Advisor: Edith Dunfee Ries, Ed.D

The Effect of Cooperative Learning Groups on the Learning, Collaboration, and Attitudes of Students in a High School Software Design Class

Franco Antonucci
Program of Curriculum and Instruction

Submitted in partial fulfillment of the requirements
for the degree of Master of Arts in the Graduate Program
Caldwell University
2019

Abstract

This study examines the implementation of a series of collaborative learning computer programming projects in a class where individual assignments were customarily assigned. Its intention was to disprove the history of teacher and student resistance to group projects by showing they would not affect student grades or attitudes towards the class negatively. The study was conducted in an advanced level computer science class composed of fourteen high-school students. The teacher/researcher assigned a series of projects that the students were required to complete using the "pair programming" paradigm, where each student in the pair assumed a defined role of either the driver or navigator. Programming in pairs was chosen due to its prevalence in both university and industry settings.

Qualitative and quantitative data was collected by the teacher/researcher throughout the course of the four-week study. Quantitative data included attitudinal Likert scale surveys conducted both before and after the intervention, as well as student project assessment scores. Observational field notes, recorded in the classroom when "pair programming" was used, comprise the qualitative data. After analyzation of the data collected during the four-week intervention, collaborative learning appears to be a successful method of instruction that does not alter the grades or outlook of students negatively compared to individual assignments. At the same time, the study suggests the significant influence of variables like group composition and student ability levels.

Acknowledgements

I would like to thank Dr. Edith Ries for all of her guidance, support, and encouragement throughout the completion of this action research project. I would also like to thank all of my instructors and professors at Caldwell University for their professionalism and assistance in the completion of my degree. Finally, I wish to give a very special thanks to all of my past and present students for inspiring me to continue my studies in education.

Table of Contents

Chapter 1 – Introduction

Chapter 2 – Review of Literature

Chapter 3 – Methodology

Chapter 4 – Data Analysis

Chapter 1
Introduction

Overview

Empowering students with effective teamwork skills is crucial in all disciplines (Marshall, 2016). Experience working in multidisciplinary teams is particularly important for technology students both to prepare them for industry and to improve their communication skills with teammates from disciplines other than their own (Pastel, 2015). While this may be the ideal scenario, one cannot ignore the fact that there is a large amount of research that addresses the challenges that pertain to team projects in the area of computer education (Borstler, 2009). Although the benefits of collaborative learning are well documented, this specific approach to teaching is still not commonplace in math or technology classes and has, historically, been meet with both teacher and student resistance in each of these disciplines (Finelli, 2018). In the teacher/researcher's experience, the main reason technology and mathematics teachers are hesitant to use group projects is because they consider the material in these courses to be cumulative. These teachers are fearful that if a math or technology student does not fully comprehend the current topic being taught, it is highly unlikely he/she will understand the next topic that is being introduced. And although computer programming is typically a group activity in the workplace, high school computing students often lack these basic collaborative skills because of resistance from both teacher and students to implement and embrace group work structures (Lingard, 2011). The teacher/researcher suggests that those who embrace this philosophy of students learning in the areas of math and technology fail to realize that in today's universities and corporate environments, students and employees are being asked to collaborate on a regular basis.

The study being proposed here is intended to disprove this theory of student unwillingness to collaborate in the areas of math and technology. Learning how to work in teams is not a natural process for many students, but the teacher/researcher suggests that mastering this skill is very important. Many people attracted to computer science and software engineering prefer to work alone. They see a program as a problem that needs to be solved and they like to develop a solution themselves from top to bottom (Waite, 2004). As a "techie" himself, the teacher/researcher understands how many programmers feel a great sense of accomplishment upon the completion of a project because they are taking an empty digital canvas and turning into a working finished product. Since the software industry is built on teams of programmers, this solitary method of teaching software design in high school is, in the eyes of the teacher/researcher, doing these high school students a disservice. It is suggested here that students who are participating in an advanced technology high school course must be trained to be a team member if they are going to be successful in the software design industry.

Need for Study

The teacher/researcher plans to introduce a "pair programming" learning structure to students in a high school computer science classroom. This collaborative learning atmosphere will be cultivated by assigning a series of well-organized group projects to the students under study. In a "pair programming" project, one student acts as the driver by typing code at the computer or writing down the basic program design. The other student, called the navigator, has many jobs. The navigator is tasked with looking for defects in the work of the driver, such as syntax errors. He/she is also responsible for being the strategic, longer-range thinker who is responsible for making sure the driver has not lost sight of the objectives of the program. Prior to this intervention, individual technology projects had been the norm for these students, as they had not shown much desire to work in groups, and generally speaking, indicated that they preferred to work alone. The teacher/researcher hypothesizes that these collaborative learning experiences will not impact the students' assessment grades in a negative way and that they will retain as much technological knowledge as they had retained previously while working in isolation. By introducing a collaborative learning structure model, the teacher/researcher believes that he will be preparing students for college, university and corporate world requirements in the area of technology. Moreover, the teacher/researcher suggests that given the grouping instincts of teenagers, this group of high schoolers will come to embrace these assigned collaborations. Consequently, it is anticipated that students will then be better prepared for technology collaborative coursework at the university and industry levels.

Having students work collaboratively on class assignments has been shown to have multiple benefits inside the classroom (Joy, 2005). Thus, the main goal for this study is to determine if a classroom structure that has students in a high school computer science classroom working collaboratively on technology projects will allow the students to learn the material and retain that information at the same level as they had when they worked on projects in an individualized structure. The teacher/researcher also plans to study the effect of collaborative learning on the attitude and outlook of the students under study with regard to computer science, information retention, and classroom experience. It is anticipated that students who were initially hesitant about working collaboratively on programming projects will find collaboration to be a benefit both academically and socially, and that they will be more likely to have positive feelings about working together in their classrooms and workplaces as they move forward in the field of technology.

Background

The study was implemented in a high school in a regional suburban school district in Northern New Jersey. The district is composed of two towns and has five elementary schools, one middle school, and one high school. The district's seven schools have an enrollment of approximately 2,600 students, and the high school where the study was implemented has an enrollment of approximately 800 students. The high school operates on a rotating-block schedule, where the teacher sees the class three out of every four days for 57 minutes.

The study was conducted in a high school advanced computer science class, which is an elective practical arts course. Students are not required to take a computer science course, but must complete a practical arts class of some kind before graduation. The teacher/researcher is the sole computer science teacher in the district and teaches four different technology courses. The class under study contains 14 students in grades 10-12 with 10 males and 4 females. One student is a classified special education student with an IEP. Only three of the 14 students have prior experience in a computer science classroom.

Research Questions

This research study will explore the effect of collaborative learning projects on students' attitudes and grades in a high school computer science class. The teacher/researcher hypothesizes that the use of group projects will not cause a negative impact on student grades or affect their attitude toward technology class. Specifically, the teacher/researcher asks:

- Will implementing collaborative learning projects in a computer science classroom affect student participation and attitude toward the course in a negative manner?

- Will the implementation of group projects in a computer science classroom which had previously relied only on individual class projects lead to lower grades on paired project assessments?

Definitions

Abstract Data Types: a mathematical model for data types, where a data type is defined by its behavior from the point of view of a user of the data, rather than the implementer.

Advanced High School Technology Class: For the purpose of this study, an advanced high school technology class is a course in a STEM-related field given the designation of Advanced Placement (AP) by the College Board.

Cognitive Skills: a term referring to a human's ability to process thoughts

Collaborative Learning: a situation in which two or more people learn or attempt to learn something together. For the purpose of this study, it will refer to technology projects in an advanced high school computer class that will be completed by a pair of students.

Linked Lists: a linear collection of data elements, whose order is not given by their physical placement in memory

Multidisciplinary Projects: For the purpose of this study, these technical projects involve other skills along with engineering skills.

