

AN ABSTRACT OF THE THESIS OF

Tai-Sheng Fan for the degree of Doctor of Philosophy in
Science Education presented on April 5, 1996.

Title: Prediction of Academic Achievement for College
Computer Science Majors in the Republic of China

Abstract approved: **Redacted for Privacy**
Margaret L. Niess

The purpose of this research was to determine whether student academic achievement in college computer science programs in the Republic of China (ROC) could be predicted by factors reported to be effective in US studies. The relationship between these factors and course performance in computer science programs was examined. Gender differences were also interrogated.

Sophomore, junior, and senior students enrolled in five universities offering computer science programs in the ROC constituted the population. A researcher-designed questionnaire was used to collect background information. Validity and reliability issues were addressed by the conduct of validity assessment, questionnaire pilot testing, and interviews with selected pilot test subjects. Scores from the College Entrance Examination (CEE) and college computer science courses were accessed through university registrar's

offices. A total of 940 questionnaires were collected, representing more than 81% of the population.

From data analysis, the predictive powers of CEE test scores in relation to subsequent college performance appeared to be limited. The CEE math component was negatively correlated to performance in college computer science programs. The positive relation of math ability to academic achievement in complete computer science programs was confirmed. High school overall achievement as well as math course averages were identified as effective performance predictors for college computer science programs. Prior computer experience showed no conclusive relationship to subsequent performance in college computer science courses.

The close relationship between performance in beginning computer science courses and performance in complete computer science programs was validated. Significant linear prediction models with limited predictive powers (R^2 ranged from 0.19 to 0.30) were generated for overall performance, but not for introductory computer science course performance. Model predictive powers were significantly improved (R^2 range from 0.59 to 0.63) when performance in introductory computer science courses was included in the models. Significant gender differences were not found for CEE performance, prior computer experience, and prediction models. However, female subjects outperformed male counterparts in course performance at both the high school and college levels.

Prediction of Academic Achievement for College
Computer Science Majors in the Republic of China

by

Tai-Sheng Fan

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Completed April 5, 1996
Commencement June 1996

©copyright by Tai-Sheng Fan
April 5, 1996
All Rights Reserved

Doctor of Philosophy thesis of Tai-Sheng Fan presented on
April 5, 1996

Approved:

Redacted for Privacy

Major Professor, representing Science Education

Redacted for Privacy

Chair of Department of Science and Mathematics Education

Redacted for Privacy

Dean of Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Redacted for Privacy

Tai-Sheng Fan, Author

Acknowledgments

When I decided to pursue further graduate study, a Ph.D. (also interpreted as PHinally Done) degree was seemingly unreachable. However, it is finally done! If I have accomplished anything during my doctoral study at Oregon State University, my deepest gratitude should be given to all who had rendered their kind help to me during the past three years.

Most of my research work could not be accomplished without the immeasurable advice of my major professor, Dr. Margaret L. Niess. With her patience and guidance, I was inspired and given the confidence to tackle the problem in my own way. My thanks are also given to Dr. Prasad Tadepalli, my minor professor for my program in Computer Science. Through his assistance, I have learned to appreciate the wonders of computer science and artificial intelligence from various perspectives.

My appreciation is also given to Dr. Norman G. Lederman for helping me to conceive a deeper meaning of learning beyond the traditional view in his research methodology class. I would also like to extend my gratitude to Dr. Dianne K. Erickson for her enthusiasm in teaching and for kind encouragement in helping me to express my thoughts clearly. A special thanks is given to Dr. Daryl Richardson, for his valuable suggestions both in my research proposal and thesis writing.

Drs. Kathy Howell, Janet Scholz, Mark Latz and Jinmmei Kuo have also been very helpful to me in clarifying several difficulties I encountered in the field of science education. Special appreciation is given to Emma Lin, my wonderful classmate in the doctoral program, and to Dr. Bill McMechan for their assistance in completing assembly of the library copy of my dissertation. Help from my colleagues at Department of Management Information Systems at the National Pingtung Polytechnic Institute is also appreciated. In particular, I must acknowledge Dr. Yaw-Huei Chen and Mr. Jing-Hwa Jeng for their enthusiastic assistance while I was writing the dissertation.

