACTIC-KIBO and IPS by one rater did not bias the primary rater, but it does not entirely rule out bias. Having different persons administer TACTIC-KIBO and IPS would be preferable as it would permit examining the correlation between more independent measures of CT. In this pilot study there were discrepancies in the ratings of individual category scores that were not accounted for when the scores of TACTIC and the IPS were calculated. This pilot study was not designed to reconcile discrepancies or determine their origins. Future studies that take these discrepancies into account may assist in improving the accuracy of the categorical ratings as well as the overall CT assessment.

Scoring

The system chosen for scoring TACTIC-KIBO impacted on the administration process because the protocol called for stopping the testing when the child was unable to provide at least three correct responses on a given level. There are pros and cons to this system of administration. In testing young children, it is important to avoid confronting them with material that is too difficult for them or that is otherwise intimidating. By stopping the testing using the three-item correct cutoff, children were not subjected to more than a few questions above their skill level. The cost of this approach is that TACTIC-KIBO can assess overall level of CT ability but it may not accurately reveal the highest level of performance in every category. In addition, because the examiner has to judge the responses as correct or incorrect on the fly, there is a potential for administration errors. This could result in the testing being stopped prematurely or continue beyond the prescribed cutoff. The choice of three correct responses as a cutoff for going to higher levels seemed to work well in practice. Higher cutoffs (requiring 4, 5, 6 or 7 correct responses) would introduce more stringency and could provide a good mechanism for extending the applicability of TACTIC-KIBO to higher grade levels. Use of alternative cutoffs in other age groups may be a worthwhile direction to explore in future studies of TACTIC-KIBO.

The primary rater scored TACTIC-KIBO during its administration and rated IPS on the basis of a review of a video recording of the session rather than during the session itself. Expert raters participating in the inter-rater reliability sub-study reviewed only the video of the IPS session and were not present during the actual administration of the tests. Some of the difference between the TACTIC-KIBO ratings and IPS rating could be related to subjectivity in interpreting video recordings rather than making live observations. To help standardize results across raters, all raters used an IPS scoring sheet that specified criteria for rating total CT abilities as well as abilities in each category (see Appendix 4). The criteria specified for this purpose were adapted from Vizner's coding development stages but were not themselves validated or generally accepted. The constructs underlying the individual categories are generally accepted but further work will be need to fully validate their applicability to CT assessment.

The lack of a true gold standard for assessment of CT in young children is a very important limitation of this study. This pilot study is built on the premise that someone who has experience and expertise in teaching robotic programming to young children can be expected to accurately assess CT abilities through observation of a child's coding performance. The IPS sessions in this pilot were a practical substitute for actual interviews by multiple raters. Some raters noted that the IPS did not provide them with enough information to rate every category with assurance in every case. It is likely that individual experts use slightly different methods of assessment when afforded the opportunity to interact with the children directly. Therefore, IPS is probably not an optimal means for all experts to rate CT abilities. It was interesting to observe that there were some significant variations in IPS scores of the various children across different raters. One rater in particular had a Cohen's kappa of zero with the other raters, indicating the absence of inter-rater reliability. This could be attributable to a variety of factors including that one particular rater was distracted while carrying out the ratings, their lack of familiarity with the IPS rating categories, less experience with the KIBO platform than the other raters, or perhaps lack of experience with young children. This variability in expert scoring suggests that teachers with less experience and multiple other competing responsibilities may not be readily able to assess CT abilities by observation alone. A highly structured assessment tool such as TACTIC-KIBO that follows a prescribed protocol and uses a simple rating system should allow teachers and researchers to make more accurate and consistent assessments regardless of their level of expertise.

Future Directions

In this study, TACTIC-KIBO proved suitable for administration to kindergarten and first grade students and showed promise as a means of assessing overall CT abilities. To further develop this instrument, some minor modifications are warranted. The level 1 Algorithm and Modularity probe did not perform according to expectations and should be altered in future versions. One possible modification would be to ask what happens when the "BEGIN" block is taken away instead of the current inquiry about the "END" block. The removal of "BEGIN" may be more easily understood by kindergarten children as a cause for the program to fail than the removal of the "END." Empiric testing will need to be carried out to confirm that this modification results in the expected response most of the time.

The Design Process probes did not differentiate levels of ability in that category for the majority of participants. This could reflect a problem with the tasks themselves or it may be indicative of a developmental ceiling effect. While lowering the level of difficult of the Design Process questions could be a solution, "dumbing down" TACTIC-KIBO in this way could make it less suitable for assessing CT in higher elementary school grades. An alternative approach is to allow some level of assistance with the Design Process tasks as a prescribed part of the administration to kindergarten and first grade students. For higher grades, no assistance would be allowed.

In designing TACTIC-KIBO, a set of acceptable and unacceptable responses was created based on the anticipated performance of young children. The set of responses obtained in this pilot study and in future studies will allow further refinement to the response list so that there are fewer ambiguities in scoring. Improving the list of acceptable and unacceptable responses may make scoring more straightforward and help reduce the number of occasions in which responses are judged to be equivocal.

TACTIC-KIBO requires further testing and validation before it can be recommended for use in schools and research studies. This should include testing a broader range of ages (4-7), experienced and inexperienced robot programmers, all genders and children of different socioeconomic backgrounds and nationalities.

In the next phase of study, testing should include administration by teachers. This will require the creation of a training program for teachers and a certification process to assure that they meet acceptable standards and have sufficient inter-rater reliability. A training curriculum along with annotated videos of TACTIC-KIBO being administered can be used for these purposes. It may also be useful to develop a software application to help in the administration and scoring of TACTIC-KIBO. This will allow easier administration by employing automated prompts, feedback and permitting remote data collection.

The current study was cross-sectional and was focused on the utility of assessing CT in young children at a single point in time. However, there are several reasons that future studies should examine the performance of TACTIC-KIBO when administered serially over time. Repeated administrations can help to establish the reproducibility of the CT assessment. If repeated administration results in stable scores, TACTIC-KIBO can be used for longitudinal assessment in its current form. However, if repeated administration is associated with a learning effect that causes scores to drift between testing session, it may be helpful to create alternative forms with equivalent questions.

Once the stability of TACTIC-KIBO's rating over repeated administration have been established, it can be used as a tool for a variety of purposes. In Figure 10, we have labeled the categories of computational learning that an established KIBO robot curriculum touches upon across lessons. By administering TACTIC-KIBO before and after sets of lessons, it may be possible to document computational learning in groups of students and in individuals. Variations on TACTIC-KIBO with more than one type of question per CT category or with questions that more directly relate to the curriculum being taught may also be useful for this purpose. In each case, validation of the approach using a similar study design comparing the TACTIC-KIBO to expert ratings can help achieve validation. Another potential benefit of using TACTIC-KIBO and curricula is that it can help guide an educator to a specific lesson. For example, if most of the children they are teaching are level 2 (early programmers) then by looking at the figure 11 they can see that lesson 3 or 4 is suitable to their classroom. Future studies should create similar recommendations within each level of all KIBO curricula.

In future versions of TACTIC-KIBO, one alternative approach might allow for some adult assistance. This could build off of Vygotsky's ZPD by using a Dynamic Assessment (DA). In DA, it is acknowledged that a person's performance is capable of changing during the assessment because the examiner teaches or helps the examinee when needed (Poehner, & Lantolf , 2005). Perhaps some provision for adult assistance would be helpful in future versions of TACTIC-KIBO to better assess ZPD. However, in a classroom setting, it may be preferable to use the approach taken in the current study of disallowing assistance other than that clearly specified in the TACTIC-KIBO instructions (e.g. assistance with scanning, mechanical problems, etc.) and only rate responses made independently by the child as correct. This could be difficult for some examiners and frustrating for the child who asks for help or is seen struggling with a task. However, since judging level of assistance can be a highly subjective process, it may be better to mandate that TACTIC-KIBO ratings reflect the child's independent performance to help reduce variability in the ratings.

Putting the TACTIC-KIBO scoring system on phone or tablet with automatic feedback could make administering the assessment less burdensome and more accurate. This could give more functionality by showing specific questions only after a child has responded correctly. This could also make scoring more accurate in that the computer would decide how many points a child receives and whether to go to the next level or not. Less training of teachers would be needed if scoring was automated and they could obtain important data such as the child's score compared to other children that age, which parts of the test the child did best in, and what lessons are recommended in future lessons. From a research perspective, having an automated system for TACTIC-KIBO could allow for data collection among a large sample of students which could assist in improving the assessment, recommendations for teaching, and the robot itself.

Eventually, I would like to investigate the possibility of combining the information obtained using this assessment instrument with automated collection of data on block choice and sequencing obtained using Scratch Jr or KIBO interface. In order to collect data like this from KIBO, some modifications to the robot would have to be made. This could have great value to educators and researchers as it could allow for one teacher to administer TACTIC-KIBO to an entire classroom at once.

