Growing Mindsets: Debugging by Design to Promote Students' Growth Mindset Practices in Computer Science Class

Luis Morales-Navarro, University of Pennsylvania, luismn@upenn.edu
Deborah A. Fields, Utah State University, deborah.fields@usu.edu
Yasmin B. Kafai, University of Pennsylvania, kafai@upenn.edu

Abstract: Mindsets play an important role in persevering in computer science: while some learners perceive bugs as opportunities for learning, others become frustrated with failure and see it as a challenge to their abilities. Yet few studies and interventions take into account the motivational and emotional aspects of debugging and how learning environments can actively promote growth mindsets. In this paper, we discuss growth mindset practices that students exhibited in "Debugging by Design," an intervention created to empower students in debugging—by designing e-textiles projects with bugs for their peers to solve. Drawing on observations of four student groups in a high school classroom over a period of eight hours, we examine the practices students exhibited that demonstrate the development of growth mindset, and the contexts where these practices emerged. We discuss how our design-focused, practice-first approach may be particularly well suited for promoting growth mindset in domains such as computer science.

Keywords: debugging, motivation, growth mindset, computer science, e-textiles

Introduction

Debugging—the process of identifying and fixing problems and errors that prevent a program from working as expected—is an inevitable and intrinsic part of learning computer programming. Furthermore, it is an essential computational thinking practice (e.g., Brennan & Resnick, 2012) that is often overlooked in K-12 classrooms (McCauley et al., 2008). As Papert (1980) remarked, "errors benefit us because they lead us to study what happened, to understand what went wrong, and, through understanding, to fix it" (p. 114). However, for some students, encountering bugs can generate emotional responses such as fear and anxiety that lead to disengagement and the avoidance of computer programming (Scott & Ghinea, 2013). Prior studies provide ample evidence of how mindsets play an important role within computer science (Burnette et al., 2020, Nolan & Bergin, 2016), as students with growth mindsets perceive bugs as opportunities for learning, and those with fixed mindsets become frustrated with failure and see it as a challenge to their intelligence. Thus, in computer science (CS), promoting growth mindset in the classroom is important since debugging is such an essential practice, and persevering through debugging is critical to students' learning.

Debugging is an area that is particularly difficult for students to learn and challenging for teachers to teach (McCauley et al., 2008). Yet, few studies provide a holistic approach to debugging that takes into account the "inextricable relationship between thinking and emotion" in learning how to persevere and handle failure (DeLiema et al., 2020, p. 210). Traditional pedagogical approaches to debugging tend to involve, highly limited, linear strategies to finding bugs in code (e.g., Silva, 2011) in constrained programming contexts that provide well-timed feedback (e.g., Luxton-Reilly et al., 2018). These approaches usually include only bugs designed by instructors, limiting students' agency over bugs. Further, they only attend to isolated, cognitive aspects of debugging in limited scenarios. In contrast, positioning learners as designers of bugs for their peers could give them greater power over bugs and allow them to be creators and not just solvers. We propose that having students design buggy (rather than functional) computational artifacts for their peers to solve may help promote students' growth mindsets in practice.

In this paper, we examine the growth mindset practices that students exhibited when creating buggy electronic textiles (hereafter e-textiles) projects for their peers to solve and, subsequently, fixing each other's projects. We report on four cases of student groups engaging in our implementation of "Debugging by Design" (DbD), an open-ended design learning activity, in a high school CS classroom over a period of eight hours (Fields & Kafai, 2020). In DbD students work with e-textiles—which involves creating wearables using programmable microcontrollers, sensors and actuators that can be sewn into fabrics (Buechley et al., 2013). Making an e-textiles project not only includes designing functional circuits but also writing code that controls interactions—thus, providing multiple opportunities for bugs in crafting physical artifacts, designing circuits and programming (Searle et al., 2018). Early evidence suggested that when students consciously created bugs, they improved their abilities to detect and fix bugs, collaboratively solved problems, and increased their confidence in debugging

(Fields et al., in press). As such, we observed students to see if and when they demonstrated growth mindset practices. Drawing on the analysis of student team interactions in designing bugs for others; we address the following research questions: (1) What practices do students exhibit that demonstrate the development of growth mindset? (2) When do growth mindset practices emerge during the DbD unit?

