Levels of Abstraction in Students' Mathematics Strategies: What Can Applying Computer Science Ideas about Abstraction Bring to Elemen....

Kathryn M. Rich
Michigan State University

Aman Yadav
Michigan State University

Jl. of Computers in Mathematics and Science Teaching (2019) **38**(3), 267-298

Levels of Abstraction in Students' Mathematics Strategies: What Can Applying Computer Science Ideas about Abstraction Bring to Elementary Mathematics?

KATHRYN M. RICH, AMAN YADAV, AND MARISSA ZHU
Michigan State University, USA
richkat3@msu.edu
ayadav@msu.edu
zhumengd@msu.edu

Moving among levels of abstraction is an important skill in mathematics and computer science, and students show similar difficulties when applying abstraction in each discipline. While computer science educators have examined ways to explicitly teach students how to consciously navigate levels of abstraction, these ideas have not been explored in mathematics education. In this study, we examined elementary students' solutions to a commonplace mathematics task to determine whether and how students moved among levels of abstraction as they solved the task. Furthermore, we analyzed student errors, categorizing them according to whether they related to moves among levels of abstraction or to purely mathematical steps. Our analysis showed: (1) students implicitly shift among levels of abstraction when solving "real-world" mathematics problems; (2) students make errors when making those implicit shifts in abstraction level; (3) the errors students make in abstraction outnumber the errors they make in purely mathematical skills. We discuss the implications for these findings, arguing they establish that there are opportunities for explicit instruction in abstraction in elementary mathematics, and that students' overall mathematics achievement and problem-solving skills have the potential to benefit from applying these computer-science ideas to mathematics instruction.

INTRODUCTION

Utilizing abstraction is considered an important skill for mathematics students to develop, as highlighted in recent mathematical practice standards (Common Core State Standards Initiative [CCSSI], 2010). Students must move among levels of abstraction—for example, shifting between a contextualized situation and a mathematical model of that situation—as they solve mathematical problems. Moreover, students face difficulties when engaging in abstraction during mathematics problem solving (Hazzan, 2003; Hazzan & Zazkis, 2005). Even so, few conversations in mathematics education have attended to the idea of moving among levels of abstraction or explored the affordances of explicitly teaching this skill to students.

Computer science has several parallels with mathematics regarding abstraction. Several scholars have argued that abstraction, and particularly, freely and consciously moving among levels of abstraction, is a critically important skill for computer scientists. Kramer (2007) suggested it is “a key skill for computing” (p. 41), and Hazzan (2008) described it as a central theme of both computer science (CS) and software engineering. The importance of abstraction in CS, combined with the challenges of teaching it (Hazzan, 2008), have led CS researchers to develop and test frameworks for helping students learn how to consciously move among levels of abstraction as they solve problems (Armoni, 2013; Statter & Armoni, 2016). Given the lack of explicit attention to moving among levels of abstraction in mathematics education, this raises questions about how the instructional emphasis on levels of abstraction in computer science might be applied to mathematics to better support the development of mathematics students’ abstraction skills.

In this paper, we examine elementary students’ work on a commonplace mathematics task, attending specifically to the ways that students are required to use abstraction as they solve the problems. We bring the idea of levels of abstraction, or more specifically, “being able to move freely and consciously between levels of abstraction” (Armoni, 2013, p. 272) to bear on our examination of the student work. We use the results to argue that attention to levels of abstraction, discussed by many computer science scholars (Armoni, 2013; Hazzan, 2005; Wing, 2006) but not by mathematics educators, could be a valuable addition to elementary mathematics curricula.

BACKGROUND

What is Abstraction? What Are Levels of Abstraction?

Abstraction refers broadly to the process of reducing complexity by ignoring irrelevant details in order to focus attention on important elements of a problem, situation, or phenomenon (College Board, 2017; Kramer, 2007; Statter & Armoni, 2016). Representations of phenomena that capture important elements but do not focus on irrelevant details are also often referred to as abstractions; thus, the term refers to both a process (creating such representations) and the product of that process. Kramer (2007) argued that abstraction is applied in many contexts and disciplines, noting that artists employ abstraction when creating simplified figures and landscapes, and jazz musicians use abstraction to identify and then embellish a core melody.

Computer science is one of the disciplines where abstraction is used extensively. CS scholars and educators have argued that CS students must learn not only to create and interpret abstractions, but to work at multiple levels of abstraction and be able to consciously and freely move among those levels (Armoni, 2013; Hillis, 1998; Wing, 2006). Levels of abstraction can be distinguished by both scope and amount of detail currently in view. When working at a higher level of abstraction, a problem solver has a wider view of the scope of the problem and the context in which it is situated, but as such, cannot necessarily consider all the fine details and complexities of the situation. For example, when computer scientists are thinking about the overall problem to identify the kind of algorithm that might be used to solve it, they are working at a higher level of abstraction. They are not necessarily considering any details about implementation. At a lower level of abstraction, a problem solver can see the fine-grained details by focusing on different aspects of the situation, but cannot attend to the full scope of the problem (Armoni, 2013). For example, when computer scientists are implementing a particular algorithm in a particular language, they are working at a lower level of abstraction, and may not be considering the full scope of the problem. Statter and Armoni (2016) described the skill of identifying and moving to the appropriate level of abstraction for different points in the problem-solving process as "[f]inding the right degree of resolution" (p. 80).

While abstraction is not discussed as a key mathematical concept as often as it is discussed as a key CS concept, abstraction is also an important concept in mathematics. This is highlighted in one of the eight Standards for Mathematical Practice (SMPs) articulated in the Common Core State Standards for Mathematics (CCSS-M; Common Core State Standards Initiative [CCSSI], 2010), *Reason abstractly and quantitatively.* The description of the standard states that mathematically proficient students should have

> the ability to decontextualize—to abstract a given situation and represent it symbolically and manipulate the representing symbols as if they have a life of their own, without necessarily attending to their referents—and the ability to contextualize, to pause as needed during the manipulation process in order to probe into the referents for the symbols involved. (CCSSI, 2010, p. 6)

Although the standard does not directly reference moving among levels of abstraction, the language implies the same general idea. When students decontextualize, they are zooming in to a lower level of abstraction to manipulate mathematical symbols without referencing the full scope of a situation. When student contextualize, they are zooming out to a higher level of abstraction to consider greater scope, but fewer details. Thus, this standard for mathematical practice calls for students to "move freely and consciously between levels of abstraction" (Armoni, 2013, p. 272), suggesting that levels of abstraction may be a worthy point of instruction in mathematics. We explore this idea further in this study by examining elementary students' mathematics work with specific attention to levels of abstraction.

Student Difficulties with Abstraction

Standards documents (College Board, 2017; CCSSI, 2010) suggest that moving among levels of abstraction is an important skill for computer science and mathematics students to develop. This raises questions about whether learning to use abstraction requires explicit instruction or can be acquired by students implicitly as they solve problems. Hazzan illustrated the ways in which mathematics and computer science students unconsciously move among levels of abstraction as they solve problems (Hazzan, 2003; Hazzan & Zazkis, 2005). Specifically, Hazzan noted a tendency for students to move to lower levels of abstraction (or *reduce abstraction*) when they were uncomfortable or unfamiliar with topics. For example, when asked to generate a proof across a set of numbers—to think at a high level of abstraction—mathematics students often begin by examining a few examples within the set, thus moving to a lower level of abstraction where more detail is visible (Hazzan & Zazkis, 2005).