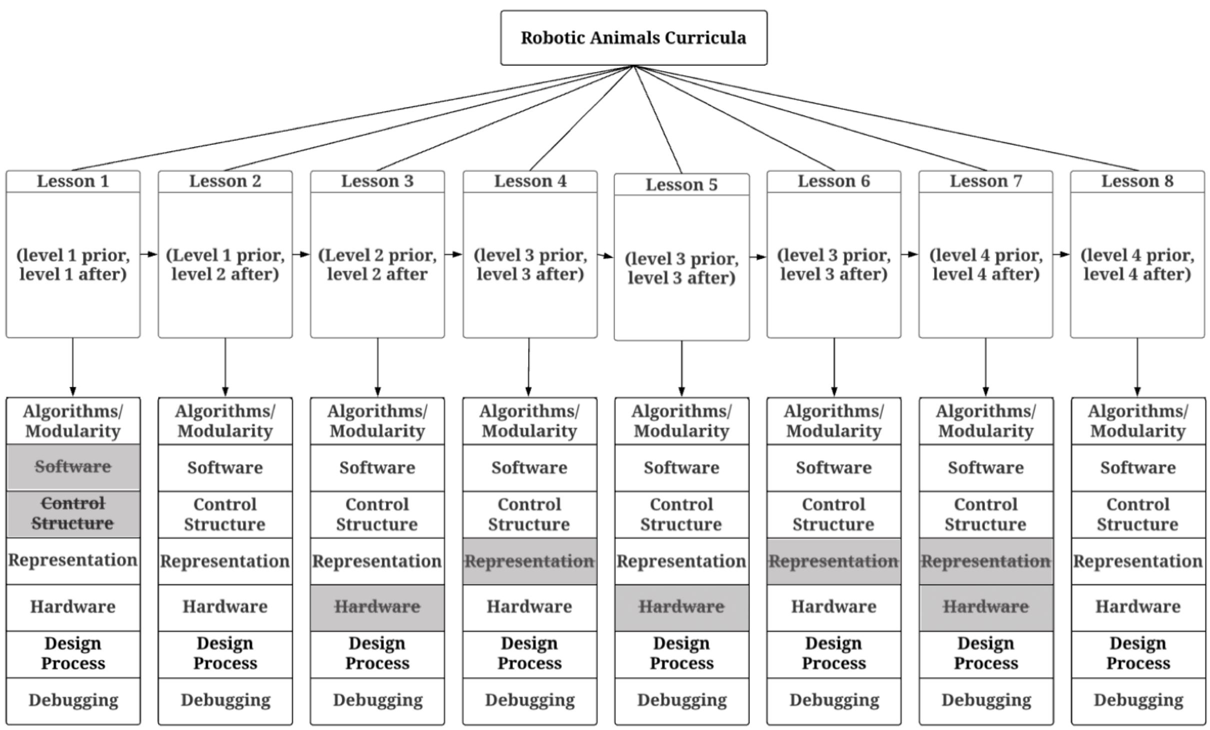
TACTIC-KIBO was designed around the KIBO robotic platform. However, there are many other robotic toys existing and under development for young children as well as adolescents. The question of the generalizability of the methods used to assess CT in this study may be best addressed by developing alternative forms of TACTIC that are applicable to other platforms. This might include the DashDot robot, LEGO WeDo, and programmable screen-based applications such as ScratchJr and Scratch. Creating alternative forms for other platforms would help to better disseminate the assessment and will also be useful in developing intuitions about lines of questioning that are more or less universal across platforms.

The TACTIC-KIBO assessment was designed specifically for the KIBO robotics kit. Therefore, if it is being applied to other robotics kits or platforms coding platforms it must be adapted. For example, ScratchJr is a screen-based app. Therefore, asking questions about hardware could potentially be less obvious to a child. Additionally, ScratchJr does not use parameter cards. Instead, children are able to write in the amount of times they want something to repeat. ScratchJr allows children to create multiple programs and have multiple different characters at the same time. This would allow for different types of questions and tasks regarding algorithms, modularity, and control structures. It also has features such as copying and pasting. This is a feature that KIBO does not have and questions of algorithms/modularity, control structures, would therefore likely differ with the ScratchJr platform.

The current study recruited young children with largely neurotypical development. It is important to understand how typical development of CT abilities occurs through the application of assessments such as TACTIC-KIBO. Once sufficient normative data is obtained, it would be interesting to extend the application to the developmentally diverse population, including children with autism spectrum disorder (ASD). A recent study provided evidence that even children with severe ASD are engaged by the KIBO robot and that they may become more communicative with their teachers as a consequence of playing with and programming the robot. In light of this, it would be interesting to study the acquisition of CT abilities in such children and compare their development with that of neurotypical children.

Figure 11

Sample Computational Learning curriculum with evaluable learning outcome suitable for assessment with TACTIC-KIBO



This graph shows how the Powerful Ideas are implemented in the "Robotic Animal Curricula" which is open source and free online. Each shaded box indicates that the powerful idea was not explicitly taught in that lesson.

References

Barrouillet, P., & Lecas, J. F. (1999). Mental models in conditional reasoning and working memory. *Thinking & Reasoning, 5*(4), 289–302)

Bers, M. U. (2012). *Designing Digital Experiences for Positive Youth Development: From Playpen to Playground (First Edition).* Oxford, New York: *Oxford University Press*.

Bers, M. U. (2018). Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom. *Routledg*e.

Braitenberg, V. (1984). *Vehicles: Experiments in synthetic psychology*. Cambridge, Massachusetts: *The MIT Press*.

Brennan, K., & Resnick, M. (2012). Using artifact-based interviews to study the development of computational thinking in interactive media design. Paper presented at annual American Educational Research Association meeting, *Vancouver, BC, Canada.*

Centers for Disease Control and Prevention (2012). *National Survey of Children's Health*

DeLuca, C., & Bellara, A. (2013). The Current State of Assessment Education: Aligning Policy, Standards, and Teacher Education Curriculum. *Journal of Teacher Education*, *64*(4), 356–372.

Denning, P., J. (2009) Beyond Computational Thinking, Communications of the ACM Vol 52, Issue 6, pp 28 – 30

diSessa, A., A., (2000). Changing minds: computers, learning, and literacy, Cambridge, MA: MIT Press.

Dewey, J. (1938). Experience and Education. New York: Collier Books.

Elkin, M., Sullivan, A., & Bers, M.U. (2016). Programming with the KIBO Robotics Kit in Preschool Classrooms. Computers in the Schools, 33:3, 169-186.

Foucault, M., Martin, L. H., Gutman, H., & Hutton, P. H. (1988). *Technologies of the self: A seminar with Michel Foucalt.* Amherst , MA: University of Massachusetts Press.

Google & Gallup. (2015). Searching for computer science: Access and barriers in U.S. K–12 education.

Google for Education. (2010). *Exploring Computational Thinking.* Retrieved from *https://edu.google.com/resources/programs/exploring-computational-thinking/index.html#!home*

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

Guzdial, M. (2008). *Paving the way for computational thinking. communications of the AMC*, 51(8), 25–27.

Hemmendinger, D. (2010). A please for modesty. *ACM Inroad*s, 1(2), 4–7.

International Society for Technology in Education (ISTE) and the Computer Science Teachers Association (CSTA), 2011. *Operational Definition of Computational Thinking for K–12 Education*

Janveau-Brennan, G., & Markovits, H. (1999). *The development of reasoning with causal conditionals. Developmental Psychology,* 35(4), 904-911.

K-12 Computer Science Framework Steering Committee. (2016). K–12 computer science framework.

Koretz, D., McCaffrey, D. F., Klein, S. P., Bell, R. M., & Stecher, B. M. (1992). The Reliability of Scores from the 1992 Vermont Portfolio Assessment Program.

Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., . . . Werner, L. (2011). Computational thinking for youth in practice. ACM Inroads, 2, 32–37

Leidl, K., Bers, M.U., Mihm, C. (2017). *Programming with ScratchJr: a review of the first year of user analytics.* In the proceedings of the International Conference on Computational Thinking Education, 2017. Wanchai, Hong Kong.

Lockwood, J., & Mooney, A. (2017). Computational Thinking in Education: Where does it Fit? A systematic literary review.

National Research Council 2010. Report of a Workshop on the Scope and Nature of Computational Thinking. The National Academies Press.

Matthew Poehner, J. L. (2005). Dynamic assessment in the language classroom.

Ming, Ming, & Bumbacher, (2014) Aligning Learning with Life Outcomes through Naturalistic Assessment. *SOSOS.*

Mioduser, D., & Levy, S. T. (2010). Making Sense by Building Sense: Kindergarten Children's Construction and Understanding of Adaptive Robot Behaviors. *International Journal of Computers for Mathematical Learning*, *15*(2), 99–127.

Mioduser, D., Levy, S. T., & Talis, V. (2009). Episodes to Scripts to Rules: Concrete-Abstractions in Kindergarten Children's Explanations of a Robot's Behavior. *International Journal of Technology and Design Education*, *19*(1), 15–36.

Moyer, K., Gilmer B.V.,H. (1953). The Concept of Attention Spans in Children. *Elementary School Journal, 54*(1), 464.

Papert, S. (1980). *Mindstorms: Children, Computers, And Powerful Ideas*. New York: Basic Books.

Papert S. (1993). *The children's machine: Rethinking schools in the age of the computer*. New York: Basic Books.

Papert, S. (1999). Papert on Piaget. *Time Magazine, special issue on "The Century's Greatest Minds*".

Papert S. (1996), *An Exploration in the Space of Mathematics Educations, International Journal of Computers for Mathematical Learning*, Vol. 1, No. 1, pp. 95-123.