Background

Our work builds on longstanding motivational research on the impact of student self-beliefs in learning. Dweck (2006) organized opposing self-beliefs on the capability to develop competencies and qualities in what she calls growth and fixed mindsets. Learners with a growth mindset, those who believe their competencies and capacity can change and be developed over time, have higher resilience in challenging environments than those with fixed mindsets (Yeager & Dweck, 2012). In CS education, growth mindset has gained relevance as learning to code is challenging and encountering bugs is inevitable. For instance, Murphy and Thomas (2008) argue that in computing, while students with a growth mindset perceive bugs as opportunities for learning, those with fixed mindsets become frustrated with failure and see it as a challenge to their intelligence. This resonates with early debugging research which identified that while some learners see bugs as inherent and exciting challenges of learning to code, others become frustrated, perceiving bugs as negative reflections of their performance (Perkins et al., 1986). Indeed, among novice learners, encountering bugs and trying to fix them can generate feelings of helplessness, frustration and anxiety, and promote a fixed mindset (Scott & Ghinea, 2014; Nolan & Bergin, 2016). Furthermore, these feelings and self-beliefs can lead to disengagement, attrition, and the avoidance of computer programming (Scott & Ghinea, 2013; Margolis et al., 2017).

The existing, but limited, research on growth mindset interventions in CS education has yielded promising results that suggest that interventions seem to increase interest, particularly among novices (Burnette et al., 2019). Most of these interventions happen within learning environments, in which teachers delivered information about growth mindset research through videos, lectures, and readings followed by reflective writing and "saying is believing" exercises (Simon et al., 2008; Cutts et al., 2010; Rangel et al., 2020). Other interventions also included testimonials of professional role models that talked about growth mindset and offered tips for success (Burnette et al., 2019; Quille & Bergin, 2020). However, all of these approaches prioritize telling students about growth mindset rather than facilitating student experiences that support its development in practice. Only one intervention proposed the use of a problem-solving checklist for students to track their progress as they programmed projects (Loksa et al., 2016).

Noticeably absent from current research are interventions that actively promote and examine growth mindsets in action. Some attention has been given to how to integrate mindset research in the classroom. For instance, Campbell and colleagues (2020) identified practices, which they call behaviors, that characterize growth mindset in learning activities and situate them within larger learning theories and Haimovitz and Dweck (2017) examined how adult and socialization practices can foster growth and fixed mindsets in children. Building on these works, we investigate growth mindset in action by analyzing how students practice it while engaging in an intervention that centers around debugging. In DbD, we promote growth mindset by putting learners, rather than teachers or researchers, in charge of creating intentional problems and turning bugs into a feature of the learning product rather than a stumbling block. Our research of DbD in this paper focuses on the growth mindset practices that students exhibited when designing bugs for others and solving bugs, as well as the situations where these practices emerged.

Methods

Context and participants

We situated our DbD activities within the e-textiles unit of Exploring Computer Science (ECS), an inquiry-based CS curriculum that is committed to broadening participation in computing through a building talent approach that addresses the structural inequities and beliefs systems that limit participation from historically marginalized groups (Margolis et al., 2017). Within ECS, the promotion of a growth mindset among teachers and students plays an important role, as it puts equity first by considering that all students with access to quality education can grow in engagement and capacity (Margolis et al., 2017). During the e-textiles unit, students work on four open-ended, interest-driven physical computing projects; a paper card with circuits, a bracelet with parallel circuits, a collaborative class mural that has light patterns controlled with switches and finally a hand-crafted sensor to control light effects (see Fields & Kafai, 2020). In these open-ended physical computing activities—that integrate coding, building circuits, and crafting—facing failure and unanticipated challenges is expected and inherent in the learning process (Searle et al., 2018). Students have to think through the distributed nature of physical computing, where problems are spread across different modalities, and must iteratively identify bugs, generate solutions, and run tests (Searle et al., 2018).

DbD was intentionally situated between the third and fourth projects in the E-Textiles unit, after the collaborative class mural described above. This allowed students to bring their experiences with bugs from earlier projects to their DbD designs, and to apply any knowledge gained from DbD to their final project. The activity was designed to take place during eight 50 minutes class sessions over a period of two weeks. At the beginning of the activity students discussed with their partners and the whole class different errors and problems they encountered when working on e-textiles projects. Then, student groups came up with a list of bugs they wanted to include in their designs. After receiving approval from their teacher, students began working on their buggy designs, or DebugIts. Once they finished the buggy projects, they exchanged them with their peers and spent one class period debugging their peers' projects before presenting their solutions to the class.

