Peer-Designed Active Learning Modules as a Strategy to Improve Confidence and Comprehension Within Introductory Computer Science

By Becky Wai-Ling Packard, Jaemarie Solyst, Anisha Pai, and Lu Yu

While research has demonstrated the links between active learning and student success, lecture remains a dominant instructional method within introductory STEM courses. In this project, we used the strategy of enlisting peer mentors to develop and facilitate active learning modules within introductory computer science. First, we describe the preparation peer mentors underwent for their role and the active learning modules that were developed. Next, we share survey data from introductory undergraduates (n = 45); they viewed peer mentors as effective facilitators and credited the active learning modules as contributing to their comprehension of and confidence with course material. In our analysis, we underscore the value of analogy and simulation as cognitive scaffolds and the contributions of near-peer perspectives when developing introductory science courses. Implications for future work involving introductory college science teaching are discussed.

Active learning is a pedagogical strategy associated with a wide variety of positive student outcomes, including stronger engagement, better grades, and reduced failure rates across STEM disciplines (Moura & van Hattum-Janssen, 2011; Preszler, 2009; Zhang et al., 2013). A meta-analysis of more than 200 studies demonstrated that students enrolled in active learning classrooms observed greater student success, whereas students in lecture-based classrooms were 1.5 times more likely to fail the course (Freeman et al., 2016). Research, drawn primarily from biology classrooms, has consistently underscored how active learning can close equity gaps for first-generation, low-income, underrepresented racial minorities, as well as female students (see Haak et al., 2011; Preszler, 2009).

Unfortunately, in a study analyzing the practice within 2,000 science classrooms across the United States and Canada, researchers found that lecture, rather than active learning, continues to be the dominant teaching method (Stains et al., 2018). Decades of prior research has documented the link between large, lecture-based introductory courses and STEM departure (Mervis, 2010; Seymour & Hewitt, 1997). As a field, computer science faces burgeoning enrollment pressures (Computing Research Association, 2017). Gender and racial disparities stubbornly remain (Beyer, 2014; Bound & Morales, 2018; Cereijido & Selyukh, 2016). The dilemma involves finding ways to keep computer science classes inclusive to a broader demographic while responding to enrollment pressures. Given that lecture formats are just as common within introductory computer science (Grissom et al., 2017), we looked to explore ways to increase the uptake of active learning strategies.

Multiple barriers impede the adoption of active learning in college science classrooms. Faculty interested in undergoing pedagogical revision need to invest time, both when initially adopting the change and longer term, as they revise. Any pedagogical change is unlikely to stick without department sponsorship or resourcing (Fagen et al., 2002; Wieman et al., 2013). In addition, faculty may not realize how much active learning already aligns with their current teaching practices, leading to a lack of adoption (Horne & Murniati, 2016). Further, students may demonstrate resistance if they do not understand the reasons for integrating active learning (Finelli et al., 2018).

In this paper, we describe a strategy to enlist peer mentors to design and facilitate active learning modules in computer science. As a potentially replicable and scalable strategy to increase access to active learning and inclusive peer models, we share the effectiveness of the peer mentors and the particular modules they designed. Next, we review the relevant literature that situates peer mentoring within STEM curricula.

Literature review

Leveraging peer mentors in curricular development and facilitation

Across STEM fields, peer mentors have played an important role in curricular support. Peer mentors are often seasoned near-peers who have recently completed the course for which they are mentoring. For example, peer mentors commonly provide supplemental instruction to newcomers across multiple STEM fields, including biology, chemistry, and physics (Peterfreund et al., 2008). A primary benefit of peer mentoring is that newer students, whether studying computer science or precalculus, learn effective strategies as they practice difficult problems collectively; peer-mentoring sessions are often offered on a weekly basis to all students, not only to those struggling with the course (Pon-Barry et al., 2017; Liou-Mark et al., 2010), which has contributed to a reduction in racial disparities (Rath et al., 2007). From a learning community perspective (Lave & Wenger, 1991), this collective engagement model can reflect a shift in culture via a change in practice that encourages collaboration. Peer mentors may also lend a valuable, complementary perspective to faculty (Mazur, 1997; Talbot et al., 2015) because peers have some knowledge of the subject, but not necessarily an expert "blind spot" (Nathan & Petrosino, 2003). Thus, peer mentoring offers both social and cognitive benefits.

