CONTRIBUTING FACTORS TO SUCCESS IN COMPUTER SCIENCE:

A STUDY OF GENDER DIFFERENCES

by

Brenda Cantwell Wilson

Bachelor of Science in Mathematics, Harding University

Master of Arts in Mathematics/Computer Science Education, University of Texas, Austin

A Dissertation
Submitted in Partial Fulfillment of the Requirements for the
Doctor of Philosophy Degree

Department of Curriculum and Instruction
in the Graduate School
Southern Illinois University
at Carbondale
June, 2000

UMI Number: 9982125

Bell & Howell Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

Dissertation Approval
The Graduate School
Southern Illinois University

June 19, 2000

I hereby recommend that the dissertation prepared under my supervision by

Brenda Cantwell Wilson

Entitled

CONTRIBUTING FACTORS TO SUCCESS IN COMPUTER SCIENCE:
A STUDY OF GENDER DIFFERENCES

be accepted in partial fulfillment of the requirements for the

DOCTOR OF PHILOSOPHY degree.

In Charge of Dissertation

Keith Hillkirk

Head of Department

Recommendation concurred in

1.
2.
3.
4.
5.

Committee for the Final Examination

AN ABSTRACT OF THE DISSERTATION OF

BRENDA CANTWELL WILSON, for the Doctor of Philosophy degree in CURRICULUM AND INSTRUCTION, presented on June 19, 2000, at Southern Illinois University at Carbondale.

TITLE: Contributing Factors to Success in Computer Science: A Study of Gender Differences

MAJOR PROFESSOR: Dr. Sharon A. Shrock

Because of the shortage of women in computer science, this study was conducted to determine factors that promote success in an introductory college computer science course and to determine what, if any, differences appear between genders on those factors. The model included twelve possible predictive factors for success in the computer science course. The factors were math background, attribution for success/failure (luck, effort, difficulty of task, and ability), self-efficacy, encouragement, comfort level in the course, work style preference, previous programming experience, previous non-programming computer experience, and gender. Subjects included 105 (19 females, 86 males) students enrolled in an introductory computer science course at a Midwestern university. The study revealed three predictive factors (using an alpha level of .05) in the following order of importance: comfort level, math, and attribution to luck for success/failure. No significant gender differences were found in these three factors. Comfort level and math background were found to have a positive influence on success, whereas attribution to luck had a negative influence. The study also revealed by studying the different types of previous computer experiences (including formal programming class, self-initiated programming, internet use, game playing, and productivity software use) that both a formal class in programming and game playing were predictive of success. Formal training had a positive influence and games a negative influence on class grade. The study revealed a significant gender difference in game playing with males reporting more experience playing games on the computer than females reported.

ACKNOWLEDGMENTS

I would like to express my sincere appreciation to Dr. Sharon Shrock for her guidance and encouragement in this research endeavor. Also, gratitude is expressed to Dr. Douglas Bedient, Dr. Bill Coscarelli, Dr. John Deichmann, and Dr. Jerry Hostetler for serving on my doctoral committee and for many helpful suggestions for my dissertation. I am indebted to Dr. Robert McGlinn for his advice and for his willingness to allow me to use his classes for my research.

It is with heartfelt gratitude that I express my love to my father, Wendell Needham, who proofread my manuscript and to my mother, Helen Needham, who constantly worried about my health during the years of work on this endeavor. Also, I am grateful to my son, Joshua Cantwell, and my daughter, Lindsay Cantwell Warren, for their support and encouragement to finish this task. It is because of them that I have gained strength to carry on with my life and to begin life again.

This work could not have been completed without the faithful and continuing support of my husband, Kim Canter Wilson. He always made me know that this was not a requirement but a choice and always encouraged me in whatever decision that I made. It is with deep-felt love and gratitude that I dedicate this book to him.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

CHAPTER II

REVIEW OF THE LITERATURE

CHAPTER III

METHODOLOGY

CHAPTER IV

RESULTS

CHAPTER V

SUMMARY, DISCUSSION, AND RECOMMENDATIONS

APPENDICES

TABLES

CHAPTER I

INTRODUCTION

Statement of the Problem

The pipeline shrinkage is a term used by many to describe a well-known phenomenon regarding women in computer science. The participation of women in computer science from high school to graduate school diminishes at an alarming rate. Not only does this "brain drain" occur throughout school but also continues in the academic faculty ranks of colleges and universities where the percentages of women computer science instructors from assistant professor through full professor also decrease. This problem is compounded by the fact that even though the numbers of women completing bachelor's degrees in general have increased, the pipeline shrinks at the bachelor's level for women in computer science over the past several years. The result is an appalling gender gap in this growing technological field. The purpose of this study was to examine contributing factors to success in an introductory computer science course and determine which of these factors affect this gender gap.

The Extent of the Problem

Pipeline Shrinkage Throughout Educational Levels

Although as many females as males take computer science classes in high school (The Condition of Education, 1998), the percentage of women receiving bachelor's degrees in computer science (including computer information systems) for the 1994-95 academic year was only 28% (Digest of Education Statistics, 1997). At the graduate level, for the same academic year, the percentages of degrees in computer science dropped to 26% at the master's level and 18% at the Ph.D. level. In addition, for women in academia, the pipeline shrinks through the ranks of faculty. During the 1993-94 academic year, only 15.6 % assistant professors, 9.4% associate professors, and 5.7% full professors were women in computer science departments of Ph.D.-granting institutions according to the Computing Research Association's Taulbee surveys (as cited in Camp, 1997).

Pipeline Shrinkage at Bachelor's Level

Not only does the pipeline shrink for women in computer science from high school to graduate school and beyond, but it also shrinks at the bachelor's level and at the graduate levels over the last decade. Within many fields of study and degree levels, women earned more than 50 percent of all degrees awarded in 1994. For example, women earned 51 percent of bachelor's and 53 percent of master's degrees awarded in biological sciences in 1994. Women also continued to earn more than half of all bachelor's degrees and master's degrees in education and health professions from 1984 to 1994. Women continued to be underrepresented within several fields of study, however. From 1970 to 1994, women continued to earn less than 50 percent of all the degrees in agriculture, business management and administrative services, engineering, computer science, mathematics, social sciences, and physical sciences (The Condition of Education, 1997). The projection from 1995-96 to 2008 for men earning bachelor's degrees in general is estimated to be a 1% increase whereas the estimate for women earning bachelor's degrees is estimated to be a 15% increase (Projections of Education

Statistics to 2008, 1998). In the 1983-84 academic year 37% of the computer science degrees were awarded to women, but in 1993-94 only 28% of these degrees went to women. In fact, computer science is the only science and engineering discipline where the percentage of bachelor's degrees for women decreased over the period 1980-81 to 1993-94. Over that same period of time, the percentage change reported for women receiving bachelor's degrees were: Bio/Life (biology and life sciences) increased 16.3%, Eng (English) increased 44.7%, Math increased 10.0% and Phy/Sc (physical sciences) increased 36.6%, but computer science decreased 12.6% (Camp, 1997).

Because the percentage of women earning bachelor's degrees in computer science is decreasing, it is not surprising that this shrinkage is occurring at the graduate levels also. Between 1970 and 1986, the difference in the proportion of men and women earning master's degrees in computer sciences and engineering narrowed each year. Since 1986, it has remained stable with males being five times more likely than females to earn a master's degree in computer sciences and engineering (The Condition of Education, 1997).

This pipeline shrinkage is particularly of interest to many in business and industry as well as in education because there is a critical labor shortage in the computer science and computer information systems field (Camp, 1997). The seriousness of this problem is stated by Fisher, Margolis, and Miller (1997), "Since computers and information technology play an increasingly pervasive role in education and careers, this underrepresentation is critical, not only for the women whose potential may go unrealized, but also for a society increasingly dependent on the technology" (p. 106).

The demand for degreed computer specialists will double in the next 10 years, which even surpasses the demands in the medical field ("Demand for," 1999).

The Nature of the Problem

The fact that women make up more than half of the population and are still significantly underrepresented in this field makes studying the factors contributing to this problem a priority. Many in the educational and business arenas have voiced concerns, hypothesized reasons, and proposed solutions to the problem. Actually the problem is two-fold. There is the problem of recruitment (actually getting high school female graduates to enroll in computer science classes) and the problem of retention (keeping the females in the computer science programs once they enroll in a class).

Recruitment

Underlying the concern about the low numbers of females who enroll in college computer science courses is the question of whether it is caused by a lack of ability or because of lack of support and encouragement to pursue high-technology careers. There is increasing evidence indicating that gender differences in computer science participation are not due to ability differences. Fennema and Sherman (1977) studied differences in math and spatial achievement scores of over twelve hundred ninth-graders and found sex differences in math achievement and spatial visualization scores only in those schools where there were also significant sex differences in students' self-perception of their ability to learn mathematics and the value that was placed on that learning. Data from the ACCCEL (Accessing the Cognitive Consequences of Computer Environments for Learning) Project showed similar findings as were found in previous gender-related mathematics research. When junior high and senior high students enrolled in computer classes were given the Raven Progressive Matrices – an ability test designed to be free of verbal and cultural bias, no gender difference in performance was evident (Mandinach & Fisher, 1985; Linn, 1985). Also, the Minnesota Educational Computing Consortium showed little evidence of sex differences in overall computer literacy and programming ability in girls and boys (Anderson, Klassen, Krohn, & Smith-Cunnien, 1982). Girls and boys were roughly equivalent in overall computer literacy as well as in programming ability.

In 1984 the number of females and males taking AP (Advanced Placement) exams were equal. By 1996, the females outnumbered the males taking AP exams 144 to 117. (These numbers represent the number of students per 1000 12th graders.) However, females outnumbered males in only three areas: social studies, English, and foreign language. The areas of calculus, computer science, and science were dominated by males. The success rate for males on the calculus AP exam was 68% and for females 60%. The males taking the computer science AP exam outnumbered the females 5 to 1 (The Condition of Education, 1998). Klawe and Levenson (1995) noted, "50% of the qualified males choose a scientific career, compared to only 16% of the qualified females" (p. 30).

The fact that so many qualified females do not choose to enter the computer science degree in college has been attributed to recruitment factors such as lack of role models and encouragement, gender stereotyping, and lack of self-esteem among females. Seeking to study and address these issues, the PipeLINK program, funded in part by the National Science Foundation, was aimed at girls and women from high school through the Ph.D. level to provide activities to encourage participation at each level, provide mentors and role models as well as to introduce to females a wide variety of computer science topics (Rodger & Walker, 1996a, 1996b). Greening (1999) examined gender stereotyping in computer science and concluded, "the biggest source of pipeline 'leakage' occurs prior to university admission" (p. 206). Anderson, Welch, and Harris explained the low level of females in computer science courses on four social factors: parental encouragement directed toward sons rather than daughters, boy and girl peer groups widening the gap, stereotyped game software (mainly directed at boys), and lack of female role models both in the classroom and in the media (as cited in Kersteen, Linn, Clancey, & Hardyck, 1988).

Retention

Attempts to examine possible contributing factors of the high attrition of female students in computer science programs in college have concentrated in several areas: previous computer experience, hostile environment and culture, and attribution theory. A related area of research includes studies of self-efficacy. In an attempt to investigate ways of retaining females in the computer science field, all of these areas of research will be discussed.

Previous computer experience. A growing body of research suggests that there are significant differences between males and females in their experience with and attitudes toward computers. Morahan-Martin, Olinsky, and Schumacher (1992) included over 600 entering freshmen in their study and found males had more experience and skills than females in specific computer usage, particularly programming and games. Gender differences were also documented in attitudes towards the computers as well. Scragg and Smith (1998) studied six possible barriers to women in computer science classes and also found that women had substantially less pre-college computing experience than did men. They concluded, "the largest barriers to retaining women in computer science may be circumstances that occur long before they enter our programs" (p. 85). Taylor and Mounfield (1991) found that having a high school programming course using structured methodology was not only statistically significant for success in a college computer science course but was also one of the best indicators of success. Applications experience only (without programming experience) did not prove to be an indicator of college computer science success. In a later study these same authors found that any type of prior computing experience for females was significant in success in the college computer science class, but only specific types of computing experience were significant for males (1994). In a study of women in introductory computer science in New Zealand, Brown, Andreae, Biddle, and Tempero (1997) found that "students who find the course difficult are intimidated by seeing other students who have prior programming experience completing the assignments very quickly" (p. 112). Other studies' findings agree that women's lack of prior computer experiences put them at a disadvantage in introductory computer science courses (Sackrowitz & Parelius, 1996; Liu & Blanc, 1996).

Hostile environment and culture. Women often find the environment and culture in computer science activities to be hostile. One reason supported by Moses (1993) is that women prefer activities where social interaction is encouraged, and collaboration is often discouraged in academic computer science. In fact, most assessment is done on a competitive basis, which is a methodology that females prefer to avoid (Howell, 1993; Moses, 1993). Frenkel (1990) stated that girls and women are "ill at ease in a field that seems to encourage "highly focused, almost obsessive behavior" (p. 38). Also women have few role models because of the small number of female computer science professors. DeClue (1997) observed that the act of working alone with a computer for long hours in obsessive "hacker" behavior is a part of many computer science programs but is a behavior uncomfortable to females (p. 4). Because of all of these factors, the female students in the program may feel isolated.

Attribution theory. Attribution theory involves explanations that people give for their successes and failures. The explanations can be of a stable nature (attributing outcome to ability or difficulty of task) or an unstable nature (attributing outcome to luck or effort). The theory suggests that when people attribute their successes to unstable causes (luck or effort) and their failures to stable causes (ability or task difficulty), the probability of persistence is low. Deboer (1984) used the framework of attribution theory to study persistence in college science courses. He found that successful science students' intention to continue in science was directly related to their attribution to ability and inversely related to task ease. Several studies have suggested that females tend to attribute their successes in computer science to luck and their failures to lack of ability (Bernstein, 1991; Howell, 1993; Moses, 1993; Pearl, Pollack, Riskin, Thomas, Wolf, & Wu, 1990). If these tendencies were substantiated, they would obviously be a barrier to an increase in motivation and self-confidence for women in computer science and certainly could, at least in part, explain the high attrition rates reported in computer science programs. Bernstein even found that males who were uncomfortable using computers attributed this feeling to "inadequate experience or poor teaching," while females tended to criticize themselves for feeling uncomfortable with the computer (p. 60).