In Spring 2019, Ben, a teacher with three years of experience in e-textiles that helped co-develop the curriculum, implemented DbD with his students at a high school located in a metropolitan school district in the West Coast of the United States. The class included 11 girls and 14 boys between 14 and 18 years old. Of these students, 72% spoke a language other than English at home, 80% had no previous CS experience, and 80% had family members with at least some college education. The class was ethnically and racially diverse, with 48% of students identifying as Latinx, 36% as Asian American/Pacific Islanders, 8% White, 4% other, and 4% not reporting their race/ethnicity. From the class, four collaborative groups were selected by the teacher for further study in order to represent a range of student interaction and performance including two groups of two students (Evelyn and Nicolás, and Liam and Sophia) and two groups with three students (Lucas, Emma and Lily, and Georgia, Gabriel and Camila) (all participant names in the paper are pseudonyms).

Data collection and analysis

To investigate the growth mindset practices that students exhibited and the situations, challenges or contexts where these practices seemed to emerge; data for analysis was drawn from in-class videologs and observations (recorded in daily field notes) of participants working on their projects as well as interviews (n=10 students). This multiple-source approach provides a fuller picture of what happened in the classroom and the growth mindset practices that students exhibited. Data sources were sorted by date, student group of interest and type.

Data was analyzed in two steps. First, a chronological reading of sources from two student groups focused on inductive descriptive coding (Saldaña, 2013), through which a coding scheme for moments of interest where students exhibited growth mindset practices was developed. These descriptive codes were then clustered into similar categories to detect patterns and interrelationships. This process of clustering was informed by prior theories and previous research on mindset practices and behaviors (Campbell et al., 2020; Dweck, 2006; Dweck, 2000). Following, a coding scheme was developed and discussed with two researchers familiar with the data. After receiving feedback, the coding scheme was revised. In the second round of analysis the coding scheme was applied across all groups through a deductive reading.

Findings

Our analysis in this paper focuses on the growth mindset practices that students engaged in while planning, designing, and solving their DebugIts. This allowed us to see intersections between the cognitive challenges in designing code, circuit and crafting bugs and the motivational and emotional challenges often associated with growth and fixed mindset. We identified five emergent growth mindset practices or behaviors. These practices include: 1) choosing challenges that lead to more learning, 2) persisting after setbacks, 3) giving and valuing praise for effort, 4) approaching learning as constant improvement, 5) developing comfort with failure. While the first four practices coincide with the practices proposed by Campbell and colleagues (2020), the last practice emerged inductively from videolog coding. In the following sections we introduce these five practices, providing examples for each, followed by a case study of how they emerged in the context of one team, and then conclude with an overview of how these practices were distributed across four teams of students.

Growth mindset practices

Choosing challenges that lead to more learning

Throughout DbD, students purposefully chose challenges that led them to learn new things, "throwing themselves wholeheartedly into difficult tasks—and sticking with them" (Dweck, 2000, p. 3). As an example, when making a DebugIt, Emma, a student with very little sewing experience sewing, decided to take on the challenge of sewing a complicated project that required creating one circuit across two different heart-shaped pieces of felt, with lights on one sheet and most of the sewn circuitry on the other sheet. As she struggled sewing the microcontroller on one sheet of felt and connecting it to the LED lights on the other sheet, Emma reflected, "probably I should do more craft," a recognition of her need for more expertise in sewing (Lesson 3, Video). Yet even knowing she had little experience in this area, Emma was the one that proposed arranging the electronic components in this spatially complex way and, mischievously, to embed a short circuit between the two layers of felt. This is just one of many examples of ways that students consciously chose challenges that led them to deeper learning.

