Peer-Led Team Learning in a Problem-Solving Course: Lessons Learned

Miguel Lara
California State University Monterey Bay

Descriptors: Peer learning, Problem solving, Computer education

Introduction

Despite the projected future demand of Computer Science related jobs in the US, there are not enough students choosing to major in this field (Bureau of Labor Statistics, 2014; NCLS, 2012). This issue is particularly evident with underrepresented minority (URM) students: among 22,273 high school students who took the Advanced Placement (AP) computer science test, 4% were African American and 9% were Hispanic Latino (College Board, 2013). URM students are significantly less likely to persevere in Science, Technology, Engineering and Math (STEM) majors (Chang, Sharkness Hurtado & Newman, 2014).

Biggers, Brauer, and Yilmaz (2008) identify "*the perception of asocial community with low levels of human interaction including student-faculty and peer-peer*" as one of the reasons students decide to leave their CS major (p. 406). A way to potentially improve the persistence of URM students in these majors is increasing the likelihood of students to engage in academic experiences, such as studying frequently with others (Chang, Sharkness Hurtado & Newman, 2014).

Peer-to-Peer Interaction and Learning

There are several instructional strategies aimed at promoting peer-to-peer interaction as part of classroom activities in STEM courses such as Peer Instruction (Mazur, 2017), Team-Based Learning (Michaelsen & Sweet, 2008) and Pair Programming, which is mainly intended for Computer Science related courses (Williams & Kessler, 2002). These strategies are based on Social Constructivism, a learning theory that suggests that knowledge is constructed through interaction with others (Vygotsky, 1978).

Peer-Led Team Learning (PLTL) is an instructional strategy that is not conducted as part of the classroom activities, as the aforementioned strategies, but as separate sessions that are lead by student mentors (called Peer Leaders). The instructor is still heavily involved in the design of the sessions (called workshops) but is not physically present in them.

PLTL was developed by Gosser et al. (2001) in the early 1990s to support students in Chemistry courses, however, it has been successfully implemented in several other areas, such as Biology (Batz, 2014), Information and Communications Technology (Sheard et al., 2011), Mathematics (Hockings, DeAngelis, & Frey, 2008), and Computer Science (Horwitz et al., 2009; Huss-Lederman, Chinn, & Skrentny, 2008; Murphy, Powell, Parton, & Cannon, 2011; Stewart-Gardiner, 2010).

Peer-Led Team Learning Characteristics

Gosser et al. (2001) identify the following characteristics: PLTL consists of a weekly workshop, which involves a peer leader and a group of six to eight students. The course instructors are closely involved with the workshops by holding weekly sessions with the peer leaders and developing challenging materials that are integrated with the course content.

The peer leaders are recruited from students who completed the course successfully. They are trained in leadership skills and teaching and learning strategies. Their role is more as facilitators and not as teaching assistants or mentors. Some of the peer leaders' tasks involve:

- clarify the purpose and goal of each workshop
- ensure full participation of the students
- create special opportunities for students
- build students' commitment and self-confidence
- practice listening skills and questioning techniques

strengthen students' skills and problem solving approaches
reflect continuously upon their process for leading the workshops

Our First PLTL Implementation

Encouraged by several studies highlighting the implementation of PLTL as a potential instructional strategy to reduce attrition for URM students (Horwitz et al., 2009, Stewart-Gardiner, 2010) and also for female students (Murphy, Powell, Parton, & Cannon, 2011), we decided to implement PLTL in CST231 Problem Solving and Programming in fall 2016. Our main research questions about the implementation of the PLTL program were:

1. What are the effects of PLTL on retention rates and grades?
2. What is the students' perception of PLTL?
3. In what ways could the PLTL program be improved?

CST231 is the first programming course that freshman students take as part of the undergraduate program in Computer Science. The class size is about 35 on average, and about 35% of them are URM students. As part of this course, students are able to edit, run, debug, and document computer programs written in C++. The course introduces basic programming skills such as conditional statements, logical expressions, loop structures, arrays, data files, and an introduction to object-oriented programming.

Following Gosser et al.'s (2001) recommendations to implement the PLTL program, we recruited students who successfully completed the course the previous semester as peer leaders; we trained them in leadership skills and teaching and learning strategies; two-hour workshops were offered every week; instructors were closely involved with the workshops, holding weekly sessions with the peer leaders and developing challenging materials that were integrated with the course content.

There were three sections of CST231 Problem Solving and Programming in fall 2016. An instructor taught two and a second instructor taught the third section. The weekly content was exactly the same across all sections. By the end of the previous semester, we had identified and recruited four students who had completed the course successfully and were willing to be Peer Leaders. Two of them had been tutors in previous semesters. We organized a two-hour training session in which we trained them to be Peer Leaders. They learned how to use several instructional strategies to facilitate the workshops, including peer problem solving and round robin techniques (Gosser et al., 2001).