Pair Programming: a style of computer programming in which two programmers work side-by-side at the same computer, continuously collaborating on the same project

Soft Skills: personal attributes that enable someone to interact effectively and harmoniously with other people

Templates: in computer programming, a template is a generic class or other unit of source code that can be used as the basis for unique units of code

Conclusion

The teacher/researcher hypothesizes that the intervention suggested here will provide valuable experience in the area of collaborative learning to high school computer science students. It is also suggested that these group projects will not adversely affect the attitudes or grades of the students in the class. This collaborative learning experience will then leave the students better prepared for computer science work in both industry and university settings. The research behind the hypothesis can be found in Chapter 2, and a breakdown of the study will follow in Chapter 3.

Chapter 2
Review of Literature

Introduction

The ability to work collaboratively is a skill that is beneficial to all students (Marshall, 2016). It has often been said, however, that collaboration is not a major component of technology education. Because of the number of missteps that have taken place in the early days of technology education, these collaboration skills have not become a staple within the everyday technology class. Some believe that this lack of interest in fostering student collaboration in technology courses has its roots in the field of engineering. While the opportunity to collaborate is a strong component of the language arts, social studies and science curricula, collaboration is still not used regularly in many math or technology classes (Matusovich, 2009). This study is attempting to understand the history of this problem, to research the issue, and to bring the implementation of collaborative activities into a high school technology advanced level computer science class.

History of Early Technology Education

The importance of group work in the computer science curriculum is something that has been studied for decades. As the demand for workers who are proficient in computing has increased, more employers found that these "techies" were lacking many required skills unrelated to technology. A reason for this skill deficiency could be that although engineering practice continued to evolve, engineering education had not changed significantly since the 1950's (Lang, 1999). From its beginnings in the early 19th century until World War II, engineering education in the United States focused strongly on engineering practice. The war itself showed the world the importance of technology on the battlefield and the soldiers returning home with their GI bills provided a large crop of engineering students and consumers of technology as well. Lang (1999) tells us that:

> Post-war industry flourished, creating demand for engineers that exceeded the supply. Newly-minted engineering Ph.D.'s joined the ranks of academia without industry experience and perpetuated the research emphasis on campus for the last forty years (p. 43).

In other words, many engineering graduates went straight into teaching, both at the high school and university levels, with no direct knowledge of how the workplace operates. As a result of this type of guidance, students were lacking many of the collaborative non-engineering related skills necessary today in the corporate world.

However, the engineering environment changed dramatically over the remainder of the 20th century. Beginning in the 1980s, international competition, along with the shift from defense toward commercial enterprise endeavors, restructured the engineering industry and significantly altered how engineers practiced their craft (Lang, 1999). Commercial enterprise now necessitated input from colleagues from fields other than engineering, forcing engineers to supplement their technical know-how with business and communication skills. Unfortunately, most traditional engineering undergraduate programs were not set up to handle this increase in the demand for a greater liberal arts education. Global competitors, who valued flexible teams with multi-talented members, imposed severe competitive pressures on the United States employers who then drastically changed their management processes. But, these changes in the business environment were not followed by changes in the world of technology education (Black, 1994). Attention to this deficit is, in the eyes of the teacher/researcher, sorely needed. Therefore, it is suggested here that high school students who are enrolled in advanced technology courses would benefit from additional exposure to collaborative learning experiences that would prepare them for the future. The teacher/researcher intends to implement a collaborative learning environment and measure the effects of this intervention in a high school elective computer science class in which there are 10th to 12th grade advanced level technology students.

Benefits of Collaborative Learning for Technology

Collaborative learning is beneficial to students in all content areas for a variety of reasons. Group work helps students develop the "soft skills" necessary for the workplace such as problem solving, self-motivation, leadership, communication, and interpersonal skills. Employers understand the need for these skills and often value them as much or more than technical skills for employees (Miller, 2019). The use of group work has many benefits, and there is a substantial body of knowledge on its effective deployment. Henry (1994) identifies five principal reasons for students to undertake group work:

- application of knowledge – students are able to put into practice the knowledge and theory assimilated in previous modules.
- motivation – a suitably chosen project is likely to be of direct relevance to the student.
- higher cognitive skills – students develop a deep understanding of the material they are working on, and develop corresponding deep learning skills.
- autonomy – Students have control over what they learn and how they learn it.
- assessment – projects are effective at distinguishing the strong students from the weaker ones.

Reynolds (1994), on the other hand, believes that there is another key reason for students to participate in group work, and that reason is an ideological one. He maintains that by participating in group work, students are prepared for participation in a society which promotes collaboration and participation. Miller (2019) agrees with Reynolds and further suggests that this type of collaborative learning, when introduced in high school, holds additional benefits for the technology classroom because it helps prepare students for an industry where projects are developed in teams. To prepare computing graduates for professional careers, their education must provide them with real-life experiences. Marshall (2016) adds to the conversation when she states the following:

> There is a need for students to be able to work in teams to meet professional accreditation requirements and employers' demands. Empowering students with effective teamwork skills is crucial in all disciplines. Team projects in Software Engineering (SE) education provide training in collaborative project development, which can be used as a vehicle for teaching teamwork skills.

And finally, Borstler (2015) reminds us that in order to prepare computing graduates for professional careers, their education must provide them with real-life experiences. Classroom collaborative opportunities do just that.

As Reynolds suggests, technical companies and technical projects are no longer solely composed of engineers. As technology has expanded into nearly every area of our society, workers from diverse academic backgrounds now collaborate together on all types of projects across a variety of industries. Thus, communication in particular has become a critical issue, as teams in modern software development projects are often multidisciplinary and distributed over cultures and time zones. Therefore, the teacher/researcher believes that it is important that technology students learn at an early age the importance of collaboration.

Miller (2019) further suggests that group work also provides academic benefits in addition to the communication and collaborative skills already mentioned. Some of the benefits to which he refers include potentially higher retention rates and grades, particularly for underrepresented students. Gonzalo (2017) shares with us his belief that group work reduces errors in programming and improves the quality of generated computer code. An example of the academic benefits referenced above can be found in a study that was conducted at Georgetown University. The results of the study found that the implementation of a group project led to significantly better understanding of the nuances in programming as well as the understanding of advanced programming techniques such as templates, linked lists, and abstract data types (Blake, 2005). But despite the vast evidence of its benefits, group learning is still drastically underused in technology classes due to both teacher and student resistance (Le, 2018).

Resistance to Group Work

Although the importance of group work in the engineering curriculum is well known and documented, there is still much resistance to its use in the classroom by both teachers and students. Collaborative projects tend to get adopted often in language arts, social studies, and even science courses, but they still are not used regularly in many math or technology classes. The main reason teachers are hesitant to implement group work in these classes is that the material covered in technology courses tends to be cumulative in nature. In other words, to understand the current concept being taught, students must clearly understand the prior concept that had been introduced. The reason for this is that in technology each concept builds upon the previous one. It is suggested here that many teachers avoid group projects because they feel that group projects do not enable each student to gain a full understanding of every concept. If that occurs then students will not understand future technology topics that are introduced. Since professional evaluations are tied to student test scores, many educators prefer to use a method which guarantees full individual student accountability.

There are a variety of other reasons a teacher may be hesitant to implement group work in the technology classroom as well. There may be personality conflicts between students that can be easily avoided when the teacher relies on individual assignments. In addition, finding a particular grouping format that works successfully for collaborative projects can take up a great deal of valuable class time. Also, students with special needs may not react well to a change in their normal work routine. Finally, grouping students may also present problems when gender is taken into account because some students may not work as well with students of the opposite gender and/or with students of a younger age. Unlike most classrooms, a typical technology class has more male students, which can sometimes make gender grouping more difficult to achieve.

In addition to the instructor's unwillingness to use collaborative learning, students have their own resistance to this classroom structure as well. Hughes and Cotterell (2002) agree with this assumption and tell us that “many people attracted to software development find working in groups difficult.” Moreover, many authorities in the computer science field admit that “learning to work in teams is not a natural process for many students, but it is nonetheless extremely important” (ACM/IEEE, 2001, p.59). Another reason for resistance is that many students feel that in a group project they will be required to do more work if another partner does not do his/her own fair share of the work (Lingard, 2011). Consequently, students feel that they may end up being penalized if they are in a group with someone who refuses to complete any tasks or has trouble with the work required. There are other types of students who prefer to be in control of the project, and working with others creates a need to compromise and perhaps do the project in a way that is different than they had originally envisioned (Waite, 2004). Finally, similar to many instructors, students may also have reservations or may be uncomfortable working with students of a different gender.

