The Effect of Cooperative Study Groups on Achievement of College-level Computer Science Programming Students

Approved by
Dissertation Committee:

The Effect of Cooperative Study Groups on Achievement of College-level Computer Science Programming Students

by

Laura Jean Baker, B.S., M. Ed.

Dissertation

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Doctor of Philosophy

The University of Texas at Austin

May, 1995

UMI Number: 9534716

Acknowledgements

To everyone who contributed so much to the completion of this work, I would like to express my sincere appreciation and gratitude:

Dr. Lowell J. Bethel, for his enduring support, guidance and expertise in the writing of this dissertation and in the completion of my studies at The University of Texas at Austin.

The members of my dissertation committee: Dr. John Huntsberger, Dr. Duane Shell, Dr. Benjamin Kuipers, and Dr. Barbara Boucher Owens for their expertise and helpful advice and assistance in completing this dissertation.

My parents, my two sisters, and my closest friends for their continued encouragement and love throughout the completion of my studies at The University of Texas at Austin.

The Computer Science faculty at St. Edward's University for their support and help during the completion of my degree.

May 1995 Laura Jean Baker

The Effect of Cooperative Study Groups on Achievement of College-level Computer Science Programming Students

Publication No. _______

Laura Jean Baker, Ph.D.

The University of Texas at Austin, 1995

Supervisor: Lowell J. Bethel

The investigation studied 81 undergraduate students enrolled in an introductory computer science programming course at a small four-year university. Students in four intact classes were assigned to either a control group (traditional individual learning method) or a treatment group (cooperative study group method) using a pretest/posttest experimental design. The length of the study was 15 weeks, equivalent to one semester. Students in the treatment group were assigned to small cooperative study groups of 4-5 students per group. Cooperative study group assignments were made randomly without regard to race, gender or academic ability. Students assigned to the cooperative study group method, N= 38, were instructed to work together on assignments, problem sets, and examination preparation.

Students in the control group, N=43, were instructed to study alone and to complete all assignments individually without help from other students. All grades of students in both groups were individually assigned, while no cooperative study group grades were given. Achievement and retention rates in the course were measured for students in both the control group and the treatment group.

Analysis of Covariance (ANCOVA) and Chi-square analyses were used to analyze the data. ANCOVA was performed on the achievement test results using the pretest scores as the covariate. There were no differences in achievement of all students in either control or treatment group. Minority students in the cooperative study group method did show a significantly higher achievement score on the posttest when compared to minority students in the traditional individual method. Retention rate of students was measured as the number of students completing the course with a grade of "C" or higher. No significant differences were found in the retention rate of all students including minority students. Female students in the cooperative study groups had a significantly higher retention rate than did female students in the control group. Cooperative study groups certainly effected minority student achievement and did improve the retention rate for female students. No negative effects on achievement or retention were found when cooperative study groups were implemented. An exit questionnaire filled out by all students revealed that students in the cooperative study group method found the cooperative study groups to be an effective learning strategy.

Table of Contents

List of Tables

Chapter I

The Problem

Introduction

The need for computer science graduates in the work force is expected to continue to grow through the year 2005 as computer usage continues to expand in business and industry. Computer programming, data communication, artificial intelligence, and information technology are areas that are predicted to provide the most employment for computer scientists (Kaspar, 1994; Occupational Outlook, 1994). Among the five occupations that are projected for the fastest growth from 1992 to 2005 are computer service technicians, computer and information systems analysts, and computer programmers (Statistical Abstracts, 1994, p.411). The work place for computer scientists continues to grow and change rapidly. Graduates of computer science programs must know several computer languages, be proficient at software design and maintenance, have knowledge of networking computers, and be able to solve business problems with information technology. Prospective employers expect new computer science graduates to be proficient in the latest computer technologies and to be able to adapt to the rapid hardware and software changes in the computer industry (Brown, 1993).

Student interest in science and engineering is declining and only 15 percent of college freshmen plan on majoring in a science or engineering field. While the number of bachelor's degrees conferred in science and engineering continues to drop the demand for engineers, scientists, and technicians is expected to exceed supply by more than 30 percent in the year 2000 (Block, 1990; Seymour, 1992; Hrabowski, F. & Pearson, W., 1993). Today, practically anyone who will have a Ph.D. in science or engineering by the year 2000 is already enrolled in college. Little progress has been made in the recruitment of women and minorities in science and engineering, even though minorities make up 30 percent of the student population today and are expected to rise to 40 percent by the year 2000 (Block, 1990).

In a study of college mathematics, science, and engineering majors, Tobias (1990) examined college students who were clearly able to pursue a degree in a science field yet switched to non-science careers by the time of their graduation. The study summarized three important factors contributing to the high attrition rate of science majors (includes computer science). These factors included the intense competition promoted in science classes, the lack of a "real-world" context for science concepts, and the absence of motivation to work harder than other students in non-science majors, and still be faced with the prospect of failure (Tobias, 1990). In another study, Seymour (1992) investigated potential science majors' reasons for leaving science to pursue degrees in other fields and found similar results. A large proportion of students leaving science did so for structural and cultural reasons, rather than lack of personal ability. The top five reasons for leaving science in pursuit of a different major included course overload of material, conceptual difficulty of subject matter, lack of support and the unapproachability of science faculty teaching courses, discouragement and loss of confidence in ability due to low grades in early years, and loss of interest in the subject matter (Seymour, 1992). The huge loss in numbers of potential students in the sciences is staggering in terms of loss of talent and productivity in the nation's future. Finding new teaching strategies and methods to retain these potentially good students in the sciences, particularly computer science, is necessary.

One of the potential problems for many students in computer science programs is their lack of success in the introductory courses. Introductory computer science courses at the college level usually have a high number of students who withdraw from the course without finishing or who fail to complete the course with a passing grade. The level of withdrawals and failures in first year programming courses is more than 25 percent (Campbell & McCabe 1984; Lipson & Tobias, 1991). Most of the withdrawals are by students who do not achieve success completing assignments and taking examinations. With the high withdrawal rate in introductory computer science courses it can be concluded that many students want to study in computer science but are unsuccessful in their introductory courses. These students must either retake the introductory course or change to a different major due to their lack of success in programming and computer science concepts (Ware, Steckler, & Leserman, 1985; Seymour, 1992).

Enrollment of women and minorities in computer science programs is less than 12 percent of the total number of student majors in computer science (Gries & Marsh, 1992). One of the challenges facing computer science programs at colleges and universities is the recruitment and especially the retainment of women and minority students because computer science is traditionally a white male-dominated field with few women and minorities as are many of the scientific disciplines (Widnall, 1988; Gries & Marsh, 1992; Seymour, 1992). Most computer science programs at colleges and universities require a rigorous degree plan of preparation for their students. While such challenging programs are necessary for the computer science graduate to be competitive in the employment marketplace, retaining potential graduates including minorities and women is often a problem in computer science departments (Frenkle, 1990; Seymour, 1992). Lack of sufficient support structures for students, unavailability and unwillingness of instructors to spend extra time helping students in computer science courses, and the traditional difficulties with mathematics and problem solving all collectively cause the high rates of attrition found in many computer science programs (Lipson & Tobias, 1991; Seymour, 1992).

One teaching strategy that has been highly successful at the elementary and secondary level is cooperative learning (Johnson, Johnson, & Scott, 1978; Johnson, Johnson, & Stanne, 1985; Yager, Johnson, & Johnson, 1985; Webb, Ender, & Lewis, 1986). Several different methods of cooperative learning have shown significant improvement in academic achievement in many fields including mathematics and science (Beilin, & Rabow, 1981; Johnson, Johnson, Stanne, & Garibaldi, 1989; Fullilove & Treisman, 1990; Borresen, 1992). With the high rate of success in improving academic achievement in various subject areas at both the elementary and secondary levels, it would seem that some form of cooperative learning at the college level would be equally successful in improving academic achievement and success of computer science students. Robert Fullilove and Uri Treisman (1990) at the University of California at Berkeley implemented a form of cooperative learning through the creation of study groups in an introductory calculus course at the University of California at Berkeley. In order to improve the success ratio of African American students in first year calculus, Fullilove and Treisman arranged study groups of four to six students to promote learning together so that students might have a support structure available to enable them to complete assignments and to learn problem solving in calculus. This leads to the current problem under investigation in this study.

Problem Statement

What effect on achievement does the use of cooperative study group methods in an introductory computer science programming course for college students have when compared to traditional individual learning methods used in similar courses?

Purpose

Because of the need for developing and maintaining successful computer science majors, it is necessary and important to investigate learning methods in computer science that may prevent or reduce the high number of withdrawals, lead to improved academic success in computer science, and encourage more students to remain in computer science programs as majors. The purpose of this study, then, is to examine a new learning strategy that may promote student success in an introductory computer science programming course. Introductory computer science courses are often rigorous and difficult for first year students. A successful experience for students in the first computer science course would motivate and encourage them to continue in the potential pursuit of a computer science major and degree. Student retention and success ratios do contribute to a strong academic program that can provide a dependable resource for employers of future computer science graduates.

Research Questions

The research questions under investigation in this study are:

1. How does the use of a cooperative study group method in an introductory college level computer science programming course effect achievement in computer science programming concepts when compared with a traditional individual study method in the same course?

2. Does achievement by minorities change in an introductory computer science programming course when a cooperative study group method is incorporated into the teaching methodology?

3. How does the achievement of women change when a cooperative study group method is incorporated into an introductory computer science programming course?

4. Will the retention rate of students in an introductory computer science programming class using a cooperative study group method be greater than, less than, or the same as the retention rate of students in a similar class using a traditional individual method?

5. Does the course retention rate of minorities improve when a cooperative study group method is incorporated into an introductory computer science programming course?

6. Does the course retention rate of women improve when a cooperative study group method is incorporated into an introductory computer science programming course?

7. Will the computer science major retention rate increase or remain the same when a cooperative study group method is used in an introductory computer science programming course?

8. Does the computer science major retention rate of minorities increase when a cooperative study group method is incorporated into an introductory computer science programming course?

9. Does the computer science major retention rate of women increase when a cooperative study group method is incorporated into an introductory computer science programming course?

Assumptions

There are several assumptions that govern the interpretation of results for this research study. Students who are enrolled in the introductory computer science programming course used in this study are attempting to successfully pass the course with the highest grades possible. Students actually intend to finish the course when they enroll. Those students who declare themselves as computer science or computer information science majors intend to complete a degree in that major. Students who are in the control group do not participate in study groups on their own, and if they do their data will be removed from the study.

Rationale for the Study

Cooperative learning, also called group learning, is defined as the establishment of groups in a class setting in which members of the group are dependent on each other to both teach and learn specific material. The group members are responsible for each others' learning and usually all group members receive the same grade for one or more assigned activities (Johnson, Johnson, & Stanne, 1985). Investigations of cooperative learning have been well documented for the past 15 years (Johnson, Johnson, & Scott, 1978; King, 1989; Sharan, 1980, 1990; Skon, Johnson, & Johnson, 1981;Slavin, 1980, 1983; Webb, 1982; and Yager, Johnson, & Johnson, 1985). Most of the research in cooperative learning has involved students at both the elementary and secondary school levels (Johnson et al, 1978, 1985; King, 1989; Sharan, 1980; Skon, Johnson, & Johnson, 1981; Slavin, 1980, 1983; Webb, 1982). Very few actual research studies exist that have investigated cooperative learning and its effects at the college level (Warring & Maruyama, 1986; Fullilove & Treisman, 1990).

Many of the studies done at either the elementary and secondary levels investigating the effect of cooperative learning on achievement in various subjects report an increase in achievement for those students participating in cooperative learning groups (Johnson et al, 1978; Sharan, 1980; Skon, Johnson & Johnson, 1981;Slavin, 1980, 1983; Webb, 1982; and Yager, Johnson & Johnson, 1985). Research at the college level is minimal in the area of cooperative learning in any subject area. While many descriptive articles appear about implementation of cooperative leaning in post-secondary education, very few quantitative research studies are documented at the college level (Brothen, 1986; Hufford, 1991; Johnson, Johnson, Stanne & Garibaldi, 1989; Radebaugh & Kazemek, 1989; Rau & Heyl, 1990; Lawrenz & Munch, 1984; Latting & Raffoul, 1991; Fitzgerald & Caulfield, 1992). Since cooperative group learning has proved successful in several studies at both the elementary and secondary school level, it would seem logical that the same success could be transferred to the college level.

The main reason for this study is to determine whether achievement improves when students are placed together in small cooperative working groups for the purpose of studying and completing computer laboratory assignments together. Several learning theories in instructional psychology apply to group learning. Not all students learn at the same rate nor in the exact same manner. Various stages of learning involve acquisition of declarative knowledge, development of cognitive skill, organization of knowledge representation, and restructuring and modification of existing schemata. Declarative knowledge refers to rules, principles, and vocabulary definitions. Development of cognitive skill involves the application of rules and principles to appropriate situations. Organization of knowledge representation relates to the means by which access to knowledge in memory is achieved through automization and integration of basic and advanced components of the knowledge itself. Restructuring and modification of existing schemata includes developing new connections between cognitive and procedural knowledge and declarative knowledge, and modification of existing structural links within the subject domain. Learning can be described as the progression from declarative knowledge to functional knowledge which is the appropriate application of declarative knowledge in various events related to the domain of the subject (Glaser & Bassok, 1989).

Working in cooperative groups provides students with opportunities to influence each other's knowledge base and to provide an avenue for developing new cognition based on other group members' ideas, concepts about the knowledge, and schemata. The opportunity to find out how others solve problems and to learn how to attain correct solutions to a particular problem in several ways would help with the organization of knowledge and the development of cognitive skills in students.

There are several different cooperative learning methods that have been developed and are in use today. The major types will be described later in this document. Research on the mechanisms of cooperative learning that actually influence achievement have been indirectly addressed in many studies. Knight and Bohlmeyer(1990) group the effects of cooperative learning on achievement into four categories including social influences, cognitive processing influences, academic task structure influences, and reward structure influences. These categories tend to overlap and interact with each other (Knight &Bohlmeyer, 1990).

Social influences include students giving feedback to one another, supporting and encouraging each other, and enjoying the social interaction with peers. In particular there are social awards when group members are dependent on each other and when the individual must work hard to perform well not only for him or herself, but for others who may be dependent on them. Cognitive processing influences results from students explaining concepts and processes to others, managing crises within the group, and restructuring concept explanations to help others in the cooperative group. Academic task structure influences include increasing time on task, interacting with other group members of differing abilities, and providing opportunities for success in achieving the highest possible grade for all members of the cooperative group. The need to work together and to communicate within the group emphasizes repetition of information and concepts. Cooperative groups allow students to clearly define problems with the assistance of others in the group preventing group members from being overwhelmed by the learning task at hand. Social interactions with peers generally keeps students on task. Reward structure influences are based on the establishment of norms for individual and group achievement and it is suggested that individuals are motivated to learn when contributing to the group's rewards (Knight & Bohlmeyer, 1990).

One form of cooperative learning found in the literature is the use of study groups in a first year college calculus course. Study groups were found to be defined in the literature by Fullilove and Treisman (1990) as groups of four to six students who met regularly outside of class to work together on assignments and problems and to study together for exams. None of the grades of students in study groups were interdependent. The research studies described later cover a wide variety of educational settings, yet almost all of them find achievement and attitude positively affected by cooperative group learning. Fullilove and Treisman (1990) researched study groups through a mathematics workshop program at the University of California at Berkeley

(UCB). Fullilove and Treisman's research was based on an informal study done on first year calculus students enrolled at UCB in 1975-1976. This investigation attempted to discover the key to Asian-American students' success in the first year calculus courses. The informal study found that the successful students worked and studied together in cooperative small groups. These students had formed informal study groups that met regularly and whose members helped one another solve problems and study together for exams. Based on this evidence, Fullilove and Treisman began a Mathematics Workshop Program (MWP) which was conducted from 1978 to 1984 at UCB. The program involved students working together in small cooperative groups of four to six students called "study groups". These study groups met outside of the regularly scheduled first year calculus class two times per week for approximately two hours per meeting. A graduate student was available at these meetings to hand out problems and to answer questions. The graduate student handed out worksheets with specific mathematics problems to students in the cooperative groups to solve during their study sessions. The primary responsibility of the students in the study groups was to help others solve the worksheet problems and to learn the basic concepts underlying the problems. Of particular interest to the researchers was the performance of African-Americans in the workshop. During the six years of the MWP, academic achievement was significantly better for participants of the workshop program, African-Americans in particular, and graduation rates in mathematics and engineering were much higher for the participants than for students not in the MWP. The MWP has been successfully adopted at other institutions such as the University of California at Los Angeles, the University of California at San Diego, California Polytechnic at Pomona (Fullilove and Treisman, 1990), and now currently at The University of Texas at Austin.

With the success of cooperative learning in elementary and secondary schools it seems reasonable to examine the use of some form of cooperative or group learning at the college/university level. There are similar problem solving concepts that students learn in both mathematics and computer sciences. It would seem reasonable to adopt some of the methodology of the MWP study groups for first year computer science programming students. While no similar investigation, formal or informal, has been found in computer sciences at the college/university level, it would seem reasonable to assume that the similarities of mathematics and computer science as both traditionally problem-solving oriented and focusing on underlying concepts would yield a similar success rate for the implementation of computer science cooperative study groups of four to five students.

With the extensive amount of research that has been done on cooperative learning groups, all but at the college level, it is evident that group learning strategies are a fairly effective strategy for improving achievement in specific subject areas for both elementary and secondary school students. College/university level studies do not provide nearly as much evidence for the use of group learning as a successful strategy to employ in the classroom, but the studies that have been done have shown significant increases in achievement. Since the use of some form of cooperative group learning has generally been found to positively affect achievement in various academic subjects including mathematics and computer programming, an investigation into the effects of the use of cooperative study groups on achievement and retention rate of computer science programming students at the college/university level seems warranted. This is especially important because of the projected needs for trained personnel in the computer science field and the large rate of attrition of potential computer science majors in colleges and universities. Since Fullilove and Treisman's investigation of cooperative study groups was very successful in improving achievement in mathematics, and in raising the science and engineering graduation rate of participants, this research would be a good model to use for implementing cooperative study groups in introductory computer science courses (Fullilove & Treisman, 1990). The proposed investigation will focus on the use of cooperative study groups of four to five students in an introductory college level computer science course. Implications can be drawn that if students are successful academically in their introductory computer science course, then the retention rate of students in computer science programs will also remain high when these academic barriers are removed. With the need for more women and minorities to fill computer science jobs in the next 10-15 years, computer sciences programs must address the need to recruit and retain women and minorities(Brown, 1993; Frenkle, 1990; Gries & Marsh, 1992; and Widnall,1988). Gender and ethnicity differences in achievement and retention rates may be affected by the use of cooperative study groups. The effects of study groups on gender and ethnicity differences of achievement and retention rate will be investigated in this study.

