Elementary Students' Understanding of Variables in Computational Thinking-Integrated Instruction: A Mixed Methods Study

Feiya Luo
feiya.luo@ua.edu
The University of Alabama

Wei Yan
University of Florida
wei1993yan@ufl.edu

Ruohan Liu
University of Florida
liuruohan1990@ufl.edu

Maya Israel
University of Florida
misrael@coe.ufl.edu

ABSTRACT

Variable is a common computer science (CS) concept and is being introduced to upper elementary students in computational thinking (CT)-integrated instruction. However, there is scant empirical evidence of when and how elementary students should learn variables. For example, national computer science (CS) standards advise introducing variables in grades 3-5 and a K-8 variable learning trajectory (LT) synthesized learning goals from the literature and hypothesized four levels of thinking in working with variables. Yet, little empirical research lies behind these. This mixed methods study examined elementary students' understanding of variables. Participants were sampled from two fourth-grade classes from a Midwestern elementary school that implemented a series of CT-integrated math lessons. Students' written responses to variables assessment items were analyzed. Additionally, cognitive think-aloud interviews were conducted with nine students to elicit students' understanding while solving the variables assessment items. Our findings suggested that most students lacked a conceptual understanding of using variables to create generalized problem solutions that could work with any set of inputs. Additionally, students had difficulty with specific mechanics of using variables such as storing user input in a variable, updating variable values, and using the values stored in variables. This study underscores the need for careful design, use, and analysis of elementary CT-integrated lessons and assessments to introduce and reinforce the conceptual understanding and specific mechanics of variables for elementary students.

CCS CONCEPTS

• **Social and professional topics → Computing education**; **Student assessment**; **Computational thinking**; *K-12 education.*

KEYWORDS

Variables, assessment, elementary instruction, computational thinking, mixed methods

ACM Reference Format:
Feiya Luo, Wei Yan, Ruohan Liu, and Maya Israel. 2022. Elementary Students' Understanding of Variables in Computational Thinking-Integrated Instruction: A Mixed Methods Study. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education V. 1 (SIGCSE 2022), March 3–5, 2022, Providence, RI, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3478431.3499323

1 INTRODUCTION

Variables, a foundational computational thinking (CT) and computer science (CS) concept, has primarily been taught in secondary schools (e.g., [3, 4, 9, 10]). In recent years, national and state CS education standards have detailed expectations for elementary students to learn the concept of variables. For example, the revised K-12 Computer Science Standards [1] listed variables (i.e., numbers or symbols used to represent information in programs) as one of the CS subconcepts to introduce in Grades K-2. In the Alabama Course of Study: Digital Literacy and Computer Science [14], students are expected to know how to identify, initialize, and update variables in Grade 5. Despite the multiple standards and prior research in teaching variables to secondary students, there lacks a consensus on when and how variables should be systematically taught in the K-5 context. Such a lack of consensus could attribute to the sparse empirical evidence for determining when it is developmentally appropriate to introduce variables to young students at the elementary level [3]. As a result, many research studies explored the introduction of the variables concept in different grades without a systematic approach of teaching variables. To further complicate the matter, variables is often integrated in different subject matters, such as science and algebra [3, 6, 21] and students often have different conceptual understanding of what variables mean in different content areas. Therefore, empirical evidence of how elementary students work with variables in CS is needed.

This study aimed to provide insight into what competencies and difficulties elementary students demonstrate in working with variables. The research question that guided the study was: What understanding of variables do elementary students demonstrate and what are some common difficulties that students face in working with variables? In the following sections, we first presented

SIGCSE 2022, March 3–5, 2022, Providence, RI, USA.

ACM ISBN 978-1-4503-9070-5/22/03…$15.00
https://doi.org/10.1145/3478431.3499323

relevant previous research that examined students' learning and understanding of the CS concept of variables in K-8. We then described the context and the design of our mixed methods study, before presenting and interpreting the findings and making suggestions for teaching variables at the elementary level.

2 UNDERSTANDING VARIABLES IN K-8

Previous research has explored the teaching of variables for secondary students (e.g., [7]) and reported variables as a difficult concept to learn for middle school students [11, 13]. There are a few common misconceptions that middle school students have in learning variables. For example, students took the concept as identical to variables in math (i.e., an unknown number, or "the letters" in math problems); Students did not know how to initialize a variable or update a variable based on various control flows (e.g., incrementing the counter variable after each iteration of the loop); Students had difficulty developing a concrete understanding of the nature of variables, i.e., how values can be stored and modified [3, 10–12].