Hazzan argued her work illustrated the utility of using reductions in abstraction as a framework for making sense of students' strategies for solving problems in mathematics and computer science (Hazzan, 2003; Hazzan & Zazkis, 2005). We see implications of Hazzan's work that go beyond establishing the utility of the reducing-abstraction framework: It also reinforces the importance of explicitly teaching abstraction to students. From Hazzan's work we know that students *unconsciously* shift to lower levels of abstraction in both math and CS. While researchers argue that *consciously* moving among levels of abstraction is an important skill (Armoni, 2013; Hillis, 1998; Wing, 2006), unconsciously shifting among abstraction levels can be problematic in at least two ways. First, when shifts are unconscious, students may have difficulty attending to whether the shifts have been carried out correctly or whether the shift allowed them to maintain key aspects of the problem. For example, shifting from thinking about arrays as a data structure to real-world examples of single objects (Hazzan, 2003) may not be a productive shift. Arrays and real-world objects have different properties. Even though a single real-world object does not need a container, a single data element can have a container and therefore can be an array.

Second, even when shifts are potentially productive, such as when students shift from thinking about a set of numbers to a single number in that set (Hazzan & Zazkis, 2005), students who make such a shift unconsciously may not realize they need to move back to the higher level of abstraction to completely solve the problems. A proof for one element of a set, for example, does not constitute a proof for the whole set. Thinking about one element may spur a full solution, but only if a student moves back to a higher level of abstraction eventually. Thus, Hazzan's studies establish the difficulty of abstraction for students, and also illustrate the need for explicit instructional attention to teaching it.

Proposed Methods for Teaching Abstraction

Given the difficulty that abstraction poses for students when they use it unconsciously (Hazzan, 2003; Hazzan & Zazkis, 2005), it is prudent to consider how students might be taught to move among levels of abstraction consciously. Writing from the perspective of computer science, Hazzan (2008) noted that abstraction is a "soft idea." In other words, abstraction is difficult to explicitly define and cannot be carried out via a predetermined list of steps, which makes it difficult to teach. Hazzan, therefore, advocated for teaching abstraction by making it explicit to students, with instructors articulating how they move among levels abstraction to solve problems. Further, Hazzan argued students should be given opportunities to reflect on the ways that they move among levels of abstraction in their own problem solving.

In computer science, Armoni (2013) developed a framework for teaching abstraction in the context of algorithm development that emphasizes explicit attention to levels of abstraction. Specifically, Armoni advocated that instructors should be persistent and precise in pointing out the various levels of abstraction they use as they solve problems and distinguish among the levels. Moreover, she suggested that instructors should use language cues to distinguish between levels and provide plenty of opportunities for students to reflect upon their own problem-solving processes and the levels of abstraction they used. A preliminary study found that use of Armoni's framework led computer science students as young as 7th graders to improve their abstraction abilities (Statter & Armoni, 2016).

In mathematics, little work has explicitly addressed how to teach abstraction. However, just as the standards for mathematical practice (CCSSI, 2010) discuss moving among levels of abstraction without explicitly calling it out, mathematics education researchers have proposed frameworks that partially echo Armoni's (2013) intent without using abstraction language. For example, Greer (1997) proposed teaching mathematics through a modeling approach where teachers would give explicit attention to the problem context, a situation model that captures important information in the context, a mathematical model that identifies the appropriate mathematical operations, and the solution interpreted in the context of all three of the former elements. These elements—context, situation model, and mathematical model—could be viewed as levels of abstraction, and Greer's recommendations could be viewed as parallel to Armoni's recommendations for explicit attention to levels of abstraction by instructors. Even so, specific changes to mathematics teaching that build on Greer's framework tend to emphasize the authenticity of the problems posed to students (Palm, 2008) and/or placing students in the role of deciding on the validity of an approach or answer (Vershaffel & DeCourte, 1997). Less attention has been given to examining students' processes of moving among levels of abstraction or focusing instruction on how to improve these processes.

CURRENT STUDY

We have established moving among levels of abstraction is an important skill for mathematics and computer science students to learn (College Board, 2017; CCSSI, 2010) and that students in each discipline exhibit difficulties with this skill (Hazzan, 2003; Hazzan & Zazkis, 2005). But whereas computer science educators have developed teaching methods and examined students' work in terms of how they move among levels of abstraction (Armoni, 2013; Statter & Armoni, 2016), these ideas have not been explored in mathematics education.

In this paper, we take preliminary steps to examine the feasibility and potential benefits of providing explicit instruction on moving among levels of abstraction in elementary school mathematics. In particular, our goal in this study was to see where and how opportunities exist to make instruction on abstraction explicit within elementary mathematics. First, we must determine whether common elementary mathematics tasks offer opportunities for discussing levels of abstraction. Assuming there are such opportunities, we must examine students' attempts to shift between levels and consider the impact of difficulties students encounter. Lastly, we must consider whether elementary school teachers and administrators could be convinced of the value of spending instructional time attending to abstraction. In this study, we examine these issues by analyzing a commonplace elementary mathematics task, and students' work on this task, through the lens of abstraction and moving between levels. We pose the following research questions:

1. How do fourth- and fifth-grade students move between levels of abstraction during mathematics problem solving?
2. To what extent are students successful at carrying out the abstraction steps of their problem-solving strategies? What errors do they make?
3. How does their level of success with abstraction compare to their level of success with the mathematics steps?

We see these questions as aligning with the three issues outlined above regarding bringing explicit instruction in abstraction to elementary mathematics. The answer to research question 1 will establish whether or not there are opportunities for addressing abstraction through work on elementary mathematics tasks. The answer to research question 2 will shed light on what difficulties students encounter as they move among levels of abstraction, the impact of these difficulties on their overall problem solving, and instructional approaches that might be used to help students more productively shift among levels of abstraction. Finally, if it can be established, via answering research question 3, that increasing focus on the abstraction involved in mathematics problem solving has the potential to increase students' overall performance in mathematics, teachers and administrators might be more inclined to implement abstraction instruction in elementary classrooms.

METHODS

Context

The data for this study was collected as part of the broader NSF-funded CT4EDU project, which is a Networked Improvement Community (NIC; Bryk, Gomez, & Grunhow, 2010) involving a School of Education in a large Midwestern university and a small cohort of elementary school teachers in a large, nearby intermediate school district. The focus of the project is to support the teachers in bringing computational thinking ideas into their mathematics and science instruction and to study how this effort impacts teachers' thinking as well as students' mathematics, science, and computer science attitudes and learning. Abstraction is one of the four key computational thinking ideas explored in the project.

Participants

A total of 204 fourth- and fifth-grade students from eight classrooms in five schools participated in this study. The students were in classes taught by one of the elementary school teachers participating in the broader project. The number of students who participated in each class, grade levels, and school demographics are given in Table 1.