Papert, S., & Harel, I. (1991). Situating Constructionism. In Constructionism. *Ablex Publishing Corp.*

Perlis. A. J. (1963). The computer in the university. In M. Greenberger, Ed., *Computers and the World of the Future,* MIT Press, Cambridge, MA, 180–219.

Piaget, J. (1959). The language and thought of the child (3d ed.). New York: Humanities Press.

Piaget, J. (1968). *Six Psychological Studies*. Anita Tenzer (Trans.), New York: Vintage Books.

Poehner, M., & Lantolf J. (2005). *Language Teaching Research, 9*(3), 233-265.

Poulin-Dubois, D., McGilly, C. A., & Shultz, T. R. (1989). Psychology of Computer Use: X. Effect of Learning Logo on Children's Problem-Solving Skills. *Psychological Reports*, *64*(3), 1327–1337.

Prakken, L. W. (1942). The Education Digest Vol 8 Page 49

Resnick, M., (2007). All I really need to know (about creative thinking) I learned (by studying how children learn) in Kindergarten in *Proceedings of the 6th Conference on Creativity & Cognition* (CC '07), pp. 1–6, ACM.

Reynolds R. C., & Kamphaus W. R. (2015). RIAS-2 Professional Manual. *PAR.*

Rich K., Pokimica J., Wherfel Q., Strickland C., & Moran C. (2017). *Building Mathematics + Computational Thinking Trajectories From Existing Literature.* American Educational Research Association.

Ruff, H. A., & Lawson, K. R. (1990). Development of sustained, focused attention in young children during free play. *Developmental Psychology*, *26*(1), 85-93.

Ryan, C. (2013). *Language use in the United States: 2011*. Washington, DC: U.S. Census Bureau

Sattler, J. M. (2014). Foundations of Behavioral. Social, and Clinical Assessment of Children. *San Diego: J.M. Sattler.*

Sattler, J. M. (2016) Assessment of Children WISC-V and WPPSI-VI. *San Diego: J.M. Sattler.*

Selby, C., Dorling, M., & Woollard, J. (2014). Evidence of assessing computational thinking [Monograph]

Shaffer, D.W., and Resnick, M. (1999). "Thick" Authenticity: New Media and Authentic Learning. *Journal of Interactive Learning Research*, vol. 10, no. 2, pp. 195-215.

Siegmund, J., Kästner, C., Apel, S., Parnin, C., Bethmann, A., Leich, T., … Brechmann, A. (2014). Understanding Source Code with Functional Magnetic Resonance Imaging. In *Proceedings of the 36th International Conference on Software Engineering* (pp. 378–389). New York, NY, USA: ACM.

Slotnick, S. (2017). In memory: Seymour Papert. *Massachusetts Institute of Technology.*

Strawhacker, A., Sullivan, A., & Bers, M.U. (2013). *TUI, GUI, HUI: Is a bimodal interface truly worth the sum of its parts?* In Proceedings of the 12th International Conference on Interaction Design and Children (IDC '13). ACM, New York, NY, USA, 309-312.

Sullivan, A., & Bers, M. U. (2016). Girls, boys, and bots: Gender differences in young children's performance on robotics and programming tasks. Journal of Information Technology Education: Innovations in Practice, 15, 145- 165. Retrieved from

Sullivan, A., Bers, M.U., Mihm, C. (2017). Imagining, Playing, & Coding with KIBO: Using KIBO Robotics to Foster Computational Thinking in Young Children. In the proceedings of the International Conference on Computational Thinking Education, 2017. Wanchai, Hong Kong.

Sullivan, A., Elkin, M., & Bers, M. U. (2015). KIBO robot demo: engaging young children in programming and engineering. *ACM Press.*

The Mathematics Teacher (1943). *National Council of Teachers of Mathematics* Vol 36

Vee, A. (2013). Understanding computer programming as a literacy. *Literacy in Composition Studies, 1*(2).

Vizner M. Z., Bers, M. U., Scarlett G., Gravel B., (2017). *Big Robots for Little Kids: Investigating the Role of Sale in Early Childhood Robotics Kits* (Master's thesis). Available from ProQuest Dissertations and Theses database. (UMI No. 10622097)

Vygotsky L. (1978). Mind in society: *The development of higher psychological processes* Harvard University Press, Cambridge, MA.

Vygotsky, L. S. (1987). Thinking and speech (N. Minick, Trans.). In R. W. Rieber & A. S. Carton (Eds.), *The collected works of L. S. Vygotsky: Vol. 1. Problems of general psychology* (pp. 39-285). New York: Plenum Press. (Original work published 1934)

Vygotsky, L. S. (1998). Infancy (M. Hall, Trans.). In R. W. Rieber (Ed.), *The collected works of L. S. Vygotsky: Vol. 5. Child psychology* (pp. 207-241). New York: Plenum Press. (Original work written 1933-1934)

Wang, D., Wang, T., & Liu, Z. (2014). A Tangible Programming Tool for Children to Cultivate Computational Thinking [Research article]. https://doi.org/10.1155/2014/428080

Werner, L., Denner, J., & Campe, S. (2015). Children Programming Games: A Strategy for Measuring Computational Learning. *ACM Transactions on Computing Education (TOCE)*, 14(4), 1-22.

Werner L., Denner J., Campe S. (2014). Using computer game programming to teach computational thinking skills. In: Schrier K., editor. Learning, education and games: Volume 1, curricular and design considerations. Pittsburgh, PA: ETC Press; pp. 37–53.

Wilson, C., Sudol, L. A., Stephenson, C., & Stehlik, M. (2010). *Running on empty: The Failure to Teach K-12 Computer Science in the Digital Age* New York, NY: *The Association for Computing Machinery and the Computer Science Teachers Association.*

Wing, J., (2006). Computational Thinking. CACM vol. 49, no, 3. March 2006, pp. 33-36.

Wing, J.M. (2008). Computational thinking and thinking about computing. Philosophical transactions of the royal society of London A: mathematical, physical and engineering sciences, 366(1881), pp.3717-3725.

Appendix 1

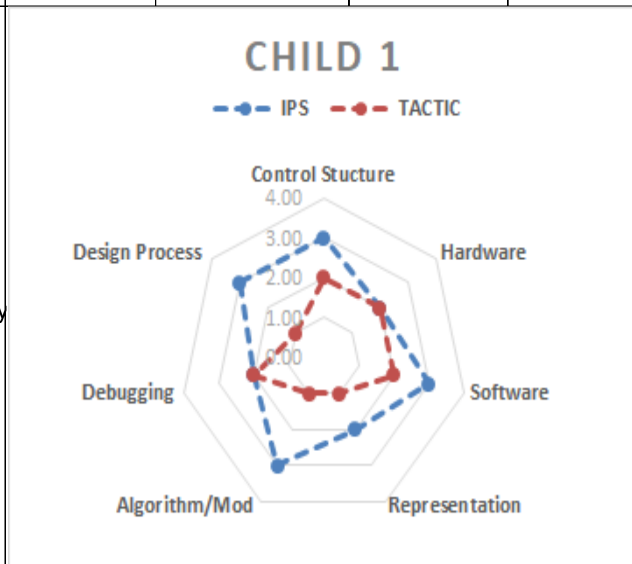
Complete Set of Individual Participant Descriptions (n=15)

Child 1

Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
5	Female	10m29s	17m11s	Yes	No	2	2

Examiner's Comments: Initially shy. Recognizes KIBO and calls it by name. Soft spoken. Can scan the blocks and follows instructions well. During IPS, child chooses to make KIBO "dance" but does not use motion blocks. Often demonstrates by carrying out action rather than expressing herself with words. Does not understand the principle of repeat blocks. Sometimes does not answer my questions.

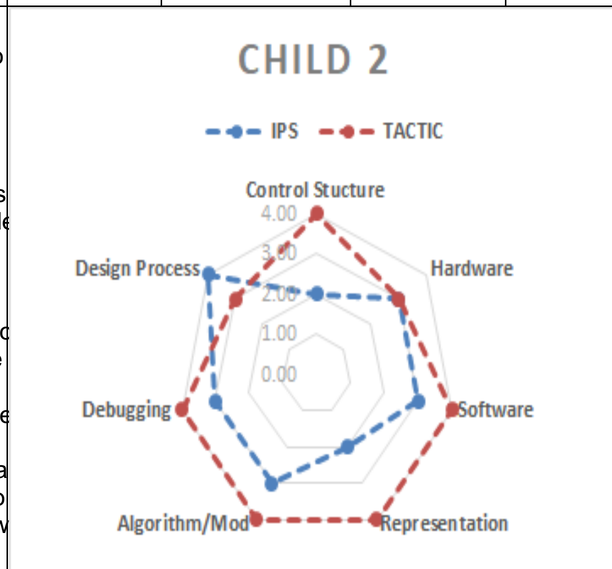
Teacher's Comments: (LM) "She actually likes building, she is a quiet worker so she does rather than tell" "She also may want it perfect sometimes… She doesn't like to put herself" (Lynne May Lim, Personal Communication 2018)



Child 2							
Age	**Gender**	**TACTIC-KIBO DURATION**	**IPS Duratior**	**PRIOR KIBO**	**PRIOR SCRATCH.JR**	**TACTIC-KIBO SCORE**	**IPS SCORE**
5	**Female**	**25m33s**	**17:56**	**yes**	**yes**	**4**	**3**

Examiner's Comments: Child was about to go outside for recess but is happy to come do testing instead. Child recognizes KIBO. At first says she doesn't know what repeats and "if" blocks are. During IPS says that they make KIBO turn. After watching me use repeat blocks and if blocks in TACTIC-KIBO-KIBO child is able to use them correctly for the most part. Child thinks that the ear sensor makes sound. Child asks what the parameters are. We took a few breaks. Child says she wants to make KIBO into a star. Child sticks with this idea throughout the IPS, Child comes up with creative ways to debug that aren't always right. For example, she mis-scans the sing block and then figures that KIBO did not sing because it did not have an ea sensor to hear it. During IPS, child attempted to draw a star unsuccessfully. Examiner then drew star and assisted child in making the robot's movement resemble a star.