Conclusion

In order to overcome the resistance mentioned above, and to prepare the students for university and corporate demands, the teacher/researcher plans to introduce a variety of collaborative learning opportunities in his advanced level technology class. It is the teacher/researcher’s belief that, for the reasons stated above, collaborative learning is beneficial for technology students. Research shows that having students work together increases many multidisciplinary skills and provides academic benefits as well (Miller, 2019). The teacher/researcher plans to conduct a four-week study in which he introduces a variety of collaborative learning opportunities and projects with fourteen high-school computer science students. The teacher/researcher believes this intervention will not negatively impact students' attitudes toward technology class and will have no significant adverse impact on their academic grades. The reason the teacher/researcher makes this statement is because of the research that appears to indicate student resistance to group work. This research is intended to debunk that idea!!! Details of the study design will be addressed in Chapter 3.

Chapter 3
Methodology

Overview

Collaborative learning is a skill that is beneficial to all students (Marshall, 2016). It is particularly important to students in a high school technology class because of the need to prepare students for successful experiences at the university and industry levels (Miller, 2019). However, despite the well-known documented benefits of student collaboration, it is still not a procedure that is used as often in the classrooms as it should be (Le, 2017). The teacher/researcher suggests that this lack of collaborative experiences in technology classes at the high school level is due to the resistance from both high school technology teachers and students. Many technology teachers and students feel that their (student) academic performance and classroom management will likely be negatively affected when they (students) are asked to participate in group projects (Lingard, 2011). In addition, it is common knowledge that technology students prefer to work individually, rather than with a partner or group. The reason for this is that working alone allows a student to complete a project exactly as he/she envisions the particular solution without the need to compromise ideas or approach (Waite, 2004).

The purpose of this study was then to determine how the implementation of collaborative learning projects in an advanced high school computer science class would affect students' attitudes and academic performance. It was hypothesized that the use of group projects would not lead to significantly lower grades on student assessments, projects or written quizzes. It was also hypothesized that the use of collaborative learning would not change student attitudes and enthusiasm toward computer science class. Qualitative and quantitative data were collected by the teacher/researcher as students participated in the newly-introduced group projects in each fifty-seven-minute class periods over the course of the four-week study period.

Participants

Participants under study were all student in a high-school advanced computer science course. There were 4 girls and 10 boys in the class which included tenth, eleventh, and twelfth grade students. The teacher/researcher had been teaching computer science in the school district for fourteen years and was one of five faculty members within the high school's industrial arts department. Although computer science courses are considered electives, they do fulfill a practical arts requirement for graduation. However, all students must successfully complete Algebra I as a prerequisite to enrollment in any computer science class in this high school. Although the course in which the study took place is designated as an Advanced Placement (AP) course by the College Board, it is an introductory computer science course and, as a result, only 3 of the 14 students had any prior experience in a computer science class. Therefore, for all of the students this was their first experience writing computer programs.

Data Collection

Prior to the start of the study, the teacher/researcher obtained consent from the administration of the high school where the study was conducted. The teacher/researcher also secured permission from the parents of each student under study through a form which assured student anonymity and confidentiality throughout the process (see Appendix A).

Pre-Intervention

All students under study were administered a pre-intervention attitudinal Likert scale survey created by the teacher/researcher (see Appendix B). The survey consisted of questions pertaining to the students' views on both collaborative learning and computer science class. It was created by the teacher/researcher to gauge the participants' feelings about group learning as well as their feelings about course content before the start of the intervention. The teacher also created a table listing individual student project grades before the intervention in order to compare pre and post-intervention grades on student group projects which were produced during the study.

Quantitative Data Collection

Throughout the four-week study, the students completed numerous teacher-assigned computer programming projects. All projects were done using the “pair programming” structure, in which each student was given a clearly defined role. The grades for these “pair programming” projects were a key component of the quantitative data collection. The project grades were used to see if students' grades had changed significantly during the transition from independent to group projects. Despite the course material being of greater difficulty than before the intervention, it was hypothesized that there would be no significant drop in student academic performance. The group project grades were placed in a table as well, for comparison purposes, along with the individual student project grades from pre-intervention.

Students under study were also administered a post-intervention attitudinal Likert scale surveys created by the teacher/researcher (see Appendix D). This survey also consisted of questions pertaining to the students' views on both collaborative learning and computer science class in general. The survey was created by the teacher/researcher to see if the participants' feelings about group learning and the course content had changed over the course of the intervention.

Qualitative Data Collection

Qualitative data for this study included teacher anecdotal records, in the form of field notes (see Appendix C). These observational notes were taken by the teacher/researcher as students worked together collaboratively on various computer programming projects throughout the four-week study. The purpose of these anecdotal records was to observe student collaboration and participation, as well as any time spent off-task (see Appendix F). Notes were also made at the end of each class regarding student involvement during lessons and class programming activities as well.

Procedure

Week 1

During the first week of the four-week intervention, the students under study and the teacher/researcher started their unit on “loops” in Java. At this point in the course, the students already had approximately one month of experience with computer programming using the language of Java. First, the teacher/researcher distributed the notes pertaining to the loops and reviewed them with the class. The class discussed the differences between the new concept of “while loops” and the previous concept of “if statements.” The teacher/researcher reviewed some example computer programming code that helped to illustrate the difference. After questioning students in order to check for understanding, the teacher and students spent the remainder of the period and the following day completing two programs together involving the new concept of “while loops.”

At the beginning of the next class period, students were assigned a partner to work with in order to complete the Fibonacci Program together as a collaborative learning technology assignment using “pair programming.” The Fibonacci Program has the user enter a maximum value and then produce the sequence of Fibonacci numbers up to the maximum entered. Each student was assigned the role of either the driver or the navigator in order to complete the project. (In “pair programming” the driver is responsible for typing the program, while the navigator observes the work of the driver for errors and makes sure the driver has not strayed from the objectives of the programming assignment.) Since each student is equally responsible for the single program being produced, each group member was given the same project grade.

Week 2

During the second week of the intervention, the students worked with the same partners. This time, the students were assigned two additional projects on which they were expected to work on with their new partner. The first assignment was the Credit Card Program where the user enters the interest rate and monthly payment for a delinquent credit card bill of one thousand dollars. The program must then output the remaining balance and total payments for each month until the total balance is reduced to zero. The students then changed their roles of driver and navigator to complete the next program. The second project of the week was the Craps Program that had the students create a simulation of a simplified version of the popular casino game. Since there were now two projects to complete, each pair was given the entire week for the completion of these group projects. While the student pairs worked on their collaborative projects, the teacher/researcher observed and took notes regarding each students' time on task, their ability to collaborate together, and the level of input of each group member toward solving the problem.

Week 3

During the third week of the intervention, the students under study were introduced to the concept of "for loops." A printout with notes and examples was again distributed to students. The printout explained the differences between "for loops" and the previous concept of "while loops" and when each should be used in a program. After the teacher/researcher showed multiple examples and checked for understanding among the students, the remainder of the class period was spent on the completion of a computer program written together as a class. The next class period continued the study of “for loops” using another class program in preparation for the upcoming group projects. At the start of the following class, the students were assigned another new partner by the teacher/researcher in order to work on the completion of a “pair programming” computer programming project which included “for loops.” For this project, each group was allowed to choose their roles. Did they wish to be the driver or navigator?

Week 4

During the fourth week of the intervention, students were permitted to choose their partner to work with for their final collaborative learning project. Each group was assigned a set of two computer programming projects to complete that involved the new concept of “for loops.” The first program was the Table of Contents Program where the user enters the names of the ten chapters in the book, and the program then creates a properly formatted table of contents. The second program was a collaborative learning project that was completely designed by the student partners. They were required to come up with the concept of the program, as well as the computer language code necessary to make it function correctly. Once again, each student was required to serve as the driver on one project and the navigator on the other. As with the preceding projects, the teacher/researcher took observational notes regarding the level of collaboration among the students and the time spent on task.