The workshops were initially offered three days a week, Monday, Tuesday, and Friday. Each workshop lasted about 2 hours and the same content was covered in the three weekly workshops. In this way, students could attend a workshop on the day that was most convenient to them. Workshop attendance was not compulsory but strongly encouraged. Each weekly course assignment consisted of five questions/problems, two of which were ill-defined problems that were addressed as part of the PLTL workshop. Consequently, students attending the workshops would then benefit by working on their homework assignment under the guidance of the peer leaders.

The emphasis of the workshops was to focus on the problem solving process. Students worked in groups of four to six, depending on attendance. It was decided that computers shouldn't be used during the sessions for several reasons: a) promoting peer-to-peer interaction by preventing students from working individually in a computer, b) preventing students from looking for answers to the problems presented, and c) forcing students to memorize and recall the syntax of programming in C++.

Unfortunately, student attendance was a major issue: out of the 108 total students in the three sections, only about 50 students (46.29%) attended at least one workshop during the semester. Only two weeks had over 20 students attending the weekly workshops but on average, there were only about 10 students attending the workshops each week.

Both instructors were adamant about not assigning extra points for attendance, fearing that students would only attend to earn the extra credit. However, we did implement several other strategies to increase student attendance, such as:

1) Instructors increased the difficulty level of one of the ill-defined problems and encouraged students to attend the PLTL sessions to work on it.
2) A survey was conducted by the end of the first month of classes to identify a better day and time that would work for more students. Based on the students' feedback, the Monday workshop was moved to a different time on Fridays.
3) PLTL leaders came to class to give a short presentation about the benefits of attending the PLTL sessions and to answer any questions or concerns that students had.

Low attendance to PLTL workshops seems to be a common issue (Gafney & Varma-Nelson, 2008). Stewart-Gardiner (2010) reports that for several semesters almost no students participated voluntarily in the workshops. Even with extra credit, just two students out of 80 attended the workshops regularly.

We conducted a survey to analyze the students' perception about the PLTL program and identify areas of improvement. In total, 67 students from the three sections submitted the survey and 14 of them indicated that they had not attended any workshop. Nine of the students who did not attend any workshop reported that they were expecting to get an A as their final grade, and the remaining five were expecting to get a grade of B. The most common reason for not attending any PLTL workshop was that students felt they did not need them because they were able to solve the assignment problems by themselves.

The number of students that dropped, failed or withdrew the class was similar to previous semesters, between 25 and 30% per section. However, due to the low workshop attendance, it was not really possible to judge the effectiveness of the PLTL program.

Revised PLTL Implementation

In Spring 2017 the PLTL program was implemented for a second semester but attendance was made compulsory. Indeed, some PLTL programs at different universities require attendance and participation (Gafney & Varma-Nelson, 2008). At Columbia University, students taking CS1 can get one unit of research credit for their participation in the PLTL workshops, with two unexcused absences resulting in a failing grade (Murphy et al., 2011).

When the class was made available for registration, students were notified that they were required to attend three days per week during the semester, and that the third day would be for a lab session (PLTL). Attendance was made mandatory, with four unexcused absences resulting in a failing grade.

There were three sections of the class, each with about 32 students; each section attended a PLTL workshop separately. There were three peer leaders attending each workshop. One of the instructors met on weekly basis with the peer leaders for around 30 minutes to discuss the problems to be covered during the following workshop.

Other than making the attendance required, everything else remained the same: students were grouped randomly in groups of six to seven; two ill-defined problems were assigned as part of the class and students worked on them in the workshops; no computers were allowed. After identifying a potential solution to a problem, one of the students in a group would write it down in a whiteboard for everybody else to see it and provide comments and suggestions.

In spite of the required attendance and the joined efforts of the instructors and peer leaders, by the end of the semester, the percentage of students who failed, dropped, or withdrew from the class was still similar to previous semesters (25% to 30% per section). These results were somehow similar as those reported by Merkel and Brania (2015), who implemented a PLTL program for a Calculus I class but did not observe any gains in students' learning or retention.

We did not conduct a survey about the perception of students towards the required PLTL workshops. However, based on the opinion and observations from some of the peer leaders, the ill-defined problems used as part of the workshops might have been problematic for two reasons: 1) the problems might have been too advanced and might have tended to just benefit stronger students, and not those students who were still struggling with conceptual understanding, and 2) some of the problems were too difficult to be broken in smaller parts for analysis and to promote collaborative learning, as suggested by the PLTL literature (Gosser et al., 2001).