The fact that middle school students struggled with working with variables in CS highlighted the importance of establishing foundational knowledge and understanding at the elementary school level. Researchers [3, 10] argued that, rather than implementing isolated activities here and there, it is necessary to provide carefully planned and targeted exposure of variables instruction. A pipeline for K-8 variables instruction will help students develop increasingly complex conceptual understanding of variables since an early age.

Rich et al.'s [18] learning trajectory (LT) for variables theorized how students may develop increasingly advanced knowledge of variables in K-8. The variables LT hypothesized four levels of thinking that students may progress through: Data User, Data Storer, Variable User, and Variable Creator. The four levels of thinking derived from synthesized learning goals from previous literature on teaching variables in K-8. For example, the first level of thinking, Data User, refers to students' knowing what data is and how data is used in a program. The next level, Data Storer, refers to students' conceptualizing variables as storage spaces to store information. The third level specifies that students begin operating on variables and modifying code to manipulate the variables. The last level is when students can initialize and update the value of variables, and implementing new variables. The authors also presented integrated instruction (a series of CT-math integrated lessons) aimed at addressing particular facts and skills in variables as well as to facilitate students' progressing along the four levels of thinking [18]. The following section provided more detail in what the integrated lessons entailed and how they were used in this study.

3 METHOD

3.1 Participants and intervention

The participants of this study were sampled from two fourth-grade classes at an elementary school in a Midwestern state in the U.S. During the 2019-2020 school year, the school had a student population that was 42.3% White, 30.3% Black/African American, 9% Asian, 6.1% Hispanic, and 12.2% Multiracial.

All participants took part in eight CT-math integrated lessons, which were implemented in the 2019-2020 school year at the elementary school. Developed under an National Science Foundation (NSF)-funded project, the eight lessons integrated a range of CT concepts, such as sequence, repetition, conditionals, decomposition, debugging, and variables in fourth-grade fractions and aligned to the Common Core State Standards for Math (CCSS-M). Four of the eight lessons have a specific instructional focus on variables with instructional activities intentionally developed for teaching the facts and skills in the variables LT [18]. For the purpose of this paper, only the four lessons focused on variables were explained.

The first variables lesson (the third in the lesson series), *Robot Boxes*, introduced an unplugged activity for fourth-grade students to store index cards with values in physical boxes, simulating how a computer program uses variables to store data. Students calculated either the perimeter or the area with the values in the index cards and output the result by performing the corresponding calculations. This lesson aimed at reinforcing students' understanding of data as pieces of information and of how computers use variables as a placeholder to store information. This lesson served to transition students from Data User to Data Storer.

The other three lessons (the fourth, the sixth, and the seventh lesson in the series), *Math Chat, Ambling Animals, Slicing Sandwiches*, were designed to move students from Data Storer to Variable User. *Math Chat* had students work with programs that asked for user input, stored the user input in variables as dimensions for a rectangle, and output the area using the user input values. In *Ambling Animals*, students interacted with a Scratch program that compared two randomly generated fractions. Students worked with user input and observed the output of the program. In this activity, students were introduced to storing and operating on user input as values of the variables. In the last lesson, *Slicing Sandwiches*, students compared randomly generated fractions with the same denominator and built conditional statements so that the program would display the correct inequality symbol based on the values stored in variables for the numerators. The complete set of lessons can be found here: http://everydaycomputing.org/lessons/action-fractions/.

3.2 Research design

Through collecting and triangulating both qualitative and quantitative data, the study used the convergent design of the mixed-method approach [5] to reveal students' understanding of variables after engaging in math-CT integrated instruction. The advantage of the convergent design is to obtain insight from a large sample size with objective measures while also maintaining details and depth with a closer examination of the experiences of the participants. The approach is appropriate when researchers want to compare and corroborate statistical results with subjective interpretations for a comprehensive understanding of the research problem or validate research findings by triangulating the two forms of data sources [5]. Following Creswell and Clark's [5] recommendations for the convergent design procedures, we conducted this mixed methods study in four steps: (1) We first collected the qualitative and quantitative data on students' understanding of variables; (2) We analyzed the two data sets separately and independently; (3) We compared quantitative and qualitative results through a narrative; (4) We made interpretations after carefully examining how the two sets of results support and relate to each other. The following sections provided more details in data collection and analysis.