Conclusion

Following the four-week intervention, a post-attitudinal Likert-scale survey (see Appendix D) and open-ended questionnaire (see Appendix E) were administered to the students in order to gauge any significant change in the students' attitudes toward group learning or the subject of computer science. The qualitative and quantitative data obtained during this four-week study will be analyzed in Chapter 4.

Chapter 4
Data Analysis

Introduction

Working in collaborative learning groups is important for technology students because it resembles the type of work done in university and industry settings (Pastel, 2015). Although much research over the years has shown that group learning is ideal for technology students, it is still not used enough in the high school classroom because of resistance from both teachers and students. The purpose of this study was to have a class of advanced-level computer science students work collaboratively on a set of group projects, rather than the individual programming assignments typically given in advanced technology classes. It was anticipated that the fourteen participating students would still be engaged while working in collaborative groups on their programs, with little time spent off-task, and their attitudes toward this class structure would not change negatively as a result of working collaboratively as opposed to individually. Over the course of the four-week project, the students completed computer programming projects using the “pair programming” paradigm. When using this collaborative method, one person acts as the “driver” who types the code to solve the task at hand and dictates their progress to their partner. Their partner assumes the role of “navigator,” who reviews the driver’s work as they type and simultaneously thinks ahead about how to solve the next part of the programming problem. The teacher/researcher assigned each participating student a partner for the first two weeks of the study, thus dividing the fourteen students in the class into seven pairs. The students worked for the first week in their chosen role of driver or navigator, and then the partners switched roles for the second week. The process was repeated for the third and fourth weeks of the study, but during those weeks, the collaborative groups were chosen by the students in the class instead of being assigned by the instructor.

Results of the Data

Quantitative Data

Attitudinal Likert Scale Surveys

The teacher/researcher created and administered an attitudinal Likert scale survey both pre- and post-intervention. The intention of the pre-intervention survey was to measure the students' feelings about this particular advanced technology class, and to determine their preferences and prior experiences with regard to working collaboratively on group projects. The results of the pre-intervention survey are presented in Table 4-1 below.

Table 4-1
Student Pre-Intervention Attitudinal Likert Scale Survey Results

	Strongly Disagree (1)	Disagree (2)	Neither Agree Nor Disagree (3)	Agree (4)	Strongly Agree (5)
I enjoy this class.		1 7%	1 7%	6 43%	6 43%
I enjoy this class more than most of my other classes.		2 14%	1 7%	7 50%	4 29%
I enjoy working on group projects in school.	1 7%	1 7%	1 7%	6 43%	5 36%
I would prefer to work on group projects in this class instead of individual assignments.	2 14%	2 14%	4 29%	5 36%	1 7%
When I have to work on a group project, I often contribute more than most of the other members.			3 21%	8 57%	3 21%
I like group projects because it means I have to do less work than on an individual project.	2 14%	3 21%	5 36%	4 29%	
All members who work on a group project should receive the same grade.	2 14%	6 43%	1 7%	4 29%	1 7%
I often get assigned group projects in math class.	9 64%	4 29%	1 7%		

The post-intervention survey asked the same questions in order to determine if the attitudes of the students had changed as a result of the collaborative learning experience intervention. Later questions were altered to inquire about their attitudes towards the "pair programming" projects they had just completed, rather than group projects in general. The results of the post-intervention survey are presented in Table 4-2 below.

Table 4-2
Student Post-Intervention Attitudinal Likert Scale Survey Results

	Strongly Disagree (1)	Disagree (2)	Neither Agree Nor Disagree (3)	Agree (4)	Strongly Agree (5)
I enjoy this class.			3 23%	4 31%	7 54%
I enjoy this class more than most of my other classes.		2 15%	2 15%	5 38%	5 38%
I enjoy working on group projects in school.		1 8%	5 38%	5 38%	3 23%
I preferred to work on these group projects instead of our previous individual assignments.	2 15%	2 15%	5 38%	5 38%	
On our group projects, I often contributed more than the other member.		1 8%	6 46%	5 38%	2 15%
I liked these group projects because I had to do less work than on an individual programming project.	2 15%	4 31%	3 23%	5 38%	
Both members of these group projects should receive the same grade.			4 31%	7 54%	3 23%
I often get assigned group projects in math class.	10 77%	4 31%			

As part of the reasoning for the study, the teacher/researcher pointed out that group projects are rarely assigned in both technology and mathematics courses. The results of the pre-intervention Likert scale survey verify these thoughts. When presented with the statement, "I often get assigned group projects in math class," every student with the exception of one (93%) gave the response "Disagree" or "Strongly Disagree." Similarly, on the post-intervention survey, all students (100%) disagreed with the statement to some degree. On the post-intervention open-ended questionnaire (see Appendix E), half of the students surveyed said they had “never” worked on a group project in a mathematics class (see Appendix G). The results of both surveys seem to justify the feelings of the teacher/researcher that collaborative learning remains underused in the high school mathematics classroom.

One hypothesis of the teacher/researcher was that the implementation of collaborative learning projects in the high school technology classroom would not affect the attitudes of the students towards the class in a negative way. Once again, the data of the Likert scale surveys seem to verify these feelings. On the pre-intervention survey, when presented with the statement “I enjoy this class,” 12 of the 14 students said “Agree” or “Strongly Agree.” One student disagreed and another said neither. The post-intervention survey was very similar, with 11 students agreeing in some fashion, and 3 saying neither. When assigned a mean score, the pre-intervention survey average for this question was a 4.21 compared to a 4.29 for the post-intervention survey, showing an actual increase in student enjoyment as a result of the collaborative learning intervention.

Student Grade Analysis

As stated earlier, one reason for student resistance to collaborative learning experiences is concern for how working with a partner may affect their project grades. Over the course of this four-week intervention, the students worked on five “pair programming” projects. All students in the class earned project averages in the A-range both during the study, as well as the six individual projects proceeding it, showing no significant change in academic performance. More specifically, five students earned a higher project average, and four students saw a negative change of three percent or less. No student had a change of over ten percent, signifying a change in letter grade. It should be noted that the majority of the projects contained extra credit options, so even the two student individuals with a project average grade of 87.38% were technically in the A-range when extra credit was factored in. Later projects completed during the intervention also covered more advanced material of greater difficulty to most students. A comparison of pre- and post-intervention project averages can be found below in Table 4-3.

Table 4-3
Student Project Average Comparison

Student #	Pre-Intervention Project Average	Post-Intervention Project Average	Change in Project Average
1	90.48%	94.17%	3.70%
2	96.83%	90.29%	-6.53%
3	93.65%	96.12%	2.47%
4	95.24%	93.20%	-2.03%
5	96.83%	95.15%	-1.68%
6	96.83%	87.38%	-9.45%
7	99.21%	96.12%	-3.09%
8	88.10%	89.32%	1.23%
9	90.48%	91.26%	0.79%
10	92.06%	94.17%	2.11%
11	96.03%	93.20%	-2.83%
12	96.83%	91.26%	-5.56%
13	96.83%	87.38%	-9.45%
14	96.83%	96.12%	-0.71%

Qualitative Data

Instructor Field Notes

In addition to the quantitative data collection performed by the teacher/researcher, there was also qualitative data recorded in the form of observational field notes. While students worked collaboratively in pairs, the teacher/researcher observed each group for an extended period of time. Notes were taken that addressed the following: level of student collaboration, ability to maintain the role of either driver or navigator, and significant spans of time spent off-task in any way. Though the teacher/researcher was often interrupted while observing in order to assist students, the field notes provided valuable insight to those aspects of the study.

One anticipated outcome of the study was that students would not spend more of their time off-task when working collaboratively as they had when working on individual projects. According to the field notes of the teacher/researcher, the students worked very diligently overall. During the course of the entire four-week invention, only two students were observed to be off-task for any length of time. These students were also the two who had been struggling the most academically with the class material before the study had started, suggesting that their off-task behavior may have had less to do with the change to collaborative learning projects and more to do with their inclination to wander off task in general. There did not seem to be any incidents of groups avoiding work by having personal conversations, even when working with the partners they had chosen themselves. In the opinion of the teacher/researcher, the students seemed to feel a responsibility to their sole partner to provide assistance, especially since there were no other group members to help.