Persisting after setbacks

Setbacks are a normal part of debugging that can cause frustration and at the same time provide opportunities for students to persist. For this practice we identified two subcategories: preceding frustration and persistence. Frustration as part of the learning process is meaningful because it creates opportunities for students to persist and, as Dweck (2006) writes, "to turn an important setback into an important win" (p. 93). We observed how frustration sometimes precedes persistence and other times co-occurs in instances of persistence. As an example of frustration preceding persistence, when fixing a circuitry bug in her sewing, Evelyn exclaimed: "This is too much… I don't know what's wrong with it." She was upset and frustrated. Her partner, Nicolás, encouraged her, but she insisted she didn't know how to fix it. Nicolás offered to help, "déjame [let me] look at it" (Lesson 7, Video). Twenty minutes later, and after working on other aspects of the project, Evelyn persisted and continued folding and bending the artifact to get it to work until the lights turned on. During this process, Evelyn discovered that the problem was in loose connections of the conductive thread. "I could have fixed it a long time ago, if I knew the connections were all loose" she reflected (Lesson 7, Video). On a different instance, Evelyn seemed upset dealing with a bug and Nicolás encouraged her: "You can fix it! Fix it!" (Lesson 7, Video). In this instance, frustration and persistence co-occurred. Despite her frustration, Evelyn kept sewing and managed to get the lights to turn on. Thus, with encouragement from her partner, as Evelyn persisted, she evaluated and analyzed the problems, looking for alternative ways to solve them. These two examples illustrate the occurrence and the co-occurrence of frustration and persistence; and how setbacks, although frustrating, became opportunities for learning in the process of creating and solving bugs.

Giving and valuing praise for effort

While working on their DebugIts, students praised each other's efforts, strategies and process. This kind of praise fosters mastery-oriented responses when facing difficult problems (Dweck, 2000). In one instance, after finding several unintentional problems in her circuit Emma reflected and suggested a possible solution: "I always forget to tie the knots, it's so annoying, I better tie them as I go along." Lucas praised her proposed strategy and the process of getting to it by saying, "Smart thinking!" (Lesson 3, Video). In a different instance, in Emma's group, praising and encouraging effort demonstrated what Love (2019) calls cultural legacy, that is the traditions and customs of culture, in self-beliefs. Emma had to add a short circuit bug in which two threads intersected. Her teammates, Lucas and Lily, were worried they would finish on time. "What time is it? Do you think I'll finish?," Emma asked. In response, Lily and Lucas started chanting Dolores Huerta's "¡Sí se puede!" (Yes we can) to encourage and praise Emma's effort (Lesson 4, Video). The "¡Sí se puede!" chant, which emerged in the 1970s during the civil rights movement for farm and immigrant worker rights in the Southwestern United States, is an expression for encouraging effort and persistence with great cultural and historical significance in Latinx communities. This praise and encouragement from peers for effort promotes growth mindset by framing setbacks and challenges as inherent aspects of learning that can be overcome with persistence.

Approaching learning as constant improvement

Student teams also embraced learning as constant improvement and looked for ways to improve their projects and their competence. As Dweck (2000) argues, this reflects a desire to learn, get smarter, and acquire new skills. We see this when, after Lucas finished writing the code and reported that he had made all the suggested changes, Emma responded by saying, "let me see what other things we can do" and proposed several ways they could improve the project (Lesson 3, Video). In a complementary episode later, after Emma tested whether the stitches and knots were strong enough by turning the project upside down, Lucas suggested they checked the wires to make sure they were not touching, taking on the role of looking for opportunities for improvement (Lesson 3, Video). In these two examples we can see how students supported each other in seeking opportunities to improve their performance and saw the goal of learning as constant improvement and not just showcasing or achieving a certain level of performance.

Developing comfort with failure

Developing comfort with failure generates opportunities for learners to embrace and overcome mistakes with proper motivation and guidance (Dweck, 2000). When learners are worried about failure they are more likely to develop a fixed mindset and question their ability while learners with a growth mindset "see their own failures as problems to be solved, and they see other people's failings that way as well" (Dweck, 2000, p. 88). In DbD, we identified that students were comfortable with failure particularly by not being afraid of being wrong when: 1) asking questions and requesting feedback, 2) sharing failure with their peers, and 3) embracing failure and imperfection as part of the process. Sophia, for example, not afraid of being wrong when sewing a circuit, asked Liam "Can positive and positive cross each other?" (Lesson 1, Video). This is one of many instances in which students constantly requested feedback from peers without any fear of being wrong. Further, when learners encountered failure while working on their projects, it was not uncommon for them to openly share failure with their peers. As an example, Lucas shared his coding mistakes saying "Oh! I messed up! Wrong in the coding!" (Lesson 3, Video), while Evelyn shared "Oh my god! I was going to keep going without sewing up the light!" when she noticed a gap in sewing and laughed at her own mistake (Lesson 4, Video). These are instances of seeing failure as something worth sharing. Furthermore, students embraced imperfection and failure as a feature of their projects—separate from the intentionally designed bugs. For instance, when Emma noted that the two heart-shaped pieces of felt of the group's project were not perfectly aligned she described them as "a little bit off, but it's okay". She also went on to offer a simple, imperfect solution, saying "we'll just cut it off, it doesn't have to be perfect" (Lesson 4, Video). These are some examples of how students were not afraid of being wrong, "messing up," making mistakes and or having imperfect projects. Developing comfort with failure helped students reframe failure as a setback they could overcome and as opportunity for learning.