Hypotheses

The major hypotheses for this investigation are:

H1 1 The use of a cooperative study group method in an introductory computer science programming course for college students increases achievement in computer science programming concepts when compared to a traditional individual learning method for college students in the same course.

Much of the research on group learning finds significant or positive effects of such learning strategies on achievement in a variety of subject matters.

Research in group learning at the college/university level is lacking as well as research of cooperative group learning in computer science programming concepts. Study group methods have been found to produce significant gains in achievement in calculus at the college/university level (Fullilove & Treisman, 1990). An investigation into study group methods in college level computer science programming might also produce significant effects on achievement for computer science programming students.

Since there is a need to retain students in computer science courses in order to meet the demand for skilled workers today and in the future, investigation into learning strategies that improve student retention in computer science classes and improve the retention of computer science majors is important. The following major hypotheses will also be investigated.

H1 2 There will be a difference in the retention rate of students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

H1 3 There will be a difference in the retention rate computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to the retention rate of computer science majors using a traditional individual learning method in the same course.

Subhypotheses

This investigation will also examine other factors related to the implementation of study group methods for a college level computer science programming course. The following subhypotheses will be investigated since there is a need to retain students, particularly women and minorities, in the field of computer science to meet the demand for workers in the future since the percentages of these two groups in computer sciences is traditionally very low (Gries & Marsh, 1992; Widnall, 1988). Subhypotheses for this investigation include the following:

H1 1-1 There will be a difference in achievement for minority students in an introductory computer science programming course participating in a cooperative study group method when compared to minority students participating in a traditional individual learning method in the same course.

H1 1-2 There will be a difference in achievement in computer science programming concepts for female students in an introductory computer science programming course participating in a cooperative study group method when compared to female students participating in a traditional individual learning method in the same course.

H1 2-3 There will be a difference in the retention rate of minority students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

H1 2-4 There will be a difference in the retention rate of female students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

H1 3-5 There will be a difference in the retention rate of minority computer science majors in a computer sciences program when a cooperative study group method is implemented in the introductory computer science programming course when compared to minority computer science majors in the same course with a traditional individual learning method.

H1 3-6 There will be a difference in the retention rate of female computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to female computer science majors in the same course with a traditional individual learning method.

The dependent variables in this study include achievement in computer science programming concepts, retention rate of students in the introductory computer science course, and retention rate of students in the computer science major. Independent variables in the study include traditional individual study, cooperative study group membership, gender, and ethnicity. Demographic data including current number of college hours, number of semesters of computer science taken (including high school), declared major, current GPA, and reasons for enrolling in the introductory computer science course will be obtained from each student in the study. Gender and ethnicity of each student will also be tabulated. Group means will be determined for each of the data items collected and a t-test will be used to compare the students in the control group and the students in the treatment group. This will be used to determine the similarities of the two groups in general abilities related to computer science, in order to be sure the treatment and control groups in the investigation have similar student characteristics.

Importance of the Study

This study is important for several reasons. Investigation of new teaching methodologies at the college/university level in computer science is lacking and appears to be necessary. The research study will help determine whether or not altering traditional individual learning methods will help retain students in the computer sciences by improving their achievement, particularly women and minorities, since they are the groups that are most under represented in the professional field of computer science. The investigation will also contribute to the necessary body of knowledge about teaching college/university students and may reveal methods that will improve the graduation ratio of all students in computer science and in particular women and minorities. Since successful use of cooperative study groups in college calculus courses has occurred, it is valuable to other similar disciplines to determine whether that success can be replicated.

Definition of Terms

The following terms will be used throughout this paper and will be defined as follows:

Cooperative Study Groups: Groups of four to five students who study together outside of class and who work on problem sets and programs together. Individuals are responsible for learning all the material and helping other group members; none of the group work is dependent on individual grade calculation, and all grades are independent of study group participation. No group grades are given to cooperative study group members for any work completed.

Traditional Individual Learning: The traditional method of students doing all class work including assignments and studying completely independent of others. No collaboration or cooperative group work is allowed, and students are expected to complete all work on their own.

College-level introductory programming course: This is a course that is taken as the first course for computer science majors at the university at which the study will take place.

Semester: A period of 15 consecutive weeks of college instruction and evaluation as recognized by many colleges/universities and the college accreditation boards. Two semesters represent one year of college.

Intact Classes: Classes that are formed according to student registration for the class with no alteration by the investigator for any purpose.

Problem Set: A group of 9 to 12 questions that typically apply a given programming concept such as looping, decision-making or subprograms to a small programming problem.

Delimitations

This study was conducted only with students enrolled in an introductory computer science programming course at a small liberal arts university located in central Texas during the 1993-1994 academic year. Students enrolled during the Fall and Spring semesters were selected for the study. The control group and the experimental group were chosen as equal representatives of the population of students enrolling in the computer science course. Students selected for the study are limited to those accepted by the particular university, a small liberal arts university located in central Texas with a student population of about 3,100 students. There are approximately 2,600 undergraduate students and 500 graduate students at the university. The undergraduate student population at the university is comprised of approximately 1,800 full-time students and 850 part-time students most of whom work full-time and attend school part-time in the evening. The ethnic background of the student population is approximately 33 percent Hispanic, 4 percent African-American, and 4 percent Asian-American.

Limitations

Several limitations exist that could threaten the validity of this study. The limitations include the sample size, the experimental design, and the use of intact classes. The number of students in the classes who are included as minorities may not be a typical representation of the minority population of the United States. The number of female students in the selected classes may not be a typical representation of the females in the population of the United States.

Measurements of retention rates may be affected by students withdrawing from a class do to unavoidable circumstances beyond the control of the instructor. It will not be known if students do not finish the course due to its difficulty or due to uncontrollable circumstances. Any student not finishing the course for any reason will still be included in the number of students used to calculate retention rates.

Several outside influences could affect the experimental variable in this study. Implementation of a cooperative study group method in classes will not be done through random assignment. Students in intact classes will be assigned to study groups of four or five students per group. During each semester, one intact class will use a cooperative study group method and the other intact class will use a traditional individual learning method. Students in the traditional class could study in their own personally chosen groups without the knowledge of the investigator. Although the exit questionnaire addresses this issue, students may study with others and not consider that to be group work or not report it as such. Additionally, students in the control group may discuss the experimental treatment with other students in the cooperative study group methods class and therefore they may confound the results by either not participating as stated or by leading others to participate in groups without reporting that action. Students in the control group may also study with others outside of the class and not report that as studying with others in groups. This could affect the outcome of achievement and retention rate.

The use of intact classes may result in a bias which is characteristic of the subjects rather than the experimental treatment. Some students may be more motivated to learn than others in the sample population.

Chapter II

Review of Literature

Introduction

Reviewing literature on cöoperative learning over the past 20 years yields a voluminous amount of information about cooperative learning methods that have been successfully developed and implemented in a wide variety of educational settings. These studies range from the elementary school level through the graduate school level and include almost every academic subject at one or more levels of education. The reports range from the documentation of local success stories in cooperative methods to pure experimental research studies and complex meta-analyses. Over the past 20 years, several general methods of cooperative learning have been developed. Some of these methods were created by a single researcher and others evolved with time and through experimental success. Practically all of these different types of cooperative learning have been researched and described at all levels of education.

The main focus for this literature review is to present an overview of the most common forms of cooperative learning that have evolved and stimulated considerable research in education in the past two decades. This research analysis will include cooperative learning at all levels of education with the primary concentration on research performed at the college/university level. While this presentation of cooperative learning methods is not exhaustive, it includes the set of issues that are applicable to almost all cooperative learning methods. These different methods of cooperative learning will be examined and presented as well as their implications for this investigation. Since the purpose of the current investigation is to examine the effect of cooperative learning on college level students, only a brief review of cooperative learning research at the elementary and secondary levels of education is done in order to provide some background for the premises of the study. Several reviews have been published with regard to cooperative learning and these will be presented with their implications for the investigation. A more extensive analysis of cooperative learning and study group methods at the college/university level will be undertaken to provide evidence and support for the current investigation.

Methods of Cooperative Learning

A number of different cooperative learning methods have been well documented in the past 15 years. Variations and modifications of these methods are covered extensively in the literature. The purpose of this section of the literature review is to present the most commonly identified methods of cooperative learning and to examine how they may influence the current investigation.

Johnson and Johnson (1975) created a rather general cooperative learning method called "Learning Together." A group goal, sharing of ideas and materials, group rewards and appropriate divisions of labor among group members generally comprise this method. Most implementations of this method have students working together to produce a single group product through sharing ideas and helping each other with answers to questions. Groups are rewarded on the basis of the performance of the group as a whole unit.

More recently Johnson et. al. (1984) renamed Learning Together as "Circles of Learning." This is the most general form of cooperative learning with groups of six or fewer members. Group members are dependent on each other for the sharing of information and for completing the academic task as a group. Generally intergroup cooperation is fostered by the teacher and learning is dependent on all members of the group. Some competition is usually present between groups to encourage members to be dependent on others in the group for learning the academic task. The concept of "Circles of Learning" provides this research study with the key element that group members are dependent on each other for sharing information and completing one or more academic tasks as a group. This method of group members helping each other to complete academic tasks is the underlying principle to be applied to students learning programming concepts in the study groups used in the investigation.

The "Jigsaw method," developed by Aronson (1978), makes each student in a learning group (jigsaw group) responsible for one part of a lesson and he or she must teach that portion of the lesson to other members of the group. Counter groups are formed of students from each jigsaw group who meet to discuss how best to teach their portion of the lesson to fellow jigsaw members. Cooperation is necessary among jigsaw groups and counter groups, however, there is no specific group reward for achievement or for the use of cooperative skills. Student grades are based solely on individual performance on examinations. Several variations of the jigsaw method have been implemented by changing incentives to be cooperative and individualistic and by adapting variations on the division of the academic task. The current investigation borrows from the jigsaw method the concept of one member of a group teaching a portion of the lesson to other members of the group. One of the purposes of the cooperative study groups in the investigation is to have group members relate problem solutions and programming skills to other members of the group. Therefore, if one group member understands a part of the problem, he or she can then demonstrate to others in the group how the solution works. Also as in the jigsaw method, members of the cooperative study groups do not receive any award for group achievement or the use of cooperative skills.

"Student Teams -- Achievement Divisions" (STAD) developed by Slavin (1980) is basically competition among groups. Students in groups practice and help one another prepare for competition among the groups. Peer support and group norms for achievement were found to result from this method. Slavin (1983a) suggests that specific group rewards be given for individual achievement as well as group achievement. Typically the reward consists of some public display of the results of the group competition through a posting or school newsletter. Since competition is foremost in this method, groups have to be formed with evenly matched teams.

"Teams-Games-Tournaments" (TGT) was created by DeVries and Slavin (1978) and is similar to STAD. The method of competition among the teams is that team members of equal ability between groups compete against one another in face to face competition. Tournament competition is used to determine which team members compete against others using a hierarchial approach in which winners advance to compete against winners and losers compete against others with the same placing in the tournaments. Individual grades for students are still given as a result of individual quizzes and tests.

Although STAD and TGT methods were not purposely incorporated in the cooperative study groups used in this investigation, it is necessary to include them in this review since they are well documented and cited throughout the body of literature on cooperative learning. The common concept of competition may overflow into study group methods naturally since problem solving tasks and assignments usually develop some form of competition among those trying to achieve a solution, whether the competition is recognized or not. Study groups may generate some form of unspoken competition among different group members to achieve the solution first and most efficiently, although this would be a side effect of the use of cooperative study groups.

"Team-Assisted Individualization" (TAI) was developed by Slavin in 1985. This method combines the incentive of group rewards with individualized instruction programs appropriate for the skill levels of the students. Small learning groups of students of varying abilities work together on individual units of mathematics instruction. Team members check each other's work and are required to obtain help from other group members before asking the teacher for assistance. Teachers remove students of similar abilities from the groups on a daily basis and provide instruction to these students according to their ability level in the curriculum.

Both individual and group rewards are built into TAI. The key element borrowed from the TAI method for this investigation is that team members are required to get help from others on their team before they may ask the teacher for help. This requirement causes discussion of concepts and peer tutoring to occur. The implementation of cooperative study groups for the current investigation relies on the foundations of the discussion of concepts and peer tutoring which comprises TAI.

"Group Investigation" (G-I)created by Sharan (1989) focuses on student self-regularization of learning activities. The class is divided into several groups and each group studies a specific aspect of some general topic. Topics are complex enough to allow groups to create a division of labor directed by the teacher which promotes interdependence among group members. Group members are required to gather information, plan, coordinate, analyze, and evaluate their work with other students in the group. Groups then present their findings to the rest of the class, thus providing a broad perspective of a general topic. This method can be related to the study groups used in this investigation because of the concept of division of labor among group members and the resulting interdependence of students in a group.

Elementary and Secondary Studies of Cooperative Learning

Several variations of cooperative learning methods have been researched thoroughly at the elementary level. David Johnson and Roger Johnson with others have researched several different aspects of cooperative learning with elementary school students. Numerous studies completed by the Johnsons and colleagues find that cooperative settings among elementary age students lead to greater achievement in various subjects and generally improve attitudes of students toward learning those subjects. Most of the studies show significant positive effects of the cooperative condition applied to learning subjects including mathematics, reading, mapping skills, and science (Johnson, Johnson, & Anderson, 1976; Johnson, Johnson & Scott, 1978; Skon, Johnson & Johnson, 1981; Yager, Johnson, & Johnson, 1985).

Sholomo Sharan has also done several studies among elementary students on various aspects of cooperative learning (Sharan, 1980; Sharan, Hertz-Lazarowitz, & Ackerman, 1980; Sharan, Kussell, Hertz-Lazarowitz, Bejarano, Raviv, & Sharan, 1984). In one particular study of 17 classes of sixth grade students, Sharan and Shaulov (1990) examined motivation to learn, task perseverance, and academic achievement using the group investigation method of cooperative learning described earlier in this chapter. The authors used a control group and an experimental group of classes and the academic subjects of mathematics, language, and the Bible over an entire school year. The results of the investigation found that those students in the experimental group who used the group investigation method of learning were significantly higher in academic achievement in every subject. Motivation to learn and task perseverance were also significantly higher for those students participating in the group investigation method of cooperative learning. The difficulty of computer science programming mandates that students demonstrate task perseverance, motivation, and achievement. Incorporation of this form of cooperative learning into computer science classes is supported by much of this research.

Investigations of the implementation of cooperative learning at the secondary level of education have also produced positive results. Noreen Webb has conducted several studies with students placed into groups of three or four members. Webb has researched the achievement of group members compared to individual instruction students' achievement and has found higher achievement for students placed in groups (Webb, 1982a; Webb, 1982b; Webb, 1983; Webb, 1984; Webb, 1985; Webb, Ender, & Lewis, 1986).

Several studies from various investigators show that achievement, problem solving ability, and attitudes of secondary students are higher when cooperative learning is implemented in some form (Humphreys, Johnson, &

Johnson, 1982; Johnson, Johnson, & Stanne, 1985; Lazarowitz, Hertz-Lazarowitz, & Jenkins, 1985; Lazarowitz & Karsenty, 1986; Okebukola, 1986; King, 1989; Lonning, 1993). One of the studies conducted by Lazarowitz and Karsenty (1986) examined high school biology students for a period of three months. The study divided 708 tenth grade students into a control group of 226 students and an experimental group of 482 students. The experimental group was instructed using peer tutoring in small investigative groups and the control group was instructed with the traditional classroom-laboratory approach. The results of the investigation found significantly higher achievement in biology by the students in the experimental group who were both quantitatively and qualitatively assessed. In addition, the learning environment as perceived by the students was significantly more satisfactory for those in the experimental group. Students in the experimental group found the learning material less difficult and the classroom environment more cohesive compared to the students in the control group (Lazarowitz & Karsenty, 1986). The current investigation employs the same research design and approximate length. Most importantly however, is the fact that students in the cooperative setting found the material less difficult and the learning environment more inviting.

Published Reviews of Cooperative Learning

Several authors have written reviews and completed extensive meta analyses of various research reports involving cooperative learning. It is relevant to examine some of these summaries as their conclusions and results furnish further support for the current investigation.

Slavin (1980) examined the results of 28 research studies of cooperative learning conducted at both the elementary and secondary school levels. All of the field experiments in Slavin's review compared cooperative techniques to control classes and had a duration of two or more weeks. The studies differed greatly in population and design so it is difficult to draw specific conclusions but general trends can be examined. Three outcomes of cooperative learning evaluated in this review were academic achievement, race relations, and mutual concern among students. For the purposes of the current investigation, only academic achievement will be explored.

Slavin (1980) concluded from his analysis that cooperative methods positively effect achievement in all subject areas except social studies. However, in his conclusions Slavin merely stated that cooperative techniques are no worse than traditional techniques for academic achievement and he indicated that there was enough evidence from field research to support the use of cooperative techniques in schools. Many of the studies found cooperative techniques such as TGT, STAD, and Jigsaw positively effected academic achievement, but had some flaws in the research design, duration, or measurement. Some of the studies either did not use random assignment of control and treatment groups, lasted only two or three weeks in duration, or did not provide accurate and reliable measurements of achievement. Slavin's review does furnish evidence of success in the use of cooperative techniques for improving academic achievement. Since Slavin's review was published in 1980, the studies included in his review were completed prior to 1980, mostly in the late 1970's. The relevance of this review is that it describes previous research indicating a need to examine cooperative learning methods in the classroom and to account for the importance of cooperative learning as an educational method worth investigating. According to Slavin (1980), further studies of cooperative methods of longer duration and tighter controls of measurements in all subject areas and age groups were needed. The current investigation is an attempt to provide further support of cooperative methods for college/university level students in the academic area of computer science.

At the same time as the previous review by Slavin (1980), another review was published that also examined cooperative learning in small groups and its effects on achievement and other variables. Sharan (1980) investigated the cooperative methods of TGT, STAD, G-I, Jigsaw, the Johnsons' cooperative learning approach, and a small group teaching approach. The applicable part of Sharan's review was his examination of the dependent variable academic achievement. Ten research experiments employing TGT as the cooperative method that examined the effect of TGT on academic achievement. Seven of the ten studies found significant positive results for academic achievement when comparing TGT to traditional classroom teaching. One of the studies involved 72 classes of junior high school English classes and superior results in English achievement were found when comparing TGT to individualized instruction. Field research of the STAD method, according to Sharan (1980), also found positive effects for achievement in three studies of elementary and junior high school students in language arts and English. The Group Investigation method (GI) and the Johnsons Learning Together method also provided evidence of higher academic achievement when compared to traditional class methods, although only two studies in Israel are mentioned concerning these techniques in the review. Sharan (1980) stated that there was an opportunity for further research of team learning effects on academic achievement to clarify and reinforce the current findings thus far, particularly at the college/university level.