Teachers Comments: No comments

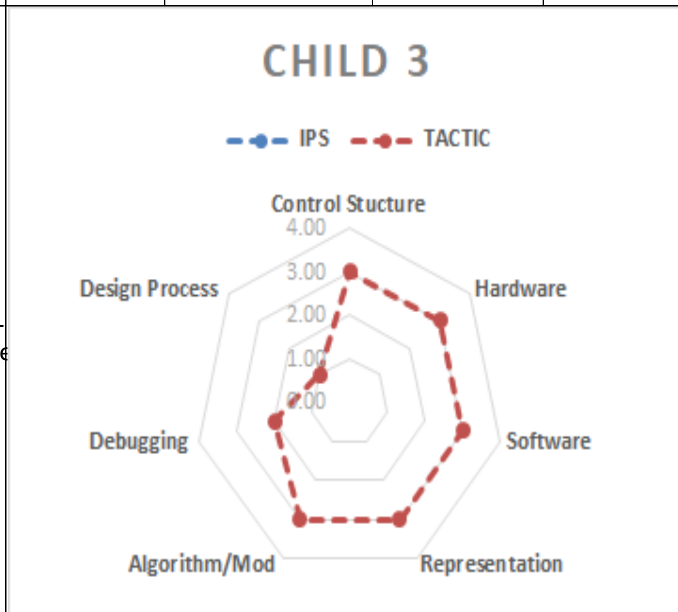


Child 3

Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
5	Male	n/a	n/a	n/a	n/a	n/a	n/a

Examiner: Child was playing outside and didn't want to come to testing room. His teacher told him to try it out and said he could leave if he didn't like the testing. Child assented to participate and listened as I explained what was going to happen. At level 4 child says "I don't know" in response to every question asked. Child says he's hungry and requires a snack, water, and multiple breaks. He wants to build his own program and seems frustrated that I keep returning to TACTIC-KIBO questions and tasks. Child left before IPS could be administered.

Teacher Comments: "He was upset that he missed recess and felt as though he was being asked too many questions. He wanted to build his own project. He has great imagination. Perhaps he was just frustrated by the process. "(On the video) "he seems fatigued at 6 minutes into testing. This is because he is not actively engaged. This format does not work for him. He is very smart, a builder, an inventor. He sometimes will lead or initiate ideas for the group. But he is a very hands on person. If you split him into two sessions … you would get more out of him then if you do everything at the same time (Lynne May Lim, Personal Communication 2018).

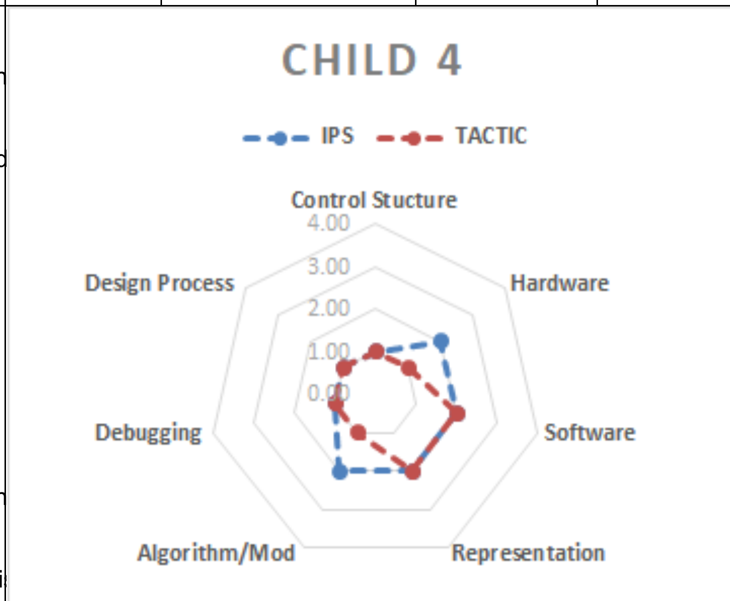


Child 4

Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
5	Male	12m30s	23m26s	no	yes	2	2

Examiner's Comments: Child is extremely excited and has a very positive attitude towards the testing. In fact, when he saw me take in another child for testing he begged to along come because he "loves robots" .Child does not know if he should start with begin or end. A lot of assistance required to complete the IPS. Child is at a low level of proficiency and requires help scanning. He is much more focused on the robots actions and material things than what he is being told.

Teacher Comments: "His eye hand coordination is challenged, like when he is counting, his words do not match his fingers. He is much more focused on the material rather than what you are doing. That is his experiment. He i focusing on the concrete materials."

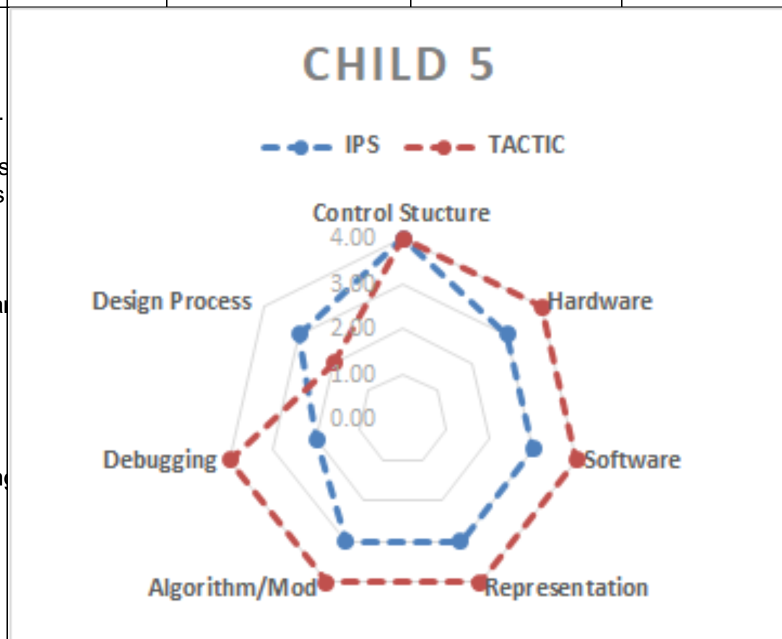


Child 5

Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
5	Male	25m10s	42m26s	yes	yes	4	3

Examiner's Comments: Child sometimes forgets how to turn on robot. Child can scan robot without much help. When asked for the difference between blocks he says that the only difference is the color. Child knows what most blocks do. Child doesn't know the difference between the telescope and eye. Child thinks that the forever block is an 8. Child thinks green circuit board is a solar panel. Child does not understand if blocks.

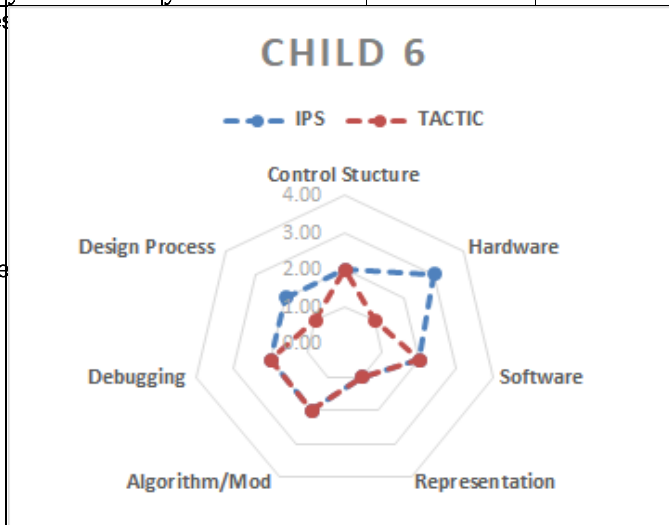
Teacher's comments: He skips over things because his mind is already ahead. He always thinks of the next thing [in reference to his forgetting simple qualities of robot but being able to do more complex tasks]



Child 6							
Age	**Gender**	**TACTIC-KIBO DURATION**	**IPS Duration**	**PRIOR KIBO**	**PRIOR SCRATCH.JR**	**TACTIC-KIBO SCORE**	**IPS SCORE**
5	Female	9m21s	16m06s	yes	yes	2	2

Examiner's Comments: When child comes into room she immediately starts to play with sensors saying that she is "putting on everything she can" Seems excited about the robot. She is able to debug simple programs. She did not know exactly how KIBO works in that she often forgets begin and end and sometimes forgets which button to press in order for it to go. During IPS, she says her idea is to put every single block together to make the "biggest program"

Teacher Comments: Child is very advanced for her age. She is young five years old and is often able to understand concepts such as reading, writing, art that her peers do not yet understand. Child is bilingual.