A case study of growing mindsets

One case study group—Georgia, Gabriel and Camila—illustrates how the five growth mindset practices described above came up in the context of the DbD unit. Growth mindset practices appeared from the very beginning of the group's work. As the group brainstormed what they wanted to make for their DebugIt and the bugs they wanted to include (see Figure 1), Camila suggested an area that went beyond the group's current expertise: making a piano and coding music. Gabriel hesitated to agree, "we've never used music before and we have two days" (Lesson 2, Video): they had not learned about coding music previously in the class and did not have much time to figure it out by themselves, but they did have access to a supplementary instructional guide on music with e-textiles. Ben, their teacher, who had overheard the conversation about coding music, encouraged the team to do it and even to create a bug related to music. They decided to play Twinkle, Twinkle, Little Star and to have Gabriel code it. This required understanding how to code notes by defining pitch and duration in numerical values with new commands in Arduino. Although Gabriel shared with his peers that he had "no clue how to code music," he took charge of this new challenge and learned about frequencies and tempo with an instructional guide. Of note, Gabriel's choosing to take on this new challenge happened in the context of both peer and teacher support.

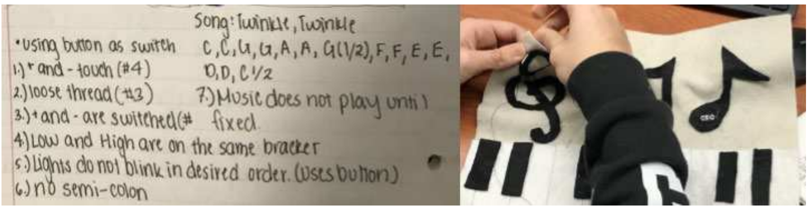

Figure 1. Georgia, Gabriel and Camila's DebugIt: (1) Left: list of bugs included in their project plus order of tones; (2) Right: Georgia sewing musical notes above the keyboard.

As Camila and Georgia worked on crafting the keyboard, Georgia became frustrated when gluing the musical notes: "I can't do this, I physically can't do this" (Lesson 3, Video). It was hard for her to calculate how much glue to apply on the felt. "I don't like glue, now it's a mess," she complained, having poured glue all over the project. Georgia's frustration created an opportunity for her to persist and overcome this setback, which she did with the support of her peers. Camila provided just-in-time praise for effort to motivate her friend to keep going: "you're almost there Georgia, that looks very good" (Video Lesson 3). Georgia persisted and tried different strategies: first using a piece of felt to apply it and then using the tip of the bottle to spread it. The latter strategy worked, and satisfied, she praised her own work "this [keyboard] looks so good!" (See Figure 1). We can see here how two mindset practices worked together: frustration generated opportunities for persistence, and just-in-time praise for effort helped reframe setbacks as inherent aspects of learning that can be overcome with persistence.

As work on the project continued, Georgia shared her mistakes when she took on a challenge that led to more learning. Georgia, who had little experience sewing and crafting, told her peers "I'm just warning you I'm not the best at sewing" as she started sewing the circuit. Not afraid of failure, Georgia shared her problems with Camila who helped her tie knots and make her connections tight. They shared their failures with each other and laughed together about the unintentional mistakes they made. Camila, for example, shared that she sewed an LED to the wrong pin of the microcontroller and Georgia shared that she sewed an LED backwards. Comfortable with failure, when students chose challenges that led to more learning, they embraced mistakes and overcame them.