Self-Efficacy. According to Bandura (1986) the way people behave is determined by their perceptions of how skilled, competent, and efficacious they are. Self-efficacy is the mechanism by which people navigate paths to achieve goals. Bandura (1991) states:

> People's beliefs in their efficacy influence the choices they make, their aspirations, how much effort they mobilize in a given endeavor, how long they persevere in the face of difficulties and setbacks, whether their thought patterns are self-hindering or self-aiding, the amount of stress they experience in coping with taxing environmental demands, and their vulnerability to depression. (p. 257)

Bandura (1977) believes that there are four important sources of information affecting perceptions of self-efficacy. The first source is performance accomplishment. People's perceived self-efficacy for an activity tends to increase if their experiences provide positive information about related competencies. "Males tend to seek out interaction with computers (through curricular and extracurricular classes and informally in video arcades), thereby creating opportunity for successful performance on the machine" (as cited in Miura, 1987, p. 305). Because computer science is a math-related subject, perceptions of self-efficacy may be affected by performance accomplishment in mathematics. The second source of information affecting perceived self-efficacy is seeing others succeed or fail. Males have numerous successful role models in math-related careers, whereas females have relatively few. The third source of information included by Bandura is verbal persuasion. Many studies have been conducted showing that girls in the United States are not actively encouraged to continue in mathematics classes and are often discouraged from pursuing math-related careers including computer science (Brody & Fox, 1980; Dachey, 1983; Hess & Miura, 1985). Finally, the fourth source of information that can affect perceived self-efficacy is emotional arousal. Several studies have shown math anxiety and lack of confidence in one's ability to do mathematics begins to emerge in girls in the junior high years and continues in the college years (as cited in Miura, 1987). Because college computer science courses have mathematical prerequisites, math anxiety may influence perceptions of self-efficacy for computer activities at the college level as well.

Problem Significance and Study Questions

With the growing need of computer professionals and the continuing decrease in participation by women in computer science, questions arise regarding the reasons why this discipline is so unattractive to females. If the ability to succeed in computer related programs is not inherent to the male gender, what can be done to attract and retain qualified females in this field? Not only is there a need to attract women to this field because of the demands of business and industry projected for the next decade but also because it raises ethical questions to have such a male-dominated discipline. If research studies can identify causes of the pipeline shrinkage, the problem of low female participation can be addressed and solved. Some of the studies in this area have successfully identified causes for the problem, but more work needs to be done to determine what efforts by educators, parents, and the business community can be directed toward a solution. Few studies which include separating the types of previous computing experiences (programming and non-programming) combined with other possible contributing factors of success in computer science have been conducted to study retention of women in computer science. This study sought to combine the study of several proposed factors in retaining women in the computer science field after they have chosen to major in the field and discover answers to the following questions:

1. What relationship exists between the factors of previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender of introductory computer programming students and their midterm course grade?

2. What factors are predictive of midterm course grade in an introductory computer science course: previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender?

3. Are certain types of previous computing experiences predictive of success in a college computer science course?

4. Of the predictive factors of success in computer science, are gender differences evident? If so, which factors demonstrate gender differences?

Definitions

As in any research endeavor, it is useful to explain the use of some terms to alleviate misunderstandings as the reader analyzes the study. The following definitions were used in this study:

Previous Computing Experience

This includes the use of the computer prior to college. The following two types of experiences can further categorize these previous computing experiences: (1) previous computer programming experiences, and (2) previous non-programming computer experiences. These two areas of previous computing experiences are also subdivided into more specific types of experiences, which were measured within each area:

Previous computer programming experiences. The specific types of experiences included in this subcategory are (1) a formal programming course in high school, and (2) self-initiated programming in which the student learned to program outside of a formal class in programming.

Previous non-programming computer experiences. The specific types of experiences included in this subcategory are (1) internet (World Wide Web) searches, e-mail, chat rooms, discussion groups; (2) games (online or individual); (3) use of productivity software such as word processing, spreadsheets, presentation software, and databases.

Attribution

Attribution is "the explanation that people give for their success or failure in achievement settings" (Deboer, p. 325). The attributions for success or failure are: (1) attribution to ability, (2) attribution to difficulty of task, (3) attribution to luck, and (4) attribution to effort.

Self-efficacy

Self-efficacy is the feeling about one's ability to perform various C++ programming tasks as measured by the Computer Programming Self-Efficacy Scale (Ramalingam & Wiedenbeck, 1998). (See Appendix A)

Comfort Level

Comfort level is a measure of how much anxiety one has in the computer science program's environment as shown by these indices: (1) likelihood of asking questions/ answering questions in class, (2) likelihood of asking questions in lab, (3) likelihood of asking questions during office hours, (4) perceived anxiety while working with the computer on programming assignments, (5) perceived difficulty of the class, (6) perceived difficulty of writing computer programs in general, and (7) perceived understanding of concepts in class compared to classmates.

Encouragement from Others

Encouragement from others is defined as the words of confidence, praise, or discussions about the computer science field and its career opportunities from sources outside of self.

Work Style Preference

Work style is the preference for learning environments categorized by competitive and individual work or cooperative and group work.

Math Background

Math background includes the number of semesters of math courses taken in high school.

Midterm Grade

The midterm grade is the midterm percent grade assigned to each student in the middle of the semester. This score was the average of computer programming assignments and an exam which consisted of both multiple choice questions about programming code and open-ended questions requiring programming code in C++ to be written.

Assumptions

In every research endeavor there are many assumptions that must be made. This project assumed the following:

1. The subjects will voluntarily participate and will give honest answers to the questionnaire.

2. Previous research can be used as a basis for the design of this project.

3. Many students will drop out before the semester ends. (The attrition rates are extremely high in introductory computer science courses.)

4. The midterm grade is a good indicator of success in the introductory computer science course. (Several seasoned computer science professors, including the one teaching the classes being studied, were questioned and all agreed that midterm grade is sufficiently predictive of how the student will do in the course.)

Limitations

Limitations that affect this project are as follows:

1. This study is limited to the computer science students in CS 202 Introduction to Computer Science at the particular university in the study.

2. As usual when studying the introductory college computer science program, the number of females will be small.

Delimitations

It may be useful to note the following areas not addressed in this study:

1. This study does not propose to make generalizations about any other educational setting other than a college computer science program.

CHAPTER II

REVIEW OF THE LITERATURE

Introduction

With the advance of computer technology in the past 50 years, the need for people with expertise in programming and controlling computer systems has increased rapidly. Despite the growing opportunities for computer-related careers, women are extremely underrepresented in computer science programs. The state of women in the field of computing has received increasing attention over the past few years, and several studies investigating the causes of this "pipeline shrinkage" (Kerner & Vargas, 1994) have been conducted. This chapter synthesizes several studies aimed at discovering causes and possible answers to this gender gap and presents them grouped by the identified problem causes discussed in chapter one. Although this study will primarily address the retention aspect of the pipeline shrinkage problem, it will include a measure (previous encouragement to study computer science) of recruitment and, therefore, it is noteworthy to include a few studies that incorporate the study of the recruitment problem as well. After discussing recruitment literature, research endeavors focusing on retention of females in computer science will be discussed. Particularly research studies in the areas of previous computing experience, hostile environment and culture, attribution theory, and self-efficacy will be discussed. Finally, a summary of this pertinent literature will be presented to give a framework for the current investigation.

Recruitment

One of the reasons for the pipeline shrinkage of women in computer science is a recruitment problem. As discussed in chapter one, many have posited that the reason for the low numbers of females enrolling in college computer science courses is a lack of support and encouragement to pursue high-technological careers. Busch (1996), while studying whether gender, group composition, or self-efficacy in computing has any impact on cooperation, found that female students had received significantly less previous encouragement to work with computers than male students had received. In this study subjects rated the level of previous encouragement on a 5-point scale the extent to which their decision to use computers had been influenced by parents, schoolteachers, and friends. He constructed a composite variable by adding the responses to these three items. The results showed a significant difference in favor of males for receiving previous computer encouragement. Liu and Blanc's study (1996) mainly addressed the problem of retention of females in the computer field; however, they did report that the majority of the women responding to their survey had not been recruited to study computer science nor had they had advanced placement courses in mathematics or computer science. A striking difference in one study was found in encouragement by parents, teachers and peers for boys over girls to use computers because the boys were both provided with opportunities to gain experience and reinforcement for demonstrating their interest and achievement (Carmichael, Burnett, Higginson, Moore, & Pollard, 1985). For students at both the primary and secondary level, student interest in learning about computers has been shown to be related to perceived parental interest in their learning about them, and to a lesser extent, to perceived teacher interest (Firkin, 1984; Clarke, 1986).

Related to the lack of support and encouragement to study computer science is the lack of role models in computer science available to girls in high school and college. Bunderson and Christensen (1995) in their study of gender differences in previous computing experiences found that the low number of females enrolled in computer science courses was related to the lack of role modeling in the traditional culture of Brigham Young University. The prevalence of male role models in the computing world has been assumed to lead to the development of sex typing of computer related activities (Kiesler, Sproull, & Eccles, 1983; Reinecke, 1983; Sanders & Stone, 1986; Firkin, Davidson, & Johnson, 1985). Clarke and Chambers (1989) reported that more fathers and brothers were the main users of computers at home, which clearly indicated that mothers and sisters were unlikely to be providing female role models for computer use. Scragg and Smith (1998) noted that women in their study thought computing was male-dominated with few female role models.

Retention

Research studies examining the contributing factors to the loss of female students who have begun a computer science curriculum have concentrated in the following areas: previous computing experience, hostile environment and culture, attributions to success and failure, and self-efficacy issues. While more research exists on the importance of previous computer experience in explaining the shortages of women in computer science and in relation to success in computer science, some studies in all the areas mentioned above will be discussed.

Previous Computer Experiences

Several studies investigating the effect of previous computer experiences on success in computer science courses were conducted in recent years. Taylor and Mounfield had two studies in 1991 and 1994. The earlier study found that not only did high school computer science have a very positive effect on college computer science, but that the teaching methodology rather than the content of the course seemed to be the major factor of the high school experience that contributed the most to success in college. The later study found that while only certain previous experiences were related to success for males, virtually all previous computer experiences were beneficial for females.

In the 1991 study, Taylor and Mounfield reported on data from two academic years of study. In both years, students enrolled in an introductory computer science course(the standard course for computer science majors) were the subjects in the investigation. Students were surveyed at the beginning of the semester, and grades were recorded at the end of the semester. The student's final grade in the course had to be a C or higher to be proclaimed "successful" in the course. The independent variables were high school computer science, gender, classification year in school, and student work hours while attending school. The first year included 709 students with 42% of the group having previous high school computer science course experience. In the follow-up study, items were developed to investigate more about the nature of the high school computing experience, separating the applications courses such as word processing or database packages from the programming experiences.

In the first reported year of data gathering, any kind of previous computer science course experience was a statistically significant factor in success in college computer science. Males not only out-numbered the females but also did much better as a group than the females. In the second year, however, the grades were more equally distributed among the males and females, with a slightly higher percentage of the females being successful. A high school programming course using structured methodology was not only statistically significant as a factor but was also one of the best indicators of success in college computer science.

In the 1994 study, subjects included 656 students enrolled in an introductory computer science course (Pascal programming) for non-computer science majors at Louisiana State University. A survey was administered to collect the data. Fifty- five percent (361) males and 45 percent (295) females were included in the study. This statistic alone shows that the group is very different than one including computer science majors where males may outnumber females by a ratio of 4 to1 (Taylor & Mounfield, 1991). The independent variable, previous computing experience, was operationalized by defining these areas: any high school course using computers, high school programming course, high school programming course emphasizing structured programming, and applications only course. Other independent variables such as gender, ownership of a computer, grade expectation, class year, and hours of work were included. The dependent variable was success in the course (defined as completion of the course with a grade of "C" or better). To minimize the differential effect of instruction experienced instructors selected for their teaching ability and positive student interactions were used as well as identical lab assignments and exams.

Computer ownership was the single most significant factor in course success for both sexes at the .01 alpha level. Previous computing experience of any type was significant for the entire group at the .01 alpha level, but was not significant for the males. Only high school programming and owning a computer were significant in predicting success in computer science for the males. A dramatic difference in the female success rate was shown for high school computer science course where 30% more females who had taken a computer science course in high school succeeded compared to those who had not taken such a course.

In 1988, Kersteen, Linn, Clancy, and Hardyck from the University of California, Berkeley, also examined previous experience with computers as a predictor of performance in college computer science courses including the possible interaction of gender with these variables. They concluded that males have more previous experience, and that much of this experience is gained through "hacking" and unguided exploration. Amount of previous computing experience was found to predict course performance for males. Since very little previous experience for females was reported, predictions could not be made for course performance.

Questionnaires were used to collect data from introductory computer science students in the spring and fall of 1985. Data was collected from 176 students and 123 students, respectively, for the two semesters. Females comprised approximately 25% of each of the samples. The questions included facts about previous computer experiences such as programming languages, application packages, ownership of a computer, non-school computer experiences, and whether computers were in the junior high and senior high schools. The dependent variable was the final letter grade in the introductory college computer science class. Interviews were also included in the study to provide depth and perspective to the results obtained from the questionnaire. In particular, the authors were interested in discovering the reason for the large difference in previous computer experience between males and females.

The mean grade for both males and females was a "B" across the two semesters. On virtually every question dealing with previous computer experience outside the classroom, males responded as having more experience than females. It is interesting to note, however, that across both semesters, an approximately equal proportion of males and females reporting having taken computer science courses in high school. This fact points to the suggestion that the experiences outside of the classroom – the self-initiated computer use – accounted for the difference found. The experience scale accounted for 14% and 25% of the variance in final grade for the males across the two semesters. For the females, however, the scale had no power in predicting final course grade. A two-tailed analysis of variance indicated that these gender differences in the self-initiated computer experience scale scores were highly significant for both the spring and fall semesters. The more self-initiated experience the boys had obtained prior to the course, the better they performed in the course.

The study conducted by Morahan-Martin, Olinsky, and Schumacher (1992) showed that gender differences exist in computer experience, skills, and attitudes among incoming freshmen in a business college. Although they found no gender differences in ever having used a computer, males had more experience and skills than females in specific areas of computer usage, particularly programming and games. They also found that females perceived computer skills as more useful for their careers than males, but males were more willing to purchase a computer than females.