Table 1
Summary of Student and School Characteristics

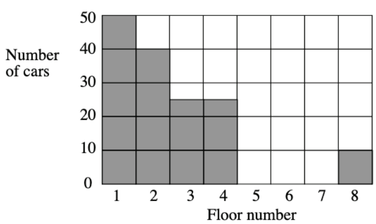
Classroom Characteristics			School Characteristics		
Class	Grade	Participating Students	% Non-white	% Free or Reduced Lunch	% English Language Learners
A	4	21	66.3	84.8	**
B	5	31			
C	5	18			
D	5	49	87.3	65.4	31.5
E	4	20	57.2	76.0	3.9
F	5	18			
G	4	25	41.8	62.5	**
H	5	22	26.3	84.8	75.9

**Percent ELL for these schools was not available.

Task

Students completed a task called *Parking Cars* (Mathematics Assessment Resource Service [MARS], 2007), shown in Figure 1. The task presents a scaled bar graph showing the number of cars parked on each of eight floors of a parking garage. This task is aligned with a Grade 3 content standard (3.MD.3) from the CCSS-M: "Draw a scaled picture graph and a scaled bar graph to represent a data set with several categories. Solve one- and two-step 'how many more' and 'how many less' problems using information presented in scaled bar graphs" (CCSSI, 2010, p. 25). It is also aligned with the practice standards *Reason abstractly and quantitatively* and *Attend to precision* (CCSSI, 2010). We chose to use a third-grade task with fourth and fifth graders in hopes that most students would attempt the task and reveal useful information about their thinking and strategies.

This bar chart shows the number of cars parked in the multi-level parking garage at a shopping center today. Each floor holds up to 50 cars.



1. On which floor are the most cars parked? How many cars are on this floor?
2. On which two floors are the same number of cars parked?
3. How many more cars are parked on Floor 1 then on Floor 8?
4. How many cars, in all, are parked on the parking garage? Show how you figured this out.
5. Fifteen more cars come into the parking garage. Show these cars on the graph, parking them in the lowest floors. Explain why you parked the cars in this way.

Figure 1. The parking cars task (MARS, 2007©; Reuse under Creative Commons License).

Questions 1–3 of the task focus on simple graph reading and have single numerical answers. Questions 4 and 5, by contrast, are more complex problems and ask students to show their work and/or explain their thinking. We chose the Parking Cars task, in particular, because we felt Questions 4 and 5 required shifts in levels of abstraction. Three pertinent levels of abstraction for this task are shown in Figure 2. At the highest level of abstraction is the overall problem context. At the lowest level are strictly numeric representations and strategies. In the middle is the bar graph, which serves as a mathematical model of the situation. Potential shifts in the level of abstraction are represented by block arrows.

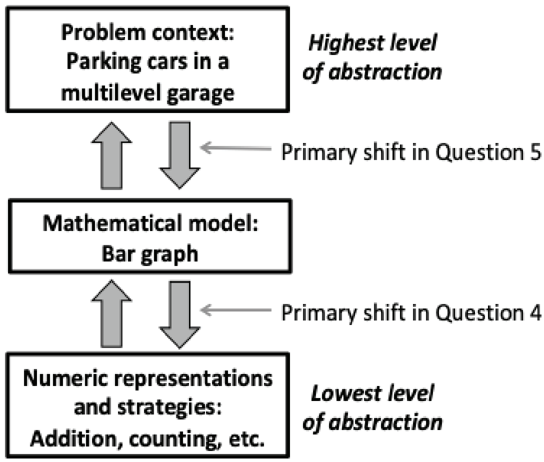


Figure 2. Three levels of abstraction involved in Parking Cars task

To answer Question 5, students had to identify the requirements of 15 cars and parking on the lowest levels as the important elements of the question's requirements and assign meaning to that information in the context of the graph—a shift from the highest level of abstraction to the middle level. To answer Question 4 (How many cars are parked in the garage?), students had to identify the pertinent information embedded in the graph and re-represent and manipulate that information to find a total (shift from the middle level to the lowest level). For these reasons, although we report student performance on the first three questions, this analysis focuses mostly on Questions 4 and 5.

Procedure

Teachers administered the mathematics task to their students near the end of the 2017-2018 school year, shortly after the broader project began. The teachers had experienced an early introduction to computational thinking, but had not yet begun using computational thinking or abstraction ideas in their instruction.

Analysis

All student work was de-identified and assigned an identification number before analysis began. We began by scoring students' responses to the first three questions on the task as correct or incorrect.

Next, the first author sorted the student work according to the strategy applied in Question 4. As she developed categories of strategies, she iteratively developed codes to describe the strategies in terms of the way students applied abstraction to identify information shown in the graph relevant to finding the total number of cars and formulate finding the total as a strictly mathematical task. The four shaded rows in Table 2 provide names and descriptions of the four identified strategies: Adding the bar heights, Counting the shaded squares, Reproducing the graph, and making a Literal drawing. Below each shaded row, each strategy is broken down into an abstraction step and a mathematics step (as applicable).

To consider accuracy of application of these strategies, the first author subsequently developed codes to describe categories of errors students made as they applied each strategy, categorizing the errors according to whether they were made during the abstraction step or the mathematics step. The error codes are also given in Table 2. Note that there are no error codes for the latter two strategies in Table 2. The reasons for this are described in the results section.

Table 2
Strategies, Abstraction Errors, and Mathematics Errors Used to Code Q4

Abstraction Step and Error Codes	**Mathematics Step and Error Codes**
STRATEGY: Adding bar heights. Student attempts to find total by adding heights of each bar.	
Abstraction step: *Read the bar heights.* ***Abstraction error codes:*** **Shifted Height:** Student records heights that do not match graph, but may reflect a compensation shift. **Misread Height:** Student records heights that do not match graph, and errors do not suggest a compensation shift. **Missing or Extra Addend:** Student records correct bar heights, but misses one or includes extra heights.	***Mathematics step:*** *Find sum of the bar heights.* ***Mathematics error codes:*** **Undetermined Error:** Student records an incorrect sum for their addends, but it is unclear where an error was made. **Place Value Error:** Student incorrectly combines digits of different place value. **Composition Error:** Student incorrectly combines pairwise sums (e.g., uses an addend twice).

Abstraction Step and Error Codes	Mathematics Step and Error Codes
STRATEGY: Counting. Student attempts to find the total by counting shaded squares.	
Abstraction step: *Determine how many cars a square represents.* ***Abstraction error code:*** **Scaling Error:** Student counts each square as 1 rather than as 10.	***Mathematics step:*** *Count shaded squares.* ***Mathematics error code:*** **Counting Error:** Student counts the number of shaded squares incorrectly (either by 1s or 10s).
STRATEGY: Reproduce graph. Student attempts to draw or otherwise reproduce the graph.	
Abstraction step: *N/A.*	***Mathematics step:*** *N/A.*
STRATEGY: Literal drawing. Student draws cars, parking spaces, or a picture of the garage.	
Abstraction step: *Create a representation of the overall problem context.*	***Mathematics step:*** *N/A.*

To examine students' shifts in abstraction in Question 5 (from the highest level, the problem context, to the middle level, the graph), the first author iteratively developed codes to describe how students' changes to the graph reflected the meaning they assigned to two important elements of the question, *15 cars* and *lowest level.* To examine how students interpreted 15 cars in the context of the graph, she first developed codes to describe how much shading they added to the graph, because correctly shifting this element of the problem context from the highest level of abstraction to the middle level requires using the graph scale to determine that 15 cars will be represented by 1.5 squares. These codes to describe the amount of shading are shown in the left column of Table 3. During coding, the first author also noted that a few students violated basic constraints of the graph, such as the requirement that shading build up from the bottom, and also violated the problem constraint of 50 cars per floor by adding shading beyond 50. These errors reflect difficulties in shifting from the problem context to the graph, and so were captured using the codes in the right column of Table 3.