Investing in a peer mentor strategy is not without cost. Departments and institutions invest in the preparation of mentors and provide financial or credit-based compensation for their role. While there are costs, introductory student success is an important benefit. In addition, evidence points to peer mentors also growing as a result of their role, as they deepen their own commitment to the field, a finding demonstrated across STEM fields (Anagnos et al., 2014; Bowling et al., 2015).

The typical method used to prepare peer mentors is a training course (Barnard et al., 2018; Bowling et al., 2015; Streitwieser & Light, 2010; Talbot et al., 2015). The peer mentors in this study enrolled in a preparatory course where they read about learning theory, metacognition, self-efficacy, and active learning within computer science. They also learned about inclusive teaching strategies, engaged in mock feedback sessions, and practiced providing feedback, with an emphasis on learners' diverse racial and gendered identities (e.g., Cohen et al., 1999). Peer mentors also pitched, developed, and revised active learning modules for introductory computer science. We recognized the possible limits of enlisting undergraduate peer mentors, as even graduate students may experience threats to their authority (Reid, 2008) and struggle with their teaching efficacy (DeChenne et al., 2012). These aspects were also emphasized in the peer mentors' preparation through exposure to various scenarios they might encounter.

Active learning approaches: Which design elements matter?

The term *active learning* typically refers to a broad set of teaching and learning activities designed to engage the learner, prompt questions, gauge initial understanding, and facilitate revision of initial understanding (Moura & van Hattum-Janssen, 2011). From a constructivist perspective, the goal is to assist learners as they develop or expand understanding and revise their mental models (Perkins, 1999). Peer instruction is one widely used active learning strategy in physics, where peers pair off, debate, and explain solutions to each other (Mazur, 1997). In biology, structured practice with frequent feedback has been emphasized (Haak et al., 2011).

In computer science, peers may engage in pair coding (Braught et al., 2011) and peer code review (Pon-Barry et al., 2017), where they work together to construct, discuss, and debug code. Teams of students may be prompted to ask questions about how they learn recursion (Hu & Shepard, 2013). Students may be prompted to write reflections about their misconceptions in order to address them (Tawde et al., 2017).

While previous research has documented the advantages of active learning over lecture, there is still a need to better understand how to optimize active learning. For example, although peer instruction is widely used, effectiveness may be reduced if the technique is implemented without the necessary components (Crouch & Mazur, 2001). When applied in computing settings, researchers found students may leave peer-to-peer discussions with unresolved misconceptions, and instructor intervention may be necessary to promote understanding (Zingaro & Porter, 2014). In this project, we explored the effectiveness of the peer-developed active learning modules. While the modules were designed with some intentionality and metacognitive scaffolds (e.g., practice, review, checks for understanding), we also acknowledged the variability in effectiveness. Thus, we aimed to learn more about which features played a role in how students perceived the learning modules' contribution to their learning.

Current project

Our research questions were:

1. How did introductory students perceive the competency and effectiveness of the peer mentors who facilitated active learning sessions?
2. How did introductory students perceive the effectiveness of the active learning sessions with regard to their comprehension and confidence? Which modules were more (or less) effective?

Method

Participants

Overall, 45 introductory students participated in the data collection (28 from fall 2017, 17 from spring 2018). Students were enrolled in "Advanced Problem-Solving and Elementary Data Structures," which is the second semester of a two-semester introductory sequence required of computer science majors. The semester was a traditional 15 weeks in duration.

This study was located at a women's college in the northeastern United States; 36 of the participants identified as female, one participant identified as gender nonbinary, one participant identified as male (this student had enrolled through a college exchange), and the remaining participants did not report their gender. Approximately half of the participants were international students. Of those who reported race, 23 identified as Asian or Asian American, eight as White, two as Black or African American, and one as Latina. Most planned to major in computer science, with a subset planning to double major in other sciences or humanities. Students spanned all years, with many first-year students and sophomores, and a few upper-level students.

Nineteen upper-level students participated as peer mentors. Most were majoring in computer science, with a few majoring in other sciences or double-majoring in social sciences. All identified as women, and the majority were international students. As for race, they self-described as Asian (9), White (5), Black (2), or Middle Eastern (2). One did not identify race.

Procedure and data sources

Following International Review Board human subjects approval, students were asked to participate in the course-based research project and sign consent forms if they agreed to have their materials analyzed for research purposes. Materials were held by the primary researcher (not the instructor) to analyze after each semester concluded. Students attended a lecture and lab section; all active learning sessions took place during one hour of a required three-hour lab.