During the summers of 1989 and 1990, incoming freshmen at a small business college in Rhode Island were asked to complete a survey, which assessed their computer experience and attitudes. Over the two years 619 freshmen completed the survey. Data was gathered about previous uses of the computer, skill level acquired previous to entering college, depth and amount of computer experience, and attitudes toward computers (using Likert scales). For each of the years frequencies were run and all variables examined to determine how students were answering the various questions on the survey. Percentages were also determined and various descriptive statistics were reported. To assess the possible effect of gender on the dichotomous variables in the survey, cross-tabulations were run for each year and analyzed. In order to assess the possible effect of gender on the rank order variables, the analysis of variance (ANOVA) procedure was run for each year. Then an analysis of variance was run on the combined quantitative data with three variables of interest: gender, year, and an interaction term of Gender x Year. Because there was no significant interaction between gender and year and because the year of the survey had no significant effect on any of the quantitative variables, results were reported on the combined data with gender being the only factor of interest.

There were no gender differences in ever having used the computer in high school or ever having taken a high school course requiring the use of a computer. There were differences, however, in the specific types of computer usage, as well as the skill level acquired and amount of experience in specific computer applications. Males were more likely than females to have experience in writing both BASIC and PASCAL programs. Males were also more likely to have used the computer to write graphics routines or to play games. Females were more likely than males to perceive the computer as an important tool in industry, while men were more likely than women to perceive computers as important only in computer-related fields. Finally, males more than females agreed that owning a computer is important.

In 1998, Scragg and Smith reported a study conducted at SUNY Geneseo to determine why few women complete their computer science major. They concluded "retention of women once they enter the major is important, but it is secondary to getting women into the major initially" (p. 82). They suggested that the most effective solutions must concentrate not on retention but on recruitment.

The researchers used focus groups to identify six hypothetical barriers to women in undergraduate computer science. Then a survey was conducted using a questionnaire that had multiple-choice questions and collected data on these research hypotheses:

1. General social pressures (e.g., attitudes of friends and family) discourage women from pursuing computer science.

2. Women face more crises of self-confidence over their performance in computer science than men do.

3. Women do not have as much chance as men to contribute ideas in classes, and their contributions are under-valued when they are made.

4. Women believe that computer science is too dominated by men.

5. Women believe math is an important part of computer science, yet suffer more than men from math anxiety.

6. Women feel more strongly than men do that they want to raise a family, but that a career in computer science is incompatible with this goal.

The researchers used a significance level of $p < .05$ throughout the analyses of the data.

There were 297 respondents (133 women and 164 men) to the questionnaire at the end of the spring semester of 1995. A follow up survey taken at the end of the spring semester of 1996 was used just to check the initial results to check for inconsistencies but the data was not pooled.

Women in the survey had substantially less pre-college computing experience than did men. They found no evidence of peer, parental, or personal perceptions that computer science is a career inappropriate for women (impact of social pressure). The study showed that women are less comfortable in class debates than men and are less willing to correct a professor. In general, the researchers did find evidence that women in introductory courses feel less self-confident than men. The authors concluded that there was no evidence to support that women's contributions are under-valued and that there was no evidence that women felt discriminated against in opportunity to contribute to class. The research hypothesis that women think computing is more male-dominated than they would like was supported. No significant difference was found that women leave the computer science program because of math anxiety, and the data did not suggest any significant difference in respondent's feelings about computing conflicting with family.

Of the six barriers studied, the results from only two –self-confidence and male dominance – yielded significant differences that might explain the high attrition of women within the field. Probably the most significant finding was that women enter with far less computing experience than do men, and most women in the introductory courses never plan to major in computer science at all. The authors concluded, therefore, that the sources of the barriers may very well be "systemic societal problems or may be caused by the early education process" (p. 85).

In 1995, Bunderson and Christensen from Brigham Young University also suggested that a key factor influencing the high rate of female attrition in the computer science program is a lack of previous experience with computers before entering the college program.

Subjects surveyed and interviewed included 275 students enrolled in one of three computer science courses, an introduction to computing course, a computer programming course, and a database modeling course (representing a wide range of expertise in computing). Only 28 females were enrolled in these courses, which was a complicating factor in comparing means of gender groups. Data about students' attitudes toward instructors, classes, and females in computer science as well as students' interactions with teachers and other students was gathered. Also, 46 former computer science students (20 males and 26 females) were sent questionnaires to collect data on reasons for leaving computer science and students' opinions about helpfulness/encouragement and gender discrimination.

The responses to the surveys and interviews were grouped into five categories:

1. Satisfaction with computer science: Women seemed to leave computer science due to influences from within the computer science major where men were pulled away by outside factors.

2. Gender discrimination: Twenty percent of the females felt that they had been treated differently due to their gender whereas only 5% of the males felt this way.

3. Previous experience with computing: one-half of the students (both male and female) thought that computer science classes were oriented toward students with previous programming experiences.

4. Traditional culture: The low number of females enrolled was attributed to the conservative nature of the religious culture at Brigham Young University.

5. Interaction with others: Females were more likely to study with other females than were males to study with any others. Students in advanced computer classes felt more confident to ask questions of the instructor than those in introductory classes, and women in all classes were more uncomfortable asking questions than were men.

The most striking finding in this study was the unrealistic expectations of previous computer expertise assumed by computer science faculty. Since females have been recognized as a group having less experience in computing than males (Klawe & Leveson, 1995), the high attrition rates of females was not surprising. Also the preference of females to study with others may suggest that opportunities for cooperative projects in computer science assignments to be advantageous to reducing the attrition rates of females.

In 1996, Sackrowitz from Middlesex County College and Parelius from Rutgers University showed that despite the proliferation of computers in our culture and daily lives, females are still entering college computer science classes with weaker programming skills and less previous computer experiences than males.

A questionnaire including questions about general academic background, mathematics background, computer experiences, perceived computer skills, and attitudes toward computers was administered to introductory computer science students at Rutgers and Princeton Universities at the beginning of the courses. Completed questionnaires were obtained from 50% of the 186 Rutgers students and from 35% of the 94 Princeton students. The sample group at Rutgers was 30% female and the one at Princeton was 45% female. This group differs from the population group at Princeton because only 29% of the students are female there. Final grades were obtained for the students in the study and for the class as a whole. The study assessed what types of gender differences in preparation and skills were significant in predicting success in the introductory course.

In familiarity with computer programming concepts, a gender difference in favor of men was observed. Investigation of computer related activities showed significant differences at the .05 alpha level in favor of the men in several areas (i.e. playing computer games, exploring the internet, reading computer magazines, and attending computer shows).

The patterns at the two schools were very similar and help to explain some of the problems that women experience in introductory computer science courses. The study showed that female students entered the introductory computer science courses with weaker computer skills and less involvement with computers than their male counterparts. Also, final grades in these courses were shown to be strongly dependent on incoming programming skill level, and high achievement in the course was very difficult without previous programming knowledge.

In 1996, Liu and Blanc from Cal Poly State University studied their computer science/engineering students and found that during the first year of the computer science curriculum (consisting mainly of programming courses) that the highest numbers of women leave the major. They made several recommendations, particularly under the control of the computer science department, for addressing the problem of retention of females in this field.

In an effort to collect data about the issues affecting the well being of the female computer science/engineering majors at Cal Poly, a survey was sent to those women. (Of the 642 students enrolled in computer science and computer engineering, only 102 (16%) were female.) Data including backgrounds, interests in computing, and feelings about the current curriculum was collected. Completed questionnaires from 46% of the female students were received and analyzed qualitatively.

The majority of the women responding had not been recruited nor had they had advanced placement courses in mathematics or computer science. On a scale of 1 to 7 (1 being none at all), the average amount of general computer experience that the women surveyed had prior to college was 2.3. They reported feeling intimidated by the terminology from the start of the class.

The combination of lack of preparation felt by many of the females and the emphasis on computer programming at the entry level of the curriculum is a "lethal combination" (Liu & Blanc, 1996, p. 34). Recommendations by the authors included incorporating information about career opportunities, offering remedial introductory courses for students who are less prepared, encourage mentoring among students, relate course material to functionality (actual applications, guest speakers), provide self-paced learning resources to assist students who do not do well in large class settings. The authors also began a student association known as FACT (Females Active in Computing Technology) to promote mentoring and mutual support among the females in the department. Cal Poly is now a member of the WCAR (Women in Computing Academic Resource List).

Hostile Environment and Culture

In Bunderson and Christensen's study (1995) discussed earlier in the chapter, women who left the computer science course reported that they did so because of the influences within the computer science major. Two of the reasons cited were so few numbers of females and feeling uncomfortable in the course. Also they reported that twenty percent of the females felt that they were treated differently due to their gender. In this study the females were more likely to study with other females and preferred cooperative work styles (a work style not encouraged in the computer science discipline). Liu and Blanc (1996) reported similar results in that the females expressed feelings of intimidation in the computer science class. They noted Cottrell's work (1992) which suggested self-paced learning resources for students who were uncomfortable in large classes and quoted some female students who echoed the sentiment that they needed additional helps to be provided, either on-line or in other supplemental modes.

Attribution Theory

In 1984, George Deboer from Colgate University conducted a study at a small liberal arts college investigating the factors related to students' decisions to continue in science courses in college. The purpose of the study was to assess the importance for men and women of the transition between the first collegiate science course experience and the decision to continue in science. Attribution theory (explanations people give for their success or failure in achievement settings) provided the framework for the study. Although this study involved the science field, instead of the computer science field, it gives insight into the possible reasons for attrition of women in computer science.

Surveys were sent to 650 freshmen that had just completed at least one science course, and 105 subjects returned fully completed forms. (If a student was taking more than one science course, one course was randomly selected for the study.) Subjects responded to one of two sets of survey items depending on whether they perceived their experience to have been successful or unsuccessful. A 5-point scale (1 = not a reason; 5 = a very important reason) was used to respond to perceived reasons for success or failure (course was easy / difficult, I was lucky/unlucky, I worked hard/ didn't work hard, my ability is high/ not high in this area).

Analysis was calculated using chi-square to determine if differences existed between the intentions of men and women to continue in science. Multiple regression was also used with the independent variables of attribution to ability, luck, effort, and task difficulty and the dependent variable of decision to continue in science. (Gender, SAT math score, and final grade in first science class were used as controls in the equation.)

The "successful" students' intentions to continue in science directly related to their attribution to ability and inversely to task ease. Gender, math aptitude, attribution to luck and effort, and performance in the initial science course (final grade) were not related to plans to continue in science. The "unsuccessful" students' intentions to continue in science *approached* statistical significance relating only to task difficulty. Gender, math aptitude, performance, attribution to luck, ability, and effort were unrelated to the decision to continue in science. No significant difference was shown between men and women to continue science for the "unsuccessful" students.

Clarke and Chambers (1989) included attribution for success and failure in their study of gender-based factors in computing. They noted that research consistently demonstrates gender-based differences in attribution processes (Maeher, 1983; Dweck & Leggett, 1988). Females are more likely to attribute success to luck and failure to lack of ability while males are more likely to attributes their successes to their own ability or effort and failure to lack of ability or other factors outside of their control. Clarke and Chambers explained:

> These gender differences in attribution processes have been used to account for the common observation that men prefer mathematics and science and women prefer humanities. In mathematics and science, answers are usually objectively right or wrong, so that the outcome is seen as related to knowledge, effort, or ability. By contrast, in humanities, assessment is more subjective and hence more consistent with attributions to external factors. Because interaction with a computer leads to definite outcomes, in that a game response or program does or does not achieve the desired outcome, computing is more likely to appeal to boys as it enables attributions more consistent with their usual style. Thus students who attribute success to internal factors and failure to external factors should be more likely to choose to study computing than students making the alternative attributions. (p. 415)

In their study, Clarke and Chambers (1989) found clear gender differences in students' perceptions of the role of their own ability. In explaining success, females gave significantly higher ratings to factors reflecting situational factors of behavior (good teaching, personal help from tutors, help from friends), and a significantly lower rating to their own ability, an important dispositional determinant of behavior. In relation to failure, females gave higher ratings to factors such as lack of ability, difficulty of the course, poor class teachings, and insufficient access to tutors. Males rated their own ability more highly as a possible reason for their success, while women rated their own lack of ability more highly as a possible source of failure. This is consistent with commonly reported gender differences in the perceived role of ability in attributions for success and failure. Given the difference in participation rates of men and women in science courses, as well as computer science courses, entering college, factors preceding college entrance must be important in explaining this differential level of participation. If, as this study shows, successful students' attribution to ability and feelings of competence are important in their persistence in further related courses, even more than measured aptitude, the pre-college experience in these fields is a factor that should be studied.

Self-Efficacy

Because self-efficacy affects academic performance and can be changed (Bandura, 1977), self-efficacy theory has been widely used in education and training. Several studies have shown that a person's efficacy beliefs in a particular domain strongly relate to skill attainment (as cited in Ramalingam & Wiedenbeck, 1998). Women are not as self-confident as men in masculine domains, where they often underestimate their ability. Bernstein (1991) and Miura (1987) found that women rate themselves lower than men for perceived self-efficacy with computers. Busch (1995) found a gender difference in favor of males in perceived self-efficacy regarding completion of complex tasks in spreadsheet and word processing but found no gender difference in self-efficacy regarding simple computer tasks. In this study, male students reported more previous computer experiences in programming and computer games and more encouragement from parents and friends than did the females. In one study, students (both male and female) incorrectly believed that male computer science majors have higher GPAs than female computer science majors (as cited in Haller & Fossum, 1998). However, several studies have been conducted that show women (who do remain in the major) performing just as well or better than their male counterparts in computer science courses (Anderson, 1987; Harrington, 1990).

Research on the relationship of gender and self-efficacy in the computer programming domain is sparse. Some work has been done in gender-efficacy relationships for end-user software skills (Murphy, Coover, & Owen, 1989) and for gender-efficacy relationship in mathematics (Lent, Brown, & Core, 1997; Lopez, Brown, Lent, & Gore, 1997) but not for computer programming. Ramalingam & Wiedenbeck (1998) developed a self-efficacy instrument specific to the computer-programming domain; therefore, studies including gender and self-efficacy in computer programming may be forthcoming.