Conclusion

We implemented a PLTL program during two consecutive semesters with the goal to increase students' performance in programming and problem solving, as well as to decrease the number of students who drop, withdraw, or fail from the Programming and Problems Solving class. Even though there was no significant differences in the percentage of students failing or withdrawing the class, it doesn't necessarily means that a PLTL program is not effective.

Given that PLTL participation was voluntary during the first semester, only very few students actually attended the workshops. The lesson we learned was to either apply more active recruitment or to make participation mandatory. During the second semester implementing PLTL, we learned that it is important to use the right kind of problems as part of the workshops (problems with the right level of complexity that can be broken into smaller sub-problems).

As part of our analysis, we should have also considered other factors that might influence students' decision to drop or withdraw the class: CST231 is the first class in the Computer Science program and it is possible that some freshmen students might still be undecided about their career goals. Perhaps the PLTL program might be more helpful in sophomore or junior years, in which more students are committed to pursuing a career in Computer Science.

References

Batz, Z. (2014). *Reaching Struggling Introductory Biology Students with a Targeted Peer Tutoring Program* (Doctoral dissertation, University of Maine).

Biggers, M., Brauer, A., & Yilmaz, T. (2008, March). Student perceptions of computer science: a retention study comparing graduating seniors with CS leavers. In *ACM SIGCSE Bulletin* (Vol. 40, No. 1, pp. 402-406). ACM

Bureau of Labor Statistics, *U.S. Department of Labor, Occupational Outlook Handbook, 2014-15 Edition*, Software Developers, Retrieved from http://www.bls.gov/ooh/computer-and-information-technology/software-developers.htm

Chang, M. J., Sharkness, J., Hurtado, S., & Newman, C. B. (2014). What matters in college for retaining aspiring scientists and engineers from underrepresented racial groups. *Journal of Research in Science Teaching,51*(5), 555-580.

CollegeBoard, (2014). *The 10th Annual Ap Report To The Nation*. Computer Science. Change in AP Exam Participation and Performance Over Time. Retrieved from http://media.collegeboard.com/digitalServices/pdf/ap/rtn/10th-annual/10th-annual-ap-report-subject-supplement-computer-science-a.pdf

Gafney, L., & Varma-Nelson, P. (2008). Peer-led team learning: evaluation, dissemination, and institutionalization of a college level initiative (Vol. 16). *Springer Science & Business Media.*

Gosser, D. K., Cracolice, M. S., Kampmeier, J. A., Roth, V., Strozak, V. S., & VarmaNelson, P. (2001). *Peer-Led Team Learning: A Guidebook* (Prentice H.). Upper Saddle River, NJ.

Hockings, S. C., DeAngelis, K. J., & Frey, R. F. (2008). Peer-Led Team Learning in General Chemistry: Implementation and Evaluation. *Journal of Chemical Education, 85*(7), 990–996.

Horwitz, S., Rodger, S. H., Biggers, M., Binkley, D., Frantz, C. K., Gundermann, D., & Sweat, M. (2009). Using peer-led team learning to increase participation and success of under-represented groups in introductory computer science. *ACM SIGCSE Bulletin, 41*(1), 163-167.

Huss-Lederman, S., Chinn, D., & Skrentny, J. (2008). Serious fun: peer-led team learning in CS. *ACM SIGCSE Bulletin, 40*(1), 330-331.

Murphy, C., Powell, R., Parton, K., & Cannon, A. (2011, March). *Lessons learned from a PLTL-CS program.* In Proceedings of the 42nd ACM technical symposium on Computer science education (pp. 207-212). ACM. New York, NY, USA, 207-212

Mazur, E. (2017). Peer instruction. In Peer Instruction (pp. 9-19). Springer Spektrum, Berlin, Heidelberg.

Merkel, J. C., & Brania, A. (2015). Assessment of Peer-Led Team Learning in Calculus I: A Five-year Study. *Innovative Higher Education*, 1-14

Michaelsen, L. K., & Sweet, M. (2008). The essential elements of team-based learning. *New directions for teaching and learning*, 2008(116), 7-27

Sheard, J., Morgan, M., Butler, M., Falkner, K., Weerasinghe, A., & Cook, B. (2011). *Experiences of first-year students in ICT courses: good teaching practices*. Retrieved from: http://www.acdict.edu.au/documents/JudySheardFirstYearStudentsICTcoursesFinal.pdf

Stewart-Gardiner, C. (2010). Using peer led team learning to assist in retention in computer science classes. *Journal of Computing Sciences in Colleges, 25*(3), 164-171.

Vygotsky, L. (1978). Interaction between learning and development. *Readings on the development of children, 23*(3), 34-41.

Williams, L., & Kessler, R. (2002). *Pair programming illuminated*. Addison-Wesley Longman Publishing Co., Inc.