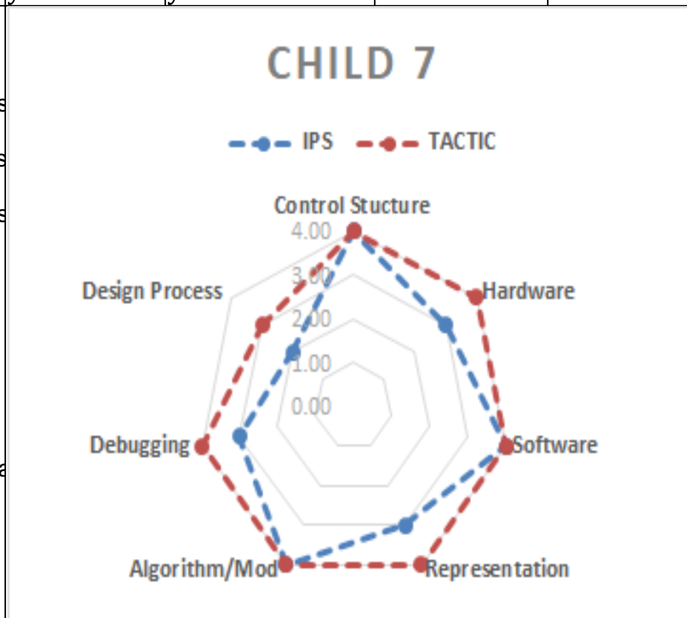


Child 7

Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
5	Male	23m44s	18m12s	yes	yes	4	3

Examiner's Comments: Child is very excited by the robot and seems to have a high level of understanding about how it works. He repeatedly asked if his answers are correct. Given positive reinforcement. Both of this child's parents are software engineers. He often asks "more questions please." Child sometimes confuses "if" and "repeat" parameters but he has a general understanding that after them you need to put an "end if" or "end repeat."

Teacher comments- He is very gregarious and chatty. He talks out loud while his is thinking and is good at figuring out problems. He seems to have a high level of why you program and what programming means in the world around us. He is able to infer what KIBO can do without being taught.

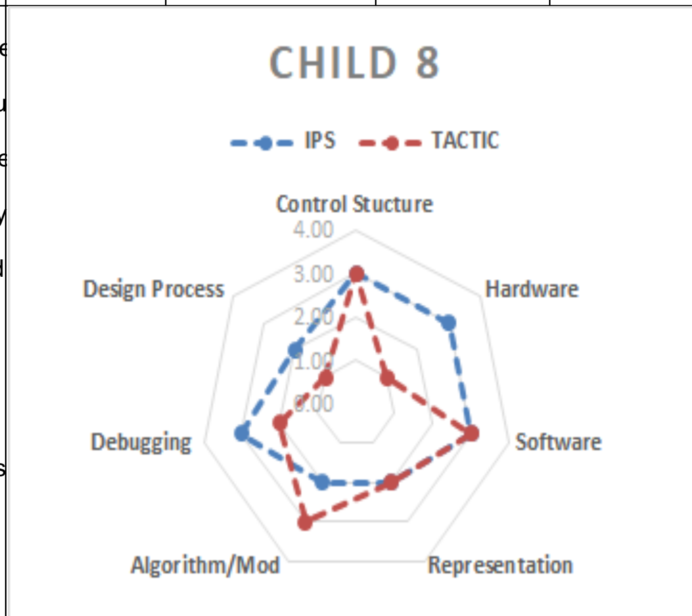


Child 8

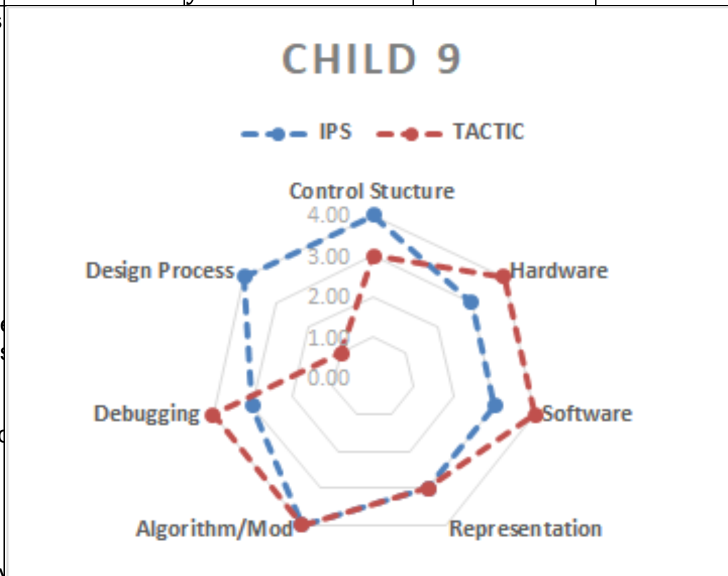
Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
5	Male	13m40s	8m26s	no	no	3	3

Examiner's Comments: Child comes up with unique and creative ways to solve the problem that aren't always right. Child does not know how to use end repeats but has no trouble finding repeat and correct number parameter cards. Child is not sure how to use If blocks and confuses if and repeat parameters. Child only wants to try out one program in IPS(which does not work because it has an improperly placed "if" in it) child debugs this by taking out if block altogether.

Teacher comments: Teacher says that he is really tactile with tools he is exploring. He often picks an idea and doesn't know how test it so he just forgets about the idea. He is creative.



<table>
<tr><th colspan="8">Child 9</th></tr>
<tr><th>Age</th><th>Gender</th><th>TACTIC-KIBO DURATION</th><th>IPS Duration</th><th>PRIOR KIBO</th><th>PRIOR SCRATCH.JR</th><th>TACTIC-KIBO SCORE</th><th>IPS SCORE</th></tr>
<tr><td>5</td><td>Male</td><td>22m</td><td>24m45s</td><td>no</td><td>yes</td><td>4</td><td>4</td></tr>
<tr><td colspan="4">Examiner comments: Child performs modularity by picking up groups of blocks at once and placing them into his program. At first, child does not understand that you need an end repeat and an end if after implementing repeat/if blocks. Child then confuses end if and end repeat. However, after watching me correct the program in TACTIC, Child learns and applies this knowledge to IPS. Child knows all blocks asked except h thinks that the ear is for smelling and i not sure what the wait for clap block does. Child comes up with a very complex program for IPS with a neste loop.

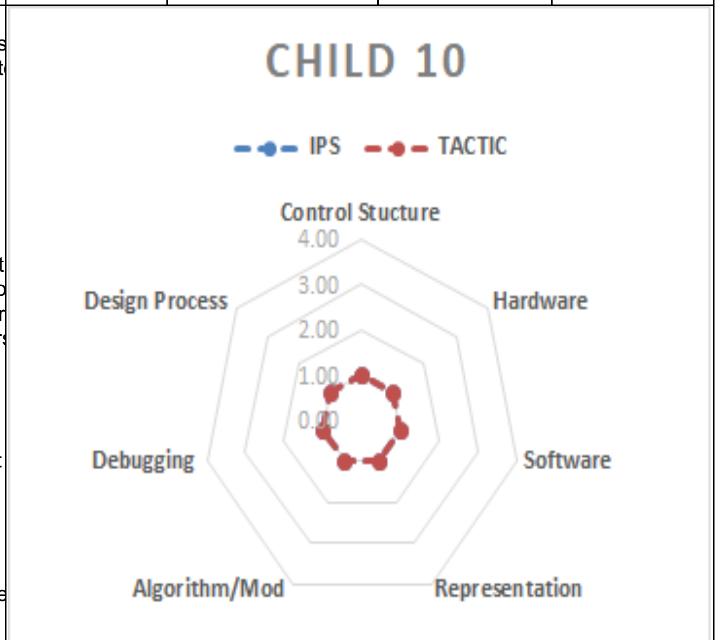
Teacher Comments: Says that the child is quiet observes and really understands concepts. He is extremel intelligent and was often explaining concepts to other students in the camp. Might speak a different language at home.</td><td colspan="4">
</td></tr>
</table>

Child 10

Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
5	Female	5m11s	18m49s	no	no	1	1

Examiners Comments: Child does not know how to use the robot. Child always approaches KIBO backwards, starting with end and going t begin. Child thinks robot worked because we only scanned "two blocks" . Knows that green button turns robot on. Thinks that the bad sound is the good sound. Thinks light bulb works automatically. Says "the batteries are trying to see something from the KIBO, they are trying to see out of the little glass pieces" points to the two bumps near the batteries that the screws come out of. Has the motor skills to scan but does not know which block to scan or what to scan. Tries to debug by putting sensor in all of the robots ports. Does not know what any of the sensors do. Child is dragging robot around. Child cannot be pushed to Zone of proximal dev. Child scans all of parameter cards at once in a line. "the light Is sticking out because of wires". Child begins scanning her shoe.