Along with having students remain on-task, the teacher/researcher was also adamant that the groups adhere to the “pair programming” paradigm due to its use at both the university and industrial levels. One issue documented repeatedly in the observational field notes was how some students had to be reminded more than once of staying in their role as either driver or navigator. The student responses to the post-intervention survey verify the observations of the teacher/researcher (see Appendix G). When asked for the most challenging aspect of the "pair programming" projects, nine of the fourteen students cited their frustration at only serving in their designated role of either driver or navigator. In the opinion of the teacher/researcher, this difficulty was due more to the inexperience of the students using the "pair programming" paradigm than it had to do with any unwillingness to work in their assigned roles. This issue of inexperience with collaborative learning in the technology classroom will be discussed further in the section below.

Limitations of the Study

There were many limitations to the study overall. A major limitation to the study was the inexperience of the students with regards to group learning, particularly within the setting of a high school technology class. Since the teacher/researcher primarily used individual programming projects in this computer science class, and most students had no prior programming experience, they had never worked on this type of project in a group setting at any level. If the study had taken place over the course of the entire school year, as opposed to only four weeks, it would have allowed the use of group projects to be gradually phased in. As documented repeatedly in the field notes of the teacher/researcher, some students had difficulty maintaining their role as driver or navigator. If the students had already been somewhat acclimated to "pair programming" assignments, these problems would have likely been avoided or at least minimized as the academic year progressed. A longer study would also have allowed for more variety in the grouping of students so as to account for variables like student personality and ability level. On the post-intervention free-response survey, 12 of the 14 students expressed a preference in some way for choosing their collaborative partner. While some spoke of personality and communication issues, others spoke of differences in ability between the members. A longer study would have allowed more time to take these issues, which are of obvious importance to high school students, into account when forming collaborative groups.

The teacher/researcher also faced obstacles when attempting to take accurate field notes. Not only were there seven pairs of students to observe, but there was still a need for the instructor to provide assistance on the programs while the groups worked on them. Even when choosing to focus on a single or a small subset of pairs, the teacher/researcher was constantly called away to assist others in some way. Since the intervention took place in an advanced-level computer science class, the pacing of the course must always be considered in order to prepare students for the end-of-course standardized exam. Therefore, only a certain amount of time can be allocated to each set of projects, and questions need to be answered to allow students to continue with the programming task at hand. Once again, a longer study would have allowed the teacher/researcher to focus on a single group for a longer period of time and spread the study over a larger set of assigned projects.

Another significant limitation to the study was due to student absences, particularly during the first phase. The study began at the beginning of the third marking period, following midterm examinations. At the high school where the study took place, there were a particularly high number of absences around this time, and the school needed to schedule nearly 150 make-up midterm exams as a result. With a school-wide absence problem, the attendance of the class was affected as well. There were 3 days of programming work in the first week, and only two of the seven groups had full attendance for those classes. In the four days of the second week, only three groups were without an absence. During the third week of the study, there was also an assembly for all twelfth-grade students during one class. Since the study took place in an advanced-level class with mostly senior students, only one group was left with both members. Most of the time, in order to keep the pace of the course, the teacher/researcher had the students work individually on the group assignments when absences occurred. Although the purpose of the intervention was to provide the same collaborative working environment as college students and industry professionals, high school students face more impediments to making up time for lost work on group projects.

A final major limitation of the study came from its interruption due to the COVID-19 virus outbreak and closing of school facilities. The closing of the school building was announced at during the fourth week of the study, after only one day of work on the "pair programming" assignments. It was announced as a closing of only one week, so it was anticipated that the study could continue after a week of remote instruction. After the school closing was extended, eventually indefinitely, the study had to be ended as some of the students of the class did not have the necessary programming or video-conferencing software installed on their home computers yet, and the due date for the completion of the research project was rapidly approaching. This interruption denied the students the ability to spend more than one day in their role as driver or navigator during the second phase of the project, and forced the elimination of the final "pair programming" assignment as well. Therefore, it is difficult to say how the results of the post-intervention surveys may have changed had the study been able to run to its planned completion.

Implications for Teaching

Collaborative learning faces a great deal of resistance from both high school students and teachers, particularly in the mathematics and technology classrooms. But the quantitative and qualitative data collected during this study show that it can be used effectively without changing the attitudes or grades of students adversely. However, the intervention also shed light on some issues in the high school classroom that are not as significant at the university and industry level. Student absences present a great hinderance to collaborative learning due to the lack of flexibility in the schedule of an adolescent high school student as compared to an adult. Any effort to implement group projects in a technology classroom must take student absences into consideration, while maintaining content pacing and progress. The data analyzed also shows an overwhelming preference among high school students for choosing their collaborative partner. While the study did not inquire as to the partner preference of those in college or industry, it stands to reasons that the psychology and emotions of a high school student would differ as compared to those of a mature adult.

Overall, the study showed the teacher/researcher the great value of a collaborative learning environment in a high school computer science classroom. While completing their group projects using the “pair programming” paradigm, students remained on task to the same degree as when working independently previously. Students seemed to feel a duty to help their one and only partner by focusing on the task at hand and providing genuine support. Also, the students’ academic performance and attitude toward the class remained relatively the same. When implemented in the future, the teacher/researcher plans to slowly phase in the use of “pair programming” to give the students a great deal of prior experience in collaborative learning and, at the same time, provide more detailed explanations of the driver and navigator roles. More consideration will also be paid as to how the students will be grouped, with major factors being similar programming ability and rapport between partners. Other factors like frequency of absence should be given consideration as well.

Conclusion

Collaborative learning is an important skill for all students to develop, particularly those in the technology classroom. The purpose of this study was to see if a series of group projects could be implemented effectively in a computer science classroom where primarily individual projects were used. The teacher/researcher was particularly concerned that the academic performance and attitudes toward the class would not be negatively impacted. Of additional interest was that of having students remain on task just as they had before when working independently. Through the use of both qualitative and quantitative data, the intervention has shown it is likely that student performance and attitudes can remain consistent whether work is done in an independent or collaborative manner. However, for maximum effectiveness in a high school setting, it seems that particular attention needs to paid to the prior experience of students with collaborative learning, as well as how the groups are selected.

References

ACM/IEEE-CS Joint Task Force on Computing Curricula. (2001). *Computing curricula 2001*. Retrieved from www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2001.pdf

Black, K.M. (1994). An industry view of engineering education. Journal of Engineering Education, 83(1), 26-28.

Blake, M.B. (2005). Integrating large-scale group projects and software engineering approaches for early computer science courses. *IEEE Transactions on Education*, 48(1), 63-72.

Borstler, J., & Hilburn, T.B. (2015). Team projects in computing education. *ACM Transactions on Computing Education*, 15(4), 16:1-16:5.

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

Gozalo, E.S., Hernandez-Fernandez, M.A., & Ferrer-i-Cancho, R. (2017). Does like seek like?: The formation of working groups in a programming project. *Journal of Technology and Science Education*, 7(2), 231-240.

Henry, J. (1994). *Teaching through projects.* London: Kogan Page.

Hughes, R. & Cotterell, M. (2002). *Software project management (3rd ed.).* London: McGraw-Hill.

Joy, M. (2005). Group projects and the computer science curriculum. *Innovations in Education and Teaching International*, 42(1), 15-25.

Lang, J.D., Cruse, S., McVey, F.D., & McMasters, J. (1999). Industry expectations of new engineers: A survey to assist curriculum designers, *Journal of Engineering Education*, 88(1), 43-51.

Le, H., Janssen, J., & Wubbels, T. (2018). Collaborative learning practices: Teacher and student perceived obstacles to effective student collaboration. *Cambridge Journal of Education*, 48(1), 103-122.

Lingard, R., & Barkataki, S. (2011). Teaching teamwork in engineering and computer science. *FIE '11 Proceedings of the Frontiers in Education Conference*, T1A-1-T1A-5, Rapid City, SD.