Webb (1982c) examined student interaction within small groups and its effect on academic achievement. In her review, Webb (1982c) focused on studies which looked at student experiences in groups and the effect that has on achievement. In particular, Webb reviewed studies in which student interactions were categorized as giving help or receiving help within the group. Significant positive differences in achievement were found among students in four of five studies in which students were giving help to other students in the group.

Another important concept discussed by Webb (1982c) was the actual mechanism that connected student interaction and academic achievement. Concluding from other studies in Webb's (1982c) review, merely verbalizing material was not enough to lead to learning. When the purpose of verbalizing material was to teach that information to a peer, then the student performing the teaching had higher achievement as a result of the teaching. This finding is relevant to the current investigation because cooperative study group members will be asked to help each other to learn the course material. Cooperative study group members have to explain or demonstrate the solution to a program or problem to other members in their group leading to more learning and thus higher achievement.

In 1983, Robert Slavin completed another review of cooperative learning research dealing with student achievement. Slavin (1983) reviewed research concerning cooperative learning and its effect on student achievement at the elementary and secondary school levels. In his review, Slavin stated an important concept about measuring individual outcomes of achievement in cooperative learning methods that is central to this particular study:

> "Learning is a completely individual outcome that may or may not be improved by cooperation, but it is clearly not obviously improved by cooperation in the same way as group problem solving is superior to individual problem solving. ... Only an individual learning measure that cannot be influenced by a group member help can indicate which incentive or task structure is best."

Slavin (1983) also wrote that when considering the vast amount of research literature published about cooperative learning, individual academic achievement was not typical of the performance outcomes that were usually measured.

In his review, Slavin (1983) only examined research studies meeting the following criteria:

1) Control and experimental groups that are essentially equivalent were used.
2) The study took place in a regular elementary or secondary school and lasted at least two weeks.
3) Achievement measures were adequate measures of learning given to both experimental and control groups after the group experience.

After analyzing 46 studies that met this review criteria, Slavin drew some conclusions from his findings. Working in small groups cooperatively with group rewards and individual accountability did consistently increase student learning in many academic areas when compared to control methods. According to Slavin, cooperative methods that did not use group rewards for individual learning did not increase student achievement more than control methods. Slavin's (1983) conclusions provide certain contrast to conclusions of other reviewers previously mentioned. The important outcome of Slavin's review was the central theme that students in groups will not show higher achievement than those not in groups unless there is a group reward associated with individual achievement. The current investigation hopes to prove the opposite, that student achievement will increase even though there are no tangible group rewards. It is possible to imply that there is some group reward associated with an individual solving a programming problem. Each person in a cooperative study group attains the solution to a problem when one or more members of the group finds the solution. Correct solutions to problems are desirable for each individual in the group, since every member is required to hand in a program solution and to take tests individually.

Slavin (1990) again examined the current state of cooperative learning and its effects on student achievement. Generally there was consensus that cooperative learning increases student achievement but controversy surrounds which specific conditions actually effect achievement. Slavin states that there was plenty of evidence to support cooperative learning in grades two through nine, but not many studies examined grades ten through twelve and even less existed at the college level. Studies of cooperative learning at the college level will be carefully examined later in this chapter but it is noted that Slavin did endorse their positive findings. One of the questions Slavin raised is the appropriateness of cooperative learning methods for higher order learning such as induction, deduction, analysis and causal effects. The current investigation focuses mainly on basic skills of programming concepts which do not necessarily require higher level thinking.

Johnson, Maruyama, Johnson, Nelson, and Skon (1981) performed a meta-analysis on cooperative learning studies in order to resolve certain controversies of previous reviews of studies. Several psychologists disagree with the conclusions of each other when comparing cooperative, competitive, and individualistic learning methods and their effects on achievement and productivity of students. Johnson et. al. (1981) reviewed 122 studies to compare the effectiveness of cooperation, competition, and individualistic goal structures on achievement of students. The results of the meta-analysis lead to several conclusions by the authors. The first conclusion was that cooperation was superior to competition with regard to achievement and productivity. The study found 109 reports that could be included in the three meta-analysis methods employed. The critical factor from this conclusion is that the subject areas of language arts, reading, mathematics, science, social studies, psychology, and physical education for all age groups including college students were contained in the analysis. Further, the tasks measured consisted of concept attainment, problem solving, retention and memory, all of which are vital to success in computer science programming.

The second conclusion of the meta-analysis by Johnson et. al. (1981) was that cooperation was superior to individualistic efforts in promoting achievement. The authors stated that their results hold for all subject areas and age groups and that the strength of the statistics made it difficult to identify any mediating or moderating variables. Since computer science programming courses have traditionally been taught with individualistic efforts exclusively, the conclusions by the authors revealed that a cooperative learning strategy might benefit computer science students at all levels of education including the college/university level.

Studies of Cooperative Learning at the College/University Level

Considerably fewer studies of cooperative learning techniques exist at the college/university level than at the elementary and secondary levels. Pure research studies of cooperative learning at the college/university level are lacking. Several descriptive articles and empirical studies exist and cooperative learning is highly praised in these articles (Radebaugh & Kazemek, 1988; Rau & Heyl, 1990; Brothen, 1986; Croteau & Hoynes, 1991; Conrad & Conrad, 1993, Posner & Markstein, 1994). Generally the common thread among these studies is that forming small groups of three to six students for the purposes of discussion, problem solving, and critical thinking procures high regard from both instructors and students in attitude and acquired high level knowledge. Pure experimental research studies of cooperative learning of some form have not been very commonly done at the college/university level. Some evidence of success with different types of cooperative learning at the college/university level has been documented and will be examined next to provide further background for the current investigation.

Frazer, Beaman, Diener, and Kelem (1977) investigated student performance in two experiments with an introductory social psychology course at the University of Washington. A control group of 172 students and an experimental groups of 212 students comprised the first study. In this study, students in the experimental group were assigned to one partner randomly and the grades of each pair of students were averaged together for the final course grade assignment. The students in the control and experimental groups took the same examinations and had the same instructor and course assignments. The results were very positive for the experimental group. None of the 212 students in the partner groups received a grade below 70% for the course and 87 percent of the students in the experimental group received an "A" or "B" in the course. The control group had 18 students receive below 70 percent for their course grade. In the second study 239 students were randomly assigned learning partners of 0, 1, 2, or 3 in number. Those students assigned no partner were the control group in the study. Partners were encouraged to study together and monitor each other's work. No extra class time was given for group discussions or group work. The results of this second experiment indicated enhanced student performance in the course with those in groups with partners averaging one half of a letter grade higher than those in the control group (Frazer et. al., 1977).

Both of the studies by Frazer et. al. (1977) provide evidence of success with group work at the college/university level and the current investigation borrows some of the methodology from these two experiments. The main difference between the current study and these two studies are the assignment of individual grades based on the group (2, 3, or 4 members) members' averages. The authors of the study noted that students protested vigorously against this grading system, but that overall grades were much higher for those in the experimental group in spite of the dislike shown by the students. The current study only calculates individuals' grades based on individual performance on examinations and assignments but replicates the idea of group study dependence for the purposes of learning the course material.

Beilin and Rabow (1981) studied the achievement of minorities in an introductory sociology course. Students in the course were assigned to either a lecture only group or to a lecture group and a discussion group. The discussion group was based on group research by Hill (1969) called "Learning Thru Discussion" (LTD). LTD, according to Hill (1969), did not increase learning of factual material but it did increase problem-solving techniques and higher order thinking skills such as analyzing and synthesizing information. Beilin and Rabow (1981) tested the relationships between interracial group learning and academic achievement among 118 male and 180 female undergraduate students. The control group (lecture only) and the experimental group (lecture and discussion) both attended the same lectures, but the experimental group students were assigned to weekly discussion groups and grading was based on student contributions to the group task. The group task was defined as group wide understanding and application of the week's reading and improvement of the group's functioning as a unit toward that task. A significant difference was found between experimental and control conditions among non-minority students but not among minority students. This result was attributed to the fact that minorities were only 17 percent of the sample population and equalization of status was not possible with so few minorities in the groups (one per group). Beilin and Rabow (1981) concluded that groups were useful for non-minorities and had potential to be positively influential for minority students if groups were established with more minority members in each group. The current investigation examines minority student achievement and success rate and does have a higher percentage of minority students than found in this report.

In a study of the effect of grouping pre-service elementary science laboratory students, Lawrenz and Munch (1984) examined science achievement, formal reasoning ability, the laboratory learning environment, and individual relationships among group members. The results of this study indicated that grouping students had significant effects on science content achievement but not on formal reasoning ability, the laboratory learning environment, or relationships among group members. This study did not have a control group for comparing individual learning to group learning so it is weak in terms of research design and provides an example of the need for better research design and methodology of cooperative learning studies at the college level.

Brothen (1986) examined group learning in a college class of 270 introductory psychology students. Students were divided into learning groups of four at the beginning of the course in an attempt to provide some educational change that could improve students' learning experiences. Students were expected to study and discuss course material in their groups throughout the semester in order to learn from one another. Only one lecture session was given per week and the remaining time was spent in learning groups. Students were assigned grades according to the entire group's work. Data collected by Brothen (1986) determined that students in learning groups were just as successful as students in previous semesters who learned only through lecture and individual study. Attrition rate was found to be very low for the learning group classes compared to previous lecture only classes. In addition student evaluations of the course gave high ratings to learning groups and students felt that their reading and study habits improved as a result of the learning group experience. The current investigation will examine retention rate of students, so the examination of attrition rate in this particular study does provide evidence that the implementation of cooperative groups may improve the retention rate of students in the class.

Warring and Maruyama (1986) examined the effect of cooperative grouping of introductory psychology students. Student test scores and individual expectations were measured over the duration of a 10-week introductory psychology course. The study was replicated over a second semester. It was found that the cooperative condition of four to six students per group was significantly different in test performance and in student expectations of their own test scores. In both variables, students showed higher scores and higher self expectation of test scores in the cooperative condition.

In a study of 152 senior nursing students enrolled in a senior level nursing course at a predominantly African American institution of higher learning, Frierson (1987) investigated academic performance according to course grades. Students in the course over a period of four semesters were divided into three groups. The first group was the control group, the second group received test-taking instruction, and the third group received test-taking instruction and participated in learning teams. The purpose of the course was to prepare students for a standardized state nursing examination. The results of the experiment supported the use of team learning and test taking instruction with significantly higher grades for students in the two experimental groups than for students in the control group. While this study did not single out team learning by itself, results still revealed higher grades for those with team learning than for those without it. The study lacked randomization, took place over four semesters, and students were almost all African-American so generalizations are limited. However, its implications for the current investigation are sound and promising for cooperative methods success at the college/university level, particularly for minority students.

Johnson, Johnson, Stanne and Garibaldi (1988) examined the effects of cooperative groups on the achievement of entering freshmen in a summer honors program at Xavier University in New Orleans. Students were instructed in a map reading and navigation unit using cooperative groups of three students per group. The results of the study indicated significantly higher achievement scores for those students in cooperative groups and the attitudes of those students in groups toward group work were improved. This study was based on only one lesson, and achievement was measured according to the success of the students in reaching their destination in the lesson and obtaining as much gold as possible during their expedition. The investigators concluded that significant differences existed between the cooperative groups and the individual students in achievement. The limitations of the study are apparent since its duration was only one two hour time period.

Hufford (1991) researched freshmen biology students and the effect of group learning on students' grades. In his study, Hufford did not formally assign students to learning groups. Instead students were strongly encouraged to form informal cooperative learning groups and to exchange information and concepts among group members. Hufford (1991) found that 20% more A's were received among those who participated in group learning and less than 3% of those in groups failed the course. No other statistical data was provided in Hufford's research.

In an empirical study of graduate level students working together in groups on a research project, Latting and Raffoul (1991) found some interesting outcomes. The desired goals of placing students in research project groups were to improve students' skills in working with others, to allow students to learn from one another, and to enrich students' experiences with the course content. The researchers investigated 97 graduate students enrolled in a required research methods course. Students were allowed to voluntarily join a group of no more than five students and those who did not form groups voluntarily were assigned to a group according to their research interests. The results of this study indicated that students who were not faced with intergroup conflict did well in the course and preferred group work. Students learned more from each other in groups that did not have conflict but also in groups that were not self-selected. Students that learned most about the course content were those who invested much time and energy into the group project. Since students in each group all received the same grade, the researchers noted that those doing the least and learning the least earned the same grade as those doing and learning the most.

Borresen (1992) studied the effects of student grouping in an introductory college statistics course. Over a period of three years, Borresen researched the achievement and attitudes of college students through three different learning methods. Six classes of the same introductory statistics course were divided randomly into individualized learning, assigned groups, and voluntary groups. The size of the groups were four to five students per group. No attempt was made to control for the ability level of the students in the groups. At the end of the study, Borresen completed an analysis of variance among the 3 groups of students using the students' GPA as the measure of variance between the groups. Although no significant differences were found among GPA's, individual test scores on exams given during the course were significantly higher for the students in groups. Student course evaluations were also analyzed using group status and a significant difference was found among the rating of the difficulty of the course material. Students in the groups found the class material much less difficult than did students working individually. This study supports the current investigation because students' individual test scores were significantly higher for those in groups and because the difficulty of the course material was rated significantly lower for those students in groups. One of the important factors in researching the use of group work in computer science is to improve the retention rate of computer science majors and the study by Borresen found that students in groups rated the course as significantly less difficult than students not in groups. This could lead to evidence of higher course retention rate for students working in groups in the current investigation.

An investigation of cooperative learning groups in a semester long computer literacy course at Rockhurst College in Missouri by Fitzgerald and Caulfield (1992) found the use of cooperative learning groups to be successful. Students in two of five laboratory sections of a computer literacy course, N= 23, were placed in cooperative groups of three students per group according to previous computing skills measured at the beginning of the study. Students in the remaining laboratory sections, N=38, were used as a control group representing the traditional individual learning style. No significant differences were found in achievement of concepts, but the cooperative learning groups did as well as those students in the control group. The investigators noted that students in the cooperative groups were much easier to manage, asked fewer questions, and appeared more confident and self-directed (Fitzgerald & Caulfield, 1992). Since the current investigation will examine a computer science programming course at a small university, the study by

Fitzgerald and Caulfield (1992) provides further evidence of success for a cooperative learning method at the college/university level.

A study of cooperative learning in a college level introductory cell and molecular biology course was found to be successful for all students, minorities in particular. Posner and Markstein (1994) examined the academic performance of students grouped together in small study teams of three to four participants per team. Study teams were lead by teaching assistants in the one hour lab that met weekly in addition to the biology lecture section of approximately 300 students. These study teams were formed so that students could work together in the laboratory on biology factual materials and on classical problems in molecular biology. During the first semester of the investigation, all laboratory sections were placed into study teams. Students met with in the lab one time per week for two hours. In the second semester all students in the course were used as a control group. No study teams were formed and the traditional lecture style format was followed in the weekly labs. The results of the study were very positive for the students in the study teams. Academic performance and retention were higher for students in the study teams. Minority students did considerably better with an average grade on a 4 point scale of 1.82 in the control group to 2.93 in the study team group. Students completed a questionnaire about their experiences in the study teams and over 70 percent responded positively toward the study teams and believed that they were useful. Another outcome of this study was that minority students enrolled in the next level biology course at a much higher rate with the implementation of the cooperative study teams (Posner & Markstein, 1994). The results of this study indicate positive outcomes for cooperative study groups at the college level, and the efforts of the current investigation are quite similar particularly with the use of cooperative study groups being very similar to the study teams of Posner and Markstein (1994).

The research that provides the strongest support for the current investigation is the Mathematics Workshop Program (MWP) implemented by Fullilove and Treisman (1990) at the University of California at Berkeley (UCB). The program began with the authors trying to determine the key to the high rate of success of Asian-American students in a first year calculus course at UCB. Asian-American students did exceptionally well in the course and the authors discovered that these students gathered in informal study groups regularly outside of class to work on problems and study together for exams. In an attempt to replicate this behavior of Asian-American students, Fullilove and Treisman (1990) started the MWP at UCB. This program involved students meeting in small groups of four to six students two times per week for the duration of a semester. At the meetings students worked on traditional calculus problems helping one another learn the basic concepts and underlying mathematics in the problems. The program was monitored for six years during which time academic achievement and graduation rates of the participants in the study groups were significantly higher than those students not active in study groups. The concept and term "study group" is taken from Fullilove and Treisman's MWP. The similarities between calculus and computer science programming include problem solving, general mathematics, rigorous and difficult subject matter, and generally low rates of success among students. A similar study seems warranted based on the success of the MWP.

Summary

The extensive number of studies of cooperative learning and group work strongly support the use of some form of group learning in academic subjects. The purpose of this chapter was to review the literature related to the current investigation and to demonstrate evidence of the usefulness and applicability of the research in regard to the research questions proposed for this investigation. Several studies at the college/university level provide support for this investigation and the use of cooperative study group methods. An important observation about the college/university studies presented is that none of them provided pure experimental research in a tightly controlled investigation. The current study will attempt to provide experimental evidence and sound statistical methods together with an improved research design in determining the answers to the proposed research hypotheses.

Chapter III

Methodology

Introduction

The objective of this chapter is to describe the research methodology and procedures that will be used in the study. A detailed description of the sample, treatment, instrumentation, and data collection and analysis is explained.

Sampling Procedure

The sample used for this investigation will consist of four intact classes of first year introductory computer science students at a four year institution of higher learning. The intact classes include students enrolled in different sections of the same course titled *Computer Science Concepts I.* The classes are all taught at the same four year institution of higher learning. Each class will have 15-25 students. The chosen sample will have a total of 81 students.

Two of the four intact classes used in this study will be selected from the Fall 1993 semester classes and the other two intact classes will be selected from the Spring 1994 semester classes. Two of the four intact classes selected will be chosen as the treatment group, and the remaining two intact classes will comprise the control group. Since the classes in each of the semesters consist of one evening section and one day section, the classes are specifically chosen for treatment and control groups. During the fall semester the treatment group will be chosen as the evening class, and the control group will be the day class. In the spring semester of the study, the day class will be purposely selected as the treatment group and the evening class will be the control group. This procedure for choosing treatment and control groups will be done in order to keep the control and treatment groups of the sample as similar as possible. Randomly choosing control and treatment groups might skew the results since the student population in day classes tends to be different than the student population in evening classes. Selecting the control and treatment groups each semester from day and evening sections will be done to try to ensure a consistent population for both the control and treatment groups.