Students completed surveys at the beginning and end of the course, which provided instructors with information about their confidence, interest, and experience. At the end of the course survey, students also rated their perception of the individual modules; a smaller (~20) subset also provided open-ended qualitative explanations about why particular modules were helpful. To supplement this, we also asked the lab instructor and faculty instructor to reflect on how they saw the contribution of the active learning modules. We also analyzed the mid-semester surveys from weeks 4 and 7, when students were asked to rate the peer mentor assigned to their lab (e.g., the mentor's knowledge, approachability, and creativity/flexibility), and whether the particular sessions contributed to their confidence or comprehension. The timing of these mid-semester surveys afforded a reflection from students closer in time to when the active learning modules occurred. These scores were averaged across the group. Due to no observed difference in the results from the fall and spring semesters, we combined the findings across both semesters.

Module creation

Each active learning module was designed as a 60-minute session. Peer mentors encouraged interaction, incorporated relevant coding language, and provided opportunities to practice. Modules were developed by peer mentors during the training courses in 2016 and 2017, and piloted thereafter; each year, less-effective modules were replaced and useful ones were revised. Next, we provide short descriptions of the modules.

Linked Lists ("Scavenger Hunt"). Students participate in a "scavenger hunt" around the lab. Students begin with a paper note with a location written on one side and a clue written on the other side. The location represents the contents of the linked list node, and the clue represents the pointer, which gives a hint to where the next note is located. The last note contains the clue "NULL" to specify that it is the end of the list. Afterward, students hold the paper notes and physically simulate different linked list operations, such as inserting a node into the middle of a list and deleting nodes in different positions. The peer mentor then draws out an ArrayList. The participants compare and contrast both list data structures in a wrap-up discussion.

Stacks and Queues ("Serving Pancakes"). The peer mentor begins with a review of terms. Then the class simulates a queue by lining up to be served from a stack of paper "pancakes." Students are then divided into small groups to discuss and write pseudocode for how objects (student and pancake) would use stack or queue data structures. Additional prompts are presented, such as how to get to the pancake at the bottom of a stack. A discussion compares stacks and queues with other data structures (e.g., arrays and lists).

Recursion ("Russian Dolls"). The peer mentor reviews a math factorial example before moving into an analogy of nested Russian dolls. Students are asked how the total number of dolls could be counted, or how to determine if a doll of a certain color exists within the set. In small groups, students write pseudocode for recursive methods; the peer mentor circulates to answer questions before groups explain their pseudocodes.

Binary Trees ("Storytelling"). The peer mentor explains binary trees and the different ways they can be created, then introduces a storytelling activity. Participants tell a chronological story by numbering sentences, each depicting a story event, and placing them in a binary tree structure; the root node is the "present," the left node is the "past," and the right node is the "future." The activity first creates a balanced binary tree, before participants create an unbalanced binary tree where are there no left nodes, so they can address insertion and traversal.

Program Design ("Let's Build a Museum"). The aim is to demonstrate how one program can be designed in several different ways using a museum curation analogy. Participants sort through a list of items that may be exhibited in a museum and group them via appropriate exhibits: Individual display pieces are variables; exhibits represent classes; and sub-exhibits (such as "airplanes" within "transportation") represent inheritance or interfaces. Students work as a whole class, then in smaller groups, and then the peer mentor facilitates a wrap-up discussion.

Mergesort ("Automotive Sorting"). Participants work with a simulation involving numbered toy cars that can change lanes on a multi-lane highway. The peer mentor demonstrates how lane changes can represent the splits and merges in the mergesort algorithm before each student takes control of a car, and the class works together to order the cars on the highway. The class then practices with pseudocode to examine the recursive nature of the algorithm before discussing common mistakes and debugging strategies.

Results

Perceptions of peer mentors as facilitators

Introductory students' ratings, drawn from weeks 4 and 7, indicated favorable views of their peer mentors. They rated highly the peer mentors' knowledge, approachability, and creativity/flexibility. Table 1 reflects the mean and standard deviation of students' responses during weeks 4 and 7.

Perceptions of peer-led active learning sessions

When asked to rate the contribution of peer-facilitated sessions to their understanding and confidence with the course material, introductory students agreed, on average, that the sessions contributed positively. Table 2 reflects the mean and standard deviation of students' responses during weeks 4 and 7.