In order to gain a better understanding of the relationship of computer programming efficacy and achievement, Ramalingam and Wiedenbeck (1998) developed the Computer Programming Self-Efficacy Scale (see Appendix A). They relied on Bandura's (1977, 1986) argument that self-efficacy is not a personality trait that can be measured by a generic test, but it is a self-judgment of what an individual perceives can be done in a specific domain of activity. Thus, an instrument specific to the activity must measure self-efficacy. Ramalingam and Wiedenbeck's instrument included the three dimensions of self-efficacy theory: (1) magnitude, which measures the level of task difficulty, (2) strength, which refers to the certainty that an individual has about an efficacy judgment, and (3) generality, which is the extent to which the belief holds across different situational contexts (e.g., different time constraints). The instrument consists of thirty-two items specific to the domain of programming in the C++ language and includes questions about designing, writing, comprehending, modifying, and reusing programs. The magnitude of self-efficacy is measured by asking questions about tasks ranging from the simplest and most specific to difficult and generic problems. The strength of self-efficacy is measured by responses on a 7-point Likert-type scale ranging from 1 (not confident at all) to 7 (absolutely confident). The generality of self-efficacy is measured by including items that probe confidence in the ability to do a task in a variety of different scenarios. The instrument was validated by administering the test first during the first week of the semester to 421 students (324 males) enrolled in eight sections of an introductory computer programming course. Then the test was given again during the thirteenth week of the semester. To evaluate construct validity the data were subjected to principle axis factoring. The results of exploratory factor analysis suggested four factors which Ramalingam and Wiedenbeck labeled: (1) independence and persistence, (2) complex programming tasks, (3) self-regulation, and (4) simple programming tasks (p. 373). Reliability coefficients were calculated for the scores on the full thirty-two item scale and the empirically derived factors which emerged in the exploratory factor analysis. Test-retest reliability was also calculated. The overall alpha reliability of the scores was .98. The corrected item-total correlations ranged from .50 to .84. The alpha reliabilities of the factors were: (1) independence and persistence = .94, (2) complex programming tasks = .93, (3) self-regulation = .86, and (4) simple programming tasks = .93. These high reliabilities are consistent with those reported for other scales in the computer domain dealing with self-efficacy and attitudes (as cited in Ramalingam & Wiedenbeck, 1998). In the second administration of the scale the alpha reliability was .97, indicating high test-retest reliability.

Summary

In studying the pipeline shrinkage of women in computer science, research has been conducted mainly in the areas of retention, although some studies have been directed toward the recruitment problem. Many studies were found which included more than one possible factor relating to success in computer science and contributing to the shortage of women in computer science; however, none were found which included all of the factors included in this investigation. The studies discussed do give a solid foundation on which to build theory about factors predictive of success in computer science and the gender differences that may exist in these factors.

CHAPTER III

METHODOLOGY

Purpose of the Study

Generally, the purpose of the study was to examine the reasons for the shortage of women in computer science programs. Specifically, the study was designed to investigate contributing factors to success in an introductory computer science class and to note any gender differences in these contributing factors in order to address and pose solutions for the pipeline shrinkage of women in this field. The study attempted to answer the following questions:

1. What relationship exists between the factors of previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender of introductory computer programming students and their midterm course grade?

2. What factors are predictive of midterm course grade in an introductory computer science course: previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender?

3. Are certain types of previous computing experiences predictive of success in a college computer science course?

4. Of the predictive factors of success in computer science, are gender differences evident? If so, which factors demonstrate gender differences?

Subjects

Approximately 130 students were enrolled in six sections of CS 202 Introduction to Computer Science at a comprehensive midwestern university (approximately 22,000 student population) during the spring of 2000. There were 105 students who voluntarily participated in the study. CS 202 is the first programming class required in the computer science major and uses C++ as the programming language. As is the case in most computer science courses, the percentage of females was low. Only 19 of the 105 students who chose to participate in the study were females (approximately 18%.) The following percentages represent how the sample was classified by year in school: 29% freshmen, 29% sophomores, 22% juniors, 12% seniors, and 8% graduate students. Of the students enrolled in the class, 54% were computer science majors, 10% were electrical engineering majors, and 7% were mathematics majors. Other various majors were also represented in the sample.

Instruments

Two instruments were used to collect data from the subjects: a questionnaire (included in Appendix B) and the Computer Programming Self-Efficacy Scale (discussed in Chapter Two and included in Appendix A).

Questionnaire

The questionnaire collected data on the following items: (1) Gender, (2) math background (number of semesters of high school math classes taken), (3) previous programming experiences, (4) previous non-programming computer experiences, (5) encouragement by others to pursue computer science as a career, (7) comfort level, (8) work style preference, and (9) attribution for perceived "success" or "failure" on the midterm exam. A pilot test was given to enable the researcher to find any ambiguities in the instrument, and revisions were made appropriately. One expert in the field of research in psychology and two experts in the field of testing and evaluation were asked to evaluate the face validity of the questionnaire. These experts were professors in the Department of Psychology, Murray State University and in the Department of Curriculum and Instruction, Southern Illinois University at Carbondale. The questionnaire was found to have high face validity. Four seasoned computer science professors examined the content of the instrument. Three of these experts were from Murray State University and one was from Southern Illinois University at Carbondale. The questionnaire was found to have high content validity for measuring the variables in the study.

A test-retest was used to examine the reliability of the questionnaire. The instrument was administered to students in an introductory computer science course at Murray State University. Because the questionnaire was intended to measure different attributes, it was necessary to determine nine correlations. The Pearson Correlation coefficients were .98 for math background, 1.0 for previous programming course, .72 for previous self-initiated programming experience, .95 for previous non-programming experience, .80 for work style preference, .88 for comfort level, .77 for attitude toward exam grade, .72 for attributions to success/failure, and 1.0 for encouragement.

Computer Programming Self-Efficacy Scale

The Computer Programming Self-Efficacy Scale was used to collect data on domain-specific self-efficacy as it relates to tasks in the C++ programming language. This instrument was developed and validated by Ramalingam & Wiedenbeck (1998) as discussed in Chapter Two. The authors reported an overall alpha reliability of .98 on the first administration and .97 on the second administration of the instrument.

Predictor Variables

Thirteen predictor variables were originally included in the study. They were gender, prior programming experience (including a high school programming course and self-initiated programming), prior non-programming computer experience (including internet, e-mail / chat-rooms /on-line discussion groups, games, and productivity software), encouragement to pursue computer science, self-efficacy in the computer science class, comfort level in the computer science environment, work style preference, math background, ACT math score, and attribution (including ability, luck, effort, and task difficulty). The way in which each variable was measured is described below: (all of this data except for self-efficacy, math ACT score, and midterm course grade were collected via the questionnaire)

1. Gender – a dichotomous variable (male or female).

2. Previous programming experience – a dichotomous variable determined by whether the subject had engaged in any programming prior to the course.

In order to study the types of previous programming experience, this variable was subdivided into two areas: (a) formal programming course taken – a dichotomous variable determined by whether the subject had a previous programming course or not and (b) self-initiated programming experience – a dichotomous variable determined by whether the subject learned to write programs separate and apart from a formal class.

3. Previous non-programming experience – a dichotomous variable determined by whether the subject had engaged in non-programming computer activities.

In order to study the types of previous non-programming experience, this variable was subdivided into three areas: (a) internet experience – a continuous variable determined by the number of hours per week each subject reported using the internet, e-mail, chat-rooms, and/or on-line discussion groups, (b) games – a continuous variable determined by the number of hours per week the subject reported spending time playing games on the computer, and (c) use of productivity software – a continuous variable determined by the number of hours per week the subject reported using productivity software such as word processing, spreadsheets, databases, and/or presentation software packages.

4. Encouragement to pursue computer science – a dichotomous variable representing whether the subject received encouragement to pursue computer science or not.

5. Comfort level – a continuous variable derived from seven questions on the questionnaire regarding asking and answering questions in class, in lab, and during office hours; anxiety level while working on computer assignments; perceived difficulty of course; perceived understanding of concepts in the course as compared with classmates; and perceived difficulty of completing the programming assignments. Numbers were appropriately assigned to the choices of the BAR (Behaviorally Anchored Rating) scale in these questions and summed for a composite score of comfort level.

6. Work style preference– a dichotomous variable (competitive or cooperative) representing the answer to a question about preference for writing computer programs and studying for exams.

7. Attributions – four continuous variables derived from the subject's rating of the possible reasons given for success or failure on the midterm exam. They are: (a) attribution to ability, (b) attribution to task ease/difficulty, (c) attribution to luck, and (d) attribution to effort.

8. Self-efficacy – a continuous variable, which is the summation of the choices made on a Likert-type scale from the Computer Programming Self-Efficacy Scale.

9. Math background – a continuous variable represented by the number of semesters of high school math courses reported by the subject.

10. Math ability – a continuous variable represented by the subject's ACT math score.

Criterion Variable

The criterion variable of the study was the midterm grade in the introductory computer science class for each student. (Because of the high attrition rates in introductory computer science courses and because of the desire to study this phenomenon as it relates to the factors contributing to success in the introductory computer science course, midterm grades were used to determine success in the course to enable the inclusion of the students who drop out of the course before the end of the semester.) This was a continuous variable representing a number between 0 and 100.

Procedure

Permission for conducting research activities involving human subjects was obtained from the SIUC Human Subjects Committee (See Appendix C). A cover letter and consent form (included in Appendix D) for collecting data along with the questionnaire and Computer Programming Self-Efficacy Scale were distributed during the spring semester after the first exam and before midterm of the semester during a class lecture session. Because so many students were absent during the class lecture section, the questionnaires were given to students who had not had opportunity to complete the questionnaire and Computer Programming Self-Efficacy Scale during the lab sessions following the lecture class. Students were asked to participate on a voluntary basis. There were 105 students who completed the questionnaires.

An unexpected phenomenon occurred when the university supplied the ACT scores for the students in CS 202. Forty-five percent of the students who completed questionnaires did not have ACT scores recorded at the university. (Transfer students, graduate students, and students with a military background do not have to supply ACT scores for admission.) Multiple regression tests were run on the 55% of the students who did complete questionnaires to determine if the ACT math scores contributed to the prediction of success in the course as defined in Chapter One. Because the ACT math scores did not produce a significant difference in the criterion variable, $p > .73$, and because math background was included as a predictor variable being studied anyway, the ACT score was dropped from the predictor variable list so that all 105 students could be included in the study.

To ascertain that the use of the midterm grade was a viable choice for determining success in the computer programming class, a correlation coefficient was generated using the midterm scores and the final scores in two sections of the first course in Computer Science from the fall semester of 1999. The Pearson Correlation Coefficient was extremely high and significant, $\underline{r} = .97173$, $N = 48$, $\underline{p} = .0001$, therefore, it seemed reasonable that the midterm grade was a good indicator of success in the class.

A correlational study was conducted in which data collected from each subject on various factors discussed above was compared to each subject's midterm grade in the class. Also data was analyzed to determine if gender differences were evident in any of the predictor variables.

Analysis of Data

Although no study could be found that combined all of the predictor variables that are included in this study, some of the previous research could be used to determine an expected hierarchy of predictor variables. Therefore, based on the literature review and on the researcher's experience of teaching computer science, a hierarchical model was generated and tested using the general linear model. The model included twelve predictor variables in the following order: math, previous programming experience, attribution to luck, attribution to difficulty of task, comfort level, non-programming experience, work style preference, domain-specific self-efficacy, encouragement to study computer science, attribution to effort, attribution to ability, and gender. This model was tested and compared to the findings of the previous research studies in computer science success. All analyses used an alpha level of .05 to determine significance and were conducted to answer the following questions:

Research Question 1

What relationship exists between the factors of previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender of introductory computer programming students and their midterm course grade? More technically stated, what is the proportion of variance in midterm course grade accounted for by the linear combination of the factors: previous programming experience, previous non-programming experience, attribution for success/failure (including 4 possible attributions), self-efficacy, comfort level, encouragement from others, work style preference, math background, and gender?

A correlation matrix was generated to examine how each of the 12 factors correlated with midterm grade and with each of the other predictor variables. By examining the R^2 and its p-value of the full-model regression equation, the proportion of variance in midterm grade accounted for by the twelve predictor variables was determined; thus Research Question 1 was answered.

Research Question 2

What factors are predictive of midterm course grade in an introductory computer science course: previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender? More technically stated, the second question was: What is the contribution of each factor over and above the contribution of the other factors in the prediction of the midterm course grade?

Using the same full model, the Type I sums of squares and Type III sums of squares with associated p-values were examined to determine the contribution of each factor over and above the other factors. The parameter estimates from the multiple regression tests were also examined to see whether each factor had a positive or negative effect on midterm grade; thus Research Question 2 was answered.

Multiple Regression Analyses. Full Model for Research Question 1 and 2:

$$Y = a_0U + a_1X1 + a_2X2 + a_3X3 + a_4X4 + a_5X5 + a_6X6 + a_7X7 + a_8X8$$
$$+ a_9X9 + a_{10}X10 + a_{11}X11 + a_{12}X12 + E$$

where Y = midterm course grade

X1 = 1 if previous programming; 0 otherwise
X2 = 1 if previous non-programming; 0 otherwise
X3 = rating for attribution to task difficulty
X4 = rating for attribution to luck
X5 = rating for attribution to effort
X6 = rating for attribution to ability
X7 = rating for self-efficacy
X8 = rating for comfort level
X9 = 1 if had encouragement; 0 otherwise
X10 = 1 if male; 0 if female
X11 = number of semesters of math courses
X12 = 0 if work style preference is cooperative; 1 if competitive
E = the errors of prediction

$a_1,a_2,a_3,a_4,a_5,a_6,a_7,a_8,a_9,a_{10},a_{11},a_{12}$ are the least squares weighting coefficients calculated so as to minimize the sum of squared values in the error vectors.

Research Question 3

Are certain types of previous computing experiences predictive of success in a college computer science course? Again, more technically stated, the third question was: Are certain types of previous computing experiences (a programming class; self-initiated programming; use of internet, e-mail, chat rooms, and/or discussions groups; playing games on the computer; use of productivity software) predictive of success in a college introductory computer science class?

Research question 3 was answered by using the full model (shown below) and four restricted models. The restricted models were constructed by dropping out one predictor variable from the full model. Each restricted model was tested against the full model to ascertain whether the contribution of each predicting factor over and above the other factors in combination was significant.