Students enrolling in the classes during each semester have to select either the day or evening section of the course according to their own scheduling needs. Students choose to enroll without any influence by the investigator. It is assumed that all of the intact classes are heterogeneous with respect to students' data in each of the classes used in the investigation. Students in the sample will be asked to voluntarily agree to be part of the investigation. Each student will sign a permission form (see Appendix A) and agree to have his or her data and test results included in the study. Every student in the sample is given the opportunity to decline participation in the study. All students in the classes have agreed to participate in the study with no students declining.

Treatment

The independent variables in this study are the learning strategy of the students, either cooperative study group membership or traditional individual study, gender and ethnicity. Students participating in the study are in either the control group or the treatment group. The control group for this experiment refers to students from two intact classes (sections) of *Computer Science Concepts I*; one class taught during the day in the Fall semester of 1993 and the other class taught during the evening in the Spring semester of 1994. The treatment group for this investigation consists of students from two intact classes (sections) of *Computer Science Concepts I* students, one class taught during the evening of the Fall semester of 1993 and the other class taught during the day in the Spring semester of 1994.

The treatment group has 39 students and the control group has 42 students for a total sample of N=81. The treatment group consists of 11 minority students, 15 female students, and 19 computer science majors. The control group consists of 13 minority students, 9 female students and 23 computer science majors.

In order to control for the instruction content and quality over the duration of the study for all four classes, the same instructor will teach each class in the study. All four intact classes will be taught in the same lecture style format and cover the same material. Identical handouts, problem sets, programs, and exams will be given to all students in both the treatment and control groups. None of the student grades in the treatment group are group dependent. Grades for all students in both the treatment and control groups will be determined by assignments and exams that are turned in individually by all students. Having the same instructor, the same course materials and examinations, the same grading policy, and the same syllabus for each class in the study is done to control for instructor effect on the dependent variables of achievement and retention rate. Due to course enrollment limitations at the chosen institution for this study, the classes chosen for the study are held in both the fall semester of 1993 and the spring semester of 1994.

The actual treatment in this study is the assignment of students to study groups for the purpose of working together on problem sets, programs, and exam preparation. All students in the treatment group will be randomly assigned to study groups of four to five members. The assignment to these study groups is completely random regardless of the gender, ethnicity, or declared major of the students. A random number generator using a poisson distribution produced a random number in the range of 1 to 14 for each student in the fall treatment group and a random number in the range of 1 to 25 for each student in the spring treatment group. The students will be ordered according to their randomly assigned number and grouped off in sets of four consecutive numbers beginning with the numbers one through four. This grouping method will create three study groups in the fall class with four members in one group and 5 members in the other two groups. Six study groups will be created in the spring class with four students per group in five of the six groups and five students in the sixth group. A total of nine different study groups will comprise the treatment group for the study.

Students in the treatment group will be instructed to work and study together with members of their assigned study group throughout the semester for a minimum of two hours per week. Students in the treatment group will be specifically directed to work with members of their assigned group for all class work which includes program assignments, problem set handouts, and exam preparation. Students in the control group will be advised to do all of their programming and class work individually. The control group students will be specifically instructed that collaborating with others is not allowed and that all class work and assignments are to be done without the assistance of other students. All programming assignments for students in the treatment group will be assigned to be done as group work and each individual in the study group will be expected to turn in a separate program for his or her individual grade although the design and implementation of the program will be a product of the cooperative study group as a whole. The program assignments handed in by the students in the treatment group will be identical for each member of the same cooperative study group. Programming assignments handed in by students in the control group are not supposed to be identical to those of any other student. In a traditional typical computer science programming class, individual work is emphasized and promoted and collaboration in writing programs is generally not allowed.

Four different typical problem sets for learning programming statements and constructs will be assigned to both the control and treatment groups throughout the semester. These problem sets contain eight to ten questions comprised of common problems that emphasized learning the programming statements for input, output, assignment, decisions, looping, subprograms, and single-dimensional arrays. The students in the treatment group will be instructed to work together with their study group in solving the problem sets and to help each other learn how to construct a correct answer to each of the problems. Students in the control group will be instructed to work individually on the problem sets and to use the problems as study material for the exams. About two weeks after each problem set is given out in class, the solutions to the problem sets will be electronically mailed to all students in both the treatment and control groups during the semester. Students in the treatment group will be directed to check their solutions with the other members of their study group for examination preparation. Students in the control group will be directed to check their solutions individually for examination preparation.

Dependent Variables and Research Hypotheses

The purpose of this investigation is to examine a new learning strategy to determine its effect on the achievement of programming knowledge and success of students in an introductory computer science programming course. Achievement of programming knowledge will be measured using a written examination. Success will be determined by the number of students completing the course with a grade of "C" or higher as a percentage of those students who began the course. The dependent variables in the study are achievement in computer science programming concepts, retention rate of students in the course, and retention rate of students in the computer science major.

The use of cooperative groups as a learning strategy has been shown in the literature over the past 20 years to significantly increase achievement in many subject areas at both the elementary and secondary levels of education. Research literature at the college/university level is less available and does show some success with the use of cooperative learning groups in some academic subject areas. The following major and minor hypotheses suggest that implementation of study groups as a learning strategy may lead to higher achievement of students in an introductory computer science programming course. The acceptance or rejection of these major and minor hypotheses will provide further experimental evidence to justify or oppose the use of study groups in college/university level introductory computer science programming courses.

H1 1 The use of a cooperative study group method in an introductory computer science programming course for college students increases achievement in computer science programming concepts when compared to a traditional individual learning method for college students in the same course.

H1 1-1 There will be a difference in achievement for minority students in an introductory computer science programming course participating in a cooperative study group method when compared to minority students participating in a traditional individual learning method in the same course.

H1 1-2 There will be a difference in achievement in computer science programming concepts for female students in an introductory computer science programming course participating in a cooperative study group method when compared to female students participating in a traditional individual learning method in the same course.

Computer science courses are traditionally difficult for many students, especially non-computer science majors required to take an introductory programming course. Implementation of new learning methodology in the classroom is valuable to computer science instructors when such methods can be shown to improve the success ratio of students in the class. The experimental results of this study can provide instructors with evidence supporting or rejecting the use of study group methods in an introductory computer science programming course. The following major hypothesis and minor hypotheses should provide appropriate experimental evidence.

H1 2 There will be a difference in the retention rate of students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

H1 2-3 There will be a difference in the retention rate of minority students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

H1 2-4 There will be a difference in the retention rate of female students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

According to the literature there is a clear need to retain students in computer sciences in order to fulfill jobs in the computer science areas of the work force over the next 10-15 years. Women and minorities will be expected to fill many of these positions. Cooperative learning and the implementation of study groups has been found in the literature to be one positive avenue leading toward success in several academic areas at all levels of education. The acceptance or rejection of this major hypothesis and minor hypotheses will provide further experimental evidence to computer science instructors when choosing learning strategies to implement in their classes.

H1 3 There will be a difference in the retention rate of computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to the retention rate of computer science majors in the same course using a traditional individual learning method.

H1 3-5 There will be a difference in the retention rate of minority computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to minority computer science majors in the same course with a traditional individual learning method.

H1 3-6 There will be a difference in the retention rate of female computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to female computer science majors in the same course with a traditional individual learning method.

Experimental Design

The data for this experiment will be gathered using a non-randomized pretest-posttest group design, using a control group and a treatment group.

Instrumentation

The test instrument that will be employed to measure achievement in computer science programming concepts is a final examination developed at The University of Texas at Austin and validated at the university's Computer Science Department at which the investigation will take place. The computer science achievement test contains 32 multiple choice items, 3 short answer coding questions, and 1 program coding question. The test was measured for validity using the rank-difference correlation coefficient. This coefficient was obtained by ranking the subjects' scores on three regular exams and correlating those scores with the subjects' final exam scores (Grunlund, 1981). The validity coefficient was 0.94 for examination number one and the final examination, 0.91 for examination two and the final examination, and 0.89 for examination three and the final examination. This test for validity was used because the three examinations test for knowledge of computer science programming concepts and have been used for the past five years for determining achievement in computer science programming concepts in the particular course used in the study. The reliability of the test was determined using the split halves method and was found to be $r = 0.85$ (Gronlund, 1981).

Data Collection

Data regarding student gender, ethnicity, GPA, college hours, computer science hours, declared major, SAT score, and reason for taking the class will be collected during the first week of class. All students will be asked to fill out a data questionnaire which is shown in Appendix B. Total college hours and GPA will be verified using student data from the administrative data base at the St. Edward's University. All students will take the pretest form of the computer science achievement test during the second week of class. The posttest form of the computer science achievement test will be administered during the final examination period in the 15th week of class as scheduled by the registrar at the university. Both forms of the test were scored by hand. The multiple choice items are worth two points each, the short answer coding questions are worth six points each and the program coding question is worth

18 points. The entire test is worth a total of 100 points. The coding questions will be uniformly scored according to correct syntax (three points) and correct semantics (three points). The program coding question was uniformly scored according to syntax(four points), semantics (four points), procedures (six points), and main program (four points). One point in each category was deducted for each error of that type found.

The number of students enrolling in the course will be determined using the official headcount class roster as calculated by total number of students enrolled in the class as of the 12th class day. The number of students successfully completing the course will be determined using the number of students on the final grade roster receiving a grade of "C" or higher in the class. A grade of "C" or higher is the standard used by the university to allow students to continue on to the next computer sciences class for majors and it is the grade a student must receive in order to satisfy a required computer science component as part of the general studies curriculum. This is the standard practice in most universities and so it will be used as the criterion for success in the course in the study.

An exit questionnaire will be completed by all students that finish the class. The purpose of the exit questionnaire is to elicit information from the students in the study about their learning processes throughout the semester and about their plans to continue on in computer sciences. The exit questionnaire is found in the Appendix C. The first two questions will be used to determine why students might not remain in the computer science major. The third and fourth questions will be used to obtain information about the course itself and to see if students will voluntarily submit that study groups were a positive or negative influence in the course. The fifth question will be used as a check to determine the amount of time students actually spent in their study groups and whether or not they believed that the cooperative study groups were effective. The sixth question will be used to determine whether students in the control group formed study groups on their own and to remove them from the control group if they did study in groups.

Statistical Analyses

In order to evaluate the first major hypothesis and the first and second minor hypotheses, analysis of covariance (ANCOVA) will be used to compare achievement means of the treatment and control groups using the pretest means as the covariate. The significance of the treatment condition will be set at the level of 0.05.

The evaluation of the second and third major hypotheses and the third, fourth, fifth and sixth minor hypotheses will be performed using a 2 x 2 Chi-square table with expected frequency values. The two rows of the table will represent the treatment and control groups respectively, and the two columns of the table will represent success (retention) or non-success (non-retention) in the course. The Pearson value will be computed to determine the significance of the retention rate of students in the course. The significance level for retention rate will be set at the 0.05 level.

Chapter IV

Results of the Investigation

Introduction

The data for these results were collected from 81 students enrolled in four different sections of the same introductory computer science programming course during the Fall semester of 1993 and the Spring semester of 1994 at a small liberal arts university in central Texas.

The IBM-370 VM computer at The University of Texas at Austin was used to analyze the data through the SPSS statistical program provided on the IBM system. Frequency distributions, t-tests, Chi -square and analysis of covariance (ANCOVA) were the statistical tests used for computation of the results of the investigation. The exit questionnaire responses were collected and quantified by accumulating positive and negative responses to answers on the questionnaire. Relevant findings in the exit questionnaire responses are described below.

Similarities Between Control and Treatment Groups

The subjects in the experiment were divided into two groups, treatment and control, by assignment according to enrollment in classes. Since individual subjects were not randomly assigned to treatment and control groups for this experiment, t-tests were performed in order to determine the homogeneity of variance of the treatment and control groups. College grade point average (GPA), number of previous semesters of computer science courses, number of college credit hours, and college entrance examination scores (SAT) were the variables used to compare the groups for homogeneity. The results of these tests are shown in tables 4-1, 4-2, 4-3 and 4-4.

Table 4-1 t-test Results of College GPA

Group	N	Mean	SD
Control	43	2.6979	.769
Treatment	38	2.9526	.710

Pooled Variance Estimate

t value	df	2-tail Prob
-1.54	79	.127

Table 4-1 shows the results of comparing the control and treatment groups on the independent variable college GPA. No significant differences were found between the groups on this variable. The t value of -1.54 was not significant at the .05 level.

Table 4-2 shows the statistical data for subjects in the control and treatment groups on the independent variable previous semester hours of computer science. The t value of 2.47 exceeds the critical level of .05. There was a significantly higher number of semesters of computer science in the control group when compared to the treatment group. Since the control group had a significantly higher number of previous computer science hours, the results of this test will not affect the treatment. If there is any effect from this variable, it will only be on the results of the control group which could strengthen the results of the experiment. Implications of this variable will be discussed further in Chapter V.

Table 4-2 t-test Results on Total Semesters of Computer Science

Group	N	Mean	SD
Control	43	2.25	1.878
Treatment	38	1.42	1.106

Separate Variance Estimate

t Value	df	2-tail Prob.
2.47	69	.016

Table 4-3 contains the descriptive statistics for the control and treatment groups on the independent variable college credit hours. There was no significant difference found between subjects in the treatment group and the control group on this variable. The t value of .24 did not exceed the critical level. The control treatment groups were statistically similar on number of college credit hours.

Table 4-3 t-test Results on College Credit Hours

Group	N	Mean	SD
Control	43	53.46	41.73
Treatment	38	51.23	39.90

Pooled Variance Estimate

t value	df	2-tail Prob.
.24	79	.807

The results of comparing the control and treatment groups on the subjects' SAT score is shown in table 4-4. No significant difference was found on this variable between the two groups. The t value of 1.04 did not exceed the critical level at .05. Not all subjects had available SAT scores for this investigation which places a limitation on the interpretation of the results of this test. Therefore, the number of subjects in the control group, N=28 and the number of subjects in the treatment group, N=19 is less than the total numbers in both groups overall on examination of the independent variable SAT score.

Table 4-4 t-test Results on College SAT scores

Group	N	Mean	SD
Control	28	1007.85	179.657
Treatment	19	955.73	150.893

Pooled Variance Estimate

t Value	df	2-tail Prob.
1.04	45	.304

Since the results of the t-tests for SAT scores, college GPA and college credit hours showed no significant differences between the groups, these variables were considered homogeneous with respect to both the control and treatment groups. The t-test for number of semesters of computer science revealed significant differences between the treatment and control groups on this variable. The control group had a significantly higher number of semester hours in computer science than did the treatment group with a t value of 2.47 which is significant at the .01 level. This indicates a highly significant difference among the control and treatment groups in semesters of computer science.

Achievement in Computer Science Programming Concepts

All subjects took an equivalent form of the achievement test as a pretest in order to determine if there were differences between the groups on the test and to use the scores as a covariate in the final analysis of the achievement test. Descriptive statistics concerning achievement in computer science programming concepts are presented in table 4-5.

Table 4-5 Descriptive Statistics for Student Achievement Scores

Pretest	Mean	SD	N
Group			
Control	31.08	10.29	35
Treatment	27.30	13.01	33
Entire Sample	29.25	11.76	68
Posttest	**Mean**	**SD**	**N**
Group			
Control	71.97	17.39	35
Treatment	71.66	11.37	33
Entire Sample	71.82	14.67	68

The first major hypothesis for this investigation was evaluated using ANCOVA at the .05 level of significance. Subject posttest scores were adjusted using pretest scores as the covariate. Data representing the results of this analysis are shown in table 4-6.

H1 1 The use of a cooperative study group method in an introductory computer science programming course for college students increases achievement in computer science programming concepts when compared to the traditional individualized learning method for college students in the same course.

The F value of .02 did not exceed the critical value. The first major hypothesis was rejected. There was no significant difference in achievement in computer science programming concepts when a cooperative study group method was used compared to a traditional individual learning method for college students.

Table 4-6 ANCOVA of Posttest Scores Between Control and Treatment Groups

Source	SS	df	MS	F	Sig. of F
Within Cells	13966.77	65	214.87		
Regression	463.54	1	463.54	2.16	.147
Between Cells	5.05	1	5.05	.02	.879

Data were collected and analyzed on achievement in computer science programming concepts using the independent variable minority. Descriptive statistics for achievement pretest and posttest scores for minority subjects are shown in table 4-7.

Descriptive Statistics for Minority Achievement Scores
Table 4-7

Pretest	Mean	SD	N
Group			
Control	30.45	14.02	11
Treatment	30.40	12.34	10
Entire Sample	30.42	12.92	21
Posttest	**Mean**	**SD**	**N**
Group			
Control	56.63	14.69	11
Treatment	70.00	8.78	11
Entire Sample	63.00	13.76	22

The first minor hypothesis was analyzed using ANCOVA at the .05 level of significance. The minority posttest scores were adjusted using the pretest scores as a covariate. These data representing the adjusted mean scores of minority subjects in the control and treatment groups on the achievement test are shown in table 4-8.

H1 1-1 There will be a difference in achievement for minority students in an introductory computer science programming course participating in a cooperative study group method when compared to minority students participating in a traditional individualized learning method in the same course.

Table 4-8 ANCOVA of Minority Achievement Scores

Source	SS	df	MS	F	Sig. of F
Within Cells	2853.71	18	158.54		
Regression	.84	1	.84	.01	.943
Between Cells	935.57	1	935.57	5.90	.026

The F value of 5.90 exceeded the critical value. The first minor hypothesis was accepted. A significant difference in achievement for minority students participating in a cooperative study group methods was found when compared to minority students participating in a traditional individualized learning method.

Data were analyzed using the independent variable of gender to determine if there were any significant differences in achievement of females when a cooperative study group method was used. The covariate pretest score was used to adjust the mean posttest scores. Table 4-9 shows the descriptive statistics for female achievement scores.

Table 4-9 Descriptive Statistics for Female Achievement Scores

Pretest	Mean	SD	N
Group			
Control	26.62	10.67	8
Treatment	22.46	10.13	15
Entire Sample	23.91	10.28	23
Posttest	**Mean**	**SD**	**N**
Group			
Control	68.87	20.15	8
Treatment	71.13	9.84	15
Entire Sample	70.34	13.86	23

The second minor hypothesis was analyzed using ANCOVA at the .05 level of significance. The female posttest scores were adjusted using the pretest scores as the covariate. These data representing the adjusted mean scores of females on the achievement test are shown in table 4-10.