Analysis of effectiveness of individual modules

Introductory students rated at the end of the semester the degree to which each active learning module was helpful for their learning. On average, students agreed that each module contributed with modest variation. Table 3 reflects the mean and standard deviation of students' responses at the end of the semester.

We noted the strong positive reviews of multiple active learning modules to include binary trees, linked lists, recursion, and stacks and queues. Several students emphasized in their surveys that visualization and simulation were positive assets of the modules. These attributes were especially salient in the open-ended comments from students. Specifically, multiple students noted the assets of the "Serving Pancakes" module.

TABLE 1

Introductory students' ratings of peer mentors.

Item	Week 4 Mean (SD)	Week7 Mean (SD)
The peer mentor who leads my active learning sessions is knowledgeable.	5.28 (0.94)	5.40 (0.65)
The peer mentor who leads my active learning sessions is approachable.	5.26 (1.04)	5.49 (0.78)
The peer mentor who leads my active learning sessions is creative and flexible about ways to help students.	5.12 (1.02)	5.35 (0.79)

Note: Scale ranged from 1 = strongly disagree to 6 = strongly agree. SD = standard deviation.

TABLE 2

Introductory students' ratings of active learning sessions at two midpoints in the semester.

Item	Week 4 Mean (SD)	Week 7 Mean (SD)
So far, active learning sessions have contributed to my understanding of course material.	4.79 (1.32)	4.96 (1.03)
So far, active learning sessions have contributed to my confidence with this course material.	4.71 (1.34)	4.91 (1.05)

Note: Scale ranged from 1 = strongly disagree to 6 = strongly agree. SD = standard deviation.

TABLE 3

Introductory students' ratings of individual active learning sessions.

Was this active learning module helpful for your learning?	Mean (SD)
Linked Lists ("Scavenger Hunt")	4.50 (1.08)
Stack and Queues ("Serving Pancakes")	4.63 (1.22)
Recursion ("Russian Dolls")	4.76 (1.00)
Binary Tree ("Storytelling")	4.85 (0.91)
Program Design ("Let's Build a Museum")	4.24 (1.20)
Merge Sort ("Automotive Sort")	4.55 (1.16)

Note: Scale ranged from 1 = strongly disagree to 6 = strongly agree. SD = standard deviation.

One student shared that having "to iterate through the stack" of pancakes helped with understanding. Another student shared that "serving pancakes really helped illustrate the difference between stacks and queues to me." Another student described how the "visualization of the stack" helped to solidify learning.

Other students emphasized the practice and feedback they received in these sessions. One student expressed, when talking about the linked list exercise, an appreciation for having the chance to practice adding and deleting. Another student, when referring to the recursion active learning module, shared that "seeing things on paper and practicing how to implement things and then having all the details defined made things a lot clearer and easier to translate from verbal to coding." While not emphasized in the open-ended comments by students, we also noted in our analysis of the modules that the binary trees activity focused on a concrete algorithm, and was illustrated via the analogy of storytelling. Students appeared to appreciate the use of analogies, as the analogy provided another way to visualize the concept.

Almost all of the modules were tied to major coding assignments, which may have increased perceived relevance. However, the module with the lowest average rating was program design ("Let's Build a Museum"), despite its use of simulation. At the introductory level, program design is taught in preparation for upper-level courses; students commented they did not appreciate the module given because it was not tied to a graded assignment.

Finally, we draw attention to the way that active learning occurred—in small groups within the lab—that provided an avenue for small group support from both peers and peer mentors. The peer mentor supported the cognitive integration through facilitation, as one student explained: “[My peer mentor] taught it very well by reiterating the takeaways at the end of each part of the exercise.” Each active learning module appeared to be most helpful to students who were struggling with that particular concept. The lab instructor underscored this: “The students who were most in need of help learning the concepts were the ones who got the most out of it.” The instructor added that the peer mentors themselves gained a great deal from the teaching: “Active learning sessions deepened conceptual understanding for the peer mentors in training, both in developing the modules, and doing the dry runs in their preparatory course.” The advantage of learning within a small group setting was also emphasized by the instructor, who shared, “Active learning contributed to students getting to know their lab peers and cohort-building.”