Multiple Regression Analysis. Full Model for Research Question 3:

$$Y = b_0U + b_1P1 + b_2P2 + b_3P3 + b_4P4 + b_5P5 + \quad + E$$

where Y = midterm course grade

P1 = 1 if programming class; 0 otherwise
P2 = 1 if self-initiated programming; 0 otherwise
P3 = number of hours/week of Internet use
P4 = number of hours/week of games
P5 = number of hours/week of productivity software use
E = errors of prediction

b_1, b_2, b_3, b_4, b_5 are the least squares weighting coefficients calculated so as to minimize the sum of squared values in the error vectors.

Research Question 4

Of the predictive factors of success in computer science, are gender differences evident? If so, which factors demonstrate gender differences? Stated in technical terms, Does a significant difference exist between genders for each of the predictive factors?

Multiple regression equations for the two genders were generated and results compared to determine if there was a significant difference in predicting the midterm grade, but because the female segment of the sample was so small, a more conservative approach was taken to answer this question by using a nonparametric Wilcoxon test to compare the genders on each of the predictive factors.

CHAPTER IV

RESULTS

In order to study the contributing factors to success in an introductory computer science class and to determine any gender differences in these contributing factors, the following questions were answered. The results were based on an alpha level of .05 to determine significant differences in all four research questions.

Prediction of Midterm Score with Full Model

A multiple regression analysis was conducted to find answers to Research Question 1: What is the proportion of variance in midterm course grade accounted for by the linear combination of the factors: previous programming experience, previous non-programming experience, attribution for success/failure (including 4 possible attributions), self-efficacy, comfort level, encouragement from others, work style preference, math background, and gender?

In addition, the multiple regression analysis was used to answer Research Question 2: What is the contribution of each factor over and above the contribution of the other factors in the prediction of the midterm course grade?

Based on the literature review and on the researcher's experience of teaching computer science, a hierarchical model was generated and tested using the General Linear Model. The model included twelve predictor variables in the following order: math, previous programming experience, attribution to luck, attribution to difficulty of task, comfort level, non-programming experience, work style preference, domain-specific self-efficacy, encouragement to study computer science, attribution to effort, attribution to ability, and gender.

In order to examine the underlying assumptions of homoscedasticity and normality, a scatter plot and histogram were generated. These can be viewed in Appendix E.

The data shown in Table 1 were used to answer Research Question 1. Table 1 shows the results of a multiple regression equation using all predictor variables. The proportion of variance in midterm score accounted for by the linear combination of the 12 factors (previous programming experience; previous non-programming experience; attribution for success/failure including luck, difficulty of task, effort, and ability; self-efficacy; comfort level; encouragement from others; work style preference; math background; and gender) was approximately .44, $\underline{R}^2 = .4443$, which was statistically significant, $\underline{F}(12, 92) = 6.13$, $\underline{p} = .0001$.

As seen in Table 2, the GLM (General Linear Model) reported in the Type I sums of squares how each factor contributed to the variance in the hierarchical order given . This showed the contribution of each factor when added after each previous factor (in order) was already included in the model. The GLM also simultaneously assessed each factor as if it was entered last therefore showing its unique contribution to the variance in midterm over and above all other factors entered into the model. As indicated in Table 2, the Type III sums of squares is important, in that three of the predictor variables contributed a significant difference in the midterm grade at the .05 level even after being considered last in the model. They were comfort level, math background, and attribution of success/failure to luck with p-values of .0002, .0050, and .0233 respectively. Two of the three significant predictive factors (comfort level and math) had positive correlations with the midterm score, but attribution of success/failure to luck had negative parameter estimation (shown in Table 1).

A stepwise regression was also conducted to determine the contribution of each factor to the proportion of variance for the model. Table 3 illustrates each step of the regression in which the best predictor is entered first, then the next best predictor entered with the possibility of removal of any predictor variable as each step progresses and continuing until no other variables can be entered at the criterion level specified. The contribution of each variable to the proportion of variance is also shown in Table 3. In this table, it is noted that after the fifth step is completed, five predictors are significant with approximately 40% of the variance accounted for by this partial model, $\underline{R}^2 = .3966$. In comparison with the three significant factors identified by the hierarchical full model, the two added predictors in the stepwise regression that showed significance were work style preference and attribution of success/failure to task difficulty. Again, Table 1 indicates that work style preference has a positive influence (indicating the preference for individual work positively correlated with midterm grade) and attribution to difficulty of task has a negative correlation with midterm score. The best five-variable model, therefore, included the top five variables in the Table 3: comfort level, math, attribution to luck, work style preference, and attribution to task difficulty.

Since multiple regression analysis was used, correlations among the variables were also determined by generating a correlation matrix. As illustrated in Table 4, the

Table 1

Summary of Multiple Regression Analysis on 12 Predictor Variables and Midterm Grade

Analysis of Variance for Model				
Source	Df	R^2	F value	P
Model	12	.4443	6.131	.0001
Error	92			
C Total	104			

Parameter Estimates

Variable	Parameter estimate	Standard error	T	P
Intercept	25.4928	11.5857	2.200	.0303
*Math Bckgrnd	1.5760	.5481	2.875	.0050
Prev.Prog.	4.4916	2.8314	1.586	.1161
*Attrib. Luck	-2.4660	1.0687	-2.307	.0233
Attrib.Difficulty	-1.8640	1.0860	-1.716	.0895
*Comfort Level	1.2526	.3225	3.884	.0002
Prev. NonProg.	6.529	6.2400	1.046	.2981
Work Pref.	4.6016	2.6057	1.766	.0807
Self Efficacy	-.0144	.0507	-.285	.7767
Encouragement	2.2484	2.1448	1.048	.2973
Attrib. Effort	1.439	1.0391	1.385	.1695
Attrib. Ability	.1470	.9886	.149	.8821
Gender	-4.1202	3.6279	-1.136	.2590

* Factors significant for alpha = .05.

Table 2

Summary of Type I and Type III Sums of Squares w/ Dependent Variable: Midterm

General Linear Models Procedure
(R^2 = .444347, F = 6.13, P = .0001)

Source	DF	Type I SS	F	P
Math	1	2348.67	15.17	.0002
Prg	1	1203.42	7.77	.0064
Luck	1	2014.73	13.02	.0005
Diff	1	766.75	4.95	.0285
Cmflv	1	3460.21	22.36	.0001
NPrg	1	51.19	0.33	.5666
WPrf	1	625.34	4.04	.0474
SE	1	53.60	0.35	.5577
Enc	1	302.55	1.95	.1654
Eff	1	359.15	2.32	.1311
Abil	1	2.20	0.01	.9053
Gnd	1	199.63	1.29	.2590

Source	DF	Type III SS	F	P
Math	1	1279.79	8.27	.0050
Prg	1	389.52	2.52	.1161
Luck	1	824.10	5.32	.0233
Diff	1	455.95	2.95	.0895
Cmflv	1	2334.38	15.08	.0002
NPrg	1	169.47	1.09	.2981
WPrf	1	482.72	3.12	.0807
SE	1	12.53	0.08	.7767
Enc	1	170.09	1.10	.2973
Eff	1	296.74	1.92	.1695
Abil	1	3.42	0.02	.8821
Gnd	1	199.63	1.29	.2590

Note: Variable
Math = math background; Prg = previous programming experience; Luck = attribution to luck; Diff = attribution to task difficulty; Cmflv = comfort level; NPrg= previous non-programming experience; Wprf = work style preference; SE = self-efficacy; Enc = encouragement, Eff = attribution to effort; Abil = attribution to ability level; Gnd = gender.

strong correlation between self-efficacy and comfort level may raise the question, Is some of the self-efficacy variance included in the comfort level variable? While their

Table 3

Summary of Stepwise Regression Analysis for Full Model w/ Dependent Variable: Midterm

Step	Variable entered	Variable removed	Partial R^2	Model R^2	F	P
1	Comfort Level		.2273	.2273	30.2939	.0001
2	Math Background		.0487	.2760	6.8589	.0102
3	Luck Attrib.		.0637	.3397	9.7481	.0023
4	WorkPref		.0309	.3705	4.9020	.0291
5	Task Diff. Attrib.		.0260	.3966	4.2674	.0415
6	Effort Attrib.		.0129	.4095	2.1453	.1462
7	Encouragement		.0096	.4191	1.6114	.2073
8	Prev. Prog Exp.		.0081	.4272	1.3531	.2476
9	Gender		.0080	.4352	1.3476	.2486
10	Prev.NonProg Exp.		.0084	.4436	1.4108	.2379
11	Self Efficacy (C++)		.0006	.4442	.1078	.7434
12	Ability Attrib.		.0001	.4443	.0221	.8821

correlation is significant, $r = .53277$, $p = .0001$, the variance they share (as shown in the multiple regression results) is not shared with midterm grade.

Table 4

Summary of Correlation Analysis on Regression Variables

Pearson Correlation Coefficients

	Sc	Gnd	Mat	Prg	NPr	Enc	Wpr	Cmf	Diff	Luc	Eff	Abil	SE
Sc	1.0	-.12	.32	.25	.27	.10	.26	.48	-.20	-.22	.19	.08	.21
Gnd	-.12	1.0	.02	.24	.11	-.20	.01	.09	.09	.16	-.19	-.08	.29
Mat	.32	.02	1.0	.21	.01	.09	.07	.21	.02	.25	-.02	-.00	.11
Prg	.25	.24	.21	1.0	.07	-.05	.20	.24	-.14	.05	.04	-.17	.21
NPr	.27	.11	.01	.07	1.0	-.03	.15	.13	.22	-.02	-.30	.02	-.01
Enc	.10	-.20	.09	-.05	-.03	1.0	-.16	-.09	-.02	.04	-.01	.06	.14
Wpr	.26	.01	.07	.20	.15	-.16	1.0	.17	-.09	-.02	.17	.09	.11
Cmf	.48	.09	.21	.24	.13	-.09	.17	1.0	-.10	-.03	-.07	-.03	.53*
Diff	-.20	.09	.02	-.14	.22	-.02	-.09	-.10	1.0	-.02	-.10	.34	-.06
Luc	-.22	.16	.25	.05	-.02	.04	-.02	-.03	-.02	1.0	-.26	-.33	.03
Eff	.19	-.19	-.02	.04	-.30	-.01	.17	-.07	-.10	-.26	1.0	.08	-.06
Abil	.08	-.08	-.00	-.17	.02	.06	.09	-.03	.34	-.33	.08	1.0	-.17
SE	.21	.29	.11	.21	-.01	.14	.11	.53*	-.06	.03	-.06	-.17	1.0

* p=.0001

Note. Variable

Sc = score on midterm; Gnd = gender; Mat = math background; Prg = previous programming experience; NPr= previous non-programming experience; Enc = encouragement, Wpr = work style preference; Cmf = comfort level; Diff = attribution to task difficulty; Luc = attribution to luck; Eff = attribution to effort; Abil = attribution to ability level; SE = self-efficacy

Previous Computing Experiences

A multiple regression equation was also used to answer Research Question 3: Are certain types of previous computing experiences (a programming class; self-initiated programming; use of internet, e-mail, chat rooms, and/or discussions groups; playing games on the computer; use of productivity software) predictive of success in a college computer science course? As indicated in Table 5, two of the variables showed significant differences in predicting the midterm score: previous programming course and games with p-values of .0006 and .0287 respectively. It was also noted that while the previous programming course variable had a positive parameter estimate on midterm grade, games had a negative parameter estimate. Also the proportion of variance accounted for by the five previous programming and non-programming variables was .15 which was significant for the sample, $p = .0041$.

Gender Differences

Because the number of females in the sample was so small, the Wilcoxon nonparametric test (Cody & Smith, 1991, p. 144) was used (instead of generating separate multiple regression equations for each gender on all predictive factors) to compare the means of each gender group on each of the predictor variables in order to answer Research Question 4: Does a significant difference exist between genders for each of the predictive factors? Table 6 illustrates the findings from these comparisons. There were no significant differences found between the genders on the twelve full-model predictors. The largest difference found was in previous programming experience where the female mean was considerably lower than the male mean but not at the significance level in this

Table 5

Summary of Multiple Regression Analysis on Previous Programming and Non-Programming Experience

Analysis of Variance for Model				
Source	Df	R^2	F value	P
Model	5	.1577	3.706	.0041
Error	99			
C Total	104			

Parameter Estimates				
Variable	Parameter estimate	Standard error	T	P
Intercept	64.8755	2.6049	24.905	.0001
PrvPrgCs	10.6138	2.9944	3.545	.0006
SiPrg	1.8392	3.3532	.548	.5846
Int	.0906	.1765	.514	.6087
Games	-.4217	.1900	-2.219	.0287
Apps	.2315	.1761	1.315	.1917

Note. Variable
PrvPrgCs = previous programming course; SiPrg = self-initiated programming; Int = use of the Internet; Games = playing games on the computer; Apps = use of productivity software

study, $p = .1142$. There was a significant difference found on the games predictor. Males reported much more experience with playing games on the computer, $M = 56.105$ (male), $M = 38.947$ (female), $p = .0218$.

Table 6

Summary of Analysis of Predictor Variables by Gender

Wilcoxon Scores Classified by Variable Gender				
Gender	Sum of scores	S.D.	Mean score	P
Variable tested: Comfort level				
F	909.5	119.859	47.8684	.4184
M	4655.5	119.859	54.1337	
Variable tested: Math				
F	973.0	115.0201	51.2105	.7709
M	4592.0	115.0201	53.3953	
Variable tested: Attribution to Luck				
F	937.0	115.5436	44.0526	.1424
M	4728.0	115.5436	54.9767	
Variable tested: Previous Programming Class				
F	842.5	103.8177	44.3421	.1142
M	4722.5	103.8177	54.9128	
Variable tested: Games				
F	740.0	116.1735	38.9474	.0218
M	4825.0	116.1735	56.1047	

Table continues

Wilcoxon Scores Classified by Variable Gender				
Gender	Sum of scores	S.D.	Mean score	P
		Variable tested: Work Style Preference		
F	1024.0	103.2486	53.8947	.8730
M	4541.0	103.2486	52.8023	
		Variable tested: Attribution to Task Difficulty		
F	953.0	116.8791	50.1579	.6441
M	4612.0	116.8791	53.6279	
		Variable tested: Self-efficacy		
F	707.0	120.1256	37.2105	.0127
M	4858.0	120.1256	56.48884	
		Variable tested: Attribution to Ability		
F	1148.5	116.3855	60.4474	.2257
M	4416.5	116.3855	51.3547	
		Variable tested: Use of Internet		
F	809.0	119.6238	42.5789	.0987
M	4756.0	119.6238	55.3023	
		Variable tested: Attribution to Effort		
F	1243.0	116.9182	65.4211	.0440
M	4322.0	116.9182	50.2558	

Table continues

Wilcoxon Scores Classified by Variable Gender				
Gender	Sum of scores	S.D.	Mean score	P
Variable tested: Previous Programming Experience (both class & self-initiated)				
F	747.5	100.5504	39.3421	.0100
M	4817.5	100.5504	56.0174	
Variable tested: Previous Non-Programming (internet, games, & apps)				
F	949.5	44.3164	49.9737	.1984
M	4615.5	44.3164	53.6686	
Variable tested: Previous Applications Software Experience				
F	992.0	119.5568	52.2105	.9035
M	4573.0	119.5568	53.1744	
Variable tested: Previous Self-Initiated Programming Experience				
F	836.5	93.0433	44.0263	.0677
M	4728.5	93.0433	54.9826	
Variable tested: Midterm Score				
F	1130.0	120.1386	59.4737	.3079
M	4435.0	120.1386	51.5698	

Note: N = 19 for females, N = 86 for males

CHAPTER V

SUMMARY, DISCUSSION, AND RECOMMENDATIONS

Summary of Study

Because of the desire to study the shortage of women in computer science, this study was designed in order to determine factors that promote success in a computer science course and to determine what, if any, differences appear between genders on those factors. Four questions were proposed:

1. What relationship exists between the factors of previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender of introductory computer programming students and their midterm course grade?