Data collection. We collected both qualitative and quantitative data using assessment items designed to solicit students' understanding of variables. These items were part of a series of paper-and-pencil assessments developed under the same NSF project where the integrated lessons were designed. The assessments reflected and mapped to a range of knowledge, skills, and abilities (KSAs) indicated by the learning goals in an LT (i.e., a CT concept) [17–20]. The assessment items aimed at collecting evidence of the competencies students articulate or demonstrate in problem-solving using the evidence-centered design [8]. Two sets of assessments, referred as the Grade 4 (G4) early and mid assessments, were administered with the students after the fourth and the eighth integrated lesson, respectively. Both assessment sets consist of items that target different CT concepts (i.e., sequence, repetition, conditionals, decomposition, and variables) and were either embedded in Scratch, simple number problems, or word problems [8]. Given the scope of this paper, only the variables items (n=4) were discussed (item IDs: V.07.c, V.12.a, V.13.a, and V.14.a). The four variables items (Figures 1-4) used in this study assessed a range of KSAs that mapped to a range of learning goals in Rich et al.'s variables LT [18]:

- Ability to use a variable to hold and reference data.
- Knowledge that variables have types related to the kind of data they hold and/or the operations available on that data.
- Ability to use a variable to accept and hold user input.
- Ability to use a variable in a boolean expression to control the program flow.

The quantitative data collected involved students' assessment performance results (i.e., written responses to the variables items). Thirty-six students took part in the G4 early assessment and 21 participated in the G4 mid. All participants provided a parental consent and an assent. The qualitative data was collected through cognitive interviews using a think-aloud protocol that elicited elementary students' thinking when solving the variables assessment items. Nine students who provided a separate parental consent and assent participated in a cognitive interview with the variables items after each written assessment. A total of 18 interview responses to the four variables items were collected.

Data analysis. Analysis of the data involved separately examining the qualitative and quantitative data before merging the qualitative themes with the statistical findings.

Quantitative data analysis. Students' written responses in the assessments were scored using criteria designed to categorize the different student responses. For example, for one of the items, V.14.a (Figure 4), the criterion involves four facets (each equals one point): Asking user to input a value, storing a value in a variable, performing a calculation using a variable, and presenting the final value of the variable. As such, a complete answer to this item will yield a total score of 4.

To ensure the reliability of scores, we first scored 30% of all data based on the established scoring criteria. Each of the students' responses to the items were scored by at least two raters. Cohen's Kappa showed that the inter-rater reliability (IRR) reached at least .93 for the 30% subset. The raters revisited and resolved disagreements to reach a final scoring decision. Then, the raters continued to score the remaining 70% of the data. Any disagreements of scores were resolved through rigorous discussion and ultimately resulted in perfect IRR. The descriptive statistics of students' scores were presented in the Results section.

Table 1: Frequency of students' assessment answers

Item ID	IDK	Blank	Scored 0	Some score	Total
V.14.a	2	21	12	1	36
V.07.c	12	18	27	0	57
V.12.a	1	1	16	3	21

Table 2: Frequency of students' assessment answers (n=21) to V.13.a

Parts/Blanks	IDK	Blank	Scored 0	Correct answer
a1	0	2	1	18
a2	0	2	3	16
b1	0	2	18	1
c1	0	1	17	3
c2	0	1	7	13

Qualitative data analysis. For analysis of the interviews, we used an adapted version of the constant comparison analysis [2] to understand students' articulation of variables at the item level. Three researchers who were also scorers of the assessments collaboratively coded and categorized students' think-aloud responses to each assessment item to understand what competencies and difficulties students demonstrated while solving the item. The three researchers rigorously discussed the patterns emerged from students' responses. The patterns were described in detail in the results section before being incorporated with the quantitative results for comprehensive interpretations.

In the next section, we presented the descriptive statistics, followed by the in-depth description of the cognitive interview findings to reveal the understanding of variables that the students demonstrated and the common difficulties they faced.