In addition, I would also like to say thanks to my dearest brothers and sisters in Christ, for their constant prayers throughout the whole period of my graduate study in the United States. A special thanks is also given to all the participants in the research and to those who helped to make this study possible.

The greatest indebtedness is to my beloved wife, Jill. She provided support, both financially and spiritually, for my graduate study through her endless encouragement and continuous prayers to our Heavenly Father. In conclusion, all the glory and honor should be dedicated to God, the Almighty One. With His boundless love and comfortness, even the darkest night will turn to sunshine bright. Praise the LORD!!

TABLE OF CONTENTS

<u>Chapter</u>	<u>Page</u>
I THE PROBLEM	1
Introduction	1
Statement of the Problem	7
Significance of the Study	13
Definition of Terms	16
II REVIEW OF THE LITERATURE	20
Introduction	20
Achievement Prediction From Results of Intro- ductory Computer Science Courses	22
Achievement Prediction From Results of Advanced Computer Science Courses	74
Summary	87
III RESEARCH DESIGN AND METHODOLOGY	94
Introduction	94
Higher Education in the Republic of China	96
Computer-Related Programs in the Republic of China .	98
Subjects	99
Instrument Development	100
Validation Process and Reliability Establishment	102
Interviews	103
Data Collection	105
Questionnaire Administration	105
Informed Consent	106
Registrar Records	107
Data Encoding	108
Data Analysis	109

TABLE OF CONTENTS (Continued)

<u>Chapter</u>	<u>Page</u>
Research Hypotheses	110
Statistical Analyses	112
IV RESULTS OF THE STUDY	119
Introduction	119
Results of Statistical Analyses	122
CEE Scores	123
Math Ability	128
Prior Computer Experience	133
Overall High School Performance	139
Introductory Computer Science Courses	141
Prediction Models	143
Gender Differences	152
Summary	155
V DISCUSSION AND CONCLUSIONS	158
Introduction	158
Discussion of the Results	160
CEE Scores	160
Math Ability	165
Prior Computer Experience	168
Overall High School Performance	170
Introductory Computer Science Courses	171
Prediction Models	172
Gender Differences	174
Limitations of the Research	176
Implications for Computer Science Education	178
Recommendations for Future Research	183
REFERENCES	187
APPENDICES	192
APPENDIX A Research Questionnaire	193
APPENDIX B Evaluation for Content Validity	199
APPENDIX C Interview Questions	203
APPENDIX D Informed Consent Form	204

LIST OF FIGURES

<u>Figure</u>		<u>Page</u>
1	Plot of CS-INTRO Model for CS-MAJOR Prediction for Senior Classes by University Enrolled.....	148
2	Plot of CS-INTRO Model for CS-MAJOR Prediction for Combined Sophomore Group.....	149
3	Plot of CS-INTRO Model for CS-MAJOR Prediction for Combined Junior Group.....	150
4	Plot of CS-INTRO Model for CS-MAJOR Prediction for Combined Senior Group.....	151

LIST OF TABLES

<u>Table</u>	<u>Page</u>
1 Research Hypotheses and Data Sources for Research Variables	113
2 Subjects by Gender, University and Class Participating in the Study	121
3 Correlation between CEE Scores and Scores in the Introductory Computer Science Courses (CS-INTRO)	124
4 Correlation between CEE Scores and Overall Performance in Computer Science Programs (CS-MAJOR)	125
5 Average Scores for Math-Related Variables and College-Performance Variables	129
6 Correlation between Math Ability and Performance in College Computer Science Programs	130
7 Correlation between Prior Computer Experience and College Performance	134
8 Average Score Differences for CS-INTRO and CS-MAJOR Between Students with Different Amounts of Prior Computer Experience	137
9 Correlation Between Overall High School Performance (HS-AVG) and CEE-TOTAL and College Performance (CS-INTRO and CS-MAJOR)	140
10 Correlation Between CS-INTRO and Overall Course Performance (CS-MAJOR and CS-NOBCC)	142
11 Regression Models for CS-MAJOR Prediction	145
12 Results of Analysis for Gender Differences	153

Prediction of Academic Achievement for College Computer Science Majors in the Republic of China

CHAPTER I THE PROBLEM

Introduction

Since computers have come into wide use in various industries, and increased numbers of high-paid-salary positions have become available in computer-related fields, computer science has evolved into an attractive field of study. Consequently, numerous universities have experienced problems of oversubscription wherein the number of eligible applicants has exceeded the number of available first-year positions in computer science programs. Faced with limited faculty and staff positions as well as computer facilities, processes for limiting the number of students have become necessary.