2. What factors are predictive of midterm course grade in an introductory computer science course: previous programming experience, previous non-programming experience, attribution for success/failure, self-efficacy, comfort level, encouragement from others, work style preference, math background, ACT math score, and gender?

3. Are certain types of previous computing experiences predictive of success in a college computer science course?

4. Of the predictive factors of success in computer science, are gender differences evident? If so, which factors demonstrate gender differences?

The ACT math score was removed from the study when it was discovered that only 45% of the sample population had ACT scores recorded at the university where the study was being conducted. Since multiple regression analysis indicated that ACT score was not significant in predicting the midterm score with those who did have the ACT scores and after considering that the math background variable included mathematical influences in the model, the decision to remove the ACT math score from the model was made so that all 105 subjects could be included in the study.

A questionnaire was used to collect data on some of the variables being studied. A Computer Programming Self-Efficacy Scale for C++, developed by Ramalingam & Wiedenbeck (1998) was used to collect data for the self-efficacy variable. The midterm grade was collected from the professor of the computer science course in which the subjects were enrolled. There were 105 subjects who volunteered to be studied with only 18% of them female.

Findings

Multiple regression equations were generated and tested using SAS (Statistical Analysis Software). An alpha level of .05 was used for the study. Major findings of this study can be summarized as follows:

1. The full model, including 12 predictor variables mentioned in question 1 above, resulted in explaining 44% of the variance in midterm score for the 105 subjects studied, which was statistically significant, $\underline{R}^2 = .4443$, $\underline{F}(12,92) = 6.13$, $\underline{p} = .0001$.

2. Three variables showed unique significant influence upon the midterm score when included in the full model. They were comfort level, math background, and attribution of success/failure to luck. The comfort level and math background variables

were positively correlated with midterm score, whereas attribution to luck was negatively correlated with the midterm score. When stepwise multiple regression was used, two more variables showed significant influence in a five factor model. They were work style preference and attribution of success/failure to task difficulty. These five variables contributed to 40% of the variance. The work style preference was positively correlated to the midterm score, which indicated that an individual/competitive work style preference had a positive influence on the midterm score. Attribution to task difficulty was negatively correlated to midterm score.

3. Two categories of previous computing experience variables showed significant difference in the midterm score when tested in a multiple regression equation. They were previous programming course and games. The previous programming course experience was positively correlated to midterm score, whereas playing games on the computer was negatively correlated to midterm score.

4. No significant differences between females and males were found in any of the full-model significant variables identified in the study at the alpha level of .05. However, a significant difference between genders was found on playing games on the computer. A fairly large mean difference in previous programming course experience was found (but it did not reach the required significance at the alpha level for this study) in which the male mean, $\underline{M} = 54.9$, was larger than the female mean, $\underline{M} = 44.3$, $\underline{p} = .1142$.

Discussion

The discovery that only 18% of the students enrolled in CS 202 were female was not unexpected, although the percentage was lower than the 4 to 1 ratio reported by Taylor and Mounfield (1991) for most college computer science classes. This supports the widely discussed concern that there is shortage of women in computer science. Furthermore, this small percentage of females enrolled in the first computer programming course supported the proposition that the pipeline shrinkage of women in computer science is a problem stemming mainly from occurrences prior to university admission. The notion, "retention of women once they enter the major is important, but it is second to getting women into the major initially" put forth by Scragg and Smith (1998) does seem to have merit. Recruitment issues involving sex stereotyping and lack of encouragement may be at play here in lessening the numbers of females who might be qualified for the difficult curriculum in computer science but who choose other fields of study instead. In post hoc analyses and shown in Table 5, it was noted that females reported having more encouragement to study computer science than the males in the sample. This may seem like a contradiction to the above statement about lack of encouragement for females. Again, one must be careful in looking at this statistic. It may be that encouragement is very important for females, and, if the encouragement were not there, the women in this study may not have chosen to pursue computer science. The fact that so few women choose to major in computer science makes studying the gender differences in the field very difficult. It should be noted that the issues of actually getting females into the computer science discipline and issues of succeeding in this discipline once they decide to major in computer science may be different phenomena with differing predictor variables.

Comfort level in the computer science class was the best predictor of success in the course. This fact, coupled with the finding that there is a moderately strong correlation found between comfort level and self-efficacy, relates well to Scragg and

Smith's study on the problem women face with low self-confidence in computer science performance compared to male self-confidence. Even though there was no significant difference found between genders on this variable in this study, one must realize that the small percentage of women who do choose to major in such a male dominated domain may have more self-confidence than women who were academically "qualified" to study in this area but who chose to study in another area. One must be careful, however, to consider that the correlation between comfort level and midterm grade does not necessarily mean causation. It could be that those students who do well in the class feel more comfortable because of their success.

Math background was second in importance in predicting success in this computer science class. It is most interesting, in this study, that comfort level was found to be more important than math background. Most of the research studied for the literature review, which included math as a predictor, concluded that math and computer programming experience were the most important factors in success in computer science, although many of these studies did not include studying comfort level as such. Although programming experience (which included both a previous programming course and self-initiated programming) was not found to be significant in the full model, when the different types of computing experiences were compared as predictors of midterm grade, the previous programming course and game playing were both significant. It should be noted that the notion that game playing gives students an "edge" in a computer science course was not supported in this study. Game playing had a negative effect on the midterm grade. This finding would be encouraging for females if indeed further studies show support for it, since females reported a significantly lower experience with playing games on the computer than males reported. As shown in Table 5, females reported less experience with self-initiated programming than males reported and almost at a statistically significant level, $\underline{M} = 44.0263$ (females), $\underline{M} = 54.9826$, $\underline{p} = .0677$.

The result for analysis of attribution to luck was also an interesting finding. To support most of the attribution research findings, attribution to luck would only be positively correlated to success in the course for those students who were unhappy with their score. In other words, if they could attribute their "low" score to an unstable cause such as luck, then they would continue to try to do better. In this study, however, attribution to luck for all students (whether happy or unhappy with their scores) was negatively correlated to midterm.

There have been several studies on self-efficacy and the gender differences that exist, particularly in science-related fields. Although self-efficacy was not found to be a significant factor in this study, it should be noted that there was a significant difference in the self-efficacy scores for male and female, $\underline{M} = 56.488$ (male), $\underline{M} = 37.211$ (female), $\underline{p} = .0127$. It was interesting to note that many males reported higher self-efficacy scores although their midterm grade did not reflect the "knowledge" they claimed to have. In post hoc analysis, correlations were generated between self-efficacy and midterm score for each gender. The results showed a higher correlation for females than for males, $\underline{r} = .32994$ (females), $\underline{r} = .23409$ (males). It was interesting to note that females scored higher on the average than did males (see Table 5), although not at a statistically significant level.

Recommendations

For Practice

Although this study did not show that higher comfort levels "cause" students to perform better in the computer science class, because of the positive correlation in this study between comfort level and success in the introductory computer science course, the notion that providing the optimum class environment for producing higher levels of comfort for students is at least warranted. It is suggested that professors of college computer science should understand the importance of providing an environment in the course which encourages students to ask and answer questions, both in class and outside of class, in a way that allows the students to feel comfortable and not intimidated. Opportunities for students to be able to consult with faculty, teaching assistants, or tutors were also indicated. The recent move in many universities to force students into large lecture sections for computer science, which by its very nature discourages dialogue between students and faculty, is an indication of the misunderstanding of the importance of the level of comfort students may need in this difficult discipline. Also, advisers should stress an appropriate mathematical background for students wanting to pursue computer science. Finally, since attribution to luck showed a negative correlation with success, professors should endeavor to match class assignments and exam questions in the hope that students will not perceive luck as a reason for success or failure on the exams. Again, this suggestion is warranted even though the study only showed a negative correlation and not causation.

For Further Research

More study on how comfort level correlates with success in the computer science class is needed. Replications of this study including at least the top five predictors (comfort level, math background, attribution to luck, work style preference, and attribution to difficulty) should be completed to find out if, indeed, these are important over and above other factors in the computer science class. In order to study the effect of comfort level, studies that investigate several different sized classes (large universities and smaller colleges) with differing styles of provided tutoring and help for students could be conducted. This type of study could be done because the first computer science course has specific guidelines put forth by the ACM (Association of Computing Machinery), which are followed by most colleges and universities offering a computer science major. Studies should also be conducted that investigate why the females who chose to pursue computer science did so. This would probably necessitate an intense qualitative research effort and could include influences even back in childhood and personality traits such as confidence, perseverance, and work style preference.

The number of non-freshmen students was not anticipated in this study because the course is the first programming course in the major. In future studies that includes mathematics background, the variable should probably include all math courses taken prior to the course instead of only high school math courses.

REFERENCES

Anderson, R.E. (1987). Females surpass males in computer science problem solving: Findings from the Minnesota computer literacy assessment. Journal of Educational Computing Research, 3(1), 39-51.

Anderson, R.O., Klassen, D.L., Krohn, K.R., & Smith-Cunnien, P. (1982). Assessing computer literacy: Computer awareness and literacy: An empirical assessment. (Available from Minnesota Educational Computing Consortium)

Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. Psychological Review, 84(2), 191-215.

Bandura, A. (1986). Social foundations of thought and action: A social cognitive theory. Englewood Cliffs, NJ: Prentice-Hall.

Bandura, A. (1991). Social cognitive theory of self-regulation. Organizational Behavior and Human Decision Process, 50, 248-287.

Bernstein, D. (1991). Comfort and experience with computing: Are they the same for women and men? SIGCSE Bulletin, 23(3), 57-60.

Bernstein, D. (1992). A new introduction to computer science. In In Search of Gender Free Paradigms for Computer Science Education. (Available from International Society for Technology in Education)

Brody, L., & Fox, L.H. (1980). An accelerative intervention program for mathematically gifted girls. In L.H. Fox, L. Brody, & D. Tobin (Eds.), Women and the Mathematical Mystique. Baltimore, MD: The John Hopkins University Press.

Brown, J., Andreae, P., Biddle, R., & Tempero, E. (1997). Women in introductory computer science: Experience at Victoria University of Wellington. SIGCSE Bulletin, 29(1), 111-115.

Bulkeley, W.M. (1994, March 16). A tool for women, a toy for men. The Wall Street Journal, pp. B1, B5.

Bunderson, E.D., & Christensen, M.E. (1995). An analysis of retention problems for female students in university computer science programs. Journal of Research on Computing in Education, 28(1), 1-15.

Camp, T. (1997, October). The incredible shrinking pipeline. Communications of the ACM, 40(10), 103-110.

Carmichael, H., Burnett, J., Higginson, B., Moore, B., & Pollard, P. (1985). Computers, children and classrooms: A multisite evaluation of the creative use of microcomputers by elementary school children. (Available from the Ministry of Education, Ontario, Canada)

Clarke, V.A. (1986). Computing in a social context: Gender differences in participation. Unpublished doctoral dissertation, Deakin University, Australia.

Clarke, V.A., & Chambers, S.M. (1989). Gender-based factors in computing enrollments and achievement: Evidence from a study of tertiary students. Journal of Educational Computing Research, 5(4), 409-429.

Cody, R.P., & Smith, J.K. (1991). Applied statistics and the SAS programming language (4th edition). Upper Saddle River, New Jersey: Prentice Hall.

Cottrell, J. (1992). I'm a stranger here myself: A consideration of women in computing. Proceedings of the 1992 ACM SIGUCCS User Services Conference, 71-76.

Dachey, K. (1983, April 8). Computing not gender neutral: Females more intimidated. The Stanford Daily, p. A5.

DeClue, T. (1997). Academic computer science and gender: A naturalistic study investigating the causes of attrition. Unpublished doctoral dissertation, Southern Illinois University, Carbondale.

Deboer, G.E. (1984). Factors related to the decision of men and women to continue taking science courses in college. Journal of Research in Science Teaching, 21(3), 325-329.

Demand for computer specialists to double in the next decade. (1999, June). OR/MS Today, 24.

Digest of education statistics. (1997). National Center for Education Statistics. Washington, D.C.: U.S. Department of Education,

Dweck, C.S., & Leggett, E.L. (1988). A social-cognitive approach to motivation and personality. Psychological Review, 95, 256-273.

Fennema, E., & Sherman, J. (1977, Winter). Sex-related differences in mathematics achievement, spatial visualization, and affective factors. American Educational Research Journal, 14(1), 51-71.

Firkin, J. (1984). Computers in schools. Victorian Institute of Secondary Education. Melbourne, Australia.

Fisher, A., Margolis, J., & Miller, F. (1997). Undergraduate women in computer science: Experience, motivation and culture. SIGCSE Bulletin, 29(1), 106-110.

Frenkel, K.A. (1990, November). Women and computing. Communications of the ACM, 33(11), 34-46.

Greening, T. (1999, March). Gender stereotyping in a computer science course. SIGCSE Bulletin, 31(1), 203-207.

Harrington, S.M. (1990). Barriers to women in undergraduate computer science: The effects of the computer environment on the success and continuance of female students. Unpublished doctoral dissertation, University of Oregon.