H1 1-2 There will be a difference in achievement in computer science programming concepts for female students in an introductory computer science programming course participating in a cooperative study group method when compared to female students participating in a traditional individual learning method in the same course.

The F value of .01 did not exceed the critical value. The second minor hypothesis was rejected. No significant differences were found among females when a cooperative study group method was used compared to a traditional individual learning method in the same course.

Table 4-10 ANCOVA of Female Achievement Scores

Source	SS	df	MS	F	Sig. of F
Within Cells	3836.43	20	191.82		
Regression	364.18	1	364.18	1.90	.183
Between Groups	1.69	1	1.69	.01	.926

Retention Rate of Students

Data gathered on number of students enrolling in the course and number of students completing the course with a grade of "C" or better were used to evaluate the second major hypothesis and the third and fourth minor hypotheses. A 2 x 2 Chi-square table was used to determine the significance of the retention rate of students in the course. Table 4-11 shows the Chi-square descriptive data on retention rate of students in both the control and treatment groups. The Pearson value was used to determine the significance of the results at the .05 level.

H1 2 There will be a difference in the retention rate of students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

Table 4-11 Chi-Square Results on Retention Rate of Control and Treatment Groups

Group	Success	Percent	Non-Success	Percent	N
Control	28	65.1	15	34.9	43
Treatment	30	78.9	8	21.1	38
Entire Sample	58	71.6	23	28.4	81

Minimum expected frequency = 10.79

	Value	df	Significance
Pearson	1.89	1	.16

The Pearson product coefficient of 1.89 was not significant at the .05 level. No significant difference was found in the retention rate of students in the course when a cooperative study group method was implemented and compared to a traditional individual learning method. The second major hypothesis was rejected.

The third minor hypothesis was evaluated using a 2 x 2 Chi-square table on data gathered about minority student retention in the treatment and control groups. The level of significance was set at the .05 level. Table 4-12 shows the Chi-square statistical data gathered for minority students.

H1 2-3 There will be a difference in the retention rate of minority students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

Table 4-12 Chi-square Results on Retention Rate of Minority Students

Group	Success	Percent	Non-Success	Percent	N
Control	7	53.8	6	46.2	13
Treatment	9	60.0	6	40.0	15
Entire Sample	16	57.1	12	42.9	28

Minimum expected frequency = 5.57

	Value	df	Significance
Pearson	.107	1	.742

The third minor hypothesis was rejected. The Pearson value of .107 did not exceed the critical level. No significant difference was found in retention rate of minority students when a cooperative study group method was implemented compared to a traditional individual learning method.

The fourth minor hypothesis was evaluated using a 2 x 2 Chi-square table on data gathered about female student retention rate in the course for both the control and treatment groups. Table 4-13 shows the results of the Chi-square test.

H1 2-4 There will be a difference in the retention rate of female students in an introductory computer science programming course when a cooperative study group method is implemented compared to a traditional individual learning method in the same course.

Table 4-13 Chi-Square Results on Retention Rate of Female Students

Group	Success	Percent	Non-Success	Percent	N
Control	7	63.6	4	36.4	11
Treatment	14	93.3	1	6.7	15
Entire Sample	21	80.7	5	19.3	26

Minimum expected cell frequency = 2.11
Cells with expected frequency less than 5 was 2 of 4

	Value	df	Significance
Pearson	3.60	1	.05

The Pearson value of 3.60 was significant at the .05 level. However, the minimum expected cell frequency was 2.115. In 50 percent of the cells in the table, the minimum expected frequency was below 5. When the minimum expected frequency is below five in more than 20 percent of the cells in the Chi-square table, the statistical tests are not reliable. Therefore, the fourth minor hypothesis was rejected due to the minimum expected frequency being below 5 in 50 percent of the cells in the Chi-square table.

Retention Rate of Computer Science Majors

The third major hypothesis for this investigation and the fifth and sixth minor hypotheses were evaluated using a 2 x 2 Chi-square table and expected frequency. The level of significance was set at the .05 level. Retention data were gathered on the number of computer science majors enrolling in the course and the number of computer science majors successfully completing the course with a grade of "C" or better. Table 4-14 shows the descriptive statistics gathered for the third major hypothesis and the results of the Chi-square test.

H1 3 There will be a difference in the retention rate of computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to the retention rate of computer science majors using a traditional individual learning method in the same course.

The Pearson value of .25 was not significant at the .05 level. There was no significant difference in retention rate of computer science majors when a cooperative study group method was implemented compared to a traditional individual learning method.

Table 4-14 Chi-Square Table on Retention Rate of Computer Science Majors

Group	Success	Percent	Non-Success	Percent	N
Control	21	75.0	7	25.0	28
Treatment	21	80.8	5	19.2	26
Entire Sample	42	77.7	12	22.3	54

Minimum expected cell frequency = 5.77

	Value	df	Significance
Pearson	.25	1	.61

The fifth minor hypothesis was evaluated using a 2 x 2 Chi-square table and expected frequency. Data were gathered on the number of minority students among computer science majors enrolling in the course and successfully completing the course with a grade of "C" or better. The level of significance was set at .05. Table 4-15 shows the Chi-square table and results of the Pearson value calculation. No significant difference was found between minority computer science majors in the cooperative study group method (treatment group) when compared to minority computer science majors in the traditional individualized learning method (control group).

H1 3-5 There will be a difference in the retention rate of minority computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to minority computer science majors in the same course with a traditional individual learning method.

Table 4-15 Chi-Square Results on Retention Rate of Minority Computer Science Majors

Group	Success	Percent	Non-Success	Percent	N
Control	4	66.7	2	33.3	6
Treatment	6	60.0	4	40.0	10
Entire Sample	10	62.5	6	37.5	16

Minimum expected cell frequency = 2.25
Number of cells with expected frequency below 5 3 of 4

	Value	df	Significance
Pearson	.071	1	.78

The fifth minor hypothesis was rejected. The Pearson value of .071 was not significant at the .05 level. The minimum expected cell frequency was less than 5 in 3 of 4 cells in the Chi-square table. This caused the results to be unreliable due to the small sample size.

The sixth minor hypothesis of the investigation was evaluated using a 2 x 2 Chi-square table and expected frequency. Data were gathered on the number of female computer science majors enrolling in the course and the number of female computer science majors successfully completing the course with a grade of "C" or better. Table 4-16 shows the results of the Chi-square analysis.

Table 4-16 Chi-Square Results of Retention Rate of Female Computer Science Majors

Group	Success	Percent	Non-Success	Percent	N
Control	4	80.0	1	20.0	5
Treatment	9	90.0	1	10.0	10
Entire Sample	13	86.7	2	13.3	15

Minimum expected cell frequency = 0.667
Cells with minimum expected frequency below 5 was 3 of 4

	Value	df	Significance
Pearson	.28	1	.59

H1 3-6 There will be a difference in the retention rate of female computer science majors when a cooperative study group method is implemented in the introductory computer science programming course when compared to female computer science majors in the same course with a traditional individual learning method.

The Pearson value of .28 was not found to be significant at the .05 level. There was no significant difference in the retention rate of female computer science majors when the cooperative study group method was implemented compared to female computer science majors in the traditional individual learning method. The small sample size for this hypothesis did not meet the minimum expected cell frequency of 5 for 75 percent of the cells in the Chi-square table. Therefore the results of this test were not statistically valid due to the small sample size. The sixth minor hypothesis was rejected.

Results of the Exit Questionnaire

All students who completed the course, N=68, filled out an exit questionnaire. The purpose of the questionnaire was to obtain qualitative data regarding student reaction to the course in general and specifically to the use of cooperative study groups. The tabulation of the results of the answers to the questions are given below each question. The results and interpretation of these responses will be discussed in Chapter V. For a more complete transcript of student responses to the questionnaire see Appendix F.

Question 1:

"Do you plan to enroll in the next computer science course (CS 25 or CIS 25)? Yes or No? Please explain your reason for choosing yes or no. "

The students in the control group and treatment group generally responded positively to this question. Nearly all the responses of "no" stated that the reason for not enrolling in the next course was that it was not required for the student in his or her major. Almost all of the responses to this question were positive for the students in the treatment group. Very few of the student responses from the treatment group gave negative feedback to this question, and several students stated that they were clearly intent on continuing on in the major. Only one student in the control group stated that the reason for not continuing was because "programming was too difficult" for that particular student. None of the "no" responses from the treatment group gave reasons that regarded dislike of the subject matter or lack of success in the course. Many students in the control group just answered "yes" to the question and gave no explanation. Generally students in both the treatment group and the control group intended to continue on to the next course if it was required for them in their degree plan.

Question 2:

"If you are currently majoring in computer science or computer information science, do you plan to continue on as a major? Please answer yes, no, or undecided, and explain the reason for your answer."

Students in both the control and treatment groups generally responded positively to this question. Two students in the control group stated that dislike for programming and computers was the reason that they would not continue as a major and two other students in the control group stated that they were transferring to other universities in the engineering field. Only one student in the treatment group gave a response of "undecided" and no explanation was given. None of the students in the treatment group indicated a strong dislike for programming or computers in their responses. After completing the course students in both groups generally planned to continue on as a computer science major. None of the students who were not computer science majors stated that they would change their major to computer science.

Question 3:

> "Please list any class activities assignments, lectures, etc., that you really enjoyed in this course and that made the course a successful experience for you."

The responses to this question showed several trends. Many students in the treatment group stated that they really enjoyed the cooperative group method and that it was a positive experience that enhanced their class experience. More than half of the students in the treatment group also indicated that they really enjoyed programming assignments. A few students in the control group also reported programming assignments as an activity that was enjoyed. Students in both the control and treatment groups indicated that the problem sets were very useful and helpful in studying for examinations. A few responses from students in both groups indicated that applying programming problems to real world applications was also enjoyable.

Question 4:

> "Please list any class activities, assignments, lectures etc. that you disliked in this course and that created a disagreeable experience for you."

No negative responses were given from the treatment group concerning the implementation of cooperative study groups. Several students in the control group gave negative responses regarding the difficulty of the programming assignments. Several students in the control group reported disliking the examinations. A few students in the control group mentioned the lack of help in the computer lab at the university. The vast majority of the students in the treatment group responded with "nothing" or left the answer to the question blank. One response from a student in the treatment group mentioned the difficulty of arranging common meeting times for the group as a dislike. Another response from a student in the treatment group indicated that the other members of the group were all males and it was difficult to work with them, but also stated that they liked the concept of the study groups.

Question 5:

> "If you were assigned to a study group, on the average, how much time per week did you study with the group? __________ hours. Did you find study groups to be an effective learning strategy? Why or why not?"

Most students in the treatment group reported studying from 1 to 3 hours per week with their group. Several students mentioned that they found the study groups to be very helpful in completing the programming assignments and in figuring out how programs worked. Many positive comments were received. A few students reported learning more by helping other group members. Several students mentioned that it was very helpful to have other students in their group to rely on for help and to understand the programming problems more easily. Several students mentioned that they thought the study groups were useful and should be continued.

Question 6:

> "If you were not officially assigned to a study group, did you study with others when completing assignments and when preparing for exams? If so, explain how much time per week you studied with others and why you chose to do so."

Only two students in the control group reported helping classmates in the lab with programming questions but did not report meeting regularly with others in the class. One student reported getting help from a student in a previous class but did not state regular meetings or amount of time spent with that other person. The responses to the exit questionnaire are summarized in table 4-17.

Table 4-17 Exit Questionnaire Summary of Responses

Question Number	Positive Responses	Negative Responses	Neutral Responses
1			
Control	70%	28%	2%
Treatment	72%	28%	0%
2			
Control	79%	12%	9%
Treatment	95%	0%	5%
3	Answers to Positive Activities		
		Programming	
Control		48%	
Treatment		55%	
		Problem Sets	
Control		15%	
Treatment		30%	
4	Answers to Negative Activities		
		Programming	
Control		12%	
Treatment		0%	
		Examinations	
Control		12%	
Treatment		1%	
		Reported Nothing	
Control		45%	
Treatment		9%	
5	Average Time per week w/group		
Treatment		2.43 Hours	
6	Positive toward group work		
Treatment		65%	

Summary

In this chapter data have been presented related to three major hypotheses and six minor hypotheses. These data were collected from 81 subjects who originally enrolled in four different sections of the same introductory computer science course. Of the 81 students who enrolled in the course and agreed to participate in the investigation, 13 of these students did not complete the course. Since retention of students was measured in this investigation, the original N=81 is used in calculating retention rate. The number of students who remained in the course and took the posttest achievement test was N=68. Therefore the data results have differing numbers depending on whether retention rate or achievement was being measured.

ANCOVA analysis using an adjusted mean pretest score as the covariate revealed no significant differences between the treatment group (using the cooperative study group method) and the control group (using the traditional individual learning method). The first major hypothesis (H1 1) was not accepted.

ANCOVA analysis was used to evaluate the first and second minor hypotheses of this investigation. The pretest mean was used as the covariate in the analysis of the posttest mean scores. The first minor hypothesis (H1 1-1) was accepted. There was a significant increase in achievement among minority students in the treatment group when compared to achievement of minority students in the control group. The second minor hypothesis (H1 1-2) was rejected. No significant increase in achievement was found between females in the treatment group and females in the control group.

Data were collected on the retention rate of students in the treatment group and control group and a 2 x 2 Chi-square table of expected frequency and Pearson value were used to analyze the second major hypothesis and the third and fourth minor hypotheses. The second major hypothesis (H1 2) was not accepted. No significant differences in retention rate of students in the course were found between the treatment group and the control group. The third minor hypothesis (H1 2-3) was not accepted. There was no significant difference in the retention rate of minority students in the treatment group compared to minority students in the control group. The fourth minor hypothesis (H1 2-4) was not accepted. No significant difference was found in the retention rate of female students in the control group compared to female students in the treatment group.

Chi-square and the Pearson value were used to evaluate the third major hypothesis and the fifth and sixth minor hypotheses of the investigation. Data were collected and analyzed on the number of computer science majors enrolling in the course and successfully completing the course with a grade of "C" or better. No significant difference was found in the retention rate of the treatment group when compared to the retention rate of the control group. The third major hypothesis (H1 3) was rejected.

Chi-square and the Pearson value were used to evaluate the fifth and sixth minor hypotheses. Data were collected on the retention rate of minority and female computer science majors. No significant difference in retention rate was found between minority computer science majors in the treatment group compared to minority computer science majors in the control group. The fifth minor hypothesis (H1 3-5) was rejected. No significant difference in the retention rate was found between female computer science majors in the treatment group compared to female computer science majors in the control group. The sixth minor hypothesis (H1 3-6) was rejected.

The responses to the exit questionnaire were reported as a summary of findings from the six questions. Common trends in responses were given to show the general kinds of responses from the control and treatment group. The results of the investigation are summarized in Table 4-18.

Table 4-18 Results of the Investigation

Summary	Of Hypotheses	Result
H1 1	No significant difference in achievement in computer science programming concepts between students in the treatment group and students in the control group.	F = .02 p = .88 Fail to Accept
H1 1-1	Significant difference in achievement in computer science programming concepts between minority students in the treatment group and minority students in the control group.	F = 5.90 p = .02 Accept
H1 1-2	No significant difference in achievement in computer science programming concepts between female students in the treatment group and female students in the control group.	F = 1.09 p = .92 Fail to Accept
H1 2	No significant difference in the retention rate between students in the treatment group and students in the control group.	Pearson = 1.89 p = .16 Fail to Accept
H1 2-3	No significant difference in the retention rate between minority students in the treatment group and minority students in the control group.	Pearson = 1.07 p = .742 Fail to Accept
H1 2-4	Significant difference in the retention rate between female students in the treatment group and female students in the control group.	Pearson = 3.60 p = .05 Fail to Accept

Table 4-18 Results of Investigation (Continued)

Summary of Hypotheses		Result
H1 3	No significant difference in the retention rate between computer science majors in the treatment group and computer science majors in the control group.	Pearson = .25 p = .61 Fail to Accept
H1 3-5	No significant difference in the retention rate between minority computer sciences majors in the treatment group minority computer sciences majors in the control group.	Pearson = .07 p = .78 Fail to Accept
H1 3-6	No significant difference in the retention rate between female computer sciences majors in the treatment group and female computer sciences majors in the control group.	Pearson = .28 p = .59 Fail to Accept

Chapter V

Summary and Conclusions

Summary

The purpose of this investigation was to determine what effect the implementation of a cooperative study group method had on student achievement in computer science programming concepts and student retention rate in a college level introductory computer science programming course. This study was designed to determine whether or not students are more successful in completing an introductory computer science programming course when they are assigned to cooperative study groups for the specific purpose of working together on assignments, problem sets, and exam preparation.

Student data gathered from four different sections of the same college level introductory computer science programming course concerning the two dependent variables, achievement in computer science programming concepts, and retention rate of students in the course were analyzed in order to obtain answers to the research questions under investigation.

The study utilized a non-randomized pretest-posttest design with a control group and a treatment group. The students in the investigation were assigned to either a cooperative study group method of learning or to a traditional individual method of learning. The treatment group refers to the students assigned to a cooperative study group method and the control group refers to the students assigned a traditional individual learning method (no study groups). The 81 subjects who began this investigation were all enrolled in the same introductory computer science course at a small liberal arts university located in central Texas. Of the four different sections of the computer science course, students in two intact classes (sections) were assigned to the treatment group and the remaining two sections of students were assigned to the control group. The study required two semesters to complete, but the data for the subjects were obtained over a 14 week period. Two intact classes in the study were two different sections of the same course in the Fall 1993 semester and the remaining two intact classes comprised two different sections of the same course from the Spring 1994 semester. Student data were gathered during both the Fall 1993 semester and the Spring 1994 semester. This was done in order to have a large enough sample size for the study.

Student achievement in computer science programming concepts was measured using a written test instrument with a split halves reliability measure of $r=0.85$ and an average rank difference correlation coefficient of 0.91. Student retention rate was measured by obtaining the number of students enrolled in the course as of the 12th class day and the number of students successfully completing the course with a grade of "C" or higher.

Evaluation of student achievement in computer science programming concepts between students in the treatment group and the control group was statistically analyzed using ANCOVA procedures on the posttest achievement score with the pretest score serving as the covariate. The level of significance was set at the .05 level.

Evaluation of student retention was analyzed using a 2 x 2 Chi-square table with expected frequencies and the Pearson coefficient. The level of significance was set at the .05 level.