Recent research has indicated that the drop-out rate for the computer science major has been increasing (Campbell & McCabe, 1984; Sorge & Wark, 1984). Additional studies have found a similar trend in beginning computer science courses (Greer, 1986; Konvalina, Wileman & Stephens, 1983b; Taylor & Mounfield, 1989). Thus, Campbell and McCabe (1984) suggested that if college counselors had access to more specific information regarding the predictive factors for student success in computer science, they

might be able to more effectively advise students concerning the reality of pursuing a computer science major. With limited resources and high attrition rates, it has become increasingly important to identify more effective criteria for the classification of those students who are likely to succeed in the computer science major, thus making the best use of available computing resources.

Prediction of student academic achievement is not a new area of educational research, and was extended to the new field of education in computer science by the late 1960s. Subsequently, considerable research has been conducted regarding performance prediction in computer science. However, most of the research conducted at the college level has focused upon the prediction of student performance for a single course, typically an introductory computer science course. Taylor and Mounfield (1989) observed that students who performed well in an introductory computer science course usually performed well in a computer science program. Moreover, a number of researchers have indicated that introductory computer science courses serve as a gateway to computer science majors, and that only those students who successfully complete these courses should be admitted into program majors (Kersteen, Linn, Clancy, & Hardyck, 1988).

Among the studies investigating performance prediction for introductory computer science courses, it has been reported that mathematical ability was strongly associated

with the performance in college entry-level computer science courses (Butcher & Muth, 1985; Dey & Mand, 1986; Dixon, 1987; Goodwin & Wilkes, 1986; Oman, 1986; Renk, 1986). In addition, it was also determined that standardized aptitude tests, such as the Scholastic Aptitude Test (SAT), can be used to predict performance in college introductory computer science courses. Still other studies have suggested that previous computer science experience contributes to achievement in beginning computer science courses at the college level (Nowaczyk, Connor, Stevenson, & Hare, 1986; Taylor & Mounfield, 1991).

However, for computer science programs, little research concerned with achievement predictions beyond the level of introductory computer science courses has been completed. Butcher and Muth (1985) argued that student cumulative grade-point averages (GPA), and not simply the grades earned in single computer science courses, should provide better measures of academic success. Shoemaker (1986) used preadmission measures (e.g., the SAT and high school GPA) for the prediction of student cumulative GPA and the major GPA for the engineering and computer science majors, and found that both high school GPA and College Board Mathematics Achievement Test results were reliable predictors.

In addition, though not focused upon computer science majors, some researchers found that grade prediction for the college level was not stable for individual class

years, and suggested the use of independent GPA for each semester as a measure of academic success (Hsu & Lin, 1982; Humphreys, 1968; Humphreys & Taber, 1973; Lunneborg & Lunneborg, 1970). Thus, the conduct of additional empirical research will be required to justify the assumption that good performance in the introductory courses also assures subsequent success in computer science programs.

Another computer science education research direction has been concerned with the disproportionately low number of females in the profession. A number of studies have indicated that women either dropped or terminated computer science training at earlier stages than did men (Campbell & McCabe, 1984; Jagacinski, LeBold, & Salvendy, 1988; Windall, 1988). As a result, researchers have attempted to determine why women tend to avoid computer science courses and do not choose computer science as a college major. Some research has indicated that women had lower perceptions of their ability and lower self-confidence than men (Clarke & Chambers, 1989; Teague & Clarke, 1991; Ware, Steckler, & Leserman, 1985). Clarke and Chambers (1989) also found significant gender attrition differences with respect to academic success or failure.