Marshall, L., Pieterse, V., & Thompson, L. (2016). Exploration of participation in student software engineering teams. *ACM Transactions on Computing Education*, 16(2), 5:1-5:38.

Matusovich, H., Streveler, R., & Miller, R. (2009). We are teaching engineering students what they need to know, aren't we?. *Proceedings of the 39th IEEE International Conference on Frontiers in Education Conference*, 1254-1259, San Antonio, TX.

Miller, L.D., Soh, L., & Peteranetz, M.S. (2019). Investigating the impact of group size on non programming exercises in cs education courses. *SIGCSE '19: Proceedings of the 50th ACM Technical Symposium on Computer Science Education*, 22-28, Minneapolis, MN.

Pastel, R., Seigel, M., Zhang, W., & Mayer, A. (2015). Team building in multidisciplinary client-sponsored project courses. *ACM Transactions on Computing Education*, 15(4), 19:1-19:23.

Reynolds, M. (1994). *Group work in education and training*. London: Kogan Page.

Waite, W. M., Jackson, M. H., Diwan, A., & Leonardi, P. M. (2004). Student culture vs group work in computer science. *ACM SIGCSE Bulletin*, 36(1), 12-16.

Appendix A

CALDWELL UNIVERSITY

CONSENT FOR PARTICIPATION IN A RESEARCH STUDY

Study Title: The Effect of Cooperative Learning Groups on the Learning, Collaboration, and Attitudes of Students in a High School Software Design Class.

Principal Investigator: Franco A. Antonucci, Candidate for Master of Arts degree in Curriculum and Instruction, Caldwell University, Caldwell, New Jersey 07006

Faculty Sponsor: Edith Dunfee Ries, Ed.D. Faculty Member, School of Education, Caldwell University, Caldwell, New Jersey 07006. Action Research Advisor

KEY INFORMATION:

- Consent is being sought for your child to participate in an action research project. Participation in the study is completely voluntary and may be withdrawn at any time.
- I am a candidate for a Masters Degree at the above University and I am conducting a four-week study on the effects of group learning in a high school computer science classroom. This is the capstone project for the graduate degree at the University. Over the course of the research project, students will continue to produce software programming projects, but with an increased emphasis on teamwork and collaboration.
- There are no foreseeable risks associated with participation in the study or anything that should cause discomfort to your child.
- Benefits to students participating in the survey include increased experience with the "pair programming" software design technique that is used extensively in universities and industry, as well as the other documented benefits of collaborative learning.

STUDY INFORMATION:

What is the purpose of this study? I am a candidate for a Masters Degree and I am conducting a four-week capstone action research project in order to study the effects of group learning in a high school computer science classroom. Collaborative software design is used extensively in universities and the workplace so I wish to study the effects of its increased use at the high school level.

Who is participating? The 14 students in AP Computer Science Principles class have been chosen to participate in this study.

What will happen during the course of this study? I am going to be the teacher/researcher of the four-week action research project. During the study, students will continue to be required to complete software design projects, but with an increased emphasis on working with a partner.

Appendix A (Continued)

CALDWELL UNIVERSITY

While collaborative learning has always been a key part of the technology classroom, "pair programming" is an innovative approach to software design that is being increasingly implemented at the university and industry levels. Rather than completing projects individually, students will be required to design a solution in collaboration with a partner. Parents may be interested in knowing that I plan to collect both qualitative and quantitative data in the following methods:

- An attitudinal survey will be given to each student before the action research project to gain insight into their feelings towards collaborative learning based on their prior experiences in other classes.
- The academic grades of the students' collaborative learning group projects completed during the action research project will be analyzed for comparison with the participants' project grades prior to the study.
- An additional attitudinal survey will be given to each student after the conclusion of the action research project to gain insight into any changes in feelings towards collaborative learning due to their experiences during the action research project.

What are the risks if your child participates in this action research ? There are no foreseeable risks associated with your child's participation in this action research project.

How might your child benefit if you participate? Participants in the study will gain valuable experience programming in collaboration with a partner, which is one of the dominant methods of software design both at university and industry levels. There are also numerous documented social and academic benefits to collaborative learning.

What will happen to your information after the study is over? I am conducting this study and collecting this data for an action research project which is the capstone requirement for completion of my Masters Degree. In May 2020 the study will be published as a four-chapter document and will be available online. The final document will be accessible to any parent upon request.

Your child's participation is voluntary. I would like to reiterate that participation in this four-week study is completely voluntary and you may withdraw your child's name from the list of participants at any time.

PARENT OR LEGALLY AUTHORIZED REPRESENTATIVE PERMISSION:
By signing this document, you are agreeing to your child's participation in this study. Make sure you understand what the study is about before you sign. I will give you a copy of this document for your records. I will keep a copy with the study records. If you have any questions about the study after you sign this document, you can contact the study team using the information provided above.

Appendix A (Continued)

CALDWELL
UNIVERSITY

I understand what the study is about and my questions so far have been answered. I agree for my child to take part in this study.

____
Printed Subject Name

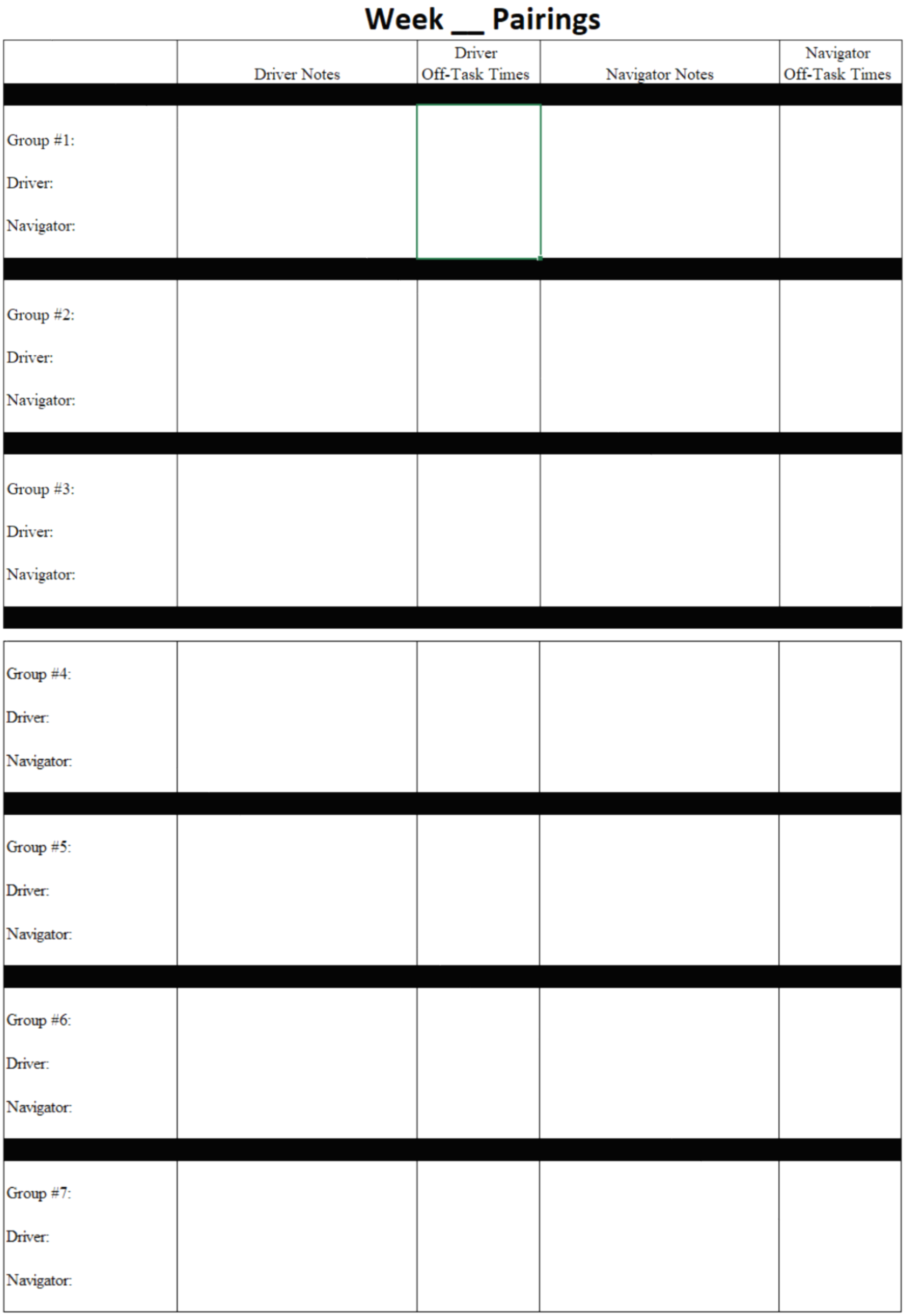
____
Printed Parent/Legally Authorized Representative Name and Relationship to Subject