Discussion

This paper describes a strategy of enlisting near-peer mentors to design and facilitate active learning modules within an introductory computer science class at the undergraduate level. Introductory students rated peer mentors as competent and approachable, and they credited the active learning sessions with contributing to improved comprehension of and confidence with the material. The preparatory course for peer mentors was an important ingredient for success, which placed an emphasis on inclusive teaching, self-regulation, and active learning. Within the preparatory course, and over a two-year period, the active learning modules were developed, piloted, and revised.

Some variability was observed in the perceived effectiveness of the modules. Most were rated as effective; the use of analogies in particular has been backed by prior research (Holyoak & Thagard, 1997). However, the module not directly related to a graded assignment was rated less positively. College students may bring a performance goal orientation to their learning, where they are concerned about demonstrating what they have learned through grades (e.g., Harackiewicz, et al., 2002).

Going forward, a better understanding of where learners tend to struggle with particular concepts would help inform curriculum designers to decide which cognitive scaffolds to include, and for which learners. In prior research, struggling students needed more instructor-led intervention (e.g., Zingaro & Porter, 2014); this paper suggests that peer mentors could also be effective in this regard, particularly if preparation for their role provides insight into anticipated misconceptions and difficulties. We are currently in the process of creating virtual modules based on these active learning sessions so that we can incorporate scaffolds that point out possible difficulties that learners can revisit at their own pace.

Introductory students were introduced to peer mentors as integral components of the course to support a community-oriented culture (Lave & Wenger, 1991). As peer mentors facilitated the active learning modules, introductory students understood this feature within a broader change to the introductory curriculum as one that incorporated peer mentoring. This transparency is important for optimizing active learning (Finelli et al., 2018). In this learning environment, introductory students are encouraged to seek out their classmates in addition to peer mentors and instructors. To some degree this reflects the culture of peer support at this particular institution. We see this orientation as an asset toward a more inclusive approach to teaching computer science, as well as other sciences.

This project had multiple limitations. We only provided ratings from introductory computer science classes across one year at one institution. Observational methods offer the advantage of capturing learning in real time rather than ratings on a self-reported survey, whereas focus groups could inform the kinds of peer-facilitated interactions that contributed most to both engagement and understanding. For future research, we are interested in learning more about the process of how peer mentors identify the struggles of novice learners; they may draw from their own struggles, or from their prior teaching experiences, or from the preparatory course. Research has shown that coping models are more effective than success models, as they tend to inspire learners who see coping strategies modeled before success (Schunk, 1984). In this study, we were not able to closely examine individual engagement with the modules in a way that allowed us to clearly connect engagement with increased comprehension. In a more controlled study, we could test the pre- to poststudy changes in comprehension using an assessment of comprehension to more carefully analyze the effectiveness of each module. In addition, given the gender and racial disparities in computer science fields, we would like to see more of this work involving peer mentors in different computer science settings to understand their contributions in varied learning contexts.

In this paper, we described the process of enlisting peer mentors as facilitators and designers of active learning modules in introductory computer science courses. Peer mentors can contribute valuable insights when developing analogies, visualizations, and other cognitive scaffolds, and contribute to a culture of support. Working with peer mentors in this capacity may be a cost-effective solution to incorporating more active learning and could be scalable in larger institutions. Such efforts can improve the quality of learning experiences in undergraduate STEM education. ■

Acknowledgment

The team acknowledges Google for funding this initiative.

References

Anagnos, T., Lyman-Holt, A., Marin-Artieda, C., & Momsen, E. (2014). Impact of engineering ambassador programs on student development. *Journal of STEM Education: Innovations and Research, 15*(3), 14–20.

Barnard, R. A., Boothe, J. R., Salvatore, J., Emerson, K., Boone, A., Sandler, C., & Coppola, B. P. (2018). Course-based support for peer-led study group facilitators in a large instructional team. *Journal of College Science Teaching, 47*(4), 21–29. https://doi.org/10.2505/4/jcst18_047_04_21

Beyer, S. (2014). Why are women underrepresented in computer science? Gender differences in stereotypes, self-efficacy, values, and interests and predictors of future CS course-taking and grades. *Computer Science Education, 24*(2&3), 153–192. https://doi.org/10.1080/08993408.2014.963363

Bound, J., & Morales, N. (2018). Commissioned study: Workforce trends in computer science. In *Assessing and responding to the growth of computer science undergraduate enrollments* (pp.171–184). National Academies Press. https://doi.org/10.17226/24926.