A lack of self-initiated prior computer science experience, especially programming experience, was also reported as a possible source of frustration and discouragement among women (Clarke & Chambers, 1989; Sproull, Zubrow, & Kiesler, 1986). Kersteen et al. (1988) suggested that it

was this lack of prior experience that made the factor of "prior computer science experience" effective only for men in the prediction of student performance in introductory computer courses. Jagacinski et al. (1988) reported that the students who did not do well in their core computer science courses were more inclined to change their major. In their study, one-third of the nonpersisters surveyed indicated that discouraging experiences in college introductory courses were the primary reason for changing majors. Lips and Temple (1990) also indicated that prior computer science experience played a stronger and more positive role in the decision of women to major in computer science than it did for men.

None of the research described above was conducted in the Republic of China (ROC). The only related study among subjects from this setting was focused upon finding the relationship between attitudes toward computers and performance in a computer science course, and was conducted at the high school level (Tsai, 1984). Due to the dramatic differences in culture as well as educational systems, the findings from studies in other western countries may not be adaptable to the ROC.

To be admitted to a four-year college or university, ROC high school graduates must pass a competitive College Entrance Examination (CEE) held nationwide annually. According to Hwang (1990), approximately 37% of all high school graduates are selected annually to continue their

education. Students are assigned to specific departments based solely upon their total score of CEE rankings. High school GPA and letters of recommendation are not taken into consideration for college admissions.

The predictive power of the CEE for academic achievement at college has become a major concern of educational research in the ROC during the last two decades. In general, Tsong et al. (1977) indicated that CEE scores were better measures of high school achievements than they were predictors of overall college performance. Hsu and Lin (1982) also reported low predictive CEE powers for college performance as measured by average scores for individual semesters. However, both studies considered the predictability of college performance in general, rather than specific programs, and the diversity of different academic disciplines was not taken into account. If students from departments with similar requirements, such as in relation to computer science programs, were selected for study, then different results may be obtained.

According to Chen (1988), there will be a shortage of from 60,000 to 110,000 information professionals in the ROC by the year 2000 if the needs of an information-based society are to be met. As a result of this study, it was proposed that the rates of productivity as well as the numbers of information professionals should be increased. Wang (1989) also called attention to this shortage of information professionals. Thus, whether potentially successful

computer science majors can be selected through application of the CEE is a critical factor in measuring the ability to fulfill the needs of the ROC as an information-based society. As a result, a study focusing upon performance predictions for college computer science majors in the ROC is needed.

Statement of the Problem

The purpose of the research is to investigate the predictability of academic achievement for college computer science majors in the ROC. Though performance prediction for complete computer science programs was the principal interest of the research, the relationship between performance in the introductory computer science courses and a number of variables, including overall performance in computer science programs, was also examined. The research design focused on responses to the following specific questions:

- 1) Are College Entrance Examination scores related to performance in college computer science programs?

As previously referenced, significant relationships between student SAT scores and performance in introductory computer science courses have been determined in the research conducted in the United States. Research regarding performance prediction in computer science programs beyond

the introductory level, though limited in extent, also demonstrated similar results. However, in the ROC, though low powers of CEE scores for the prediction of student average scores in general college course work has been reported, none of this research has dealt specifically with performance predictions for computer science majors, neither for introductory computer science courses nor for overall performance in the computer science programs.

Effective predictors for particular disciplines may not provide predictive powers for other academic fields. Thus, the questions remains as to the extent that total CEE score and scores for specific subject areas (especially for math, English, physics, and chemistry) relate to student performance in computer science programs.

2) Is math ability related to performance in college computer science programs?

Math ability has frequently been reported to be positively related to performance in the introductory computer science courses in studies among US subjects. Various measures for math ability were used in these studies, including the number of math courses taken in high school and college, performance in these math courses, or scores from the SAT math component. Although some researchers also used self-rating for measuring math ability, the validity of this measure is questionable without providing other supporting data. In addition, none of the studies reviewed found a significant relationship between number of

college math courses taken and performance in the computer science programs. Thus, the relationship of math proficiency to student overall performance in computer science programs has not been conclusively demonstrated.

In the ROC, without any research justification for the practice, certain universities have set minimum CEE math scores, besides total CEE scores, as corequirements for admission (College Entrance Examination Board, 1994). Therefore, an investigation of the relationship between math ability and performance in computer science courses may provide empirical evidence on the appropriateness of such a hypothesized relationship. Based upon the consideration that ROC students take the same number of math courses during high school, for this study math ability is measured by CEE math component scores, high school math course average scores, and college math course average scores (if taken).