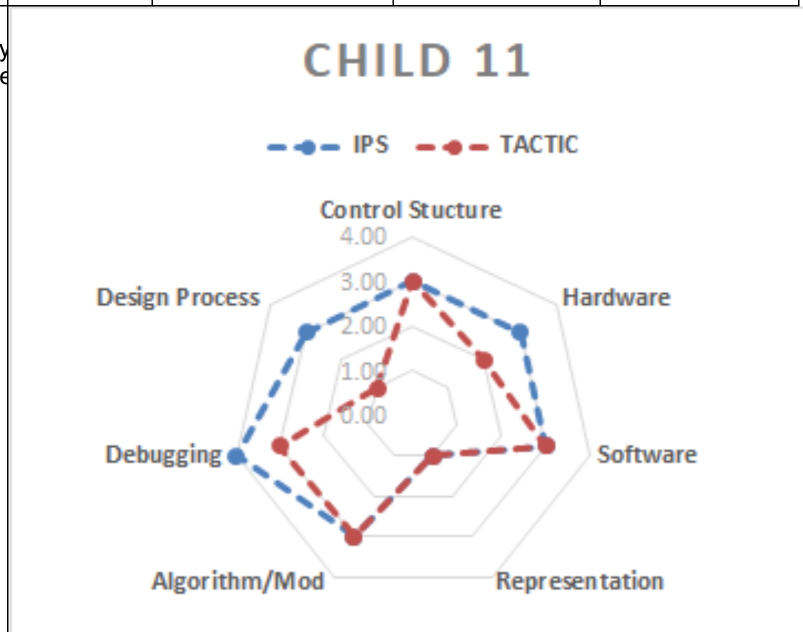
Teacher Comments Says that child is on the Autism Spectrum (I did not know this prior) she says that her mom said that she sometimes has outbursts, recedes, and has trouble following instructions but that she did not observe this happening in the camp. However, teachers says that they could tell with their research testing that she was very regimented and often repeated activities. She has a high reading capacity for someone her age. Teache brings up that the child scans from end to begin. "That's something that I have seen before, when they are just developing KIBO awareness... I thought it was really interesting because she had such high reading ability so you would think she would scan from left to right… I think some children just work backwards"(Strawhacker, Personal Conversation 2018).



Child 11							
Age	**Gender**	**TACTIC-KIBO DURATION**	**IPS Duration**	**PRIOR KIBO**	**PRIOR SCRATCH.JR**	**TACTIC-KIBO SCORE**	**IPS SCORE**
5	Female	13m	17m30s	yes	yes	3	3

Examiner's Comments: Child has some trouble scanning by herself. Child originally thinks that both sound/recording blocks are the same even though they have different pictures on them. She has trouble figuring out the record/playback sensor but during IPS can figure it out with adult help. Child can differentiate between most blocks/sensors. Calls if block is mystery. Child puts end repeat after repeat.

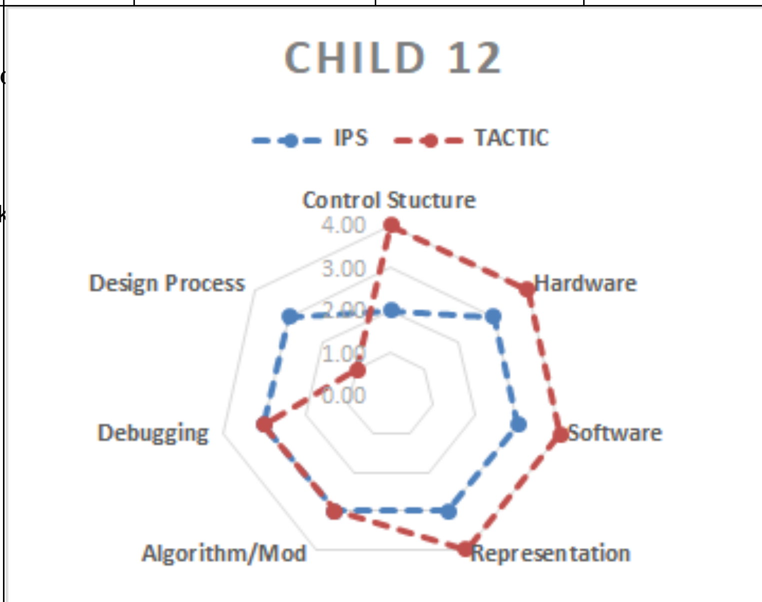
Teacher's Comments: No comment



Child 12							
Age	**Gender**	**TACTIC-KIBO DURATION**	**IPS Duration**	**PRIOR KIBO**	**PRIOR SCRATCH.JR**	**TACTIC-KIBO SCORE**	**IPS SCORE**
7	Female	28m20s	12m14s	no	no	4	3

Examiner's Comments: Child is engaged by robot. Child has basic understanding of that blocks control robot and can understand simple blocks and sensors. Child refers to wait for clap block as "making KIBO clap block" child can scan blocks without help and knows how to correctly sequence blocks. Child skips scanning the "sing" block during the IPS. In order to debug the program she moves the sing block to a different area in hopes that it will work if the blocks are in a different order.

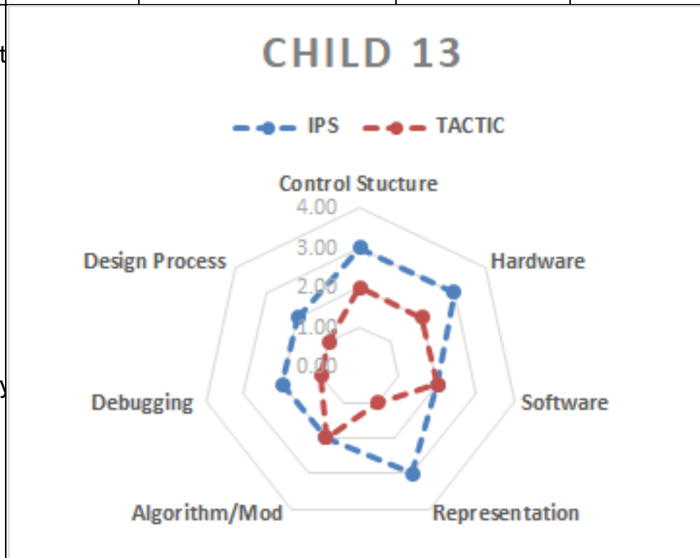
Teacher comments: I am a little bit surprised because I don't know how much reading has to do with this type of thinking when you are playing with KIBO but just generally she's very slow when it comes to reading and I have never seen her put puzzles or anything together but because she has a great interest in KIBO that could have something to do with her being able to sequence and recognize parts… When reading a book, I think you are using a different part of your brain. Since we don't really do anything similar to this in our class she might have some experience at home.



Child 13							
Age	**Gender**	**TACTIC-KIBO DURATION**	**IPS Duration**	**PRIOR KIBO**	**PRIOR SCRATCH.JR**	**TACTIC-KIBO SCORE**	**IPS SCORE**
6	Female	13m14s	20m15s	no	no	2	2

Examiner's comments confident when it comes to programming and naming blocks. Child thinks wait for clap makes robot clap. Child can read/ name most blocks and sensors. At first doesn't not know difference between end repeat and repeat. Difference is "that one is green and that one is red" but with more prompting says that red one stops it and green one starts it. Child confuses if and repeat parameter cards. When asked for what she wants to do she says "I'm making a big one" child connects as many blocks as possible. Child recognizes that KIBO's sound isn't right but doesn't recognize that its not going to perform action after that.

Teacher comments: I consider her very good at naming and sequencing. However, she gets very nervous, even when she is reading and has to decode words. If she's in a group or even in front of me she can't focus well on tasks. She i a ELL (English language learner).

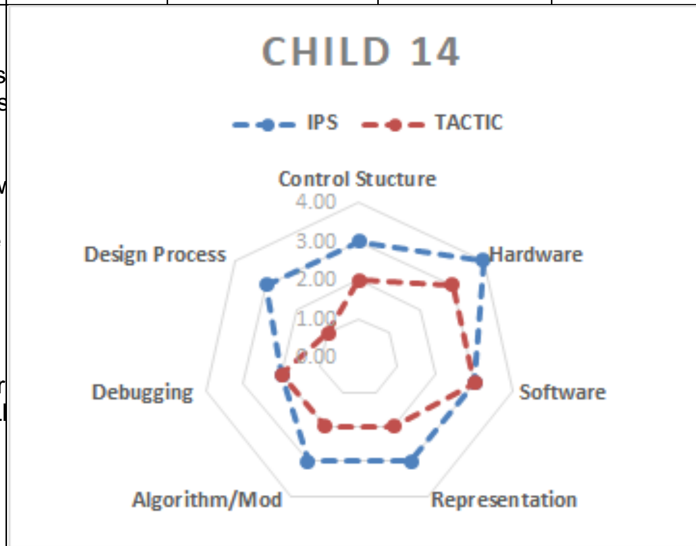


Child 14

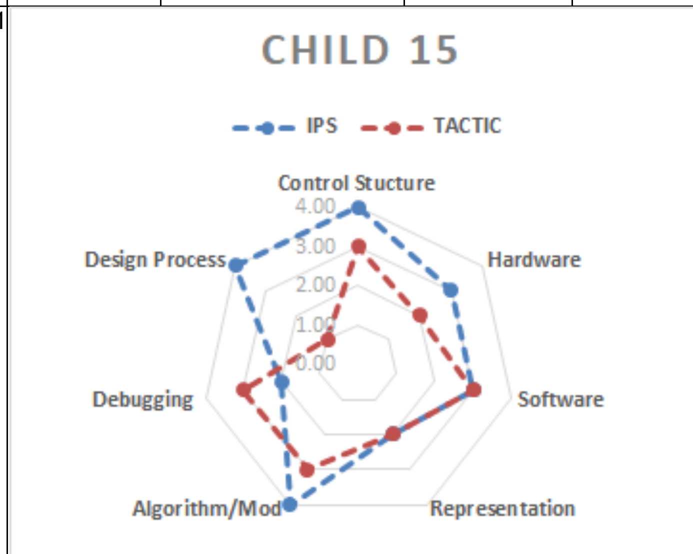
Age	Gender	TACTIC-KIBO DURATION	IPS Duration	PRIOR KIBO	PRIOR SCRATCH.JR	TACTIC-KIBO SCORE	IPS SCORE
7	Male	13m25s	17m26s	no	no	3	3

Examiner's comments: Child knows what most parts are. Calls telescope his binoculars. Guesses correctly that it is his eyes. Thinks that one of the play blocks is a camera. But then corrects himself. Thinks wait for clap makes KIBO clap. Child figures out with some adult help how to use If blocks. Child tries to use forever at first for if block but then decides to use light. Child can do a lot with a little bit of adult help.