Students finishing the course completed an exit questionnaire in order to obtain qualitative data on student reactions to the treatment. A summary of the results of this questionnaire is presented.

Summary of Findings

The nine research questions were answered as a result of the analysis of data presented in Chapter IV. The significant effects of the treatment are examined after the research questions are presented.

Question 1. How does the use of a cooperative study group method in an introductory college level computer science programming course effect achievement in computer science programming concepts when compared with a traditional individual study method in the same course?

Analysis of the data indicated that the use of a cooperative study group method in an introductory college level computer science programming course did not significantly increase student achievement in computer science programming concepts when compared to a traditional individual study method in the same course. The results of the data analysis found that students achieved at the same rate when participating in a cooperative study group method as when participating in a traditional individual learning method.

Question 2. Does achievement by minorities change in an introductory computer science programming course when a cooperative study group method is incorporated into the teaching methodology?

Achievement by minorities participating in the cooperative study group method was significantly higher compared to minorities participating in the traditional learning method. Analysis of the data indicated that the implementation of a cooperative study group method was clearly an effective means of increasing achievement for minority students.

Question 3. How does the achievement of women change when a cooperative study group method is incorporated into an introductory computer science programming course?

Achievement by females participating in the cooperative study group method was not significantly higher than achievement by females participating in the traditional individual learning methods. Females did attain higher average achievement scores when participating in the cooperative study group method that could be educationally significant although not statistically significant.

Question 4. Will the retention rate of students in classes using a cooperative study group method be greater than, less than, or the same as the retention rate of students in non-cooperative study group classes of introductory computer science programming?

The implementation of a cooperative study group method did not significantly increase the retention rate of students when compared to a traditional individual learning method. The data did indicate higher retention percentages for all students participating in the cooperative study group methods although not statistically significantly higher. The implementation of study groups did increase the percentage of success of students and educationally this can be useful when choosing instructional methods since increasing retention is a desirable educational objective in computer science courses.

Question 5. Does the course retention rate of minorities improve when a cooperative study group method is incorporated into an introductory computer science programming course?

There was no significant difference in the retention rate of minority students when a cooperative study group method was implemented. Minority students did have a higher percentage of retention when a cooperative study group method was implemented, however the difference was not significant.

Question 6. Does the course retention rate of women improve when a cooperative study group method is incorporated into an introductory computer science programming course?

Female students participating in the cooperative study group method did not have a statistically significantly higher retention rate than those participating in the traditional individual learning method. The data indicated a strong positive effect on retention for females when a cooperative study group method was implemented. This demonstrates that cooperative study groups are clearly a positive way to increase retention of females in introductory computer science programming courses.

Question 7. Will the computer science major retention rate increase or remain the same when a cooperative study group method is used in an introductory computer science programming course?

The cooperative study group method did not significantly increase the retention rate of computer science majors. The percentage of retention was higher for those computer science majors in the cooperative study group method than for those computer science majors in the traditional individual learning method class. Pedagogically it seems practical and beneficial to implement cooperative study groups as a means to increase retention rates. Although the increase was not statistically significant, retention percentage was higher for those students who participated in the cooperative study group method.

Question 8. Does the computer science major retention rate of minorities improve when a cooperative study group method is incorporated into an introductory computer science programming course?

The retention rate of minority computer science majors did not improve significantly with the implementation of a cooperative study group method. No differences were found among minority computer science majors' retention rate in the cooperative study group method when compared to the retention rate of minority computer science majors in the traditional individual learning method.

Question 9. Does the computer science major retention rate of women improve when a cooperative study group method is incorporated into an introductory computer science programming course?

Female retention rate for computer science majors was not significantly higher for the cooperative study group method students. A higher retention rate was seen for women participating in the cooperative study group method than for those in the traditional individual learning method. While not statistically significant, improvement was seen in the retention rate of women when a cooperative study group method was implemented.

Discussion and Conclusions: Student Achievement

Students participating in the cooperative study group method did not have significantly higher achievement than students participating in the traditional learning method. The overall mean achievement score was almost exactly the same for both learning methods after adjusting for the pretest covariate. The standard deviation, indicating the amount of variance of the achievement test scores, was considerably lower for the cooperative study group methods students at SD = 11.37 when compared to the control group's SD = 17.39. The difference of 6.02 is a large difference in the achievement test score accounting for more than one half of a letter grade difference since letter grades are calculated on a traditional 10 point scale per grade.

The large difference in standard deviations could be interpreted in several ways. Overall, the students in the cooperative study groups had a more even distribution of scores around the mean and much less variance. This is noteworthy since the consistency of the achievement scores was much more reliable for the students in the cooperative study group method. Students in the traditional individual learning method had a much larger variance in the posttest achievement score, indicating the presence of several very high and several very low scores. A smaller standard deviation among achievement scores indicates that most students are performing close to the average and few are at the extremes of doing exceptionally well and exceptionally poorly. While it is most desirable to have all students do exceptionally well, it is also a goal to have as few students as possible be unsuccessful. The cooperative study group treatment did not effectively increase achievement on the posttest scores, but the those students in that group did show more consistency in performance.

When examining the achievement of minority students and female students within the cooperative study group method and the traditional individual learning method, some important findings are revealed. Minority students had a significant increase on the posttest achievement score when cooperative study group methods were implemented. This is consistent with the non-experimental observations of Fullilove and Treisman (1990) in their

Mathematics Workshop Program (MWP). Minority students did consistently better in a first year college calculus course when participating in study groups arranged by the MWP. The standard deviation of the posttest score for minority students in the cooperative groups was SD = 8.78. This is much lower than the SD = 14.69 for minority students in the traditional individual learning methods. Minority students performed consistently better on the achievement posttest when participating in cooperative study groups when compared to the traditional individual methods. Since one of the problems facing computer science departments at this time is the lack of minority students, it would seem imperative to use teaching methods that increase the achievement and therefore the success of minority students. Cooperative study group methods have seem to increase achievement of minority students which is one of the first steps in leading to their success in the computer science field (Seymour, 1992).

Female students that participated in the cooperative study group method did perform higher on the average on the achievement posttest score than did females in the traditional individual method, although the difference was not statistically significant. While the higher mean achievement score of females in the cooperative study group method may not be significantly attributed to the treatment effect of using cooperative study groups, females did perform better when participating in the cooperative study groups. This is useful information to college/university instructors because the cost associated with implementing cooperative study groups is very low while the benefits are positive.

With the need for more minority and female students in computer science now and in the future (Widnall, 1988; Massey, 1989; Block, 1990) cooperative study group methodology would seem to benefit the very populations that are most desired in the discipline. With the ease of implementation of cooperative study groups, the very low cost of using cooperative study groups, and the fact that all students in cooperative study groups do as well or better than students not using cooperative study groups, computer science educators should consider implementation of cooperative study groups for introductory computer science programming courses. The implementation of cooperative study groups appears to be a beneficial teaching strategy for instructors of computer science at a very minimal cost.

Discussion and Conclusions: Retention Rate

In examining the retention rate of students participating in the cooperative study group method several observations are educationally important. Retention rate is associated with success in the course, since many students are taking the course to continue on in computer science or to fulfill a requirement for a degree. Success in this context indicates that students received a grade of "C" or higher in the course.

Although no significant difference in retention rate was found when comparing all students in the cooperative study group method to all students in the traditional individual learning method, the percentage of success was higher in the treatment group than it was in the control group. There were 30 of 38 students who were successful under the cooperative study group method while the number of students who were successful in the traditional individual method was 28 out of a total of 43. The Chi-square value was not statistically significant for this measure. However, the fact that more students in the treatment group were successful (78.9 percent versus 65.1 percent) is certainly noteworthy from a pedagogical view. The treatment group certainly did have a higher success rate even though statistically that higher percentage was not significant. Educationally, observing that a larger percentage of students were successful with the treatment effect merits consideration for the use of cooperative study group methods and continued research. It is also important to recognize that the implementation of cooperative study groups in no way decreased the success rate of students.

The retention of minority students in the treatment group was not significantly increased by the implementation of study groups. Several factors should be mentioned. The minority students in the cooperative study group methods did have a higher than expected number of students retained overall, but the number was not statistically significant. Educational significance depends on the context and perspective of the results. Since retention rate of minority students did not show significant increase or decrease with the implementation of cooperative study group methods, the use of these cooperative study group methods may not effectively increase or decrease retention. With 9 of 15 minority students successful in the treatment group and 7 of 13 minorities successful in the control group, the percentages of success do not vary greatly but the students in the cooperative study groups were more successful.

The retention rate of female students participating in the cooperative study group methods was not significantly higher than the retention rate of female students in the control group. The statistical significance of the results was negated by the low number of female students in the study. When using the Chi-square distribution, the reliability of the statistical measurement depends on not having 20 percent or more of the cells with a minimum expected frequency below 5. The Chi-square requirement was violated in the analysis of female students in the study. In 50 percent of the cells in the 2 x 2

Chi-square table used for analyzing female retention, the minimum expected frequency was less than 5. This caused the results of the test to possibly be unreliable since the number in the sample was too low. The Pearson coefficient of 3.60 would have been significant at the .05 level of significance had the minimum expected frequency been 5 or higher in all cells. It is useful to examine the rate of success (completing the course with a grade of "C" or higher) of females in the cooperative study group methods. Out of 15 female students in the treatment group, only 1 was considered not successful. In the control group, 4 of 11 female students were not successful. The percentages of success have less meaning because of the small sample size. However, from an educational viewpoint, cooperative study group methods did have a higher retention rate(success) for females when compared to females participating in the traditional individual learning method.

The findings regarding the third major hypothesis concerning the retention rate of all computer science majors indicated no statistical difference among computer science majors retention in the cooperative study group method when compared to computer science majors in the traditional individual learning method. Of 21 majors in the treatment group, 5 were not successful. Of the 21 majors in the control group, 7 were not successful. The difference in these success ratios is not significant, but it is important here to state that no decreases in success ratios were caused by the treatment. The retention rate was about the same for both groups. The results reveal that when a cooperative study group method is implemented with computer science majors, there is no decrease in the retention rate percentage. Any other benefits possibly gained by the cooperative study group method can still occur since students have the same rate of success in either learning method.

The retention rate of minority computer science majors was not significantly higher with the implementation of cooperative study group methods. The percentages of successful students were higher in the treatment group, but the nature of the statistical analysis did not confirm statistical significance. It is important to note that there was no decrease in the minority computer science retention rate when cooperative study group methods were implemented.

The retention rate of female computer science majors was not significantly higher for female computer science majors participating in the cooperative study group method when compared to female computer science majors participating in the traditional individual learning methods. The number of computer science females in the treatment group who were successful was 9 out of 10. The number of female computer science majors in the control group who were successful was 4 out of 5. Statistically it is not possible to conclude that there is a significant difference in those ratios of success, however, pedagogically, it seems meaningful that only 1 student in 10 was unsuccessful in the cooperative study group method. Again, it is noteworthy to state that the percentages of success clearly did not decrease with the implementation of a cooperative study group method when compared to a traditional individual learning method for female computer science majors.

Exit Questionnaire: Discussion and Interpretation

The purpose of the exiting questionnaire was to solicit information from students as to their perceptions of the class in general and specifically how successful cooperative study groups appeared to be to students. The intention of the questionnaire was to determine why students liked or disliked the cooperative study groups and what they liked or disliked about the class.

Questions 1 and 2 were developed to determine whether or not computer science major students would enroll in the next course in the computer science sequence and continue in the computer science major. Responses from students in the control group indicated that a few of the computer science majors would not enroll in the next course in the sequence and would not continue as a computer science major. The reason that was given by those students was the difficulty of programming and dislike of the computer science subject matter. The investigation was performed to address these particular reasons for students not continuing on in the computer science major and to prevent or reduce the attrition of computer science majors because of those factors. Responses from the treatment group indicated that all computer science majors would enroll in the next course in the computer science sequence and continue majoring in computer science. None of the computer science majors who participated in the cooperative study group method stated that they would not continue as a major. This is exactly the response that was expected from the implementation of cooperative study groups. The purpose of the investigation was to find a learning method that would lead to success and retention of students in the computer science program. Responses to this question clearly indicated that the cooperative learning method did achieve this goal, which is very important to computer science educators. The implementation of a cooperative study group method in this investigation did lead to higher success and appeared to have prevented high attrition rates of students in the computer science major.

Question 3 was designed to determine which class activities seemed to be most useful to the students. The wording of the question was such that students could mention any activity that they thought was particularly useful during the class. No mention of study groups was made in the actual question itself in order to determine if students would voluntarily mention the study groups as being a positive experience. There were several responses to this question indicating that the assignment of cooperative study groups was perceived as useful to the students. Some of the specific comments received were extremely positive and complimentary to the implementation of cooperative study groups. Comments ranged from students stating that they could not have completed the course without others in their group, to actually learning more from teaching and showing others how problems and programs worked. Students stated that study groups helped them through difficult assignments and that it was useful to have other students to turn to for help, particularly in the computing lab. All of the responses from students who mentioned the use of study groups as a positive class activity indicated that those students perceived study groups to be useful in completing the course and learning the material. These responses strengthen the rationale for implementing cooperative study groups in introductory computer science programming courses.

Other responses to question number 3 regarding positive class activities are also interesting to computer science educators. The programming problems were mentioned as very beneficial to learning from students in both the treatment group and the control group regardless of their major. This indicates that students do perceive programming problems as necessary activities and positive methods for learning. Therefore, any activities that can improve programming success can also lead to success in the course. Students did indicate that cooperative study groups helped with completing programming assignments and therefore it seems logical to implement cooperative groups for this purpose. It should also be noted here that the use of cooperative groups did not cause achievement to decrease, so getting help from other students in completing assignments did improve students' perceptions of learning and did not cause them to learn any less according to the achievement score results.

Several students from both the treatment group and the control group stated that the problem sets were very beneficial in helping learn the programming concepts throughout the course. While various key problems have generally been incorporated into that particular course curriculum throughout the semester, no actual "problem sets" were utilized in the course until this study. The concept of using problem sets was modelled from the Mathematics Workshop Program of Fullilove and Triesman (1990). The availability and assignment of problem sets seems to be very useful to students in an introductory computer science programming course. Computer science instructors should attempt to include such problem sets in their introductory courses.

Question 4 was designed to determine which class activities students particularly disliked. From examining the responses to question 4, very little negative feedback was received. Some of the students in the control group stated that the programming assignments were too difficult and too long, that the lack of help in the computer lab was very frustrating, and that the examinations were too difficult. Only a very few negative responses were received from students in the treatment group and they all mentioned disliking the examinations. No other negative responses were given by the students in the treatment group. This indicates that students in the cooperative study group method were possibly more successful with the programs due to help and support from their cooperative study group peers. Again it is important to note that none of the students in the treatment group mentioned the cooperative study group participation negatively nor did they state that they disliked being part of a cooperative study group.

Question number 5 asked students who participated in the groups how much time they spent with their group per week and whether or not they perceived the study groups to be effective in learning. The students in the cooperative study groups had only two different negatives responses to participating in cooperative study groups. The first and most common negative feedback regarding cooperative study group work was that students stated that time conflicts among the group members made it difficult to meet and sometimes the entire group could not meet together, so members met in pairs or groups of three. The other negative feedback toward the cooperative study groups involved a female student stating that the other members were all male and too advanced for her to work with and to learn from. Positive comments were received on almost all of the questions. These comments ranged from "group work was excellent and useful" to "groups helped me learn when I showed others how programs worked." The most important concept to state is that the students in the groups generally had positive comments about groups and found them beneficial. None of the responses to question 5 stated that the participants disliked group work nor did they state that it was a detriment to their learning in any way. This is significant to the study, since student perception and attitude could effect student success in the course. The questionnaire responses generally had very positive comments toward group work and no significant negative responses.

Conclusions

Cooperative study groups were helpful to many of the participants in learning the course material. This learning may or may not have been reflected on the achievement test scores. Student perception of learning may be very important in retaining students in the major and in keeping them interested in the course material. This could be researched further through attitude and motivation testing with cooperative study groups. With the need for more computer scientists in the future labor force, particularly minorities and women (Massey, 1989; Block, 1990), computer science educators should consider implementing cooperative study groups in introductory computer science classes. The benefits of placing students in cooperative study groups may not be overwhelming in the numerical assessment of achievement and retention, but in student perception and attitude it seems that students had a very positive experience in cooperative study groups as stated by students in their responses to the questionnaire and questions posed.

The traditional individual learning method that has been in place for so long in introductory computer science courses is not promoting student success and retention in computer science (Widnall, 1988; Massey, 1989; Block, 1990; Seymour, 1992). Exploring the use of a cooperative study group method in an introductory computer science course has shown that minority students do achieve better and that students do intend to continue on in the computer science program when cooperative study groups are implemented. Previous research has shown that minorities and women tend to learn better in a noncompetitive environment that promotes cooperation and socialization (Turkle, 1984; Sproull & Kiesler, 1986; Fullilove & Triesman, 1990; Howell, 1993). Computer science educators need to find ways to encourage students to remain in the computer science major program and to achieve success in the program. The implementation of cooperative study groups in an introductory computer science program appears to lead in that direction. While much of the statistical analysis of this study was not significantly successful with the implementation of cooperative study groups, student attitudes and perceptions of cooperative study group learning is certainly positive and perceived as useful to those students who were in cooperative study groups.

There is relatively no cost associated with the implementation of a cooperative study group method in an introductory computer science course. The instructor is still able to cover the same material, give the same assignments, and give the same examinations as in a traditional individual method class. The benefit of the cooperative study group method is in the students' attitude toward learning the material and in their perceptions of success in the course as well as their achievement in the course. Having interaction with other students in the course, helping others solve problems, communicating with other students, and the socialization process of working together appears to be very beneficial to students according to this study.

Educators in computer science need to find techniques to retain majors, particularly women and minorities. The use of cooperative study groups is one method that is cost effective, easy to implement, and beneficial to all students, women and minorities in particular.

Recommendation for Future Research

The results of this investigation indicate the need for further research into the implementation of cooperative study group methods in college/university level classes in computer science. Several problems could be addressed in this future research.

One of the problems students in the cooperative study group method stated was that they had difficulty arranging group meeting times. Students in cooperative study groups might have designated meeting times when all group members can attend. This could be pre-arranged with the course schedule description. Alternatively, the arrangement of cooperative study groups according to meeting time availability instead of random assignment might be examined. Either of these techniques might provide further evidence to educators as to the benefits of cooperative study groups and improve those benefits for students.