- 3) Is prior computer science experience related to performance in college computer science programs?

In the US, prior computer science experience has been related to performance in computer science courses at the college level. However, some researchers have argued that structured programming experience, and not experience with general computer applications, is the most significant contributor to student performance in college level computer science courses (Dey & Mand, 1986; Greer, 1986). It is also noted that most of the studies conducted did not

differentiate among the varieties of computer experience when using "prior computer science experience" as a potential factor for the performance prediction of computer science majors. Thus, "structured programming experience," in addition to the number of computer courses taken, are used for measuring student prior computer science experience for this study. If prior computer experience, such as programming experience, is beneficial to student performance in college computer science programs, then students may be advised to take additional computer courses, thus solidifying their knowledge in computer science before entering college.

- 4) Is overall high school performance related to performance in college computer science programs?

In the US, high school GPA has been reported to be a good performance predictor for college computer science programs. However, high school performance has never been taken into consideration for college admissions in the ROC. It is of particular concern whether student high school performances can be used to predict college achievement. If a relationship between high school performances and success in college computer science programs does exist, findings of this study may provide supporting evidence for future changes in college admission policies in the ROC, as specifically related to the field of computer science.

- 5) Is performance in introductory computer science courses related to overall performance in the computer science programs?

Commonly, US universities have used introductory computer science courses as "gateways" for entering the computer science major. Thus, students who satisfy preset requirements in an introductory course are then qualified for admission into a computer science program. The underlying assumption for this selection process is that good performance in the entry-level courses suggests future success in computer science programs. However, this hypothesized relationship has never been empirically verified. It is important to determine the validity of the hypothesis to justify whether this selection process should be continued.

- 6) Can reliable models be developed to predict performance in (a) introductory computer science courses, and (b) complete computer science programs? If so, can the equivalency of the two models be demonstrated?

It has been demonstrated that the CEE scores provide only low predictive powers for successful college performance (Chen, 1975; Hsu & Lin, 1982; Lu & Jien, 1976; Tsong et al., 1977). Thus, a primary concern for this study is whether reliable models, based upon the introduction of a number of potential factors in addition to the CEE, can be developed to predict performance of overall computer science programs.

While some factors, such as math ability, have been reported to be good predictors for performance in introductory computer science courses, the long-term predictability of performance in computer science majors beyond the level of introductory computer science courses has not been fully investigated. It is unclear whether math ability should continue to be related to overall computer science program performance, as to the performance in the introductory computer science courses. If predictors related to performance in introductory computer science courses are not effective for predicting overall performance in computer science programs, then different models may need to be employed. The issue is basically whether the predictive model for introductory computer science courses is equivalent to the model for an overall computer science program?

- 7) Are there gender differences in performance predictors for computer science majors?

Although little evidence has been provided that men outperform women in computer science courses, many of the studies conducted in the US have found that some predictors are effective only for male students (Clarke & Chambers, 1989; Kersteen et al., 1988). With respect to prior computer science experience, men were reported to have more self-initiated computer experience than women. Therefore, since women tended to have little prior computer science experience, this factor could not be an effective predictor of female performance in computer science courses.

If gender differences do exist among performance predictors for computer science majors in the ROC, then future changes in the selection process for college entrance, at least for the computer science majors, must also take these gender differences into consideration. However, if the contrary is the case, then additional research may be required to determine the reason for different findings for gender differences between the US and the ROC.

Significance of the Study

According to Borg and Gall (1989), the purpose of prediction research is "to select students who will be successful in a particular setting . . . [and] identify students who are likely to be unsuccessful at a subsequent point so that prevention programs can be instituted" (p. 6). While some prediction research in the area of interest has been conducted in the US, almost none has been completed in the ROC. In addition, an initial view of potential predictive factors in ROC must focus upon the CEE since it is currently the only means for admission into a university.