4 RESULTS

4.1 Assessment performance results

Item V.07.c (Figure 1), appearing in both the G4 early assessment and mid assessment, received 57 responses. Item V.14.a (Figure 4), in the G4 early assessment, received 36 responses. Items V.12.a (Figure 2) and V.13.a (Figure 3) from G4 mid received 21 responses each. For each of the items, students' written responses were categorized into four groups: Did not know the answer (IDK), no answer provided (Blank), answers did not earn any points (Scored 0), and partial or complete answer (Some Score). The frequencies for each category were presented in Tables 1 and 2.

For items V.14.a, V.07.c, and V.12.a, the majority of the participants in this study either acknowledged that they did not know the answer, or left the answer sheet blank, or provided an incorrect answer that earned no score. Only one participant provided a partially correct answer to V.14.a and three provided completely correct answers to V.12.a (Table 1).

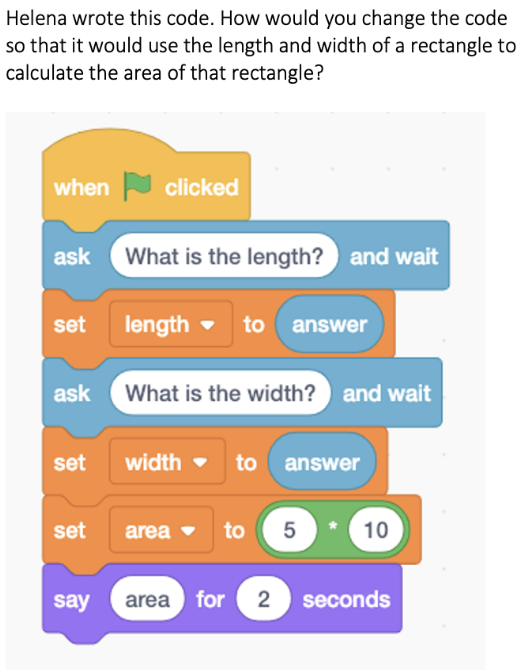

Figure 1: Item V.07.c

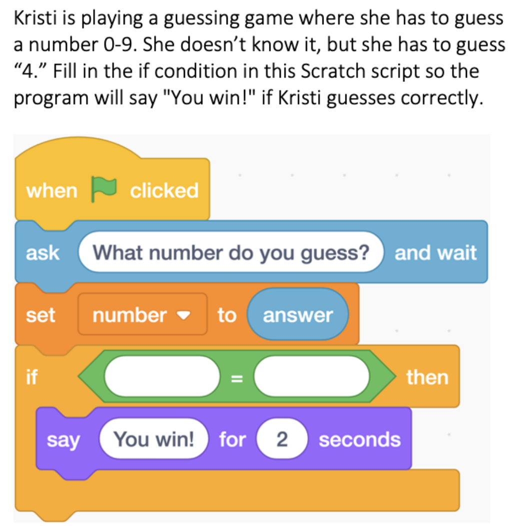

Figure 2: Item V.12.a

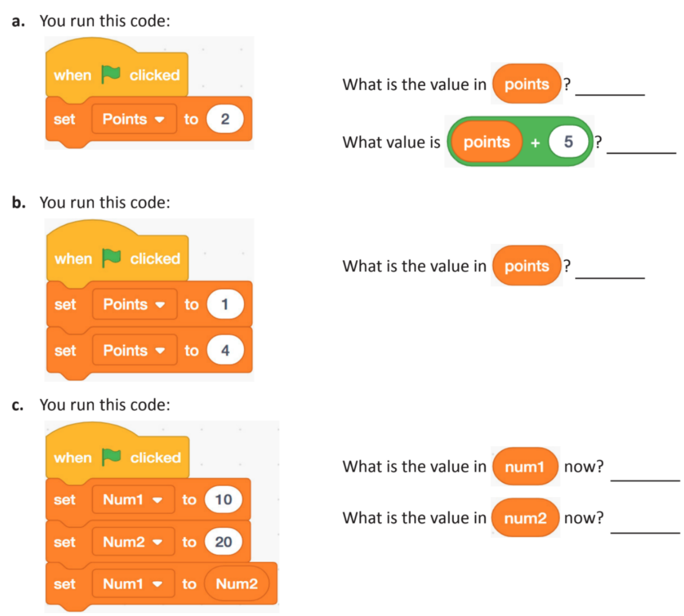

Figure 3: Item V.13.a

For item V.13.a, which had five parts/blanks for students to fill in the value of variables, more participants provided a correct answer in parts a1, a2, and c2 than in the other two parts (Table 2).