When coding *Twinkle, Twinkle, Little Star*, Gabriel persisted after encountering many setbacks and frustration. He did not know how to trigger the song to play, but after fixing many unintentional bugs in his code and asking the teacher many questions he managed to make the microcontroller play music. When it worked, with frustration, he said "I have no clue why it sounds so hideous, this is devilish trap music" (Video Lesson 4). His peers agreed and did not like the sound. Yet, he persisted and continued working on improving the song and on the last day of designing he was able to play it to the teacher and his peers. The music became an Easter egg in the project: they used the light sensor on the microcontroller to trigger the song. Gabriel explained: "if you take the flash on your phone and put it on your CP [microcontroller] it'll make sounds happen." This is an example of how when students chose challenges that led to more learning, their frustration became an opportunity to overcome setbacks, persist when things do not go as expected, and approach learning as constant improvement. The team embraced learning as constant improvement in other instances as well, when they looked for ways to improve their project and their performance. When Gabriel was done with the code he insisted they had to test it to see if it worked as expected and if not to fix it. They tested several times, and every time they came up with new ideas of how to improve the project.

Growing mindsets across groups

This case illustrates how students engaged with growth mindset practices in the context of DbD. We identified 75 instances in which Georgia, Gabriel and Camila engaged with these practices. They were not the exception. The four groups we analyzed engaged in all identified growth mindset practices in 230 instances. Figure 2 shows the frequency of instances of each practice by group of interest as well as information about the context of each instance, when they occurred (e.g., when planning the project), and who was involved (e.g., teacher and learner). As we saw in our case study, students engaged with growth mindset practices through peer-to-peer and learner-teacher interactions when planning their projects (e.g., making a circuit diagram), making buggy projects, and debugging their peers' projects. Most of the instances of growth mindset practices identified (56%) occurred when learners made buggy projects for their peers to solve. Most of the instances coded (78%) occurred when peers interacted with each other, although we also identified growth mindset practices when students worked alone or in learner-teacher interactions.

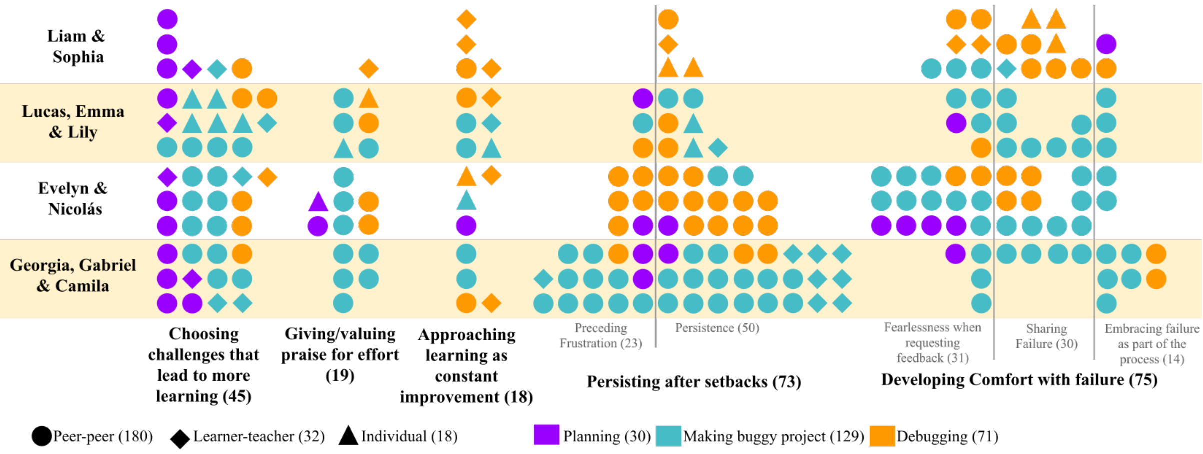

Figure 2. Visualization of instances of growth mindset practices by student group; color indicates when the instance occurred, and shape indicates who was involved.

Discussion

We analyzed the growth mindset practices that students exhibited in DbD, an intervention designed to empower learners in becoming more familiar, comfortable and capable with debugging. In the following sections, we discuss our instructional design rationale, what we learned about developing students' growth mindset practices, and directions for further research.