Teacher comments: I think the reason that these children are getting the wait for clap block wrong is that they are in an ELL class so they probably see the hands or the word clap and think that is what it does.



<table>
<tr><th colspan="8">Child 15</th></tr>
<tr><th>Age</th><th>Gender</th><th>TACTIC-KIBO DURATION</th><th>IPS Duration</th><th>PRIOR KIBO</th><th>PRIOR SCRATCH.JR</th><th>TACTIC-KIBO SCORE</th><th>IPS SCORE</th></tr>
<tr><td>7</td><td>Female</td><td>11m38s</td><td>18m12s</td><td>no</td><td>no</td><td>3</td><td>3</td></tr>
<tr><td colspan="4">Examiner's Comments: goes on long 11 minute tangent while eating her snack about her family issues (poor, often no electricity, bother dropped out of highschool) refers to KIBO's ear as "hot pepper block" "oval block " is the eye block. Refers to if block as "a forward". Child has clear plan for IPS. She sticks with the plan as she builds her program. She confuses the end if and end repeat blocks

Teachers Comments: In math unless i explicitly model everything and show her how to do it she often gets confused.</td><td colspan="4">
</td></tr>
</table>

Appendix 2: CT Assessment Tool

SUBJECT NUMBER	DATE	TIME
____________	______________	______________

KIBO TACTIC-KIBO LEVEL 1

BEGIN FORWARD END

Number	Category	Question/Task	Satisfactory (+1)	Unsatisfactory (0)
.1	ontrol Structure	**Scan blocks and press triangle on KIBO]** Watch the KIBO robot. Vhat is the robot doing and why is it doing it?	he robot is moving because you programmed it, the robot is moving forward because you told it to, the robot is leaving the cave because its scared, the robot is running away, or equivalent ( )	t's magic, no answer, I don't know, or equivalent (
.2	Iardware	**Flip robot over]** Which one of these is called the batteries?"	hild points to the batteries, Child describes what the batteries are, or equivalent (	don't know, child points to the wrong thing, or equivalent (
.3	oftware	**[Point to 3 blocks]** Which block told KIBO to move forward?"	hat blue block (points to forward block), those blocks (points to all three block) (	Aagic, it wanted to go, you told it, no answer or equivalent (
.4	epresentation	**Point to red block]** Vhat does the red block do?	t makes it stop, end, ends the program, it finishes it, or equivalent ( )	hild describes any function except stop, I don't know or equivalent (
.5	lgorithms/ Aodularity	**Take away end block and show remaining 2]** What would happen if I just used these two blocks?	t won't work, it will beep with KIBO's error sound, or equivalent (	t won't change, its okay, it'll just keep going, I don't know or equivalent (
.6	ebugging	**Reverse order of start and stop blocks]** Iow can I fix this so the KIBO works?	hild places start and stop in correct positions (	hild does not restore correct order, does nothing or equivalent (
.7	esign Process	**Say: I want the KIBO to light up this light bulb & show light].** Vhich block should I use? (give choice of three)	hild points to white light on block, child describes the white light on block, or equivalent (	ny other response or no response (

Level 1 Total Satisfactory: _________
(If Score is 3 or higher, continue to Level 2)

SUBJECT NUMBER ______________ **DATE** _________________ **TIME** _________________

KIBO TACTIC-KIBO LEVEL 2

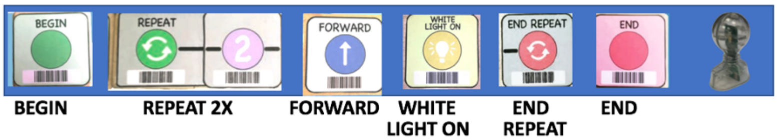
BEGIN FORWARD WHITE LIGHT ON END

Number	Category	Question/Task	Satisfactory (+1)	Unsatisfactory (0)
2.1	Control Structure	**Scan blocks and press triangle on KIBO]** **"Watch the KIBO robot. "** "What is the robot doing now that is different from the first program?"	The robot is now flashing a light, or equivalent (	It's magic, no answer, I don't know, or equivalent (
2.2	Hardware	**Show KIBO and programming blocks** **"**Here is a program, I scanned it last time. Could you scan it this time? If not, could you describe how to scan it? "	Child correctly scans the blocks or describes how to scan (	Child cannot correctly scan or describe how to scan a program, does not try or equivalent. (
2.3	Software	**[Point to 3 blocks]** "Which one of these blocks tells KIBO that you are finished with the program?"	Child identifies end block ou of the three block options.) (	Child is unsure of which block ends the program, child points to the wrong block or equivalent (
2.4	Representatior	Show 1 blue and one yellow block "What is the difference between the blue blocks and the yellow blocks?"	It makes it stop, end, ends th program, it finishes it, or equivalent. (	Child describes any functio except stop, says colors are different, I don't know, equivalent (
2.5	Algorithms/ Modularity	**[Show blocks, switch order to begin, white light, forward, end]** "What will be different about this program if I do this?"	The light will go on before the forward instead of after, the order of the program is different. (	Nothing changed, I don't know, or equivalent (
2.6	Debugging	**[Rescan without white light on block]** "I want the robot to go forward, put its light on, and end."	You can add the white light on block or equivalent. (	You can't fix it, its broken, don't know, or equivalent (
2.7	Design Process	"I want KIBO to repeat going forward and flashing white light twice? Which blocks, modules, and parameters woul I need?"	Child adds repeat block and parameters (	Child doesn't know, child adds in the wrong block (

Level 2 Total Satisfactory: _________
(If Score is 3 or higher, continue to Level 3))

SUBJECT NUMBER	DATE	TIME
________	________________	_______________

KIBO- TACTIC-KIBO LEVEL 3



Number	Category	Question/Task	Satisfactory (+1)	Unsatisfactory (0)
3.1	Control Structure	**Scan blocks and press triangle or KIBO]** “Watch the KIBO robot. What is the robot doing now that is different?”	Robot is going further, The light is flashing twice, The robot is repeating itself, Robot is going further and flashing, the robot is going further from the cave and flashing more, it’s running longer, or equivalent (	It’s doing the same thing, It has more blocks, no answer, or equivalent
3.2	Hardware	**[Flip robot over]** “What does the green board do?”	Child correctly identifies that the green board i the circuit board, brain, controller circuit, computer in the KIBO or the equivalent (	I don't know, child points to the wrong thing, or equivalent (
3.3	Software	**[Point to if blocks and repeat blocks]** Which two of these things do you need to make the robot repeat something?”	Child correctly identifies repeat blocks (	Child is unsure of what the green board does, child has the wrong answer,, or equivalent (
3.4	Represent-ation	**[Show gray block and yellow]** “What is the difference between the gray blocks and the yellow block?”	The difference is that the gray blocks make the robot repeat something and the yellow lights make the KIBO turn a light on, or equivalent (	I don't know, child describes the wrong thing, or equivalent (
3.5	Algorithms/ Modularity	“What would I change if I wanted it to repeat 4 times?”	the parameter, scan the forward block twice, scan the white light on block twice, or equivalent (	Scan the repeat block twice, i don’t know, or equivalent (
3.6	Debugging	**Move the white light on block t before the repeat blocks**. “ I want the white light on to flash twice. How can I do that?”	You have to move the block back to within the repeat two times loop. Scan the white light on twice, get another white light on block, or equivalent (	You can’t fix it, its broken, I don’t know, or equivalent (
3.7	Design Process	“What if I want KIBO to go forward and put the light on only if it is near something else? Which blocks, modules and parameters would I need?”	Child correctly uses if near blocks to build thei program. (	Child doesn't know, child adds in the wrong blocks, modules, or parameters to the program (

Level 3 Total Satisfactory: _________
If Score is 3 or higher, continue to Level 4))