Thank-you, Mr. Franco A. Antonucci

Appendix B
Pre-Intervention Likert Scale Survey

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
I enjoy this class.					
I enjoy this class more than most of my other classes.					
I enjoy working on group projects in school.					
I would prefer to work on group projects in this class instead of individual assignments.					
When I have to work on a group project, I often contribute more than most of the other members.					
I like group projects because it often means I have to do less work to do than on an individual project.					
All members who work on a group project should receive the same grade.					
I often get assigned group projects in math class.					

Appendix C
Instructor Field Notes



Week __ Pairings

	Driver Notes	Driver Off-Task Times	Navigator Notes	Navigator Off-Task Times
Group #1: Driver: Navigator:				
Group #2: Driver: Navigator:				
Group #3: Driver: Navigator:				
Group #4: Driver: Navigator:				
Group #5: Driver: Navigator:				
Group #6: Driver: Navigator:				
Group #7: Driver: Navigator:				

Appendix D
Post-Intervention Likert Scale Survey

	Strongly Disagree	Disagree	Neither Agree nor Disagree	Agree	Strongly Agree
I enjoy this class.					
I enjoy this class more than most of my other classes.					
I enjoy working on group projects in school.					
I preferred to work on group projects in this class instead of individual assignments.					
On our group projects, I often contributed more than the other member.					
I liked these group projects because I had to do less work than on an individual project.					
Both members of these group projects should receive the same grade.					
I often get assigned group projects in math class.					

Appendix E
Post-Intervention Open-Ended Questionnaire

Do you enjoy working on group projects in general? Why do you like about them? What do you not like about them? (Be specific)

Did you prefer working with a partner on these recent pair programming assignments instead of individual programming assignments? Why or why not?

Did having the option of choosing your partner instead of an assigned partner change how you felt about these projects? Why or why not?

What was the biggest challenge of the pair programming projects? Be specific.

How often have you worked on group projects in math class?

Appendix F
Observational Field Notes

Week 1		
Group	**Driver Notes**	**Navigator Notes**
1	Day 2: Partners are working well together. Driver is explaining her thought process while she is coding. Good job asking navigator for verification and more ideas. Day 3: Absent	Day 1: Absent Day 2: Great communication between partners, both are on the same page. Navigator needs to provide more guidance about how to proceed. Most assistance to the driver is being provided by the teacher.
2	Day 2: Tells the driver what she is doing. Asks for help when she needs it and accepts advice of the navigator. Tries to work with partner to figure things out instead of asking teacher. Day 3: Absent	Day 2: Has a good idea of what the driver is doing and is able to assist with syntax. Essentially performing the same role as driver though. Both are thinking about same portion of the programming task.
3	Day 1: Making good progress on program and recognizes how to accomplish the task at hand. Needs to explain more of his process and reasoning to the navigator as he is typing. Day 2: Absent Day 3: Absent	Day 1: Lack of input overall. Student is not recognizing small errors by the driver and has no advice on how to proceed to the next section of the program. (Off-task 9:39 - 9:43) Day 2: Absent
4	Day 1: Needs to tell the navigator what he is doing. This lack of explanation could be why the navigator may be having a problem giving assistance. Day 2: Absent Day 3: Absent	Day 1: Very quiet, providing little input to the driver on how to proceed and not seeing any of the multiple syntax errors of the driver. Driver has a clear concept of what to do and where to go next and navigator may be intimidated by speaking up.

Appendix F (Continued)

5	Day 3: Great job coding the program but needs to dictate his thinking to the navigator. Navigator will then have a better idea of where to go next on the program.	Day 1: Took control of the keyboard from the driver. Reminded to explain what to do to the driver. Day 3: Much better job overall staying in role of navigator. Student is providing feedback as well as assistance on how to proceed.
6	Day 2: Very good job explaining to the navigator. Clearly conveys what he is doing as he is doing it.	Day 2: Great job backing up the driver. Understands the driver's thought process and explains what to do next. Also recognized errors like counter out of bounds. Needs to show more patience - let the driver make mistakes so there's no interruption of his process. Then, go back and fix mistakes.
7	Day 2: Great job explaining his thoughts and reasoning to the navigator. Very accepting of advice and input of the navigator as well. Day 3: Absent	Day 2: Good assistance but he is helping the driver too much with the current task. Driver should focus on the current task and explain his progress while the navigator reviews and thinks ahead.
Week 2		
Group	**Driver Notes**	**Navigator Notes**
1	Day 1: Absent Day 2: Absent Day 3: Explains her process well. Has a good idea about how to proceed and is explaining each step to the navigator. Quick to ask the teacher a question rather than work through it with partner. Day 4: Still working very well together. Getting much more vocal as time progresses.	Day 1: Absent Day 3: Good assistance overall. Following along with the driver and providing error checking when needed. Not as helpful with how to proceed on program.

Appendix F (Continued)

2	Day 1: Serving in last week's role to complete work missed due to absence. Good communication with navigator. Does a good job of asking for verification as she is typing, resulting in little time lost.	Day 1: Good suggestions to driver on how to proceed. Have a good back-and-forth going. Got out paper to work through strategy with driver resulting in little need for teacher assistance.
3	Day 1: Absent Day 2: Absent Day 3: Went directly to bathroom despite 2 prior days of absences. (Off-task 9:23-9:29) Day 4: Had to be told to work with partner. No effort to assume the role of driver and take the lead. Partner is serving in the role and getting almost no assistance. (Off-task 8:27-8:31)	Day 3: Took over role of driver. Probably used to little interference due to partner absences. Reminded to now serve as navigator but continues driving when teacher is not monitoring. Day 4: Started work with no effort to work with partner. Continuing to drive and getting little assistance even with roles reversed.
4	Day 1: Serving in last week's role to complete work missed due to absence. Day 4: Worked on extra credit for earlier program while partner worked on current program. Split work instead of working together as a pair. (Off-task 9:32-9:35)	Day 1: Confusing roles. Acting as both driver and navigator at different points. Might be used to working along due to partner absences of prior week. Day 2: Absent Day 3: Took over driving instead of explaining how to proceed. Difference in ability level between partners seems evident and navigator is getting impatient.

Appendix F (Continued)

5	Day 1: Serving in last week's role due to absences. Great job navigating. Day 2: Good job explaining process as he types. Partner is able to follow along and is accepting of suggested changes. Day 3: Continue to work well together. Work seems to be progressing at a smoother pace as well.	Day 2: Great job working with navigator. Able to review driver's work while simultaneously thinking ahead about next part of program. Day 3: Great collaboration continues. Despite being paired by teacher, students function very well as a team.
6	Day 1: Good collaboration continues, even with roles reversed. Driver is dictating his thoughts as he types. He asks questions to the navigator making sure he is on the right track. Day 3: Finished all work and extra credit.	Day 1: Provides great advice to the driver about where to go next. Good job maintaining the role overall. Interacts with partners extremely well. Day 3: Finished all work and extra credit.
7	Day 2: Great job progressing on the program and working with the navigator. Needs to verbalize his thought process more to the navigator. Day 3: Worked on extra credit	Day 1: Absent Day 2: Great support. Thinks ahead to next part of program but also gives positive feedback to driver's progress.
	Week 3	
Group	**Driver Notes**	**Navigator Notes**
1	Day 1: Giving much more input into program this time around. May be due to working with partner of choice now. Able to finish first project.	Day 1: Good job with assistance and has clear idea of how to proceed. Needs to wait for driver to make mistake and then provide help. Don't interrupt the driver's progress, let them finish their idea.
2	Day 2: Partners have a great back-and-forth going. Driver is focusing on current part of program and explaining as he types.	Day 2: Doing a great job of verbalizing where to go next. Asking for input where necessary but mostly knows what needs be done.