Bowling, B., Doyle, M., Taylor, J., & Antes, A. (2015). Professionalizing the role of peer leaders in STEM. *Journal of STEM Education, 16*(2), 30–39.

Braught, G., Wahls, T., & Eby, L. M. (2011). The case for pair programming in the computer science classroom. *ACM Transactions on Computing Education, 11*(1).

Cereijido, A., & Selyukh, A. (2016, December 21). *Why aren't there more women in tech? A tour of Silicon Valley's leaky pipeline* [Audio broadcast]. Retrieved from https://www.npr.org/

Cohen, G. L., Steele, C. M., & Ross, L. D. (1999). The mentor's dilemma: Providing critical feedback across the racial divide. *Personality and Social Psychology Bulletin, 25*(10), 1302–1318. https://doi.org/10.1177/0146167299258011

Computing Research Association. (2017). *Generation CS: Computer science undergraduate enrollments surge since 2006.* http://cra.org/data/Generation-CS/

Crouch, C., & Mazur, E. (2001). Peer instruction: Ten years of experience and results. *American Journal of Physics, 69*(9), 970–977. https://doi.org/10.1119/1.1374249

DeChenne, S. E., Enochs, L. G., & Needham, M. (2012). Science, technology, engineering, and mathematics graduate teaching assistants teaching self-efficacy. *Journal of the Scholarship of Teaching and Learning, 12*(4), 102–123.

Fagen, A. P., Crouch, C. H., & Mazur, E. (2002). Peer instruction: Results from a range of classrooms. *The Physics Teacher, 40*, 206–209. https://doi.org/10.1119/1.1474140

Finelli, C., Nguyen, K., DeMonbrun, M., Borrego, M., Prince, M., Husman, J., Henderson, C., Shekhar, P., & Waters, C. (2018). Reducing student resistance to active learning: Strategies for instructors. *Journal of College Science Teaching, 47*(5), 80–91.

Freeman, S., Eddy, S. L., McDonough, M., Smith, M. K., Okoroafor, N., Jordt, H., & Wenderoth, M. P. (2016). Active learning increases student performance in science, engineering, and mathematics. *PNAS, 111*(23), 8410–8415. https://doi.org/10.1073/pnas.1319030111

Grissom S., Mccauley R., & Murphy, L. (2017). How student centered is the computer science classroom? A survey of college faculty. *ACM Transactions on Computing Education, 18*(1), 1–27. https://doi.org/10.1145/3143200

Haak, D. C., HilleRisLambers, J., Pitre, E., & Freeman, S. (2011). Increased structure and active learning reduce the achievement gap in introductory biology. *Science, 332*(6034), 1213–1216. https://doi.org/10.1126/science.1204820

Harackiewicz, J. M., Barron, K. E., Tauer, J. M., & Elliot, A. J. (2002). Predicting success in college: A longitudinal study of achievement goals and ability measures as predictors of interest and performance from freshman year through graduation. *Journal of Educational Psychology, 94*(3), 562–575. https://doi.org/10.1037/0022-0663.94.3.562

Holyoak, K. J., & Thagard, P. (1997). The analogical mind. *American Psychologist, 52*(1), 35–44. https://doi.org/10.1037/0003-066X.52.1.35

Horne S., & Murniati C. (2016). Faculty adoption of active learning classrooms. *Journal of Computing in Higher Education, 28*(1), 72–93. https://doi.org/10.1007/s12528-016-9107-z

Hu, H. H., & Shepherd, T. D. (2013). Using POGIL to help students learn to program. *ACM Transactions on Computing Education, 13*(3). https://doi.org/10.1145/2499947.2499950

Lave, J., & Wenger, E. (1991). *Situated learning: Legitimate peripheral participation*. Cambridge University Press.

Liou-Mark, J., Dreyfuss, A. E., & Younge, L. (2010). Peer assisted learning workshops in precalculus: An approach to increasing student success. *Mathematics and Computer Education, 44*(3), 249–260.

Mazur, E. (1997). *Peer instruction: A user's manual*. Prentice Hall.