SUBJECT NUMBER:	DATE:	TIME
______________	_________________	_________________

KIBO-TACTIC-KIBO LEVEL 4

Number	Category	Question/Task	Satisfactory (+1)	Unsatisfactory (0)
4.1	Control Structure	**[Scan blocks and press triangle on KIBO]** "Watch the KIBO robot (once near an object, once far away, once again near.) What is the robot doing now that is different?" "Which of these blocks/parameters made the robot change what it did?"	The robot is sensing when it is near an object and then putting its light on and moving, the robot only starts moving and flashing its light when it's near something You added an if block, you added an end if block, you added an if near parameter, or equivalent (	It only works sometimes, it's broken, i don't know, it's doing the same thing, no answer, or equivalent You didn't change anything, you added a light sensor, forward, end, begin, white light on, you made the program longer, I don't know, or equivalent. (
4.2	Hardware	**[Fill all four spaces with sensors/modules.]** " "How does KIBO know if it is near something?"	Child correctly identifies that robot needs the distance sensor to tell if an object is near/far. Child says it uses sensor to tell if it is near (	Child is unsure how to know if the robot is near something. (
4.3	Software	**[Point to 3 blocks]** "Which one of these things do I need to tell KIBO to go only if it is near something?"	Child says it uses the telescope sensor to tell if it is near (	child cannot identify which sensor is the distance sensor. (
4.4	Representation	"What is the difference between the purple (or light blue in some cases) blocks and the yellow blocks?"	The difference is that the light blue blocks make the robot do a conditional yellow lights make the KIBO turn a light on, or equivalent (	I don't know, child describes the wrong thing, or equivalent (
4.5	Algorithms/ Modularity	Move "white light on "block to after begin block " What will the KIBO do differently if move this block to after the begin block?"	it will go forward no matter what, or equivalent (	I don't know, nothing would be different, the white light will change, or equivalent (
4.6	Debugging	**[Take out the if block and just scan the near parameter**] " I want the robot to work the way it did before. How can fix this?"	You have to move the scan the if block first, or equivalent (	You can't fix it, its broken, I don't know, or equivalent (
4.7	Design Process	"What if I want my KIBO to go forward if it's near something or put its light on if it's far?"	You have to nest the if blocks so that there is an if near and an If far block (	*You can't do it, its broken, I don't know, or equivalent* (

Level 4 Total Satisfactory: _________

End of KIBO-TACTIC-KIBO)

Appendix 3: Interactive Play Session Protocol

Protocol

Children will be given a short break after they complete the CT assessment and then invited to an Interactive Play Session (IPS). The IPS will be video recorded with the child seated in the play area. The IPS is structured into three parts:

1. **KIBO Name Game**: Children are shown parts of the KIBO, blocks, sensors and have to name them

2. **Construction Session**: 5-10 minutes to program the KIBO robot to make either an animal robot, a dancing robot or a helper robot. Children may choose other options if reasonable for a 5-10 minute session.

Script:

Today we are going to make something with the KIBO robot. You get to pick what we make! Do you have any ideas?

If child does not pick an activity then the examiner suggests:

Should we make KIBO move like an animal, should we make it do dance, or should we make it do something that can help people?

Basic blocks, advanced blocks and sensors will be organized into three boxes.

Which of the blocks would you like to use? Child allowed to choose and work with the blocks of their choice.

Examiner may assist with scanning of blocks but cannot help plan, create or debug a program
ing

Length of construction session is maximum of 10 minutes, at which time the program must be scanned and tested before proceeding to challenge

3. **Construction challenge**: Child is challenged to enhance the robot to the "next level" for example, adding a loop or a conditional or a sensor if none were used in the program they created during the construction session. Goal is to get child to their zone of proximal development:

If program from construction session does not work, challenge is revision or debugging

If program does work, challenge to add in loop, if then statement, sensors with appropriately worded questions:

Can you make the KIBO light up while it dances?

Can you make the KIBO move forward twice as much as it did?

Allowed interventions during the IPS (with notation of occurrences):

1. Child unable to scan: Examiner may scan if child is unable to do so
2. Child is distracted: Examiner may reorient to the task of programming.
3. Child asks for help: Examiner must try to reflect questions back (what do you think?) If child is stumped, multiple choices can be given.

Appendix Four: Interactive Play Rating System

Computational Thinking with KIBO

Interactive Play Rating Scales

Please complete the following:

1. Rater name: ____________________

2. How many years of experience do you have working with the KIBO robot? (Please round up to nearest whole number)

   ____

3. Please estimate how many children you have interacted with in the context of teaching KIBO? (circle best answer)
   a. 10 or less
   b. Between 10 and 50
   c. Between 50 and 100
   d. Between 100 and 500
   e. Over 500

Instructions: Circle the level (1 – 4) in each category that best describes the child's highest level of ability in each category based on observation of their activities during interactive play with KIBO. If there is insufficient evidence to score a particular category, circle "NS" (Not Scorable). Do not give more than one rating per category or write in fractional ratings (e.g.: "Level 2.5").

Subject Number:		**Rater Name:**
Category	**Level**	**Typical Behaviors**
Algorithms / Modularity	**NS**	Not Scorable (child did not exhibit any programming behaviors related to algorithms/ modularity in the play session)
	1	Places blocks in random or nonsensical order • Doesn't create meaningful sequences in KIBO's blocks
	2	Defines/ names some blocks individually (programs "on the fly") Correctly combines 2 blocks but does not create or show understanding of longer sequences Does not break programming tasks into smaller parts
	3	Connects 3 or more blocks in syntactically correct sequences Begins to divide more complicated programming tasks into simpler steps
	4	Breaks up programming task into parts that are inter-dependent or recursive Uses clusters of blocks as units in larger programs
Hardware	**NS**	Not Scorable (child did not demonstrate their knowledge of hardware in the play session)
	1	Does not operate scanner or start KIBO independently Correctly names and describes function of no more than one part of KIBO
	2	Names and describe function of 2 parts of KIBO Names and explain function of 2 sensors / actuators Scans blocks and start KIBO with assistance
	3	Names and describe function of 3 KIBO parts, 3 sensors / actuators and 3 parameters Scans blocks and starts KIBO without assistance but may make occasional scanning errors
	4	Correctly names and describes the functions of nearly all parts of KIBO, including most sensors, actuators, and parameters Scans independently and starts KIBO successfully most of the time
Software	**NS**	Not Scorable (child did not demonstrate software skills in this play session)
	1	Doesn't relate blocks to robot's actions Treats programming blocks as building blocks (e.g.: stacks them instead of linking into program, doesn't connect blocks)
	2	Can create a simple program (at least 3 blocks) that is syntactically correct Tries to create the "biggest program in the world" without knowing what all th blocks do
	3	Programs a syntactically correct sequence with at least 4-5 blocks Correctly names / uses parameter cards but may confuse if and repeat parameter
	4	programs syntactically correct complex sequences with 6 or more blocks withou assistance Correctly names/ uses parameter cards, loops and conditionals

Subject Number:		**Rater Name:**
Category	**Level**	**Typical Behaviors**
Control Structur	NS	Not Scorable (Child did not demonstrate behavior relating to control structures in the play session)
	1	No understanding or use of repeats or conditionals
	2	Scans the same block repeatedly instead of using repeat blocks
	3	Integrates sensors into program to control robot
	4	Integrates parameters and blocks that control complex KIBO behaviors with feedback from sensors
Debugging	**NS**	Not Scorable (child did not debug in play session)
	1	Does not recognize when programs are not working or attempt to fix errors
	2	Recognizes a program is not working but usually unable to fix errors without assistance
	3	Recognizes problems in program and corrects simple errors with little or no assistance
	4	Recognizes errors in complex program sequences and successfully correct errors without assistance
Design Process	**NS**	Not Scorable (child did not use design process in play session)
	1	No evidence of a design plan or iterative process in creating programs
	2	Design is spontaneous ("On the fly") with little or no iterative revision
	3	Design is an iterative process involving testing and revising single steps
	4	Design is an iterative process in which multiple programming steps or modules may be meaningfully revised in each iteration
Representation	**NS**	Not Scorable (there were no examples of representation in the play session)
	1	Explains differences between symbols on blocks – may or may not be able to read words on blocks
	2	Identifies symbols on basic blocks and demonstrates ability to explain their meaning May not identify more complex blocks, conditional cards or classify categories of block
	3	Identifies symbols on most all blocks and some conditional cards Relates symbols and cards to actions of KIBO
	4	Child identifies all symbolic blocks, sensors, and parameter cards, Correctly explains use of blocks, sensors and parameter cards in relation to KIBO or more abstractly in relation to programming in general

OVERALL RATING OF COMPUTATIONAL THINKING ABILITY

Instructions: Please circle the level (1-4) that best reflects your OVERALL IMPRESSION of the child's Computational Thinking abilities based on review of the Interactive Play video and the typically observed behaviors listed in the table below. If you are between two levels please choose one which most accurately represents the child's performance.

Subject Number:	**Rater Name:**

Level	**Typically observed behaviors**
Proto Programmer (1)	Plays with blocks as blocks not bits. Does not scanning properly. presses the green triangle button without programming KIBO first
Early Programmer (2)	Starting to develop knowledge of hardware and software, simple programming concepts like start with begin, end with end. Child programs without an idea. The child tries to use as many blocks as possible and may scan blocks that are not part of a meaningful program sequence.
Programmer (3)	3-6 instruction/ symbolic idea represented with a program. Can debug and use repeats with adult help
Fluent Programmer (4)	Solves 6+ instruction maze with repeats, advanced debugging by checking work and revising