Appendix F (Continued)

3	Day 3: Partners working well together. Keeping same partners seems to have been a good choice. Both students should seek extra help though.	Day 3: Providing more assistance to partner. May be due to being more comfortable with driver after two prior weeks of working together.
4	Day 2: Driver is fulfilling both roles. Has a discussion with partner before coding, then asks for review, then discuss where to go next. Flow should be more seamless especially for partners who already worked together.	Day 2: Confusion of roles on both sides. Partners discuss task at hand, coding takes place, then discuss next steps. Leads to delays. (Off-task 9:38-9:40)
5	Day 3: Not dictating much of his process to the navigator. May not feel that it is necessary because program is mostly written and he already knows what to do next.	Day 1: Absent Day 2: Absent Day 3: Providing checking of syntax to the driver, but not much else. Probably due to inexperience with program after recent absences.
6	Day 1: Driver has a clear idea of what to do. Needs to provide more explanation to the navigator. Driver is mostly working independently.	Day 1: Needs to provide more input to driver. Not helpful or giving guidance. Content to sit and watch. May be due to driver having a clear idea what to do and navigator does not.
7	Day 1: Great communication from both sides. Program is progressing with no delay. Finished first project.	Day 1: Both partners constantly on task. Work extremely well together and can discuss their process as they are working with no issues at all.
Week 4		
Group	**Driver Notes**	**Navigator Notes**
1	Day 1: Needs to explain more to navigator. Has a clear idea of what needs to be done but navigator does not. Clear difference in ability level between partners.	Day 1: Driver is pretty much doing all the work. Navigator not giving much input or assistance. Seems to be more a lack of knowing what to do than laziness. (Off-task 8:40-8:43)

Appendix F (Continued)

2		Day 1: Providing good support. Keeping an eye on what the driver is doing but also thinking ahead. Gets partner started and lets him proceed.
3	Day 1: (Off-task 8:50-8:54)	
4		Day 1: Went home 30 minutes into class period.
5	Day 1: Great job talking through each step.	Day 1: Great advice about next task to accomplish. Excellent cooperation and teamwork overall.
6	Day 1: Need to tell the navigator what you're doing. Student knows what to do, just seems uncomfortable with verbalizing it.	Day 1: Good support. Even with lack of talking from partner, he is able to follow along.
7		

Appendix G
Post-Intervention Open-Ended Questionnaire Responses

Question	**Student Responses**
Did having the option of choosing your partner instead of an assigned partner change how you felt about these projects? Why or why not?	Generally, being allowed to choose a partner makes the project easier, or at least less frustrating. I think this holds up with these partner projects but doesn't affect the overall project too dramatically.
	Yes, having the option of who my partner was changed how I felt about the assignment. Even though I did not change partners the second time having a partner I knew made me much more comfortable. Sometimes if you are assigned a partner who you do not know or who messes around too much can change the amount of effort that a project gets.
	I liked being able to choose my partner much more. I hate when I am randomly assigned a partner.
	It did change how I felt, I enjoyed working on the project more with someone who I knew.
	Yes, but only because I prefer picking a partner rather than being assigned one
	Yes that definitely did because when I was able to choose, I was able to choose someone who was at my skill level. This way we contributed more equally than the previous project where I felt like I was teaching a lot.
	Being able to choose a partner didn't affect how I felt about these projects because I honestly have no friends in this class anyway so I just went with whoever else needed a partner
	I enjoyed choosing my partner more because it allowed me to be more comfortable so that I did not feel like holding back any comments
	Choosing a partner is immensely better than being assigned one because I would rather pick a partner that could help me and contribute not the opposite.
	No, I feel like was good that we had a choice. It felt the same because I have the same partner.
	Yes I liked choosing my partner instead of being assigned because I felt more comfortable and wasn't nervous to add my input.

Appendix G (Continued)

Did having the option of choosing your partner instead of an assigned partner change how you felt about these projects? Why or why not?	Choosing your partner instead of an assigned partner did not change how you felt about these projects because everyone is trying to complete the same task no matter who I am paired with. However, I do believe that it is easier to talk and communicate to someone who you know as you are more comfortable with him/her.
	Absolutely, choosing a partner was a well needed change. Although this is an intro to computer science based class, there are many kids who have already taking other computer science classes. For this reason, some are more knowledgeable than others. So if a true beginner students got matched with a more experienced computer science student, the true beginner wouldn't have much to contribute as the navigator, and when they were the driver they were constantly barked commands of how to do what. But when you get to choose a partner, it's much better to seek out someone who works at the same pace as you and has a similar understanding of the subject.
	It did not because the aspects that I did not like had nothing to do with the person I was working with.
What was the biggest challenge of the pair programming projects? Be specific.	The biggest challenge as the navigator was trying to tell the driver where exactly to type, as we couldn't just take the computer for a moment to place them at the right spot (especially considering the number of similar statements and braces the program had). As a driver, the hardest part was figuring out what the navigator was saying, as they often were vague.
	The biggest challenge of the pair programming projects were remembering to explain what you were working on and to let your partner try to fix their own mistakes before you explain to them what they did incorrectly.
	I believe that not being the driver aggravated me because it was difficult to do a program without being able to type it.
	The biggest challenge was trying to get my vision of the program to the other person at the beginning.

Appendix G (Continued)

What was the biggest challenge of the pair programming projects? Be specific.	When I was the navigator, it was difficult for me to be able to see the code on the screen
	The challenge is being the navigator and having to think what to say when you can't just type it out.
	The fact that only one person could "drive" made it difficult. The person who was not "driving" was forced to be very specific when directing the driver what to fix or what to add. I felt that it would've been much easier and more efficient if both partners could type out the program.
	The biggest challenge is making sure that both members of the group are sticking to their assigned roles.
	Mainly deciding which way to go about things but that was easily solved with a discussion.
	The biggest challenge was figuring out what little details we missed to get it right.
	The biggest challenge of pair programming projects was when the other partner was absent and I wasn't able to work on the program we had been working on the previous day and then we were set back.
	The biggest challenge is agreeing on an idea when you both have different ones. You don't want to be the person who has the wrong idea, so it is challenging to choose between different ones because you don't want to look like the bigger "fool". The best way to work this challenge out is to talk gently to each other about the process to see which one works in this particular project.
	The biggest challenge was only having one computer, it made for a very uncomfortable situation. Although multiple people working on a program seems very realistic, they would never do it in this fashion. One person would work on one aspect of the program and another would work on a different part of the program. If this method was conducted in the real world, it would be very inefficient since only half the amount of work would get done.
	Trying to understand one another when thinking of code. I found it difficult to put my ideas into words since I hadn't before.

Appendix G (Continued)

How often have you worked on group projects in math class?	Never.
	Never
	Almost never
	When either our math teacher is absent or we are reviewing the day before a test, we will tend to split into groups to work or review; however, these aren't really "projects" per se, as opposed to just working with a group. We rarely get actual projects in math class, let alone group ones.
	Not often.
	Almost never to be honest.
	Pretty much never since we rarely do projects in math class.
	I've never worked on group projects in math class.
	In my experiences, I have never worked on group projects in math class throughout my time in school. Math classes are mostly lectures, learning the material, and tests. Since I have been mostly in the advanced math classes, teachers have had no time for group projects in class.
	I have never working on a group project in math class. Only partner project when the teacher says to work with the person sitting next to you to complete the assignment. In math class if you are only working with one other person it is great when you add more the risk for distractions is higher and less work is bound to get done.
	Almost never.
	Extremely Rarely
	Sometimes we collaborate in class on homework but other than that never. We don't usually do projects in math; they are very rare.
	Never.