Fullilove and Triesman (1990) used study groups with designated group mentors in their Mathematics Workshop Program. These group mentors were graduate assistants who were available during group meeting times to answer questions and help group members solve problems. A replication of this current investigation with the addition of cooperative study group mentors who were students previously successful in the course or computer science graduate students if available, might provide further improvement in retention and achievement of students participating in cooperative study groups.

Other studies similar to this one should be conducted to assist in the generalizability of these results. Larger samples to include more women and minority students would improve the external validity of the study and would allow for more statistically reliable results regarding student retention.

A replication of this study that includes the examination of student attitudes and motivation toward success in computer science might provide further evidence as to the benefits of implementing cooperative study groups in computer science.

Since the purpose of this study was to investigate teaching strategies that improve student learning in computer science and lead to greater student success, further research might investigate the reasons for students not finishing this type of computer science course. This might be done by having students fill out an exiting questionnaire if they withdraw from the course before the end of the semester, and to have all students fill out a similar exiting questionnaire regarding their own perceptions of their success or failure in the course. This could eliminate many external conflicts from the results of the study on student retention in order to make a clearer examination of retention due to the treatment effect.

Appendix A

Consent Form for Participation in Research Study at St. Edward's University

Consent Form for Participation in Research Study at St. Edward's University

Study Title: The Effect of Cooperative Study Groups on Achievement of College-level Computer Science Programming Students

You are invited to participate in a study being conducted in this introductory computer science course. All information obtained in this study will remain confidential. No information identifying you will be released except to you at the conclusion of the study.

This is to certify that I ______________________________________ hereby agree to allow my responses to be used in a scientific study. I understand that my data and responses will remain confidential with regard to my identity. I have been given an opportunity to ask whatever questions I may have had and all such questions and inquiries have been answered to my satisfaction. Furthermore, I understand that my withdrawal from this study at any time will not jeopardize my status in this course nor will it affect in any way my status at St. Edward's University or The University of Texas at Austin. I hereby grant permission to Laura J. Baker, instructor at St. Edward's University to use my test results and personal data in a research study conducted during the 1993-1994 academic school year.

Signed______________________________Date__________________________

Appendix B

Biographical Data Sheet

Biographical Data Sheet

Please give the last 4 digits of your social security number

___ ___ ___ ___

Circle or fill in the response that best describes you.

1. Sex: Male Female

2. Declared Major: Computer Science Computer Information Sciences

   Other ________________ Undecided

3. Reason for taking this course: Required for major Elective

   Other

____

4. Ethnic background: African-American Caucasian

   Hispanic Asian

   Other ________________________

5. Number of semesters of previous computer science including high school:

6. Number of hours of college credit at all institutions:

7. Number of hours of college credit taken at St. Edward's:

8. Current GPA:

9. SAT score:

Appendix C

Exiting Questionnaire

Exiting Questionnaire

This is NOT a test. Please take your time and carefully answer the following questions.

Your 4 digit identification number ___ ___ ___ ___

1. Do you plan to enroll in the next computer science course (CS 25 or CIS 25)? Yes or No? Please explain your reason for choosing yes or no.

2. If you are currently majoring in computer science or computer information sciences, do you plan to continue on as a major? Please answer yes or no or undecided and explain the reason for your answer.
(if you are not a major but now plan to become a major, please indicate !)

3. Please list any class activities, assignments, lectures etc. that you really enjoyed in this course and that made the course a successful experience for you.

4. Please list any class activities, assignments, lectures etc. that you disliked in this course and that created a disagreeable experience for you.

5. If you were assigned to a study group, on the average, how much time per week did you study with the group ? ____________ hours. Did you find study groups to be an effective learning strategy? Why or why not?

6. If you were not officially assigned to a study group, did you study with others when completing assignments and when preparing for exams? If so, explain how much time per week you studied with others and why you chose to do so.

Appendix D

Computer Science Achievement Test Instrument

and

Answer Key

Computer Science Achievement Test

CS23 Final Exam   Name________________

Part I. Multiple Choice -- choose the single best answer for each item below (2 points each -- 64 points)

____ 1. Any Pascal program **must** have
a. a type declaration
b. a function
c. a loop
d. an identifier
e. all the above

____ 2. A step-by-step finite process for solving a problem is
a. an axiom
b. a protocol
c. a dialogue
d. an algorithm
e. none of the above

____ 3. Which of the following is **not** a built-in Pascal function?
a. PRED
b. SQR
c. ROUND
d. SUCC
e. none -- they all are

____ 4. If there are two different conditions which could cause a loop to stop, which Pascal construct(s) should you use?
a. WHILE loop
b. REPEAT loop
c. FOR loop
d. a and b only
e. a and c only

____ 5. The value of the Pascal expression 11 MOD 3 / 5 is
a. 1.0
b. 0.4
c. 0.2
d. 0.733
e. error

____ 6. Which of the following is a legal Pascal identifier?
a. 2nd
b. begin
c. Second
d. first-choice
e. all of the above

____ 7. Which of the following may appear on the left hand side of an assignment operator?

a. a procedure name d. MAXINT
b. a literal e. none of the above
c. a variable identifier

____ 8. What is the value of

TRUE AND ((Y>Z) OR (NOT (X>10)))

a. TRUE d. not enough information
b. FALSE e. error
c. -10

____ 9. The value of the Pascal expression TRUNC (106.8) is

a. 106 d. error
b. 107 e. none of the above
c. 1.064 e+2

____ 10. Every Pascal procedure

a. is accessible from the main program
b. must have at least one variable parameter
c. must contain at least one local variable
d. must end with a semicolon
e. none of the above

____ 11. A FOR loop is a special case of a/an

a. event controlled loop
b. sentinel controlled loop
c. count controlled loop
d. flag controlled loop
e. none of the above

____ 12. The result of a function is communicated through

a. value parameters d. global variables
b. variable parameter e. all of the above
c. the function name

____ 13. Given the following subprogram heading, you can tell that
FUNCTION Exam (A,B : integer) : boolean;
a. no global variables are used in Exam
b. no local variables are used in Exam
c. the result type of EXAM is **boolean**
d. B is an export (out) parameter
e. none of the above

Given the following code, answer questions 14-16:

```
TYPE ExArray = ARRAY [-6 .. 10] OF char;
VAR  ChArray : ExArray;
         i : integer;
BEGIN
     ChArray [0] := 'M';
     FOR i := 1 TO 10 DO
          ChArray [ i] := PRED (ChArray[i-1]);
END.
```

____ 14. How many elements may be stored in ChArray?
a. 4 **b.** 6 **c.** 10 **d.** 16 **e.** 17

____ 15. What is the value of ChArray [3]?
a. 'J' **b.** 'P' **c.** 'M'
d. undefined **e.** none of the above

____ 16. What is the value of ChARray [-3]?
a. 'J' **b.** 'P' **c.** 'M'
d. undefined **e.** none of the above

____ 17. The Pascal type BOOLEAN is
a. printable
b. enumerated
c. ordinal
d. pre-defined
e. all of the above

____ 18. What is the value in Pascal of the expression:
(14 DIV 5 + 10 / 4)
a. 3
b. 4.5
c. 5.3
d. 4
e. None of the above

____ 19. A selection/exchange sort
a. requires an array
b. is most effective with small amounts of data
c. can be used to sort in ascending order
d. will have one less pass than the number of elements
e. all of the above

____ 20. An example of a heterogeneous data type is
a. a record which has a character field and a real field
b. a one dimensional array
c. a real number
d. an enumerated data type
e. all of the above

Use the following syntax diagram to answer question 21.

```
------> Even ----------> Odd -------> Even ---------->
                         ^                |
                         |                |
                         |                |
                         ------------------

        Even                           Odd
  >-------------------          >--------------------
  |       |    |    |           |     |    |    |   |
  |       |    |    |           |     |    |    |   |
  |       |    |    |           |     |    |    |   |
  v       v    v    v           v     v    v    v   v
  2       4    6    8           1     3    5    7   9
  |       |    |    |           |     |    |    |   |
  |       |    |    |           |     |    |    |   |
  |       |    |    |           |     |    |    |   |
  v       v    v    v           v     v    v    v   v
 --------------------->         ---------------------->
```

____ 21. Which of the following series would be correct according to the above syntax diagram?

a. 212	d. both a and b
b. 2376	e. both a and c
c. 6135894	f. both b and c

____ 22. Which of the following **writeln** statements will print the value of an integer variable Degrees (including formatting instructions) after the words **The angle is**?

a. writeln ("The angle is ", Degrees:3:0)
b. writeln ('The angle is ', degrees:3)
c. writeln ('The angle is ,' Degrees:3)
d. writeln ('The angle is ', Degrees:3:0)

Use the following code fragment and data to answer question 23

```
VAR A, B, C, D : integer;
...
...
read (A);
readln (B,C);
read (D);
```

and this data:

```
24  72
46  55 62
18  4
```

____ 23. Which of the following will be true after the above code is executed?

a.	A = 24	B = 46	C = 55	D = 18
b.	A = 24	B = 72	C = 46	D = 18
c.	A = 24	B = 72	C = 55	D = 62
d.	A = 24	B = 46	C = 55	D = 62
e.	none of the above			

Based on the following Pascal code, what is printed for the values specified in questions 24 and 25?

```
CASE N DIV A OF
     0: writeln ('Moody');
     1: IF N < 2 THEN write ('Andre')
                 ELSE write ('Fleck');
     2: {Do nothing}
     3,4: writeln ('A is ', A:1)
END
```

____ 24. Given N = 2, A = 7 :
a. Moody
b. Andre
c. Fleck
d. A is 7
e. error

____ 25. Given N = 4, A = 2 :
a. Moody
b. Fleck
c. A is 2
d. error
e. nothing

____ 26. 1011_2 = ____________________$_{10}$
a. 22
b. 11
c. 1011
d. B
e. none of these

____ 27. 964_{10} = ____________________$_2$
a. 10111_2
b. 101101011_2
c. 1111000100_2
d. impossible
e. none of these

The following definitions and declarations are referred to in question 28.

```
TYPE NameType = string[20];
     Disc = RECORD
       Title, Artist : NameType;
       Year : 1900..2000;
       RPM : 16 .. 78
     END;
     Album = ARRAY ['a' .. 'z'] OF Disc;
VAR Phonorecord : Disc;
    Star : NameType;
    Collection : Album;
```

____ 28. Which is an invalid (illegal) statement?
a. Phonorecord.RPM := 33;
b. Star := Disc.Artist;
c. Phonorecord.Year := 1958;
d. Collection['z'].Title := 'White Christmas';
e. none of the above (all are correct)

Questions 29-30 are based on the following Pascal code: (Assume that X and Y are integer variables, C is a boolean variable)

```
X := 2;
Y := 6;
C := FALSE;
WHILE (X<Y) AND NOT C DO
BEGIN
    IF (Y MOD X) = 0 THEN
        C := TRUE;
    X := X + 1;
END;
```

____ 29. How many times will the body of the loop be executed?

a. 0
b. 1
c. 2
d. 3
e. 6

____ 30. What is the final value of the variable C?

a. TRUE
b. FALSE
c. MAYBE
d. 9
e. not enough information

____ 31. Parameters that will change in a procedure or function are called

a. Variable parameters
b. Reference parameters
c. Address parameters
d. Value parameters

____ 32. Parameters that will not change in a procedure or function are called

a. Variable parameters
b. Reference parameters
c. Address parameters
d. Value parameters

Part II. Coding.

Given: type realarray = array [1..max] of real;
var salescom:realarray;

33. (6 points) Write a **function** called MaxSales(SalesCom, N) that returns the largest value in the array of N real values called SalesCom (assume data is already placed into the array).

34. Write a procedure called Add100(salescom, n) that adds 100.0 to all values in the array salescom that are BELOW 500.00, returns the array with the new values added appropriately.(6pts)

35. (6 points) Write a procedure called practice that asks the user to input two values. The procedure then outputs the product of the two inputs. The procedure repeats this practice until the user responds with 'n' when asked if he/she wants to repeat the process again.

36. (18 points) Write a complete Pascal program to do the following: Ask the user to input a set of test scores for a class.(NO MORE THAN 25 students are in any given class.) The scores will be terminated by the integer value -999 when entered by the user. The program outputs the average of the set of scores, a separate list of scores that are greater than the average, and a separate list of those scores whose values were less than the average. You should write at least 4 subprograms:

1. to get the numbers
2. to calculate the average
3. to print a list of values less than the average
4. to print the list of values above the average

All output is to the screen and the numbers are input from the keyboard. Make sure your arguments from the main program are passed appropriately to each subprogram where they are needed.

```
(i.e., If the input were:  70 60 100 90 80 80 30 100 40 100 -999
     Output:     Average = 75.0
                 Scores above average: 100 90 80 80 100 100
                 Scores below average:  70 60 30 40        )
```

Answer Key to Test Instrument

1. D	10. D	19. E	28. B
2. D	11. C	20. A	29. B
3. E	12. C	21. D	30. A
4. D	13. C	22. B	31. A
5. B	14. E	23. B	32. D
6. E	15. A	24. A	
7. C	16. D	25. E	
8. D	17. E	26. B	
9. A	18. B	27. C	

33). (1 point deducted each syntax, semantic error)

```
Function Maxsales(salescom:realarray; n: integer):real;
Var i:integer; max:real;
Begin
  max := salescom[1];
  for i:= 2 to n do
     if salescom[i] > max then
          max:= salescom[i];
  maxsales := max;
End; (* Maxsales *)
```

34). (1 point deducted each syntax, semantic error)

```
Procedure Add100(var salescom:realarray; n:integer);
Var
     i:integer;
Begin
  For i:= 1 to n do
     if salescom[i] < 500.00 then
        salescom[i] := salescom[i] + 100.00;
End; (* Add100 *)
```

35). (1 point deducted each syntax, semantic error)

```
Procedure Practice;
Var
   a,b,product:integer;
   ans: char;
Begin
   Repeat
     write('Enter 2 integers please: ');
     readln(a, b);
     product := a * b;
     writeln('The product is : ', product:5);
     write('Repeat again? y/n ');
     readln(ans);
   Until (ans = 'n') or (ans = 'N');
End; (* Practice *)
```

36). 1 point deducted each syntax, semantic error
3 points each procedure written correctly
4 points main program and var declarations

Appendix E

Problem Sets

CS 23 Integer Conversion Exercises

Using an 8 bit word length, convert the following values

	1's Complement	2's Complement
1) -94	______________	______________
2) +110	______________	______________
3) -23	______________	______________
4) +49	______________	______________

Using a 10 bit word length convert following values

	1's Complement	2's Complement
5) +278	______________	______________
6) -190	______________	______________
7) -2	______________	______________
8) +87	______________	______________

Convert following values to Decimal assuming they are in form

	1's Complement	2's Complement
9) 10111011	______________	______________
10) 00111101	______________	______________
11) 11101101	______________	______________
12) 01011000	______________	______________
13) 011011010111	______________	______________
14) 101101101101	______________	______________
15) 000101101010	______________	______________
16) 111111011011	______________	______________

CS 23 Practice Problems for Decisions

1. Write a segment of code to input a letter grade, output the total grade points earned for the letter grade.

2. Input a quantity sold, calculate discount on that quantity, calculate price at $25 per unit sold, output the discount price using the following chart:

Quantity	Discount
0 -50	10%
51-100	20%
101-200	25%
201+	30%

3. Input 3 numbers, output the largest of the 3 numbers.

4. Input hours worked and pay rate, output net pay, giving time and a half to all hours over 40 and tax withheld as 23% of gross pay.

5. Input the age, registration status (y or n) of a person. Output the appropriate message as to whether or not they may vote in an upcoming election.

6. Input a number, output the following information about the number:
1. Is it even or odd? 2. Is it positive, negative or zero? 4. Is it divisible by 3?

7. Ask the user to input the temperature and wind velocity in mph. Print whether or not they should go sailing. Sailing is okay if wind is below 30 mph and the temperature is above 30 degrees. If the wind is 30 to 40 mph and the temperature is over 45 degrees, sailing is still okay.

8. Write a menu program to print an appropriate greeting:
1. Happy Birthday
2. Merry Christmas
3. Happy Valentines Day
4. Founder's Day Celebration

9. Write a segment of code to ask the user to input 3 characters. Output each character and a message stating it was a letter, digit, punctuation or other.

10. Write a program to calculate cost of a banquet, costs are as follows: 1-20 persons with chicken or beef 8.95 per person, seafood or pasta 9.95 per person 21-40 persons with chicken or beef 7.95 per person, seafood or pasta 8.50 per person , over 40 persons 7.50 per person for anything. Dessert and beverage add 2.50 per person for up to 50 people, 2.00 per person if over 50 people. Your program should input the number of people, type of food desired and output the cost.

11. Input 4 test scores, homework score, and quiz average. Calculate letter grade. (To calculate letter grade, find their average ==> drop lowest test score, give test avg 70%, homework 20%, quizzes 5%., then use a traditional scale for the grade assignment, rounding off the average.)

CS 23 **Repetition Practice Problems**

Write a segment of Pascal code to do each of the following.

1. Input a set of numbers (integers) terminated by the value -999. Calculate the average, the largest and smallest value in the set of numbers.

2. Print a metric conversion chart which shows pounds converted to kilograms.
Chart should print pounds from 1 to 50 and their equivalent kilograms. (there are 2.54 pounds in 1 kilogram.

3. Prompt the user to enter a sentence terminated by the enter key. Report the number of vowels, digits, question marks, commas, periods and letters in the sentence.

4. Ask the user to input 2 numbers, output the sum of the squares of the 2 numbers. Continue to get numbers until the user answers 'n' when asked to repeat again.

5. Find the sum of the numbers from 100 to 200 inclusive.

6. Write a menu driven program that has the following options:
   1. Print Thanksgiving greeting
   2. Print Halloween greeting
   3. Print Birthday Greeting
   4. Quit

7. Write a program that does the following: Finds the greatest common divisor of 2 integers as input from the user. Program repeats the process until the user wants to quit. (To find the greatest common divisor, divide the smaller number into the larger number, get the remainder. As long as the remainder is not zero, the old divisor becomes the dividend, the old remainder becomes the divisor, and get the next remainder. Keep doing this until you get a zero remainder, when that happens the last divisor (causing the 0 remainder) is the greatest common divisor.

8. Write a program to count the number of words, letters, digits, and punctuation on an input file. Words are terminated by a blank space, a period, or an exclamation point or the end of a line.

9. Print a chart of the decimal ascii codes from 32 to 126 showing the char and its numeric code.

10. Print a list of the positive integers less than 00 that are divisible by either 5 or 6. When the list is complete, print a count of the number of integers that were found.