Table 3
Codes for How Students Shifted "15 cars" from Problem Context to Graph

Codes for Amount of Shading	Codes for Violations of Graph and Context Constraints
Shaded 1.5 squares: Student added one and a half squares worth of shading to the graph. **Shaded 15 squares:** Student added 15 squares of shading to the graph, suggesting a scaling error. **Overshaded or Undershaded:** Student adds a different amount of shading to the graph, typically between 1.5 and 15 squares.	**Shaded from top:** Student added shading at the top of the graph. **Ignored maximum:** Student added shading beyond the 50-car limit for a floor.

To examine how students interpreted "lowest floors" in the context of the graph, the first author next developed codes to describe where students placed the shading on the graph, because correctly shifting this element of the problem context from the highest level of abstraction to the middle level requires identifying floors 2 and 3 as the lowest floors in the parking garage that are not full. The codes to describe the position in which students added shading to the graph are shown in the left column of Table 4. The first author also examined students' written explanations in Question 5 for why they placed the cars (or the shading) where they did and developed codes to describe ways that students appeared to be interpreting "lowest floor." These codes are in the right column of Table 4. During coding the first author noticed that a number of students had provided an explanation that referenced a practical consideration for the drivers, such as getting out of the parking garage more quickly. She added this code to the codebook as a separate category, and simply coded as whether a practical consideration was referenced or not.

Table 4

Codes for How Students Shifted "lowest cars" from Problem Context to Graph

Codes for Placement of Shading	Codes for Explanations of Lowest
Floors 1-3: Student added cars in some combination to floors 1-3. **Floors 5-8:** Student added shading on floor 8 (and 7, 6, and 5 if student made a scaling error). **Empty or near-empty floors:** Student added shading to empty floors or floor 8 (nonempty floor with fewest)	**Lowest as starting with 1:** Student's explanation reflects interpretation of Floor 1 as the lowest floor. **Lowest as starting with 8:** Student's explanation reflects interpretation of Floor 8 as the lowest floor. **Lowest as emptiest:** Student's explanation reflects interpretation of lowest to mean the floors with the fewest numbers of cars.

The initial coding scheme described in Tables 2-4 was presented to the third author, including a description of each code and one example piece of student work for each code. The third author used the codebook to code 20% of the data (randomly selected). After checking for initial inter-rater reliability, the first author made minor changes to the codebook descriptions and returned the codebook to the third author. They had a short conversation about the meaning of the codes without referencing any specific examples of student work. The third author then adjusted her codes based on the revised codebook. After this adjustment, the coders had high reliability, with an average Kappa of 0.81 across codes. This was deemed sufficient agreement to proceed with the first author's original coding.

The first author next calculated the percent of students with work reflecting each code for Questions 4 and 5 and wrote summaries and reflections on patterns across codes. She also mapped the various codes onto particular shifts in levels of abstraction. This analysis is presented in the section that follows.

RESULTS

The percentage of students who answered each part of the first three task questions correctly are shown in Table 5, aggregated by grade level. As might be expected, a higher percentage of fifth-grade students answered each question correctly than fourth-grade students, although this difference was very low in the case of Questions 1a and 1b. Questions 1a and 2, which involved holistic comparisons of bar heights, were answered correctly by the most students. Question 1b, which involved reading the height of a single bar on the graph scale, was answered correctly by slightly fewer students. Performance dropped markedly for Question 3, which involved reading two bar heights and operating on them.

Table 5
Student Performance on Basic Graph-Reading Questions

Grade Level	Percent of Students Answering Correctly			
	1a. On which floor are the most cars parked?	**1b.** How many cars are there on this floor?	**2.** On which two floors are the same number of cars parked?	**3.** How many more cars are parked on Floor 1 than on Floor 8?
Grade 4	93.94	78.79	81.82	46.97
Grade 5	94.20	79.71	89.13	65.22
Overall	94.12	79.41	86.76	59.31

This pattern of results suggested two preliminary conclusions that guided our analyses of responses to Question 4. First, the high levels of success on Questions 1 and 2 suggested that students have some successful strategies for mapping contextualized questions onto appropriate information in the graph (moving from the highest level of abstraction to the middle). As such, even though Question 4 required use of these same skills, we felt comfortable focusing our Question 4 analyses on discerning whether and what strategies students used for moving from the middle level of abstraction (mathematical model) to the lowest level (numerical representation). Second, students had greater difficulty with Question 3, which involved a shift in abstraction level—extracting the appropriate numeric information from the graph, a shift from the middle level to the low level—as well as a mathematical computation. It was unclear if students' difficulty with this question was more directly related to the abstraction or the mathematical computation. As such, we examined errors in abstraction and errors in mathematical computation separately when analyzing student responses to Question 4.

Question 4

Question 4 Strategies. A total of 105 students (51.47%) found the correct total of 150 cars for Question 4. This included 31 fourth graders (46.97%) and 74 fifth graders (53.62%). A total of 76% of students applied at least one of the four strategies (Adding bar heights, Counting, Reproducing graph, and/or Literal drawing; see Table 2) while attempting to solve the problem. The remaining 24% did not provide a response or provided a response we were unable to interpret. The four strategies are shown in Figure 3 mapped onto the relevant shift among levels of abstraction. The numbers of students who used each strategy are shown in Table 6.

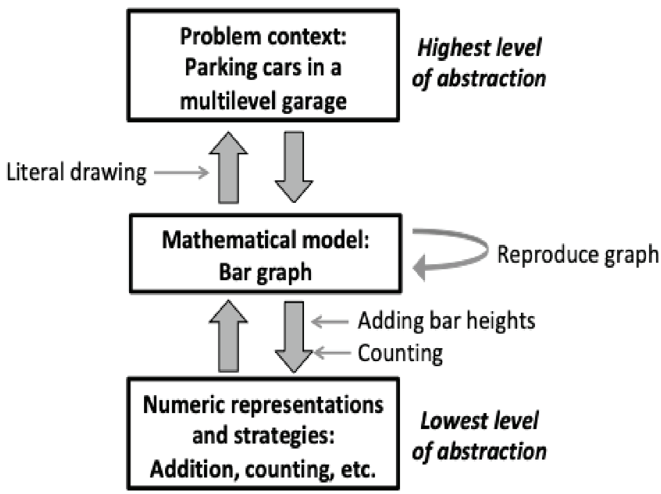


Figure 3. Question 4 Strategies Mapped onto Shifts in Level of Abstraction.

The two most commonly used strategies for finding the total number of cars had an abstraction step for shifting from the mid-level graph to a low-level mathematical representation, followed by a mathematical step. The *Adding bar heights* strategy was the dominant strategy at both grade levels. As described in Table 2, the abstraction step of this strategy involved reading and recording the heights of each of the five bars to transform the problem into an addition problem. The subsequent mathematics step involved adding the resulting numbers.

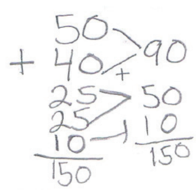
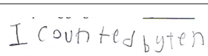
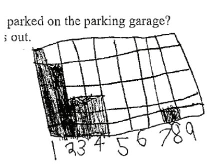
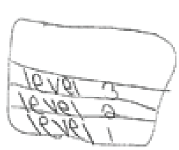
The second most common strategy, used by about 6% of students, was to count the shaded squares on the graph to find the total number of cars. As noted in Table 2, the abstraction step of this strategy involved determining the number of cars that each shaded square represented to transform the problem into a low-abstraction counting problem. The subsequent mathematics step was to count the squares (either by 10s or by 1s).