As previously addressed, numerous universities in the US have experienced a serious attrition problems in computer science programs. That is, a great proportion of students dropped the major after their first year of academic training (Campbell & McCabe, 1984; Jagacinski et al.,

1988; Sorge & Wark, 1984). Changes of academic major are more easily completed in the US than in the ROC, and this reason, among others, could account for the high attrition rate of the computer science programs in the US.

Moreover, a great number of freshmen enter university in the US as undecided majors. Thus, enrollment in an introductory computer science course may be an initial experience with the discipline of computer science. Once these students experience the reality of the discipline and discover it is not as "fun and exciting" as they had expected, they have tended to drop from the courses or to change their majors (Sorge & Wark, 1984).

In the ROC, the college enrollment in computer-related programs has increased by more than 121% since 1985 (Hwang, 1990). From a sample university examined, only two students dropped from the program within the years between 1985 and 1989. However, changing academic majors in the ROC is not as easy as in the US, and if a student reluctantly remains in an undesired program, then learning may become a painful and unproductive process for that individual. The student may ultimately leave the field of computer science upon graduation.

As previously stated, at the present time and for some years to come, the ROC faces a shortage of information professionals (Chen, 1988; Wang, 1989). If computer science graduates do not work in the field for which they have been trained, then the productivity of the educational resources

would be further limited. In addition, the failure to meet national needs for the information profession may create problems for the entire economy. Hence, the findings of this study may provide some insights into the adequacy of the CEE selection process of students for computer science programs.

Factor identification is also helpful to counselors in providing better advice to and placement of students for planning their future educational needs, as well as early counseling for appropriate career paths (Hunt, 1977; Renk, 1986; Stephens, Wileman, & Konvalina, 1981). Unless changes are made, high school graduates in the ROC will still compete within the current matriculation policy for higher education. Therefore, guidance in determining appropriate college majors is extremely important for high school graduates in the ROC prior to taking the CEE. The results of this study will provide information to assist counselors in working with high school students as they determine intended college majors and prepare for these disciplines.

In addition, the identification of factors linked to potential academic success is also important in curriculum development (Ralston & Shaw, 1980). For example, if a strong mathematics background is identified as a significant factor of success for computer science programs, then more mathematics courses might be required as prerequisites or corequisites for certain computer science courses.

Based upon their findings, Kersteen et al. (1988) also suggested that the factors linked to academic success can be used to redesign the precollege curricula to encourage more women to consider computer science as a major in college.

Since curricular development or change (at either college or high school levels) in the ROC is subject to the authority of the Ministry of Education, individual curricular modifications are not typical (Hwang, 1990). However, uniform standards for the required courses for each college program are revised periodically. Thus, the results of this study may also provide valuable information for computer science curriculum developers in the ROC.

Definition of Terms

To avoid confusion, the terms used in this study have been carefully defined, as follows.

Academic success: Academic success is interpreted differently in the US by various researchers within different areas of research. It has been suggested that persistence in computer science programs may be a good measure of academic success (Butcher & Muth, 1985; Campbell & McCabe, 1974; Shoemaker, 1986). However, student dropout rates in computer science programs in the ROC are too small to effectively differentiate possibly successful students from others. Therefore,

to fulfill the purpose of this study, course work performance in computer science programs is employed as the measurement of academic success in the ROC.

Though studies emphasizing performance predictions for single courses as well as for complete computer science programs are reviewed, this definition of academic success is used for both groups. However, when performance predictors for introductory computer science courses are emphasized, the grade or score earned in that particular course is used as the academic success measure. Where overall program performance is the primary interest, the major GPA or average score of computer science core courses is employed as the academic success measure.

Average score of computer science core courses for individual semester or year: This score is computed as described below for computer science core courses (as defined below), but only for those courses offered in a specific semester or year.

Average score of computer science core courses: The scores for all computer science core courses are summed and then divided by the total number of courses taken by that student.

Computer science core courses: These courses include all the computer science courses required by computer science departments for computer science majors among all the universities participating in this study.

Cumulative GPA: This GPA is the grade-point average earned by a student from all courses taken in a university.

Introductory computer science courses: These courses include entry-level courses offered and required by computer science departments. Typically, these courses are referred to as "Introduction to Computer Science."

Major GPA: This GPA is the grade-point average from all computer science courses in the major subject and the service courses (i.e., mathematics and physics) required by computer science departments.