11. If an organism doubles its population every 12 hours, and you start with 100 organisms, how many hours will it take to have 1 million organisms?

12. Calculate the factorial of a number, what is the biggest integer number without overflow you can get the factorial of in HP Pascal? floating point?

13. Given a file of test scores ending with -1, count the number of a's, b's, c's, d's and f's on the file, and calculate the overall average of all the scores.

CS 23 1 Dimensional Array Practice Problems

Use the following type definitions and variables for the next several questions:

```
type    list100 = array[1..100] of integer;
        data100 = array [1..100] of real;
var
        ids:list100; prices:data100;
```

1. A data file, data.inp, contains no more than 100 records. Each record contains an id number followed by a price (real). Sentinel value is any negative id number on the file. Write a procedure readdata(ids, prices, n) that reads the data file and stores the data in the arrays ids and prices respectively, returns n as the number of actual records read from the file.

2. Now write a procedure called printprices(ids, prices, n, currentprice), that is passed the arrays, n, and a current price. The procedure prints all ids that are less than or equal to current price.

3. Write a function called Count100(price, n) that returns the number of prices that are over 100.00 and returns that number as the value of the function.

4. Write a function called searchid(ids, prices, n, currentid) that searches the list of ids for currentid, if found, the function returns the price associated with the currentid, if not found, the function returns the value -1.0

5. Write a function called averageprices(prices,n) that returns the average price of all the prices.

Now use the following types and variables

```
type        idlist = array [1..50] of integer;
            taxlist = array [1..50] of char;
var
       employees:idlist; category:taxlist
      num:integer;
```

6. A data file, employee.dat, contains data for no more than 50 employees. The first record of the data file contains the number of records on the rest of the file, followed by the data, employee id(integer), and taxcategory(char), one blank space separates the employee id from the tax category. Write a procedure, getdata(employees, category,num) that reads the data from the file into the arrays employees and category, returns num as the number of actual sets of data read.

7. Write a procedure that prints all employees in tax category of 'E'.
PrintAllEs(employees, category, num);

8. Write a function that counts the number of employees in a particular tax category that is sent to the function, call looks like

count := countcategory(category, taxcategory, num);

9. Write a procedure to print all tax rates for all employees. Printrate(employees, category, num). Tax rate is 33% if category is 'E', 28% if category is 'S' and 22 % if category is 'M'. Any other category has a rate of 19%.

Appendix F

Questionnaire Transcripts of Open-Ended Questions

Questionnaire Transcripts of Open-ended Questions

Question 3. Please list any class activities, assignments, lectures etc. that you really enjoyed in this course and that made this course a successful experience for you.

"the programs, the homeworks...really helped me to understand the course and how it works."

"Program assignments have really helped me put what I learned in the lecture to practice."

"I enjoyed all the activities, in college it isn't that often that you get to use what is taught in class, I see it's actual value."

"Good class participation throughout. Instructor gives good lectures. Enjoyed class."

"The practice problems were very helpful ... I enjoyed the programming assignments, they're definitely the best way to understand the material."

"The well-presented lectures and the easy-to-understand programs we did made it interesting and challenging. I thought I'd be lucky to pull off a "C", now I am confident I will get an "A". "

"The practice problems are very useful and the programs assigned are appropriate."

"Programming is enjoyable, lectures are not."

"Enjoyed all aspects of the class, lectures were well prepared and valuable, practice problems were very useful, good programming assignments."

"All lectures were enjoyable, 3 hour classes are too long. The group work is an excellent idea."

"I think that the assignments helped drastically in the area of getting a feel for the language. The study groups helped."

"The practice problems were a challenging way of learning the material, and proved to be effective and satisfying when completed."

"Basically the teaching style was outstanding. The drilling at the end of class, and then the review of the drill the next class made studying almost not necessary."

"I thought the group assignments were very successful, when we got together. The practice assignments were also successful."

"The programs were very good and letting me know whether or not I knew the material. Your use of diagrams to help explain things really helped."

"I think the program assignments are very useful and I think that the extra practice sheets are useful."

"The programming assignments I liked the most. I enjoy sitting at my computer and making things work correctly."

"Writing the programs was the best part of the class."

"The class itself was nicely set up. The homework reflected the class perfectly."

"All of the programs. They were challenging at times. The lectures were very good and made understanding the material much easier."

"The course in general was interesting but I really enjoyed attempting to work on several of the programs."

Control
"All the programs were a challenge and they helped me understand the concepts introduced. The practice problem handouts were very helpful! And the lectures were great also. You are a very good instructor."

"Writing programs made the course most successful for me. It's what forced me to learn the material and apply it."

"Time limits on testing should be reviewed. "

Question 4. Please list any class activities, assignments, lectures etc. that you disliked in this course and that created a disagreeable experience for you.

Control
"Fine tuning the programs was often times difficult but I realize this is an important part in mastering programming."

"I believe it was difficult sometimes to get help in doing the computer programs."

"I basically liked everything but the tests. Some of the programs were very long."
"Exams!"

"More time on tests."

"None."

"Lectures were boring."

"I have never taken a Pascal class before and I really knew nothing about it. The first few programming assignments were very difficult for me."

Treatment
"None really, the programs take a lot of time but that is how you learn the material."

"Nothing really comes to mind."

"The HP lab is no place to enjoy yourself."

"Nothing disagreeable."

Question 5. If you were assigned to a study group, on the average, how much time per week did you study with the group? _________ hours. Did you find study groups to be an effective learning strategy? Why or why not?

Treatment
"Yes, most of the time they helped me when working on program assignments but due to time constraints at work and home I couldn't make the most of them."

"Yes, study groups would be an effective learning strategy because it allows us to brainstorm to come up with the answers to questions that might cause trouble for one or more of us."

"Yes, because I pick up little things that I miss with the help of my group members. That way, don't have to get stuck on something for a long time."

"Although we only officially met about one hour per week, it was helpful to know that there were people to call if I ran into problems. Without the groups it could have been very frustrating."

"YES, team work is essential for us in the FUTURE."

"Yes if somebody is having trouble with a program, they can ask somebody in the group."

"It can go both ways. If you are stuck on something, meeting with the group can be a big help. But if you are up to speed on your programs then the group meeting wasn't as helpful. The times I did spend with the group was very helpful and I suggest you continue doing it."

"My experience in a study group helped me more to understand Pascal programming and helped me to see what the problems were asking. The study groups helped to improve my style in programming."

"Yes and No. Yes, because they helped me learn and understand the material, helped me out on programming assignments when I got stuck. No because of difficulty meeting with others in the group."

"Yes. This kind of class can be frustrating at times and it's helpful to work together."

"I think that study groups are very effective in helping to understand the course material. "

"I believe the groups helped to analyze the programs."

"Yes, I studied with others in my group because it helped me remember better once I explained to others what I am doing in a program or a certain procedure. Also seeing how others do their programs helps to expand the ways of doing the programming."

"Yes, not everyone has a complete idea of what is going on, it helps to study with others and to learn together."

"Groups work!! Teaching each other and learning from one another is better than staring at code all night."

"Yes, it really was helpful when getting together to study for tests. We worked out problems together and used different methods to solve them."

"The study groups were helpful-they gave you someone to talk to and help and you realize that the same problems you had are shared by others."

"Yes, without my group I don't think my understanding of Pascal would be hardly as clear."

"Yes. Having someone to help out when your program crashes is a big help."

"Study groups were great because I could share their knowledge with mine to get ideas on how to solve the computer problems."

Control

"I was not in a study group but that's a good idea."

"I wasn't assigned to a study group, but if I were, my experience in the class may have been a little better."

References

Arnson, E. (1978). The Jigsaw Classroom. Beverly Hills, CA: Sage Publications.

Beilin, J. and Rabow, J. (1981). Status value, group learning, and minority achievement in college. Small Group Behavior, 12(4), 495-508.

Block, E. (1990). Education and human resources at the National Science Foundation. Science, 249, 839-840.

Borg, W. E. & Gall, M. D. (1989). Educational Research. 5th Ed. , White Plains, NY: Longman Inc.

Brothen, T. (1986). Using learning groups in introductory psychology. ERIC Document ED268913.

Borresen, R. (1992). Success in introductory statistics with small groups. College Teaching, 30(1) 26-28.

Conrad J. & Conrad P. (1993). Small groups and research projects in science. College Teaching, 41(2), 43-46.

Croteau, D. and Hoynes, W. (1991). A transitional teaching experience: learning groups and the first-time teacher. Teaching Sociology, 19, January, 1991, 28-33.

DeVries, D. & Slavin, R. (1978). Teams-Games-Tournaments: A research review. Journal of Research and Development in Education, 12, 23-38.

Frierson, H. T. (1987). Academic performance in predominantly black nursing classes: effects associated with intervention designed for standardized test preparation. Journal of Research and Development in Education, 20(3), 37-40.

Fitzgerald, S, & Caulfield, J. (1992). Cooperative group learning: A viable strategy for computer-based laboratories? The Journal of Computing in Small Colleges, 7(5), 111-121.

Fullilove, R., & Treisman, P.U. (1990). Mathematics achievement among African American undergraduates at the University of California, Berkeley: An evaluation of the Mathematics Workshop Program. Journal of Negro Education, , 59, 463-478.

Glaser, R. and Bassok, M. (1989). Learning Theory and the Study of Instruction. Annual Review of Psychology, 40, 631-666.

Gronlund, N. E. (1981). Measurement and Evaluation in Teaching. New York: Macmillan Publishing Co.

Henderson, C. and Ottinger, C. (1985). College degrees...still a ladder to success? Journal of College Placement, Spring, 1985, 35-41.

Hill, W. (1969). Learning Thru Discussion. Beverly Hills, CA: Sage Publications.

Howell, K. (1993). The experience of women in undergraduate computer science: What does the research say? SIGCSE Bulletin, 25(2), 1-8.

Hrabowski, F. & Pearson, W. (1993). Recruiting and retaining talented African-American males in college science and engineering. Journal of College Science Teaching, 22(4), 234-238.

Hufford, T. L. (1991). Increasing academic performance in an introductory biology course. BioScience, 41(2), 107-108.

Humphreys, B., Johnson, R., & Johnson, D. W. (1982). Effects of cooperative, competitive, and individualistic learning on students' achievement in science class. Journal of Research in Science Teaching, 19, 351-356.

Johnson, D. W., Johnson, R., Johnson, J. & Anderson, D. (1976). The effects of cooperative versus individualized instruction on student pro-social behavior, attitudes toward learning, and achievement. Journal of Educational Psychology, 68, 446-452.

Johnson, D. W., Johnson, R. T., Holubec, E. J., & Roy, P. (1984). Circles of Learning Cooperation in the Classroom. Alexandria VA: Association for Supervision and Curriculum Development.

Johnson, D. W., Johnson, R. T., & Scott, L. (1978). The effects of cooperative and individualized instruction on student attitudes and achievement. Journal of Social Psychology, 104, 207-216.

Johnson, D. W., Johnson, R. T. & Stanne, M. B. (1985). Effects of cooperative, competitive, and individualistic goal structures on computer-assisted instruction. Journal of Educational Psychology, 77, 668-677.

Johnson, D. W., Maruyama, G., Johnson, R., Nelson, D., & Skon, L. (1981). Effects of Cooperative, Competitive, and Individualistic Goal Structures on Achievement: A Meta-Analysis. Psychological Bulletin, 89(1), 47-62.

Jones, R. (1994). First-year science students: Their only year? Journal of College Science Teaching, 23(6), 356-362.

Kaspar, P. (1994). Tomorrow's Jobs. Future Outlook 1994-1995, 35-36.

King, A. (1989). Verbal interaction and problem-solving within computer-assisted cooperative learning groups. Journal of Educational Computing Research, 5(1), 1-15.

Knight, G. & Bohlmeyer, E. (1990). Cooperative learning and achievement: Methods for assessing causal mechanisms. In Cooperative Learning Theory and Research, edited by S. Sharan, New York, NY: Praeger.

Latting, J. & Raffoul, P. (1991). Designing student work groups for increased learning: An empirical study. Journal of Social Work Education, 27(1), 48-59.

Lawrenz, F. and Munch, T. (1984). The effect of grouping of laboratory students on selected educational outcomes. Journal of Research in Science Teaching. 20(7), 699-708.

Lazarowitz, R., Hertz-Lazarowitz, R., & Jenkins, J. (1985). The effects of modified jigsaw on achievement, classroom social climate and self-esteem in high school science classes. In Learning to Cooperate, Cooperating to Learn, edited by R. Slavin et. al., pp. 231-253. New York: Plenum.

Lazarowitz, R. & Karsenty, R. (1986). Cooperative learning and students' academic achievement, process skills, learning environment, and self-esteem in tenth-grade biology classrooms. In Cooperative Learning Theory and Research, edited by S. Sharan, New York, NY: Praeger.

Lipson, A. & Tobias, S. (1991). Why do some of our best college students leave science? Journal of College Science Teaching, 21(2), 92-95.

Lonning, R. A. (1993). Effect of cooperative learning strategies on student verbal interactions and achievement during conceptual change instruction in 10th grade general science. Journal of Research in Science Teaching, 30(9), 1087-1101.

Massey, W. (1989). Science education in the United States: What the scientific community can do. Science, 245, September 1989, 915-921.

Okebukola, P. (1986). Cooperative Learning and Students' Attitudes to Laboratory Work. School Science and Mathematics, 86(7), 582-590.

Posner, H. & Markstein, J. (1994). Cooperative learning in introductory cell and molecular biology. Journal of College Science Teaching, 23(4), 231-233.

Radebaugh, M.R, & Kazemek, F. E. (1988). Cooperative learning in college reading and study skills classes. Journal of Reading, 32(5), 414-418.

Rau, W. & Heyl, B. S. (1990). Humanizing the college classroom: Collaborative learning and social organization among students. Teaching Sociology, 18, 141-155.

Seymour, E. (1992). 'The problem iceberg' in science, mathematics, and engineering education: student explanations for high attrition rates. Journal of College Science Teaching, 21(4), 230-238.

Sharan, S. (1980). Cooperative learning in small groups: recent methods and effects on achievement, attitudes, and ethnic relations. Review of Educational Research, 1980, 50, 241-271.

Sharan, S., & Hertz-Lazarowitz, R., & Ackerman, Z. (1980). Academic achievement of elementary school children in small group versus whole-class instruction. Journal of Experimental Education, 48, 125-129.

Sharan, S., Kussell, P., Hertz-Lazarowitz, R., Bejarano, Y., Raviv, S., & Sharan, Y. (1984). Cooperative Learning in the Classroom: Research in Desegregated Schools. Hillsdale, NJ: Lawrence Erlbaum and Associates.

Skon, L., Johnson, D. W., & Johnson, R.T. (1981). Cooperative peer interaction versus individual competition and individualistic efforts: effects on the acquisition of cognitive reasoning strategies. Journal of Educational Psychology, , 73, 83-92.

Slavin, R. E. (1980). Cooperative Learning. Review of Educational Research, 50, 315-342.

Slavin, R. E. (1983). When does cooperative learning increase student achievement? Psychological Bulletin,94(3), 429-455.

Slavin, R. E. (1990). Research on cooperative learning: Consensus and controversy. Educational Leadership, 47(4), 52-54.

Sproull, L., Aubrow, D. & Kiesler, S. (1986). Cultural socialization to computing in college. Computers in Human Behavior, 2(4), 257-275.

Statistical Abstract of the United States. (1994). 114th Ed., United States Department of Commerce.

Tobias, S. (1990). They're Not Dumb, They're Different: Stalking the Second Tier. Tucson, AZ: Research Corporation.

Turkle, S. (1984). The Second Self: Computers and the Human Spirit. New York: Simon and Schuster.

Ware, N., Steckler, N. and Leserman, J. (1985). Undergraduate women: Who chooses a science major? Journal of Higher Education, 56(1), 73-84.

Warring, D. & Maruyama, G. (1986). Expectations, attributions, and achievement of college students under cooperative, group competitive, and individualistic goal structures. Paper presented at the Annual Meeting of the American Educational Research Association San Francisco, CA, April 16-20, (ERIC Document Reproduction Service No. ED 277415).

Webb, N. M. (1982a). Group composition, group interaction, and achievement in cooperative small groups. Journal of Educational Psychology, 74, 475-484.

Webb, N.M. (1982b). Peer Interaction and learning in cooperative small groups. Journal of Educational Psychology, 74, 642-655.

Webb, N. M. (1982c). Student interaction and learning in small groups. Review of Educational Research, 52(3), 421-455.

Webb, N. M. (1983). Predicting learning from student interaction: Defining the interaction variables. Educational Psychologist, 19, 33-41.

Webb, N. M. (1984). Microcomputer learning in small groups: Cognitive requirements and group process. Journal of Educational Psychology, 76, 1076-1088.

Webb, N., Ender, P., & Lewis, S. (1986). Problem-solving strategies and group processes in small groups learning computer programming. American Educational Research Journal, 23, 243-261.

Yager, S., Johnson, D. W., & Johnson, R. T. (1985). Oral discussion, group-to-individual transfer, and achievement in cooperative learning groups. Journal of Educational Psychology, 77, 60-66.

VITA

Laura Jean Baker was born in Cleveland, Ohio on August 19, 1958, the second of three children of John and Nancy Baker. After graduating from Charles F. Brush High School in Lyndhurst, Ohio in 1976, Laura attended Indiana University in Bloomington, Indiana for two years from 1976 to 1978. In January, 1979 Laura transferred to Texas A&M University in College Station, Texas where she completed a B.S. in Parks and Recreation in August, 1981. Upon graduation, Laura took a teaching/coaching position at Iola High School in Iola, Texas for two years from 1981-1983. In August of 1983, Laura moved to San Marcos, Texas and enrolled in graduate school at Southwest Texas State University. In August, 1985, Laura became an Assistant Instructor in Computer Science at Southwest Texas State University. In August, 1987, Laura received a Master of Education degree in Secondary Education and Computer Science at Southwest Texas State University. In August, 1987, Laura was hired as an instructor in the Computer Science Department at Southwest Texas State University where she taught for two years. In January, 1989, Laura enrolled in the Ph.D. program in Science Education at The University of Texas at Austin. In August, 1989, Laura was hired as an instructor in the Computer Sciences Department at St. Edward's University in Austin, Texas, where she continues to teach. In August, 1995, Laura will be placed in a tenure-track position in the Computer Sciences Department at St. Edward's University. Laura has been a doctoral candidate at The University of Texas at Austin since 1993.

Permanent Address: 629 West Hopkins St.
San Marcos, Texas 78666

This dissertation was typed by the author.