Hess, R.D., & Miura, I.T. (1985). Gender differences in enrollment in computer camps and classes. Sex Roles, 13, 193-203.

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

Jagacinski, C.M., LeBold, W.K., & Salvendy, G. (1988). Gender differences in persistence in computer-related fields. Journal of Educational Computing Research, 4(2), 185-202.

Kerner, J., & Vargas, K. (1994). Women and computers: What can we learn from science? SIGSCE Bulletin, 26(2), 53-56.

Kersteen, Z.A., Linn, M.C., Clancey, M., & Hardyck, C. (1988). Previous experience and the learning of computer programming: The computer helps those who help themselves. Journal of Educational Computing Research, 4(3).

Kiesler, S., Sproull, L., & Eccles, J. (1983, March). Second class citizens? Psychology Today, 41-48.

Klawe, M., & Leveson, N. (1995). Women in computing: Where are we now? Communications of the ACM, 38(1), 29-44.

Lent, R.W., Brown, S.D., & Gore, P.A.,Jr. (1997). Discriminant and predictive validity of academic self-concept, academic self-efficacy, and mathematics-specific self-efficacy. Journal of Counseling Psychology, 44(3), 307-315.

Linn, M.C. (1985). Fostering equitable consequences from computer learning environments. Sex Roles, 13, 229-240.

Liu, M., & Blanc, L. (1996). On the retention of female computer science students. SIGCSE Bulletin, 28(1), 32-36.

Lopez, F.G., Brown, S.D., Lent, R.W., & Gore, P.A., Jr. (1997). Role of social-cognitive expectations in high school students' mathematics-related interest and performance. Journal of Counseling Psychology, 44(1), 44-52.

Maeher, M.L. (1988). On doing well in science: Why Johnny no longer excels; why Sarah never did. In S.G. Paris, G.M. Olsen, & H.W. Stevenson (Eds.), Learning and Motivation in the Classroom (Chapter 8). London: Lawrence Erlbaum Associates.

Mandinach, E., & Fisher, C.W. (1985). Individual differences and acquisition of computer programming skill. ACCCEL Report, Lawrence Hall of Science, University of California, Berkeley.

Miura, I.T. (1987). The relationship of computer self-efficacy expectations to computer interest and course enrollment in college. Sex Roles, 16(5), 303-311.

Morahan-Martin, J., Olinsky, A., & Schumacher, P. (1992). Gender differences in computer experience, skills, and attitudes among incoming college students. Collegiate Microcomputer, 10(1), 1-8.

Moses, L.E. (1993). Our computer science classrooms: Are they "friendly" to female students? SIGCSE, 25(3), 3-12.

Murphy, C.A., Coover, D., & Owen, S.V. (1989). Development and validation of the computer self-efficacy scale. Educational and Psychological Measurement, 49, 893-899.

Pearl, A., Pollack, M.E., Riskin, E., Thomas, B., Wolf, E., & Wu, A. (1990). Becoming a computer scientist. Communications of the ACM, 33(11), 47-57.

Projections of education statistics to 2008.(1998). National Center for Education Statistics. Washington, D.C.: U.S. Department of Education.

Ramalingam, V., & Wiedenbeck, S. (1998). Development and validation of scores on a computer programming self-efficacy scale and group analyses of novice programmer self-efficacy. Journal of Educational Computing Research, 19(4), 367-381.

Reinecke, I. (1983). Microcomputers. Melbourne: Penguin.

Rodger, S.H., & Walker, E.L. (1996, March). Activities to attract high school girls to computer science. SIGCSE Bulletin: The Proceedings of the Twenty-seventh SIGCSE Technical Symposium on Computer Science Education, 28(1), 373-377.

Rodger, S.H., & Walker, E.L. (1996, June). PipeLINK: Connecting women and girls in the computer science pipeline. Proceedings of the National Educational Computing Conference (NECC '96), 378-384.

Sackrowitz, M.G., & Parelius, A.P. (1996). An unlevel playing field: Women in the introductory computer science courses. SIGCSE Bulletin, 28(1), 37-41.

Sanders, J., & Stone, A. (1986). The neuter computer: Computers for girls and boys. (Available from the Women's Action Alliance, New York)

Scragg, G., & Smith, J. (1998, March). A study of barriers to women in undergraduate computer science. SIGCSE Bulletin, 30(1), 82-86.

Taylor, H., & Mounfield, L. (1991). An analysis of success factors in college computer science: High school methodology is a key element. Journal of Research on Computing in Education, 24(2), 240-245.

Taylor, H., & Mounfield, L. (1994). Exploration of the relationship between prior computing experience and gender on success in college computer science. Journal of Educational Computing Research, 11(4), 291-306.

The condition of education. (1997). National Center for Education Statistics. Washington, D.C.: U.S. Department of Education.

The condition of education. (1998). National Center for Education Statistics. Washington, D.C.: U.S. Department of Education.

Turkle, S., & Papert, S. (1992). Epistemological pluralism and the reevaluation of the concrete. Journal of Mathematical Behavior, 11, 3-33.

APPENDICES

Appendix A

Computer Programming Self-Efficacy Scale

Computer Programming Self-Efficacy Scale
Ramalingam & Wiedenbeck

Rate your confidence in doing the following C++ programming related tasks by placing a circle around the choice using a scale of 1 (not at all confident) to 7 (absolutely confident). If a specific term or task is totally unfamiliar to you, please circle 1.

	Not at all confident	Mostly not confident	Slightly confident	50/50	Fairly confident	Mostly confident	Absolutely confident
1. I can write syntactically correct C++ statements.	1	2	3	4	5	6	7
2. I understand the language structure of C++ and the usage of the reserved words.	1	2	3	4	5	6	7
3. I can write logically correct blocks of code using C++.	1	2	3	4	5	6	7
4. I can write a C++ program that displays a greetings message.	1	2	3	4	5	6	7
5. I can write a C++ program that computes the average of three values.	1	2	3	4	5	6	7
6. I can write a C++ program that computes the average of any given number of values.	1	2	3	4	5	6	7
7. I can use built-in functions that are available in the various C++ libraries.	1	2	3	4	5	6	7
8. I can build my own C++ libraries.	1	2	3	4	5	6	7
9. I can write a small C++ program given a small problem that is familiar to me.	1	2	3	4	5	6	7
10. I can write a reasonably sized C++ program that can solve a problem that is only vaguely familiar to me.	1	2	3	4	5	6	7
11. I can write a long and complex C++ program to solve any given problem as long as the specifications are clearly defined.	1	2	3	4	5	6	7
12. I can organize and design my program in a modular manner.	1	2	3	4	5	6	7

	Not at all confident	Mostly not confident	Slightly confident	50/50	Fairly confident	Mostly confident	Absolutely confident
13. I understand the object-oriented paradigm.	1	2	3	4	5	6	7
14. I can identify the objects in the problem domain and declare, define, and use them.	1	2	3	4	5	6	7
15. I can make use of a pre-written function, given a clearly labeled declaration of the function.	1	2	3	4	5	6	7
16. I can make use of a class that is already defined, given a clearly labeled declaration of the class.	1	2	3	4	5	6	7
17. I can debug (correct all the errors) a long and complex program that I had written and make it work.	1	2	3	4	5	6	7
18. I can comprehend a long, complex multi-file program.	1	2	3	4	5	6	7
19. I could complete a programming project if someone showed me how to solve the problem first.	1	2	3	4	5	6	7
20. I could complete a programming project if I had only the language reference manual for help.	1	2	3	4	5	6	7
21. I could complete a programming project if I could call someone for help if I got stuck.	1	2	3	4	5	6	7
22. I could complete a programming project once someone else helped me get started.	1	2	3	4	5	6	7
23. I could complete a programming project if I had a lot of time to complete the program.	1	2	3	4	5	6	7
24. I could complete a programming project if I had just the built-in help facility for assistance.	1	2	3	4	5	6	7
25. I could find ways of overcoming the problem if I got stuck at a point while working on a programming project.	1	2	3	4	5	6	7

	Not at all confident	Mostly not confident	Slightly confident	50/50	Fairly confident	Mostly confident	Absolutely confident
26. I could come up with a suitable strategy for a given programming project in a short time.	1	2	3	4	5	6	7
27. I could manage my time efficiently if I had a pressing deadline on a programming project.	1	2	3	4	5	6	7
28. I could mentally trace through the execution of a long, complex, multi-file program given to me.	1	2	3	4	5	6	7
29. I could rewrite lengthy confusing portions of code to be more readable and clear.	1	2	3	4	5	6	7
30. I can find a way to concentrate on my program, even when there were many distractions around me.	1	2	3	4	5	6	7
31. I can find ways of motivating myself to program, even if the problem area was of no interest to me.	1	2	3	4	5	6	7
32. I could write a program that someone else could comprehend and add features to at a later date.	1	2	3	4	5	6	7

Information about this instrument can be reviewed in the Journal of Educational Computing Research, Vol. 19(4), pp. 367-381, 1998.

Appendix B

Questionnaire

Questionnaire CS 202

No. _____

Thank you for your willingness to participate in the study of an introductory computer science course and the factors pertinent to success in this type of course. Your individual responses to this questionnaire will be kept confidential, will be kept in a data base by an identification number (no names), and will be destroyed after the collective data is gathered and analyzed. Please be as honest and accurate as you possibly can so that the data given to the researcher can be relied upon to analyze the computer science discipline.

Demographic & Background Data:

Gender: Male or Female (Circle one)

Classification: Freshman Sophomore Junior Senior Other (Circle one)

High School Math Courses: Indicate how many semesters of math you took in high school. _____

1. **Previous Programming Experience**: Place a check mark by each statement that applies to you.

____ I took a computer programming course prior to this class (high school, computer camp, community college, etc.)

If you checked having a prior computer programming course, please check any of the statements that apply to that course:

____ The course emphasized using subprograms and structured approaches to the design of the programs (no GO-TO statements, use of while loops, top-down design, etc.)
____ BASIC (QBasic) programming language
____ PASCAL or ADA programming language
____ COBOL programming language
____ C, C++ programming language
____ Other programming language...what was it? ________________

____ I learned to program on my own by reading books, talking with others, etc. (You should check this if you participated in self-initiated programming without the help of a formal class. Even if you took a computer programming class, you may have written programs that went above and beyond the scope of the class "on your own," so put a check mark at this item.)

2. **Previous Non-Programming Experience**: Tell how many average hours per week you spent on the following items **prior** to taking this class (use 0 if none, ½ if 30 minutes, etc.).

____ Number of hours per week I surfed the web for information I needed, used e-mail, chat-rooms, and/or on-line discussion groups.

____ Number of hours per week I played games (either on-line games or individual games) on the computer.

____ Number of hours per week I used application software such as word processing, spreadsheets, or databases, etc.

3. Encouragement to Study Computer Science:
Did someone give you encouragement and/or the desire to study computer science either by words or role modeling? Yes No

If yes, put a checkmark by the **most important person** that gave you this encouragement and/or role modeling. (Choose only one)
____ father/step-father
____ mother/step-mother
____ other family member
____ teacher, guidance counselor, high school personnel
____ friend / peer
____ professional in the computer field (does not have to be an acquaintance)
____ college recruitment

4. Work / Study Style Preference:

When given a programming assignment or when a test is coming up which method would you prefer? (check one)

____ individual / competitive work or study

____ cooperative / group work or study

Read each of the following items completely and decide which of the five statements most accurately describes your feelings. Circle one number for each item.

5. DIFFICULTY OF THIS CLASS:

5 very easy; I find the concepts easy to understand and the assignments simple to complete on my own.

4 somewhat easy; Most of the time I understand the concepts and can complete assignments without much help.

3 neutral; I find some concepts easy and some difficult to understand. There are times when I need help completing an assignment and other times when I do fine alone.

2 somewhat difficult; The concepts are difficult to understand. I need help most of the time with assignments.

1 very difficult; It seems no matter what I do, I just cannot grasp the concepts. I would not be able to complete assignments without help.

6. WRITING COMPUTER PROGRAMS:

5 very easy; I can design the logic of the program and code the program without help and with no real problems.

4 somewhat easy; Most of the time I can design the logic and write the code with very little help.

3 neutral; I find some programs easy and some difficult to write. There are times when I need help designing the logic and/or writing the code and other times when I do fine alone.

2 somewhat difficult; The programs are difficult to write; I need help most of the time with designing the logic and/or writing the code.

1 very difficult; It seems no matter what I do, I just cannot see how to design the logic and code the program. Without help I would not be able to complete the programs.

7. WORKING AT THE COMPUTER ON A PROGRAMMING ASSIGNMENT:

5 very relaxed; I am completely at ease working at the computer on my assignment. I know if something does not go as planned I will be able to work the problem out.

4 fairly relaxed; I am pretty much at ease working at the computer on my programming assignment. I know if something goes wrong I can get help.

3 neutral; Although I do not feel really nervous working at the computer to complete my programming assignment, I also am not really relaxed either. I work at the computer on my programs like I would on homework in other courses.

2 somewhat anxious; I am uneasy about completing my assignments and working with the computer. I am aware that problems might occur that would make my program not work.

1 very anxious; I am very aware of nervousness. I have tightness in my stomach or clammy hands or other symptoms of being nervous. I am afraid I will not know what to do if I do something wrong.

8. MY LEVEL OF UNDERSTADING IN THIS CLASS:

5 Overall, my level of understanding is higher than any of the others in the class.

4 Overall, my level of understanding is higher than most of the others in the class.

3 Overall, my level of understanding is probably about the average of the class.

2 Overall, my level of understanding is lower than most of the others in the class.

1 Overall, my level of understanding is lower than any of the others in the class.

9. PARTICIPATION IN CLASS:

5 I am very comfortable asking questions and enjoy volunteering to answer questions in class.

4 I will sometimes ask a question in class and sometimes will volunteer to answer a question.

3 I usually do not ask questions in class but will answer a question when called on by the professor.

2 I refrain from asking questions in class and prefer that the professor not ask me questions in class.

1 I will not ask questions in class and sit in fear that I will be called on to answer a question in class.

10. PARTICIPATION DURING LAB:

5 I am very comfortable asking questions and enjoy volunteering to answer questions in LAB.

4 I will sometimes ask a question in LAB and sometimes will volunteer to answer a question..

3 I usually do not ask questions in LAB but will answer a question when called on by the professor.

2 I refrain from asking questions in LAB and prefer that the professor not ask me questions in LAB.

1 I will not ask questions in LAB and sit in fear that I will be called on to answer a question in LAB.

11. USING OFFICE HOURS for help with assignments or course content:

5 I am very comfortable going to the professor's office to ask questions about the class and/or assignments.

4 I am fairly comfortable going to the professor's office to ask questions although I will usually try to get my answer from someone else first.