4.2 Cognitive interview results

V.07.c. asked students to make the necessary changes so that the program will use the user input stored in the "length" and "width" variables to calculate the area rather than using constants (Figure 1). Students needed to know how to store user input in variables and perform operations on the variables, instead of the constants (5 and 10). Seven responses were collected through think-aloud. One student’s think-aloud provided evidence for his lack of understanding of user input: “[The] set the length to the answer [block] doesn’t make sense” (P5). Two students, in their response, described the program without identifying the changes required, showing no understanding of user input (P8 & P16). Other students expressed non-relevant solutions, for example, changing either the positions of the blocks (P6 & P16) or the amount of seconds in the “say area” block (P16), also showing no understanding of user input.

V.12.a. asked students to fill in the conditional statement so that the program outputs a message when the user guesses the correct number "4." Students needed to understand how to store user input in a variable, check equivalency of the user input to a predetermined value, and evaluate the true-or-false state of the boolean expression. Three responses were collected through think-aloud. One student changed the boolean expression to “if 4, [then say. . .]” (P6). Another student put “If correct = ding,” indicating that if guessed correctly, “[play the] ding [sound], then, say you win” (P7). The third student put “if 4 = 8, then...” explaining that “4 plus 2 equals 8” (P15).

V.13.a. This three-part item asked students to identify the initial value of a variable and perform operations on that variable. Four think-aloud responses were collected with evidence showing students’ understanding of deciding a variable’s initial value and how to do simple addition on the variable (P1 & P8). However, there was no evidence of students’ understanding of updating the value of variables. Students thought that all assigned values should be added together, rather than updating the value with the most recently assigned value. For example, one participant said “That will be 5 because it sets points to 1, and then set point to 4, 1 plus 4 equals to 5. So..five” (P1). Students’ responses also showed that in part c,

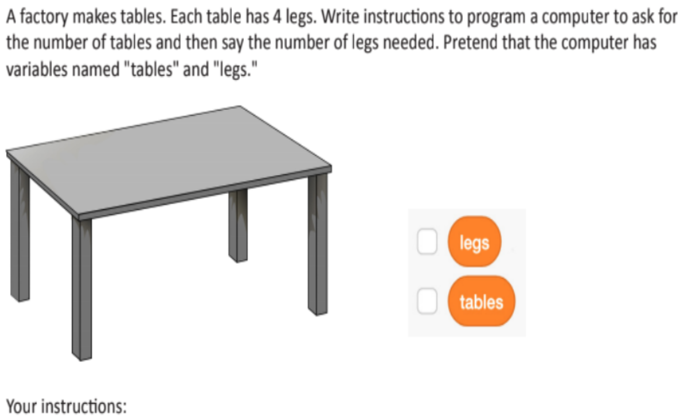

Figure 4: Item V.14.a

when multiple variables are initialized, assigned, and updated, students tended to treat codes separately (line-by-line tracing) without understanding that the entire program needed to finish running before deciding the outputs.

V.14.a. asked students to use variables in a program to calculate the number of legs needed depending on how many tables are needed. Students needed to know how to write instructions to ask for user input, take the user input as the assigned value of the variables, and do operations on the variables, instead of constants. Four think-aloud responses were collected. Students tended to think that the item asked for solving "how many tables they need to make and how many legs for each table" (P7 & P1) rather than constructing a program that calculates the number of legs based on the user-determined number of tables. Other students either talked about how to build a table (P5), or did not understand what the question was asking (P15).

5 DISCUSSION

5.1 Students' competencies and challenges

In this section, we first made interpretations of the descriptive statistics of students' performance results. We then explained how the cognitive interview results corroborated students' assessment results and examined the common difficulties students faced.