Students showed evidence of using two other strategies (reproduce graph and literal drawing) in service of answering Question 4. First, around

5% of students attempted to reproduce the graph as part of their answer to Question 4. We note that that this strategy suggests difficulty with selecting and extracting important information from the graph. It does not reflect a change in the abstraction level, but rather stays at the middle level of abstraction. Because there is no translation to a low level of abstraction, there is no mathematics step in this strategy. Second, around 5% of students made a literal drawing of the floors of the parking garage, arrangements of parking spaces, or cars themselves. While these literal drawings were not an immediate step toward finding the total number of cars, they do reflect a shift in abstraction level, from the middle level to the highest level. Students who made literal drawings may have been trying to make sense of the situation the graph represented, perhaps as a precursor deciding what information they needed to abstract from the graph. However, because the shift in abstraction is upward to the problem context rather than downward to a mathematical representation, there is no mathematics step in this strategy, either.

Table 6

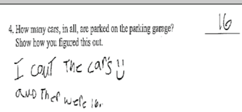
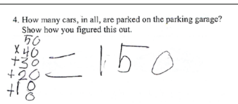
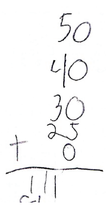
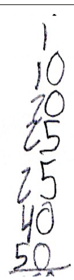
Number of Students Using Each of the Question 4 Strategies

Strategy Name	**Example**	**No. (%) of G4 Students**	**No. (%) of G5 Students**	**Overall No. (%) Students**
Adding bar heights		33 (45.45)	102 (73.91)	135 (64.71)
Counting		9 (13.64)	4 (2.90)	13 (6.37)
Reproduce graph		7 (15.15)	2 (1.45)	9 (4.41)
Literal drawing		7 (10.61)	5 (3.62)	12 (5.88)

Question 4 abstraction errors. As noted above, the *Adding bar heights* and *Counting* strategies described in Table 2 each involved an abstraction step followed by a mathematics step. We examined the student work for errors in each of these phases of problem solving to discern which parts of the problem-solving process students had difficulty with and to what extent. Table 7 summarizes the errors students made in the abstraction level shift involved in each strategy. For the *Counting* strategy, five fourth graders made a scaling error and counted each square on the graph as 1 car. The remaining eight students (4 fourth graders and 4 fifth graders) who used the counting strategy correctly abstracted that each square on the graph stood for 10 cars.

Table 7

Number of Students Who Made Various Abstraction Errors in Question 4

Error Name	**Example**	**No. (%) of G4 Students**	**No. (%) of G5 Students**	**Overall No. (%) Students**
Counting: Scaling error		5 (7.58)	0 (0)	5 (2.45)
Bar heights: Shifted height		4 (6.06)	13 (9.42)	17 (8.33)
Bar heights: Misread height		11 (16.67)	28 (20.29)	39 (19.12)
Bar heights: Missing or extra addend		6 (9.09)	15 (10.87)	21 (10.29)

About 43% of the students who attempted the *Adding bar heights* strategy correctly completed the abstraction level shift, as evidenced by recording all five bar heights as directly read from the graph. Three additional students wrote that they used addition to solve Question 4, but did not record any work that allowed us to evaluate their use of the strategy. The remaining students who used the *Adding bar heights* strategy made three kinds of errors. First, about eight percent of students recorded bar heights that did not match the heights on the graph, but may have reflected a mental compensation shift among bars. For instance, in the example shown in Table 4, the student recorded 20 and 30 as bar heights in lieu of recording 25 and 25. It is possible that these shifts were not abstraction errors, but we are unable to tell for certain based on our data and so report these shifts as possible abstraction errors. Nearly 20% of students recorded bar heights that did not match the graph and did not suggest a compensation shift. Lastly, around 10% of students either missed a bar height when recording them or added an extra bar height.

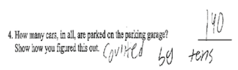
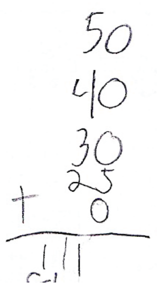
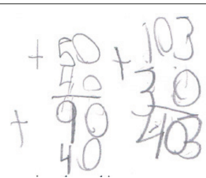
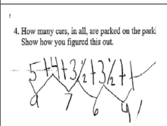
In summary, around 62% of the students who used the *Counting* strategy and 43% of the students who used the *Adding bar heights* strategy to solve Problem 4 completed the abstraction level shift involved in their chosen strategy correctly. The remaining students—38% of those who used the *Counting* strategy and 57% of those who used the *Adding Bar Heights* strategy—made an error during the abstraction shift from middle to lowest level.

Question 4 mathematics errors. Table 8 summarizes students' errors in the mathematics step of the *Counting* and *Adding bar heights* strategies. Four of the 13 students who attempted the *Counting* strategy made a counting error after deciding whether to count by 1s or 10s. Of the 135 students who attempted the *Adding bar heights* strategy, 113 (84%) executed the addition step correctly. The remaining 22 students (16%) had errors in their addition. For most of these students, their work did not reveal the specific nature of the error. In two cases, we were able to pinpoint an error relating to place value. In an additional two cases, we were able to pinpoint an error related to composition of pairwise sums.

Thus, roughly the same proportion of students who attempted the *Counting* strategy made mathematics errors as made abstraction errors. However, the vast majority of students who attempted the *Adding bar heights* strategy executed the mathematics step correctly. This is in stark contrast to errors in the abstraction step, which were made by roughly half of the students. To further explore the relationship between the mathematics errors and the abstraction errors, we examined the work of the 58 students who both recorded an incorrect total number of cars in Question 4 and provided work we were able to interpret in terms of any abstraction and mathematics errors. Of these 58 students, 15 (26%) made errors in both the abstraction and mathematics steps, 7 (12%) made errors only in the mathematics step, and 36 (62%) made errors only in the abstraction step. This further demonstrates the impact of abstraction difficulties on students' problem solving.

Table 8

Number of Students Who Made Mathematics Errors in Question 4

Error Name	Example	No. (%) of G4 Students	No. (%) of G5 Students	Overall No. (%) Students
Counting error		4 (6.06)	0 (0)	4 (1.96)
Addition: Undetermined error		7 (10.61)	11 (7.97)	12 (9.31)
Addition: Place value error		1 (1.52)	1 (0.72)	2 (0.98)
Addition: Composition error		0 (0)	2 (1.45)	2 (0.98)

Question 5

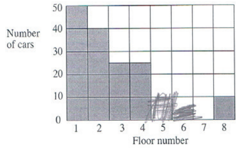
To further investigate these students' difficulties with abstraction, we examined their work on Question 5. We organize the results according to the two pieces of information that had to be shifted from the highest level of abstraction (the context) to the middle level (the graph): 15 cars and the lowest levels of the parking garage. Only 26 students (12.75%) added shading to the graph reflecting 10 additional cars on floor 2 and 5 additional cars on floor 3, which is the intended solution to the problem. To arrive at this solution, students had to discern how to represent 15 cars on the graph and decide where on the graph the lowest floors were represented.