3 I feel indifferent about going to the professor's office to ask questions. I will do it if I need to but probably will use other means.

2 I am uncomfortable enough that I refrain from going to the professor's office to ask questions unless circumstances are serious.

1 I feel too uncomfortable going to the professor's office to ask questions. I will just let my question go unanswered before I will go.

12. USING E-mail to professor for help with assignments or course content:

5 I am very comfortable e-mailing the professor to ask questions about the class and/or assignments.

4 I am fairly comfortable e-mailing the professor to ask questions although I will usually try to get my answer from someone else first.

3 I feel indifferent about e-mailing the professor to ask questions. I will do it if I need to but probably will use other means.

2 I am uncomfortable enough that I refrain from e-mailing the professor to ask questions unless circumstances are serious.

1 I feel too uncomfortable e-mailing the professor to ask questions. I will just let my question go unanswered before I will use e-mail.

13. Exam grade:

Please rate your feelings about your grade on your exam by checking **one** answer:

____I am **happy** with my grade on the first exam – **go to question 14, skip question 15.**

____I am **unhappy** with my grade on the first exam – **skip question 14, go to question 15.**

14. Reasons for Exam Grade: Answer this question only if you were **HAPPY** with your grade.

Please rate the importance of each possible reason by using the scale below. Just circle the number.

	Definitely not a reason	Not a very important reason	Neutral	Somewhat important reason	A very important reason
The exam was easy.	1	2	3	4	5
I was lucky. *(studied just the right material, guessing paid off, etc.)*	1	2	3	4	5
I worked very hard studying for the test.	1	2	3	4	5
My ability in this area is high.	1	2	3	4	5

15. Reasons for Exam Grade: Answer this question only if you were UNHAPPY with your grade.

Please rate the importance of each possible reason by using the scale below. Just circle the number.

	Definitely not a reason	Not a very important reason	Neutral	Somewhat important reason	A very important reason
The exam was difficult.	1	2	3	4	5
I was unlucky. *(studied the wrong material, got sick just before the test, etc.)*	1	2	3	4	5
I didn't work hard enough studying.	1	2	3	4	5
My ability in this area is low.	1	2	3	4	5

Appendix C

Human Subjects Approval Form

SIUC HSC FORM A

REQUEST FOR APPROVAL OF RESEARCH ACTIVITIES INVOLVING HUMAN SUBJECTS

This approval is valid for one (1) year from the approval date. Researchers must request a renewal to continue the research after that date. This approval form must be included in all Master's theses/research papers and Doctoral dissertations involving human subjects to be submitted to the Graduate School.

PROJECT TITLE: Contributing Factors to Success in Computer Science

CERTIFICATION STATEMENT:

In making this application, I(we) certify that I(we) have read and understand the University's policies and procedures governing research activities involving human subjects, and that I(we) shall comply with the letter and spirit of those policies. I(we) further acknowledge my(our) obligation to (1) accept responsibility for the research described, including work by students under my(our) direction, (2) obtain written approval from the Human Subjects Committee of any changes from the originally approved protocol **BEFORE** making those changes, (3) retain signed informed consent forms, in a secure location separate from the data, for at least **three** years after the completion of the research, and (4) report immediately all adverse effects of the study on the subjects to the Chairperson of the Human Subjects Committee, Carbondale, Illinois, (618) 453-4533, and to the Director of the Office of Research Development and Administration, Southern Illinois University at Carbondale, (618) 453-4531.

Brenda Wilson

Brenda Wilson — 2-15-00

RESEARCHER(S) or PROJECT DIRECTOR(S) — DATE

Please print or type name below signature

Sharon Shrock — 2-15-00

RESEARCHER'S ADVISOR (required for all student projects) — DATE

Please print or type name below signature

The request submitted by the above researcher(s) was approved by the SIUC Human Subjects Committee.

Robert C Koethe — 2-25-00

CHAIRPERSON, SOUTHERN ILLINOIS UNIVERSITY HUMAN SUBJECTS COMMITTEE — DATE

Appendix D

Cover Letter and Consent Form

The attached questionnaire & Computer Programming Self-Efficacy Scale you are being asked to complete are parts of a study of first semester computer science students at this university. The study seeks to discover contributing factors for success in a computer science course. The study's design and purpose are based on research reported in scholarly literature. The researcher is associated with Southern Illinois University – Carbondale. The safeguards related to your rights as a subject of the study have been approved by the Human Subjects Committee at Southern Illinois University – Carbondale. This study includes the collection of data from four sources: (1) the questionnaire, (2) the Computer Programming Self-Efficacy Scale, (3) your grade in this class, and (4) your ACT scores.

You have been asked to participate in this study because you are or have been enrolled in an introductory computer science course. Before and during the completion of the questionnaire, you should be aware of your rights as a participant. They are listed below:

- You may decide not to participate in the study.
- You may decide to stop participation after the study is underway.
- Completion and return of this questionnaire indicates voluntary consent to participate in the study. The professor of your class will supply to the researcher the grade of each participant in the study.
- The information you provide will be kept confidential. All reasonable steps will be taken to protect your identity. An identification number (not social security number) will be assigned to each participant. The database that stores the data will only contain identification numbers (not names) of participants in the study. The researcher will be the only one who has access to the code list of identification numbers and names. (This is necessary to be able to connect the data from the questionnaire to the grade in the class and ACT scores for each participant.) Upon completion of the study, the code list with participant's names will be destroyed.

Thank you for participating in this study by completing the questionnaire. Your help is greatly appreciated.

If you have any questions about this study, you may contact:

Brenda Wilson
Murray State University
(270) 762-6210

Dr. Sharon Shrock
Southern Illinois University -- Carbondale
(618) 453-4218

Human Subjects Committee Review Board
Southern Illinois University – Carbondale
Carbondale, Illinois 62901
(618) 453-4543

CS 202 Consent Form

Project Title: Contributing Factors to Success in Computer Science

Researcher: Brenda Wilson
Phone: 270-762-6210

I agree to be a participant in this study. I give my permission to Brenda Wilson to obtain my ACT math and verbal scores from Southern Illinois University and to obtain my grade in this class from my professor.

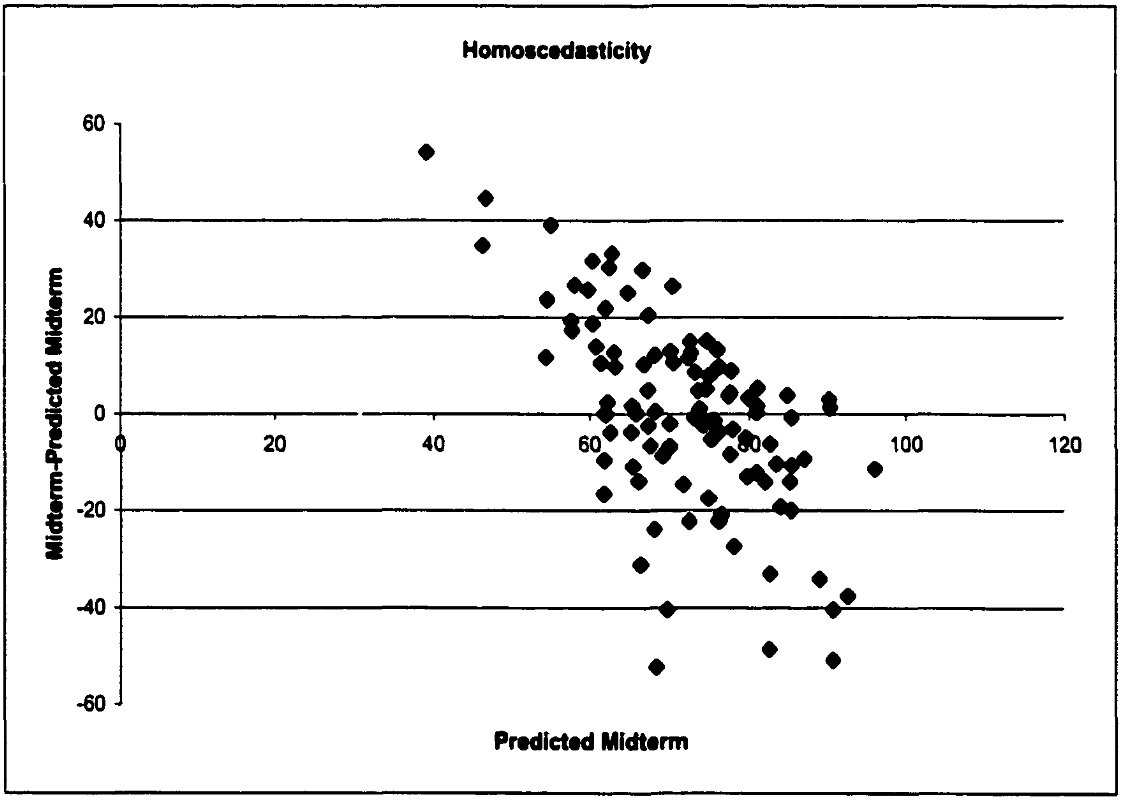
____
Print your name

________________________________ ________________
Your Signature Date

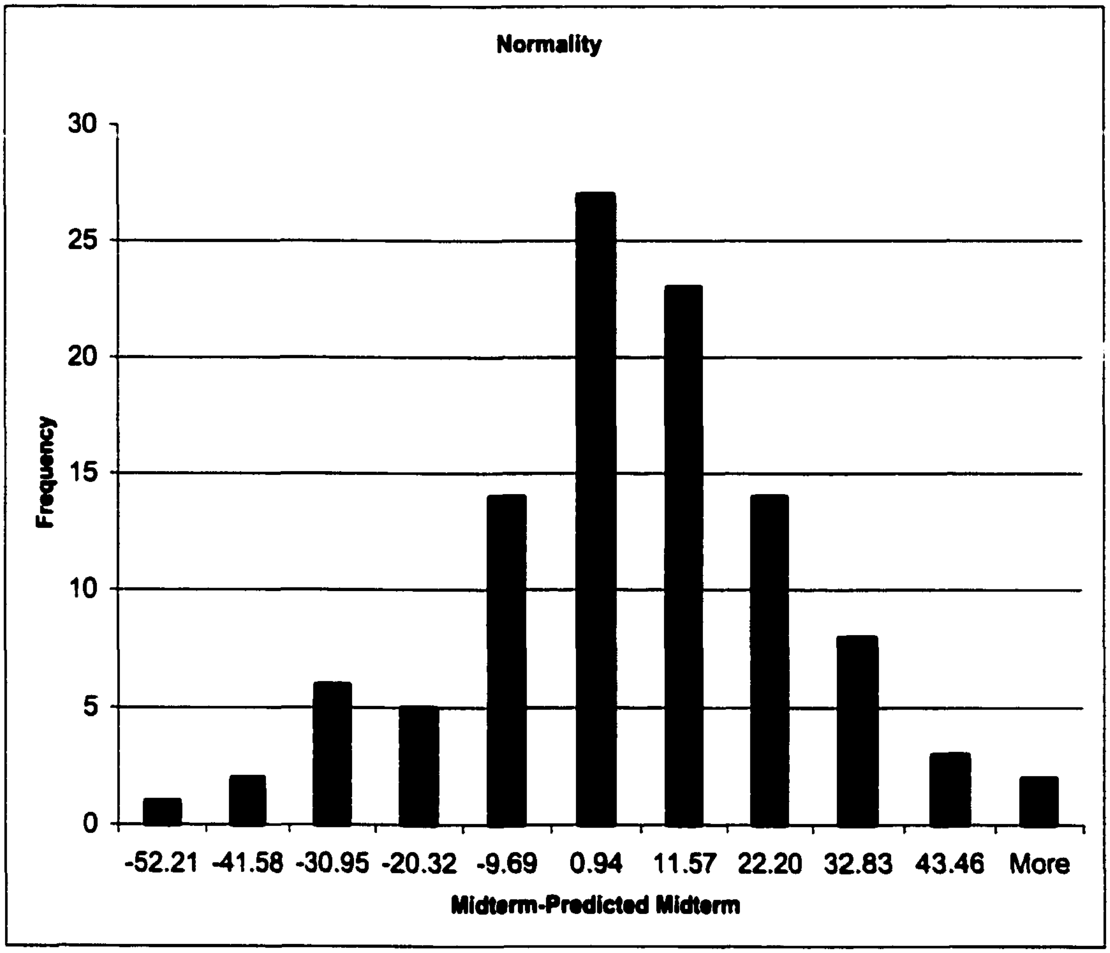
Appendix E

Scatter Plot and Histogram

Scatter Plot



Histogram


Normality
Frequency
30
25
20
15
10
5
0
-52.21
-41.58
-30.95
-20.32
-9.69
0.94
11.57
22.20
32.83
43.46
More
Midterm-Predicted Midterm

VITA

Graduate School
Southern Illinois University

Brenda Cantwell Wilson — Date of Birth: August 28, 1948

595 East Tucker Road, Mayfield, Kentucky 42066

Harding University, Searcy, Arkansas
Bachelor of Science, Mathematics, May 1969

University of Texas at Austin
Masters of Arts in Mathematics/Computer Science Education, Dec 1982

Special Honors and Awards:

1980-1982	National Science Foundation "Women in Science" Accelerated Computer Science Program, University of Texas at Austin
1987	Lucille Speakman Excellence in Teaching Award, Westark College, Fort Smith, Arkansas
1988	National Excellence in Teaching Award, NISOD, University of Texas at Austin
1994	Distinguished Service Award, Project IMPAC (Instructional Microcomputer Project for Arkansas Classrooms), Arkansas State Department of Education
1997 & 1998	Outstanding Teacher Award, Department of Computer Science and Information Systems, Murray State University

Dissertation Title:
Contributing Factors to Success in Computer Science: A Study of Gender Differences

Major Professor: Dr. Sharon A. Shrock

Publications:

"Hardware and Software Needs of Businesses in the Western Kentucky Region", with Lila Waldman, KBEA Journal, Spring 1997.

"Survey of Computer Usage", with Lila Waldman, OSRA Journal, Spring 1997.