Mervis, J. (2010). Better intro courses seen as key to reducing attrition of STEM majors. *Science,330*(6002), 306. https://doi.org/10.1126/science.330.6002.306

Moura, I. C., & van Hattum-Janssen, N. (2011). Teaching a CS introductory course: An active approach. *Computers & Education, 56*(2), 475–483. https://doi.org/10.1016/j.compedu.2010.09.009

Nathan, M. J., & Petrosino, A. (2003). Expert blind spot among preservice teachers. *American Educational Research Journal, 40*(4), 905–928. https://doi.org/10.3102/00028312040004905

Perkins, D. (1999). The many faces of constructivism. *Educational Leadership, 57*(3), 6–11.

Peterfreund, A. R., Rath, K. A., Xenos, S. P., & Bayliss, F. (2008). The impact of supplemental instruction on students in STEM courses: Results from San Francisco State University. *Journal of College Student Retention: Research, Theory & Practice, 9*, 487–503. https://doi.org/10.2190/CS.9.4.e

Pon-Barry, H., Packard, B. W., & St. John, A. (2017). Expanding capacity and promoting inclusion in introductory computer science: A focus on near-peer mentor preparation and code review. *Computer Science Education*, *27*(1), 54–77. https://doi.org/10.1080/08993408.2017.1333270

Preszler, R. W. (2009). Replacing lecture with peer-led workshops improves student learning. *CBE—Life Sciences Education, 8*(3), 182–192. https://doi.org/10.1187/cbe.09-01-0002

Rath, K. A., Peterfreund, A. R., Xenos, S. P., Bayliss, F., & Carnal, N. (2007). Supplemental instruction in introductory biology I: Enhancing the performance and retention of underrepresented minority students. *CBE—Life Sciences Education*, *6*(3), 203–216. https://doi.org/10.1187/cbe.06-10-0198

Reid, E. S. (2008). Mentoring peer mentors: Mentoring education and support in the composition program. *Composition Studies, 36*, 51–69.

Schunk, D. H. (1984). Self-efficacy perspective on achievement behavior. *Educational Psychologist, 19*(1), 48–58. https://doi.org/10.1080/00461528409529281

Seymour, E., & Hewitt, N. M. (1997). *Talking about leaving: Why undergraduates leave the sciences*. Westview Press.

Stains, M., Harshman, J., Barker, M. K., Chasteen, S. V., Cole, R., DeChenne-Peters, S. E., Eagan Jr., M. K., Esson, J. M., Knight, J. K., Laski, F. A., Levis-Fitzgerald, M., Lee, C. J., Lo, S. M., McDonnell, L. M., McKay, T. A., Michelotti, N., Musgrove, A., Palmer, M. S., Plank, K. M., … Young, A. M. (2018). Anatomy of STEM teaching in North American universities. *Science, 359*(6383), 1468–1470. https://doi.org/10.1126/science.aap8892

Streitwieser, B., & Light, G. (2010). When undergraduates teach undergraduates: Conceptions of and approaches to teaching in a peer led team learning intervention in the STEM disciplines: Results of a two year study. *International Journal of Teaching and Learning in Higher Education, 22*(3), 346–356.

Talbot, R. M., Hartley, L. M., Marzetta, K., & Wee, B. S. (2015). Transforming undergraduate science education with learning assistants: Student satisfaction in large enrollment courses. *Journal of College Science Teaching, 44*(5), 24–30.

Tawde, M., Boccio, D., Kolack, K. (2017). Resolving misconceptions through student reflections. *Journal of College Science Teaching, 47*(1), 12–17. https://doi.org/10.2505/4/jcst17_047_01_12

Wieman, C., Deslauriers, L., & Gilley, B. (2013). Use of research-based instructional strategies: How to avoid faculty quitting. *Physics Education Research, 9*(2). https://doi.org/10.1103/PhysRevSTPER.9.023102

Zhang X., Zhang C., Stafford T., & Zhang, P. (2013). Teaching introductory programming to IS students: The impact of teaching approaches on learning performance. *Journal of Information Systems Education, 24*(2), 147–155.

Zingaro, D., & Porter, L. (2014). Peer instruction in computing: The value of instructor intervention. *Computers & Education, 71*, 87–96. https://doi.org/10.1016/j.compedu.2013.09.015

Becky Wai-Ling Packard (bpackard@mtholyoke.edu) *is professor of psychology and education at Mount Holyoke College in South Hadley, Massachusetts.* ***Jaemarie Solyst, Anisha Pai,*** *and* ***Lu Yu*** *all completed their computer science undergraduate degrees from Mount Holyoke College.*