The four variables items used in this study assessed a range of learning goals in the variable LT, including using a variable to hold and reference data, manipulating the variable (i.e., initializing, updating, and operating on variables), using a variable to accept and hold user input, and using a variable in a boolean expression to control the program flow. Few students earned scores on items V.07.c, V.12.a, V.14.a and the two sub-parts of V.13.a (b1 and c1), meaning that most students had not progressed beyond "Data Storer," the second level of the variables LT. The two aspects that many participants demonstrated an understanding of were: assigning value to a variable (i.e., setting the "Points" variable to 2 and setting the "Num2" variable to 20) and doing simple arithmetic operation on a variable (i.e., adding 5 to the "Points" variable).

The cognitive interviews corroborated the quantitative results and provided rich details of the nature of students' assessment performance. Generally speaking, students had difficulty in the following aspects: (1) articulating how to initialize or update a variable, (2) accept and store user input in a variable, and (3) reference data/user input using a variable, or reference variables in a boolean expression to control the program flow. In terms of updating variables, students generally did not demonstrate the understanding that a variable holds only one value at a time. This means that, if a second value is set to the same variable, that variable will now carry the most recently-assigned value. Cognitive interviews revealed a misconception carried by the students, which was that values set to a variable were to be added together.

The think-aloud also revealed that the students tended to perform calculations with an arbitrarily-decided constant value, rather than using a variable. This tendency was likely due to students' familiarity with concrete values and objects. Students at this age may not have had much experience with abstract representations, such as using a variable to represent changing values. In conclusion, the cognitive interview results showed that most students lacked the knowledge and skills necessary for them to move towards "Variable User," the third level of the variables LT.

5.2 Implications for instruction and assessments

While national and state CS standards advise introducing variables in grades 3-5 and that the K-8 variable LT synthesized learning goals from the literature, research has mainly focused on variables teaching at the secondary level. Previous studies [15, 16] have discussed how LT-based instruction and assessments can help identify learning gaps and misconceptions that students have when learning CS concepts such as sequence, repetition, conditionals, and decomposition. However, there is little empirical research in how elementary teachers should teach and how students learn variables. Through triangulating students' assessment performance results with cognitive interviews, this study collected empirical evidence of students' competencies working with variables and identified some common difficulties that students faced. Based on the quantitative and qualitative findings, we made the following recommendations in regard to LT-based instruction and assessment of variables at the elementary level. We also illustrated example unplugged activities that may be adopted by teachers and researchers in variables instruction and research.

Increased exposure on the mechanics of variables. The results of this study suggested that students likely needed more exposure around the mechanics of variables, such as the different data types, rules of initiating and updating variables, working with user input, and so on. The integrated lessons by Rich et al. [18] set a good starting point to explore how to introduce the CS concept of variables to elementary students. These integrated lessons demonstrated some deliberate activities and assessments designed based on the variables LT to move students towards more advanced understanding of variables. However, the lack of evidence of students' understanding of variables could mean that increased exposure is needed. In addition to previous research that argued for systematic and structured instruction on variables [3, 9], we also suggested increasing instructional exposure with various targeted activities to reinforce students' conceptual understanding. Future research is encouraged to explore how extended instruction using targeted activities reinforces students' understanding of on the mechanics of variables.

We illustrated different unplugged activities that can be used in instruction below. Following the learning goals in the variables LT, instruction should first focus on having students understand the different types of data that a variable can store and how to reference information by the variables instead of by the information itself. Given students' limited experience with the abstract variables concept, concrete objects such as boxes may be used to represent variables used by computer programs to store information. Also, elementary grades-appropriate language should be used when referencing the types of variables, for example, "numbers" and "texts" instead of "integers" and "strings." At this stage, students will need to differentiate a variable in CS (i.e., a piece of information that can be changed and referenced in a computer program) from a variable in math (i.e., usually refers to an unknown quantity). To reinforce students' understanding of referencing data using variables, an activity can have students place an object or a message in the boxes and practice referencing to the object or message by the box containing them (e.g., box1, box2, and so on).

At the next stage, students may be introduced to how to seek and store user input and reference it with variables. Again, using the box illustration, an activity can be where students work in groups and take turns in controlling the box and ask a peer to put in an object or a message. Then, have students discuss the object or the message by referencing to the box (i.e., box1, box2, and so on).