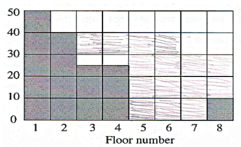
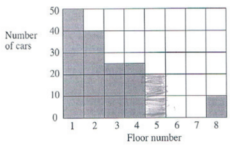
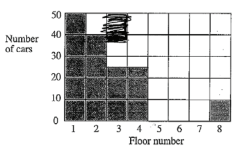
Question 5 representations of 15 cars. In order to translate the 15-car requirement to shading on the graph, students had to interpret the scale of the graph and determine that 15 cars would be represented by 1.5 squares. As shown in Table 9, about 45% the students in our sample did not make any change to the graph. Of the remaining 113 students, 79 (70%) added 1.5 squares of shading. Small numbers of students in each grade level added 15 squares of shading, suggesting an error in interpreting the graph scale. Additionally, less than 10% of students added a different amount of shading to the graph other than 1.5 or 15 squares, which we considered over- or under-shading that did not necessarily suggest a misinterpretation of the graph scale.

Even smaller numbers of students made other errors not related to the amount of shading that suggest difficulty in interpreting the graph. Three students added shading to the floor 1 column, despite that column already showing the graph's maximum of 50 cars. (Note that this maximum is also stated in the text of the task.) Two students added shading starting at the top of the graph, rather than building up from the bottom of the graph or the current level of a bar. This suggests a significant misunderstanding of how to read and interpret a bar graph.

Table 9

Number of Students Adding Each Amount of Shading in Question 5

Change	Example	No. (%) of G4 Students	No. (%) of G5 Students	Overall No. (%) Students
No change		34 (51.52)	57 (41.30)	91 (44.61)
Shaded 1.5 squares		19 (28.79)	60 (43.58)	79 (38.73)

Change	Example	No. (%) of G4 Students	No. (%) of G5 Students	Overall No. (%) Students
Shaded 15 squares		8 (12.12)	9 (6.52)	17 (8.33)
Over or Un-der shaded		5 (7.58)	14 (10.14)	19 (9.31)
Ignored maxi-mum		1 (1.52)	2 (1.45)	3 (1.47)
Shading from top		0 (0)	2 (1.45)	2 (0.98)

In summary, of the students who made a change to the graph, a vast majority made changes that reflected understanding how to use the graph scale to translate the 15-car problem requirement into 1.5 squares of shading—a successful shift from the highest level of abstraction to the middle. Even so, about 8% of students made a scaling error that reflected difficulty with this abstraction shift. Moreover, 45% did not make any change to the graph. We are unable to tell from our data whether this reflects lack of understanding or of time to complete the task.

Question 5 interpretations of the "lowest floors" constraint. The above analysis of shading on the graph did not take account of either the place on the graph where the shading was added or the explanation that students gave for this placement. In this section, we address students' placement of shading and explanations to gain insight into how they shifted the "lowest floors" constraint to a position on the graph. We analyzed students' explanations for their work on Question 5 with an eye toward evidence for their interpretation of the meaning of lowest.

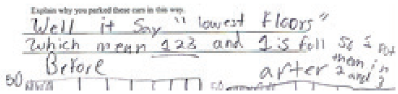
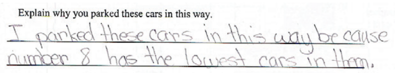
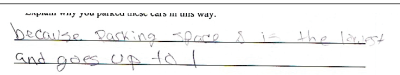
As shown in Table 10, about 13% of students did not provide any explanation, and an additional 39% provided an explanation that did not reference the lowest floor constraint. Interestingly, a significant number of students (14%) gave a practical reason (from the perspective of the drivers of the cars) for their placement, such as explaining that the drivers would be able to get out of the garage easier. This suggests that these students did not shift from the problem context to a lower level of abstraction, as shown in Figure 4.

Figure 4 also shows the three codes that reflect a shift from the problem context (highest abstraction level) to the graph (lower abstraction level). A total of 71 students (35%) provided an explanation that made some reference to the idea of lowest floors. Nine students referenced "lowest" in their explanations but did not say anything else that would allow us to infer how they interpreted it. The remaining 62 students provided explanations that suggested one of three interpretations of lowest. The highest number of students appeared to interpret floors 1, 2, and 3 as the lowest, as intended by the problem. A slightly lower number of students interpreted "lowest floors" to refer to the floors that had the fewest cars on them, including empty floors or floor 8, which was the nonempty floor with the lowest number of cars on it. Finally, three students appeared to interpret floor 8 as the lowest floor. Thus, although analysis of the student work revealed three interpretations that reflect a translation of "lowest floors" into a position on the graph, only one of those interpretations is the one intended by the problem. The remaining two, *Lowest starting with 8* and *Lowest as least cars,* could be considered abstraction errors.

Table 10

Number of Student Explanations Reflecting Each Interpretation of Lowest

Explanation code	Example	G4 Students	G5 Students	Overall Students
No explanation		13 (19.70)	13 (9.42)	26 (12.75)
No connection to lowest	Explain why you parked these cars in this way. I Would park the car in floor five Because 15 can go wet 25.	29 (43.94)	50 (36.23)	79 (38.73)
Practical consideration	Explain why you parked these cars in this way. So They Can get out easier.	8 (12.12)	20 (14.49)	28 (13.73)
Unclear reference to lowest	Explain why you parked these cars in this way. it is 50 uo 30 cars are the lowst	2 (3.03)	7 (5.07)	9 (4.41)

Explanation code	Example	G4 Students	G5 Students	Overall Students
Lowest starting with 1	 Explain why you parked these cars in this way. Well it Say "lowest Floors" which mean 123 and 1 is full Before after	8 (12.12)	25 (18.12)	32 (15.69)
Lowest as least cars	 Explain why you parked these cars in this way. I parked these cars in this way because number 8 has the lowest cars in them.	5 (7.58)	22 (15.94)	27 (13.24)
Lowest starting with 8	 Explain why you parked these cars in this way. because Parking space 8 is the lowest and goes up to 1	2 (3.03)	1 (0.72)	3 (1.47)

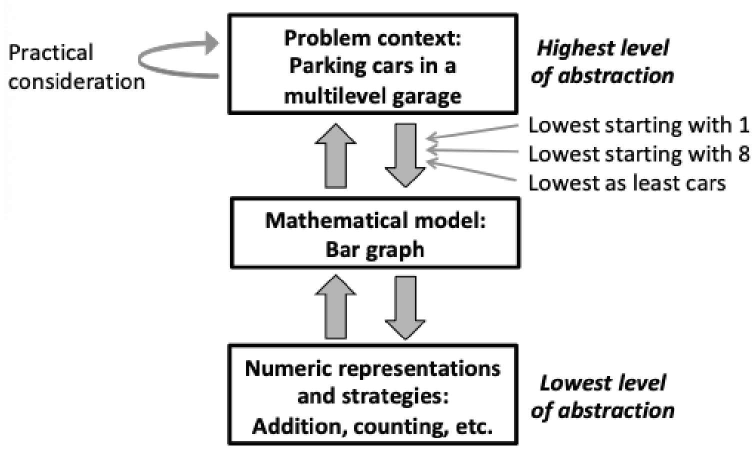


Figure 4. Lowest Floor Interpretations Mapped onto Abstraction Level Shifts.