In designing DbD, we built on a longstanding tradition of constructionist activities that put learners in control of their own learning by designing applications for others (Harel & Papert, 1990, Kafai, 1995) and extended this concept by shifting the focus from designing functional artifacts to designing non-functional, or buggy, projects. By making learners designers rather than finders of bugs, we created an environment in which students were in control of naming and creating mistakes. DbD provides a compelling example of what Dweck and Yeager (2019) called for, "an examination of the mindsets conveyed by or embodied in the environments that students (or adults) are in" (p. 490). This proactive design approach contrasts with previous growth mindset interventions where teachers explicitly instructed students on growth mindset theory or practices (e.g., Simon et al., 2008). While explicit instruction and reflection on growth mindset can certainly help students, we suggest that our design-focused, practice-first approach may be particularly well suited to domains like CS that involve iterating, testing, and debugging. Putting students in charge of designing mistakes for their peers to solve could be applied to other areas of CS education as well other disciplines.

Furthermore, our analysis of four groups of students provided insights into how learners develop their growth mindsets and become more comfortable with debugging. Students engaged in practices such as choosing challenges that led to more learning, praising effort, approaching learning as constant improvement, persisting after setbacks, and developing comfort with failure. This last practice complements those practices mentioned in Campbell and colleagues' (2020) framework of practices for growth mindset learning activities, acknowledging mistakes and failures as "an intrinsic part of the learning process" (Papert, 1980, p. 153). Combined, these practices promote debugging as a holistic process that involves thinking and emotion (DeLiema et al., 2020), rather than a basic skill for identifying isolated problems. Of further note are the critical roles that peer collaborations and teacher support played in the DbD unit. The development of the observed growth mindset practices mainly occurred through interactions with peers, while they made buggy projects together. But these peer-supported practices did not happen accidentally, our implementation of DbD was conducted by a highly experienced teacher that promoted a classroom culture that encouraged student collaborations, legitimized student expertise, and modeled comfort with failure (Fields et al., in press).

We reported on the first classroom-based implementation of DbD. Further studies should examine how other teachers can foster growth mindset practices when applying the unit in their classrooms. In addition, we must examine the impact of DbD on students' continued programming efforts and compare it to students who did not engage with the unit. Furthermore, we need to examine how growth mindset practices exhibited in the classroom relate to the perceived growth mindset of students and teachers as measured in self-reported instruments. Lastly, the idea of making students designers of bugs, or mistakes, is not limited to CS education and could also be applied to other STEM contexts in which students design artifacts. Students can benefit not only from designing applications but also from designing mistakes—both are and should be an intractable part of any learning production.

References

Buechley, L., Peppler, K., Eisenberg, M., & Yasmin, K. (Eds.) (2013). *Textile Messages: Dispatches from the World of E-Textiles and Education. New Literacies and Digital Epistemologies*. Peter Lang.

Burnette, J. L., Hoyt, C. L., Russell, V. M., Lawson, B., Dweck, C. S., & Finkel, E. (2020). A growth mind-set intervention improves interest but not academic performance in the field of computer science. *Social Psychological and Personality Science*, 11(1), 107-116.

Campbell, A., Craig, T., & Collier-Reed, B. (2020). A framework for using learning theories to inform 'growth mindset' activities. *International Journal of Mathematical Education in Science and Technology*, 51(1), 26-43.

DeLiema, D., Dahn, M., Flood, V. J., Asuncion, A., Abrahamson, D., Enyedy, N., & Steen, F. F. (2020). Debugging as a context for collaborative reflection on problem-solving processes. *Deeper learning, communicative competence, and critical thinking: Innovative, research-based strategies for development in 21st century classrooms*, 209-228.

Dweck, C. S. (2000). *Self-Theories. Essays in Social Psychology*. Psychology Press.

Dweck, C. S. (2006). *Mindset: The new psychology of success*. Random House.

Dweck, C. S., & Yeager, D. S. (2019). Mindsets: A view from two eras. *Perspectives on Psychological Science*, 14(3), 481-496.

Fields, D. A., & Kafai, Y. B. (2020). Debugging by Design: Students' Reflections on Designing Buggy E-Textile Projects. *Proceedings of Constructionism 2020*.

Fields, D. A., Kafai, Y. B., Morales-Navarro, L., Walker, J.T. (in press). Debugging by Design: High School Students' Learning Experiences and Reflections about Programming Failure Artifacts. *British Journal of Educational Technology*.