When students are ready, the instructional focus can then move to the updating of variables (i.e., a variable holds the last value that is set to it). Instruction may center around how each of the physical boxes can take one object or message at a time. We recommend that teachers pay attention to the misconceptions revealed by this study. One of such misconceptions was that, even though students started to conceptualize that variables can hold values, they mistakenly thought that variables took values in an accumulating effect, rather than replacing the old value with the new. This means that teachers may need to emphasize that a variable holds only the last value set to it and the value previously set would be replaced. For example, remind students that whenever a new object or message is placed in the box, what is previously in the box is now discarded. Future research is encouraged to explore how extended instructional exposure using such activities help upper elementary students meet the learning goals in the variables LT.

Strategic design and use of assessments. The written and cognitive interview results provided additional validation evidence for the design of the variables items and suggested the need to revisit specific items, such as the one that received few correct answers (V.07.c). This item was revised after our validation study [8] and the revision resulted in the version used in this study. However, data analysis in this study still showed a lack of evidence that supported students' understanding of user input. We also suspected that for item V.13.a, the position of the questions in part c with blanks may have misled students to examine the outcomes of the codes line by line rather than finishing running the entire program. It was also possible that the Scratch syntax in some items posed challenges to students' problem-solving. Therefore, future research may explore how students solve those items with simplified code.

The assessments designed based on the LT provided one way to assess students' learning outcomes. However, the assessment results may not "tell the full story," especially in the cases of blank or incorrect answers that lacked elaboration. In real life, the strict curriculum schedule in elementary schools may not always allow the use of cognitive interviews as a diagnostic approach to inform teachers which learning goals in the LT that students are not meeting. Therefore, we called for careful and strategic design of assessment items that target individual learning goals. We acknowledged the need to also design items that assess students on a group of learning goals or on higher-order problem-solving. However, researchers and teachers should be cautioned that such items, when used as a summative assessment, may not always yield the most useful understanding of students' knowledge gaps. Hence, a combination of items that target both a single learning goal and higher-order problem solving may grant a more meaningful interpretation of assessment results.

Given that previous research has reported the rarity of middle school students' meaningful use of variables [9], the purpose of elementary instruction may focus on preparing students for foundational, conceptual understanding and familiarity of the language associated with variables. It is recommended that variables instruction focuses on the specific mechanics of variables in elementary grades before engaging students in higher-level abstraction, as partial understanding and misconceptions may prohibit students from successfully applying the knowledge in problem-solving.

6 CONCLUSION & LIMITATIONS

This mixed-methods study examined fourth-grade students' understanding of variables by collecting both assessment performance results and cognitive think-aloud as students engaged in a series of math-CT integrated lessons. The cognitive interviews complemented and corroborated students' performance results and provided rich insight in how students worked with variables. Specifically, this study generated empirical evidence of the competencies that the students demonstrated, identified common difficulties that the students faced, and discussed how we can address those difficulties with instruction. The results and implications of this study are expected to be beneficial to researchers, instructional designers, and elementary teachers.

Due to the sampling of this study, generalization of the results and interpretations findings should be made with caution. A limited number of students participated in the cognitive interviews, so while the qualitative results were descriptive, they may not be entirely representative. Last but not the least, this study was part of the early implementation of the integrated lessons and assessments and initial validation of the early and mid assessments was available[8]. However, it is acknowledged that additional validity evidence of the variables items is needed.

7 ACKNOWLEDGEMENTS

This study was supported by the National Science Foundation under Award Number 1932920.

REFERENCES

[1] Computer Science Teachers Association. 2017. K-12 Computer science standards. *https://portal.ct.gov/-/media/SDE/CTE/CSTA-K12-Computer-Science-Standards-Revised-2017.pdf* (2017).

[2] Hennie Boeije. 2002. A purposeful approach to the constant comparative method in the analysis of qualitative interviews. *Quality and quantity* 36, 4 (2002), 391–409.

[3] Philip Sheridan Buffum, Kimberly Michelle Ying, Xiaoxi Zheng, Kristy Elizabeth Boyer, Eric N Wiebe, Bradford W Mott, David C Blackburn, and James C Lester. 2018. Introducing the computer science concept of variables in middle school science classrooms. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. 906–911.

[4] Michelle Craig and Diane Horton. 2009. Gr8 designs for Gr8 girls: a middle-school program and its evaluation. In *Proceedings of the 40th ACM technical symposium on Computer Science education*. 221–225.