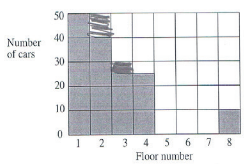
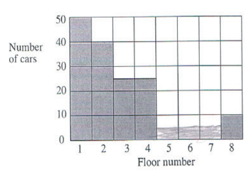
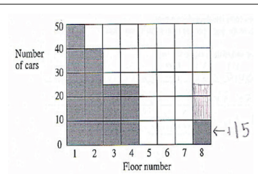
With only one exception, the students who provided an explanation that allowed us to infer their interpretation of lowest and also made changes to the graph made changes that matched their explanations. For example, all 32 students whose explanations reflected understanding of floors 1-3 as the lowest floors added shading only to those floors. Given this consistency, we examined the placement of shading on the graph for the students who provided no explanation or an explanation that did not elucidate an interpretation of lowest. The numbers of students who added shading consistent with each interpretation of lowest, with and without an accompanying explanation to support that interpretation, are shown in Table 11. For simplicity, we did not separate these by grade level. Note that students who added shading to floor 8 could have been seeing that floor as lowest because it had the fewest cars or because it was last on the graph, so we list the students who added shading to floor 8 without a supporting explanation to both of the latter two rows of Table 11.

Based on Table 10, it appears that roughly the same proportion (13-16%) of students interpreted "lowest floors" to be the bottom floors (1-3) as interpreted "lowest floors" to mean the floors with the fewest cars, with comparatively few interpreting floor 8 as the lowest floor. Table 11, however, suggests that more students (34% versus 19%) may have interpreted "lowest" in terms of the number of cars rather than the position of the floor in the garage.

In summary, students appear to have significant difficulty mapping the intended meaning of lowest floor onto a location on the graph. Although about 60% of students show some form of evidence (placement of shading, sometimes accompanied by an explanation) of attempting to translate the lowest floors requirement into a position on the graph, less than 20% of students showed evidence of translating it correctly.

Table 11

Patterns of Shading Reflecting "Lowest" Interpretations

Explanation code	**Example**	**No. (%) Students with Explanation**	**No. (%) Students without Explanation**	**Overall No. (%) Students**
Lowest starting with 1		32 (15.69)	7 (3.43)	39 (19.12)
Lowest as least cars		27 (13.24)	43 (21.08)	70 (34.41)
Lowest starting with 8		3 (1.47)	5 (2.45)	8 (3.92)

DISCUSSION

This study examined fourth and fifth graders' work on a mathematics task to discern when and how they shift among levels of abstraction during mathematics problem solving. Our chosen mathematics task required three such shifts: (1) transforming the graph into an addition, counting, or other mathematical problem in Question 4, (2) shifting the 15 cars requirement into an amount of shading on the graph in Question 5, and (3) shifting the lowest floor requirement into a position on the graph, also in Question 5. We organize our discussion around three main findings that correspond to our three research questions. The discussion of each finding connects each question and its results to implications for how abstraction instruction might best take place in elementary mathematics classrooms and what needs there may be for teacher education.

Finding 1: Fourth and fifth grade students make shifts in levels of abstraction as they solve mathematics problems.

Our first research question was: How do fourth- and fifth-grade students move between levels of abstraction during mathematics problem solving? Our analysis revealed that a significant number of students showed evidence of shifting among levels abstraction. In Question 4, 65% of students shifted from the middle-level bar graph to a low-level mathematics representation by transforming the problem of finding the total to the addition of bar heights. An additional 6% of students made the same shift by transforming the interpretation of the graph into a counting problem. In Question 5, 38% of students added shading to the graph that suggested a successful shift from the high-level problem context to the graph via evidence of understanding of the graph and its scale, with an additional 9% making only minor over- or undershading errors that may reflect carelessness rather than mistakes in abstraction. Lastly, 35% of students provided an explanation in Question 5 that reflected an attempt to abstract the meaning of the "lowest floor" constraint and transform it to a particular position for shading on the graph. An additional 27% added shading in a location consistent with one of the interpretations found in the corpus of student answers, although these 27% did not corroborate this reasoning in their explanations. In sum, at least a third of students attempted to shift among abstraction levels in all three of the aspects of the task identified above. This was the case even though these classes did not receive any explicit instruction in abstraction.

The purpose in answering research question 1 was to establish whether or not there are opportunities for addressing abstraction through work on elementary mathematics tasks. Our results replicate and extend the findings of Hazzan and Zazkis (2005) in showing that elementary students, like middle school students, do shift among levels of abstraction, either consciously or unconsciously, as they solve mathematical problems. Further, we established that elementary students use multiple strategies when shifting abstraction levels. For example, in Question 4, transforming the problem of finding the total number of cars into an addition problem was one way to shift to a lower level of abstraction, while transforming the problem into a counting problem was another. This establishes that elementary mathematics problem solving presents opportunities for discussing abstraction with students.

Given that students have strategies for shifting the level of abstraction when reading and interpreting graphs, it would be fruitful to explore what other types of mathematical tasks prompt elementary school students to shift levels of abstraction. When common shifts made by students are identified, they could be leveraged to integrate preliminary instruction about abstraction into elementary mathematics courses. Successful leverage of these shifts in abstraction will also require that teachers be educated on what levels of abstraction are, how to recognize when students are making shifts among levels, and how to effectively discuss those shifts in the classroom. Such teacher education efforts would allow teachers to explicitly call out shifts in levels of abstraction to students, as advocated Armoni (2013). Educating teachers to recognize shifts in levels when made spontaneously by students may be particularly important in light of our finding that a small number of students made literal drawings of the situation when working on Question 4 (see Figure 2). This was an unexpected shift in abstraction not explicitly required by the problem. Such a shift could reflect productive preliminary sense-making, or problematic abstraction shifts. If teachers were able to discuss these shifts in the classroom, it would provide opportunities to highlight student thinking and potentially remedy difficulties.

Finding 2: Although they have starting points for thinking about abstraction, many students struggle with making abstraction shifts.

Our second research question was: To what extent are students successful at carrying out the abstraction steps of their problem-solving strategies? What errors do they make? Our results show that although many students attempted to use abstraction, they made several kinds of errors when doing so. About 38% of students who transformed Question 4 into a counting problem made an abstraction error by misinterpreting the scale, and a little more than half of students who transformed Question 4 into an addition problem made an abstraction error when reading or recording the bar heights. Moreover, about six percent of students attempted to answer Question 4 by reproducing the graph, suggesting they did not have an abstraction strategy that allowed them to get started on the problem. In Question 5, a small number of students (less than 10%) made shading errors that reflected lack of understanding of the graph scale, its maximum, or its general manner of representation, suggesting serious difficulty in interpreting the graph as an abstraction and translating the 15-car requirement into an amount of shading.

Lastly and most strikingly, only 15-20% of students showed evidence of mapping the lowest-floor constraint onto the graph in the way intended by the problem (with floors 1-3 as the lowest). Half of students did not provide any evidence of attempting to operationalize the lowest floor constraint. At least 13%, and as many as 34%, of students interpreted lowest floors to mean those with the fewest cars on them, which could also sometimes reflect understanding lowest to refer to the heights of the bars on the graph.

The purpose in answering research question 2 was to shed light on what difficulties students encounter as they move among levels of abstraction, the impact of these difficulties on their overall problem solving, and instructional approaches that might be used to help students more productively shift among levels of abstraction. Our data provides clear examples of how errors in abstraction can impact students' success with problem solving. For example, students who made a scaling error when using the *Counting* strategy to solve Question 4 produced answers that were incorrect by a factor of 10, and students who misread the bar heights while using the *Adding bar heights* strategy often produced incorrect totals (see Table 7). The high amount of errors in abstraction illustrates that bringing abstraction instruction into elementary school will take more than simply making all shifts in abstraction level explicit. Students will need guidance in evaluating the accuracy and effectiveness on their shifts. We see this as akin to Armoni's (2013) and Hassan's (2008) calls for students to be provided opportunities to reflect on their abstraction efforts. As of now, we know little about how elementary school students might think about shifts in abstraction level when introduced to the idea, how they might be best engaged in reflection, and how teachers might develop skills in helping students through these reflection experiences. Further research is needed to explore these questions.