Haimovitz, K., & Dweck, C. S. (2017). The origins of children's growth and fixed mindsets: New research and a new proposal. *Child Development*, 88(6), 1849-1859.

Harel, I. & Papert, S. (1990). Software design as a learning environment. *Interactive Learning Environments*, 1(1), 1-32.
Kafai, Y. B. (1995). *Minds in play: Computer game design as a context for children's learning*. Norwood, NJ: Ablex.
Loksa, D., Ko, A. J., Jernigan, W., Oleson, A., Mendez, C. J., & Burnett, M. M. (2016, May). Programming, problem solving, and self-awareness: effects of explicit guidance. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems* (pp. 1449-1461).
Love, B. L. (2019). *We want to do more than survive: Abolitionist teaching and the pursuit of educational freedom*. Beacon Press.
Luxton-Reilly, A., McMillan, E., Stevenson, E., Tempero, E., & Denny, P. (2018, July). Ladebug: an online tool to help novice programmers improve their debugging skills. In *Proceedings of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education* (pp. 159-164).
Margolis, J., Goode, J., & Flapan, J. (2017). A Critical Crossroads for Computer Science for All: "Identifying Talent" or "Building Talent," and What Difference Does It Make?. In Moving Students of Color from Consumers to Producers of Technology (pp. 1-23). IGI Global.
McCauley, R., Fitzgerald, S., Lewandowski, G., Murphy, L., Simon, B., Thomas, L., & Zander, C. (2008). Debugging: a review of the literature from an educational perspective. *Computer Science Education*, 18(2), 67-92.
Murphy, L., & Thomas, L. (2008, June). Dangers of a fixed mindset: implications of self-theories research for computer science education. In *Proceedings of the 13th annual conference on Innovation and technology in computer science education* (pp. 271-275).
Nolan, K., & Bergin, S. (2016, November). The role of anxiety when learning to program: a systematic review of the literature. In *Proceedings of the 16th koli calling international conference on computing education research* (pp. 61-70).
Papert, S. A. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic books.
Perkins, D. N., Hancock, C., Hobbs, R., Martin, F., & Simmons, R. (1986). Conditions of learning in novice programmers. *Journal of Educational Computing Research*, 2(1), 37-55.
Quille, K., & Bergin, S. (2020, June). Promoting a Growth Mindset in CS1: Does One Size Fit All? A Pilot Study. In Proceedings of the 2020 *ACM Conference on Innovation and Technology in Computer Science Education* (pp. 12-18).
Rangel, J. G. C., King, M., & Muldner, K. (2020). An Incremental Mindset Intervention Increases Effort During Programming Activities but Not Performance. *ACM Transactions on Computing Education* (TOCE), 20(2), 1-18.
Scott, M. J., & Ghinea, G. (2013). On the domain-specificity of mindsets: The relationship between aptitude beliefs and programming practice. *IEEE Transactions on Education*, 57(3), 169-174.
Scott, M. J., & Ghinea, G. (2014, July). Measuring enrichment: the assembly and validation of an instrument to assess student self-beliefs in CS1. In *Proceedings of the tenth annual conference on International computing education research* (pp. 123-130).
Silva, J. (2011). A survey on algorithmic debugging strategies. *Advances in engineering software*, 42(11), 976-991.
Searle, K. A., Litts, B. K., & Kafai, Y. B. (2018). Debugging open-ended designs: High school students' perceptions of failure and success in an electronic textiles design activity. *Thinking Skills and Creativity*, 30, 125-134.
Simon, B., Hanks, B., Murphy, L., Fitzgerald, S., McCauley, R., Thomas, L., & Zander, C. (2008, September). Saying isn't necessarily believing: influencing self-theories in computing. In *Proceedings of the Fourth international Workshop on Computing Education Research* (pp. 173-184).
Yeager, D. S., & Dweck, C. S. (2012). Mindsets that promote resilience: When students believe that personal characteristics can be developed. *Educational psychologist*, 47(4), 302-314.

Acknowledgments

This work was supported by a grant from the National Science Foundation to Yasmin Kafai (#1742140). Any opinions, findings, and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of NSF, the University of Pennsylvania, or Utah State University. Special thanks to Lindsay Lindberg and Ammarah Aftab for support in data collection and Gayithri Jayathirtha for feedback on this paper.