Predictors, influencing factors, prediction variables: These are terms used interchangeably to denote factors that contribute to student academic success.

Prior computer experience: This experience include any formal training in computer science taken by students prior to entering a university. Computer courses lasting more than 20 hours for instruction period, taken either at a school or a

private-funded institute, are treated as formal
computer training.

CHAPTER II REVIEW OF THE LITERATURE

Introduction

Most studies related to the prediction of academic success in computer science have focused on student performance in introductory computer science courses. The factors investigated in these studies have generally included sex, age, math ability, previous computer experience, previous academic achievement, and previous educational background. However, various researchers have adopted different approaches to the definition of student academic success. Some have focused upon identifying those factors that affect student performance for final grades/scores in introductory computer science courses (Dey & Mand, 1986; Goodwin & Wilkes, 1986; Nowaczyk et al., 1986; Oman, 1986; Renk, 1986; Taylor & Mounfield, 1989; Thronson, 1985). Others have sought to identify those factors that differentiated persisters from those who eventually withdrew from these courses (Greer, 1986; Konvalina et al., 1983b; Ramberg & Caster, 1986). Still others measured student performance by using such criteria as specially designed computer science aptitude tests (Dixon, 1987; Konvalina, Stephens, & Wileman, 1983a).

Relatively little research has sought to determine the predictability of student performance in computer science programs beyond the introductory level. Among those that

adopted this focus, some used student cumulative grade-point averages (GPA) or major GPA for specific class years as measures of academic success, whereas others focused on the identification of factors that affected student persistence in computer science programs (Anyanwu, 1988; Butcher & Muth, 1985; Campbell & McCabe, 1984; Shoemaker, 1986; Sorge & Wark, 1984). However, the factors examined in these types of studies were similar to those investigated in the studies based upon performance in introductory computer science courses, subject to additional concentration upon such preadmission measures as SAT scores, high school GPA, and class rankings (Anyanwu, 1988; Butcher & Muth, 1985; Shoemaker, 1986; Sorge & Wark, 1984). Moreover, certain investigations were also directed at determination of potential gender differences in the prediction of student performance in the computer science programs (Campbell & McCabe, 1984; Jagacinski et al., 1988).

The literature review presented in this chapter summarizes research in both categories: (a) achievement prediction as determined from the results of introductory computer science courses and (b) achievement prediction as determined from the results for courses beyond the level of introductory computer science courses. These two categories are considered, respectively, in the following two sections, followed by a summary of this review.

Achievement Prediction From Results of Introductory Computer Science Courses

Thronson (1985) was interested in identifying those factors from background information which could be used to predict achievement in a beginning computer science course. The factors examined included age, sex, high school performance, programming experience, previous computer science education, languages available for programming, access to computers, years of high school math, number of college math courses, GPA, college class level, and academic major.

The sample included those students enrolled in all the sections of three entry-level computer courses at Montana State University: Introduction to Scientific Computing (CS 101), Introduction to General Computing (CS 111), and Computers in Elementary Education (ED 451). A 13-item check-list questionnaire was used to elicit subject demographic as well as background information during the first week of the class. Data from 298 students enrolled in these classes during Winter quarter of 1984 who properly completed the questionnaire were used for statistical analysis (Thronson, 1985).

A stepwise-multiple regression was performed to generate a prediction model, using final letter grades as the dependent variable. Six variables, including high school performance, college GPA, college class level, self-perception of programming ability, years of high school

math, and sex, were entered in the prediction model. Both "having the capability to program" in COBOL, BASIC or Pascal and "non-programming computer experience" were found to be negative factors in achievement prediction (Thronson, 1985). Gender (particularly female) was found to be a significant predictor of academic achievement in the beginning computer science courses. The model accounted for 27% of the variance in predicting letter grades for the courses. This low prediction power led the author to question the usefulness of the model obtained.

Dey and Mand (1986) investigated the relationship between student performance in introductory computer language courses and student mathematics backgrounds, both at the high school and college levels. There was also an interest in determining the effect of student performance in one computer language course upon learning another language to determine an appropriate sequence of prerequisites for programming courses.