For example, future research could explore whether and how students are able to identify abstraction errors in their own and other students' work when explicitly prompted to do so, and if and how they attempt to correct the errors once they discover them. The results of such studies could be used to devise instructional methods for abstraction that are responsive to student thinking.

Finding 3: Students made more errors with abstraction than with pure mathematics.

Our third research question was: How does students' level of success with abstraction compare to their level of success with the mathematics steps? Our results show that when students needed to engage both in abstraction and strictly mathematics steps to solve a problem, they made more mistakes during abstraction than when executing the mathematical steps. This is most evident in the analysis of Question 4. Whereas only 14% of students who used the *Adding bar heights* strategy made an error in addition, half of the students made an error when abstracting the numbers to add. Moreover, for the 58 students who produced an incorrect final answer to Question 4 and interpretable work, 62% made errors in abstraction but not in the pure mathematics step. This is in contrast to 12% who made mathematical errors only and 26% who made errors in both abtraction and mathematics.

The purpose in answering research question 3 was to explore whether increasing focus on the abstraction involved in mathematics problem solving had the potential to increase students' overall performance in mathematics. That the vast majority of errors students made were in abstraction steps suggests that instruction focused on abstraction has the potential to improve mathematics achievement. While additional research is needed to explore this idea, this finding suggests a potential avenue for encouraging teachers and administrators to implement abstraction instruction in elementary classrooms. Explicit instruction about shifting levels of abstraction may focus student attention on the aspects of the problem where they make the most errors. Moreover, providing teachers information about how to separate abstraction shifts from purely mathematical steps when examining student work could provide them with new ways of identifying and remedying sources of student difficulty.

CONCLUSION

This study serves as an illustrative example of how elementary school mathematics tasks ask students to make shifts among levels of abstraction and the ways in which students make those shifts. The current study has yielded the following key insights: (1) students implicitly shift among levels of abstraction when solving "real-world" mathematical problems; (2) students make errors when making those implicit shifts in abstraction level; (3) the errors students make in abstraction outnumber the errors they make purely mathematical skills. Further research should examine how levels of abstraction could provide a useful lens for examining other kinds of elementary mathematics tasks and student strategies for solving those tasks. Additionally, further research should examine elementary school teacher thinking about their students' work in relation to abstraction to better understand how instruction on abstraction might fit into the elementary school classroom.

Given the broad similarities between the importance (College Board, 2017; CCSSI, 2010) and difficulties (Hazzan, 2003; Hazzan & Zazkis, 2005) of levels of abstraction in mathematics and computer science, we believe this study may also have implications for computer science education. The CS education is field is increasingly exploring ways to connect computer science education with other disciplines (Schanzer, Fisler, Krishnamurthi, & Felleisen, 2015; Weintrop, 2016). A recent analysis of the CCSS-M for grades K-5 (Rich, Spaepen, Strickland, & Moran, 2019) suggested that elementary mathematics concepts offer opportunities to begin a spiral curriculum (Bruner, 1977) focused on computational thinking ideas, and that ideas first explored in elementary mathematics could be revisited later in computer science contexts. This paper has established that there are also opportunities in elementary mathematics for students to explore moving among levels of abstraction, another important computer science concept (Armoni, 2013). Furthermore, Rich, Yadav, and Schwarz (2019) leveraged elementary teachers' ideas about how computational thinking fit into their mathematics and science teaching to help the teachers integrate computational thinking in their classrooms. If explicit instruction in moving among levels of abstraction is taken up in elementary mathematics classrooms, research could similarly examine ways in which the ideas about abstraction elementary students develop in mathematics instruction can be leveraged during their later instructional experiences with computer science.

Given the importance and difficulty of abstraction for mathematics and computer science students alike, we hope this study spurs new lines of research and development that help all students gain access to these important and powerful ideas.

Acknowledgement

This work is supported by the National Science Foundation under grant number 1738677. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Armoni, M. (2013). On teaching abstraction in computer science to novices. *Journal of Computers in Mathematics and Science Teaching*, *32*(3), 265–284.

Bruner, J. S. (1977). *The process of education* (2nd ed). Cambridge, MA: Harvard University Press.

Bryk, A. S., Gomez, L. M., & Grunow, A. (2010). Getting ideas into action: Building networked improvement communities in education. *Carnegie Foundation for the Advancement of Teaching,* Stanford, CA. Retrieved from http://www.carnegiefoundation.org/spotlight/ webinar-bryk-gomez-building-networked-improvement-communities-in-education

College Board. (2017). *Course and exam description: AP computer science principles including the curriculum framework.* Retrieved from https://apcentral.collegeboard.org/pdf/ap-computer-science-principles-course-and-exam-description.pdf

Common Core State Standards Initiative [CCSSI]. (2010). *Common Core State Standards for Mathematics.* Retrieved from http://www.corestandards.org/Math/

Greer, B. (1997). Modelling reality in mathematics classrooms: The case of word problems. *Learning and Instruction*, *7*(4), 293–307.

Hazzan, O. (2003). How students attempt to reduce abstraction in the learning of mathematics and in the learning of computer science. *Computer Science Education*, *13*(2), 95–122.

Hazzan, O. (2008). Reflections on teaching abstraction and other soft ideas. *ACM SIGCSE Bulletin*, *40*(2), 40–43.

Hazzan, O., & Zazkis, R. (2005). Reducing abstraction: The case of school mathematics. *Educational Studies in Mathematics*, *58*(1), 101–119.

Hillis, W. D. (1998). *The pattern on the stone.* New York: Basic Books.

Kramer, J. (2007). Is abstraction the key to computing? *Communications of the ACM*, *50*(4), 37–42.

Mathematics Assessment Resource Service [MARS]. (2007). *Parking cars.* Performance assessment task retrieved from the *Inside Mathematics* web site: http://www.insidemathematics.org/assets/common-core-math-tasks/parking%20cars.pdf

Palm, T. (2008). Impact of authenticity on sense making in word problem solving. *Educational Studies in Mathematics*, *67*(1), 37–58.

Rich, K. M., Spaepen, E., Strickland, C., & Moran, C. (2019). Synergies and differences in mathematical and computational thinking: Implications for integrated instruction. *Interactive Learning Environments*. DOI: 10.1080/10494820.2019.1612445

Rich, K. M., Yadav, A., & Schwarz, C. V. (2019). Computational thinking, mathematics, and science: Elementary teachers' perspectives on integration. *Journal of Technology and Teacher Education, 27*(2), 165-205.

Schanzer, E., Fisler, K., Krishnamurthi, S., & Felleisen, M. (2015, February). Transferring skills at solving word problems from computing to algebra through Bootstrap. In *Proceedings of the 46th ACM Technical symposium on computer science education* (pp. 616-621). New York: ACM.

Statter, D., & Armoni, M. (2016, October). Teaching abstract thinking in introduction to computer science for 7th graders. In *Proceedings of the 11th Workshop in Primary and Secondary Computing Education - WiPSCE '16* (pp. 80–83). New York: ACM.

Verschaffel, L., & De Corte, E. (1997). Teaching realistic mathematical modeling in the elementary school: A teaching experiment with fifth graders. *Journal for Research in Mathematics Education*, *28*(5), 577–601.

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