[5] John W Creswell and Vicki L Plano Clark. 2017. *Designing and conducting mixed methods research*. Sage publications.

[6] Jill Denner, Linda Werner, and Eloy Ortiz. 2012. Computer games created by middle school girls: Can they be used to measure understanding of computer science concepts? *Computers & Education* 58, 1 (2012), 240–249.

[7] Kathi Fisler, Emmanuel Schanzer, Steve Weimar, Annie Fetter, K Ann Renninger, Shriram Krishnamurthi, Joe Gibbs Politz, Benjamin Lerner, Jennifer Poole, and Christine Koerner. 2021. Evolving a K-12 Curriculum for Integrating Computer Science into Mathematics. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*. 59–65.

[8] Brian D Gane, Maya Israel, Noor Elagha, Wei Yan, Feiya Luo, and James W Pellegrino. 2021. Design and validation of learning trajectory-based assessments for computational thinking in upper elementary grades. *Computer Science Education* 31, 2 (2021), 141–168.

[9] Shuchi Grover and Satabdi Basu. 2017. Measuring student learning in introductory block-based programming: Examining misconceptions of loops, variables, and boolean logic. In *Proceedings of the 2017 ACM SIGCSE technical symposium on computer science education*. 267–272.

[10] Shuchi Grover and Roy Pea. 2013. Computational thinking in K–12: A review of the state of the field. *Educational researcher* 42, 1 (2013), 38–43.

[11] Shuchi Grover, Roy Pea, and Stephen Cooper. 2015. Designing for deeper learning in a blended computer science course for middle school students. *Computer science education* 25, 2 (2015), 199–237.

[12] Athanassios Jimoyiannis. 2013. Using SOLO taxonomy to explore students' mental models of the programming variable and the assignment statement. *Themes in Science and Technology Education* 4, 2 (2013), 53–74.

[13] Marja Kuittinen and Jorma Sajaniemi. 2004. Teaching roles of variables in elementary programming courses. In *Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education*. 57–61.

[14] Digital Literacy, Computer Science Course of Study Committee, and Task Force. 2018. The Alabama Course of Study: Digital Literacy and Computer Science. *https://www.alsde.edu/sec/sct/COS/Final%202018%20Digital%20Literacy%20and%20Computer%20Science%20COS%205-14-19.pdf* (2018).

[15] Feiya Luo, Maya Israel, and Brian Gane. Accepted. Exploring elementary students' computational thinking: A learning trajectory perspective. *Transactions on Computing Education* (Accepted).

[16] Feiya Luo, Maya Israel, Ruohan Liu, Wei Yan, Brian Gane, and John Hampton. 2020. Understanding Students' Computational Thinking through Cognitive Interviews: A Learning Trajectory-based Analysis. In *Proceedings of the 51st ACM Technical Symposium on Computer Science Education*. 919–925.

[17] Kathryn M Rich, T Andrew Binkowski, Carla Strickland, and Diana Franklin. 2018. Decomposition: A K-8 Computational Thinking Learning Trajectory. In *Proceedings of the 2018 ACM Conference on International Computing Education Research*. ACM, 124–132.

[18] Kathryn M Rich, Diana Franklin, Carla Strickland, Andy Isaacs, and Donna Eatinger. 2020. A Learning Trajectory for Variables Based in Computational Thinking Literature: Using Levels of Thinking to Develop Instruction. *Computer Science Education* (2020), 1–22.

[19] Kathryn M Rich, Carla Strickland, T Andrew Binkowski, Cheryl Moran, and Diana Franklin. 2017. K-8 learning trajectories derived from research literature: Sequence, repetition, conditionals. In *Proceedings of the 2017 ACM conference on international computing education research*. ACM, 182–190.

[20] Kathryn M Rich, Aman Yadav, and Christina V Schwarz. 2019. Computational Thinking, Mathematics, and Science: Elementary Teachers'Perspectives on Integration. *Journal of Technology and Teacher Education* 27, 2 (2019), 165–205.

[21] Linda Seiter. 2015. Using SOLO to classify the programming responses of primary grade students. In *Proceedings of the 46th ACM technical symposium on computer science education*. 540–545.