A total of 467 students enrolled in introductory programming courses at two south-central universities and a community college were surveyed. A questionnaire specifically designed for the study was distributed during Fall and Spring semesters to collect data, including current computer course, mathematics and computer science courses previously completed, attitudes toward computer science and mathematics, and additional demographic data. Students were also asked to report their average grade for mathemat-

ics courses in both high school and college, along with their expected final grade for their current programming course (Dey & Mand, 1986).

Between average grades in high school mathematics and expected grades in current computer science courses, there was an overall correlation coefficient of $r_{(451)} = .36$ ($p < .001$). Whereas for the different courses taken, the highest correlation was found for Pascal ($r_{(94)} = .48$, $p < .001$), followed by BASIC ($r_{(207)} = .34$, $p < .001$) and COBOL ($r_{(138)} = .25$, $p < .001$). An overall correlation coefficient of $r_{(414)} = .40$ ($p < .001$) was found between average grade in college mathematics courses and expected grade in current computer science courses. Unlike the results from high school mathematics courses, the highest correlation was found in BASIC ($r_{(180)} = .48$, $p < .001$), followed by Pascal ($r_{(91)} = .37$, $p < .001$), and COBOL ($r_{(131)} = .25$, $p < .002$). The trend that student performance in mathematics courses, both at high school and college levels, was correlated least with expected grades in the COBOL course was also reported (Dey & Mand, 1986).

Subjects were then classified into two groups, those with three or fewer mathematics courses previously completed and those who had completed more than three mathematics courses. Expected grades were coded in numerical values (eight for A, seven for A-/B+, six for B, etc.) for further analysis. A significant difference ($t_{(454)} = -3.76$, $p < .001$) for average expected grade values was reported

between groups completing varying numbers of high school math courses, with a mean score of 5.65 (one to three courses completed) versus 6.37 (four or more courses completed) (Dey & Mand, 1986). No significant difference was found between groups completing college math courses (5.94 for one to three courses completed versus 6.22 for group of four or more courses completed; $t_{(411)} = -1.02$, $p = .39$).

A nonsignificant finding was observed for the preexposure effects of COBOL upon learning Pascal. The mean expected grade value for COBOL-experienced students was 6.31, with 5.42 for students without experience in COBOL ($t_{(95)} = 1.51$, $p = .13$). Only 31% of the students with BASIC experience expected a grade of B or higher in their current Pascal course. However, 61% of those who had previously completed BASIC expected a grade of B or higher in a current COBOL course. This finding led to the suggestion that "prior exposure to BASIC seems to have some positive effect on the learning of COBOL, but not on Pascal" (Dey & Mand, 1986, p. 146). In addition, 73% of the students who had completed Pascal and who were currently enrolled in a COBOL course expected a grade of B or higher. A significant difference in expected grade values was found for students currently enrolled in the COBOL course between those with or without Pascal exposure (6.87 versus 6.05, $t_{(139)} = 3.09$, $p = .002$). This finding was said to "reinforce the belief that an initial exposure to a structured language increases

a student's likelihood of success in subsequent computer science courses" (p. 146).

Dey and Mand (1986) concluded that background mathematics skills contributed to success in introductory computer science courses. They also asserted that there was a stronger complementary relationship between COBOL and Pascal than between other language pairs. However, some conclusions were inconsistent with the findings of the study. It was stated that "students with prior exposure to COBOL or Pascal have higher grade expectations in a subsequent Pascal or COBOL course than students with no prior computer language experience" (p. 148). Yet, only an insignificant p -value of 0.13 was found in the t -test for a preexposure effect between COBOL and the current Pascal course. Of greater concern, since grades in previously completed computer science courses were obtained, it was unclear how the "effect of the student's performance in one computer language course on the learning of another" (p. 144) could be tested and how subsequent conclusions could be drawn.

Goodwin and Wilkes (1986) conducted a study to examine the relationship between student background characteristics and achievement in beginning computer science courses. In the Fall of 1984, 322 students enrolled in "Beginning Pascal Programming" at Worcester Polytechnic Institute were asked to complete a questionnaire at the second class meeting. Among these subjects, 63% were either computer science or electrical engineering majors, whereas the ratio of

male to female subjects was three to one. Subject background characteristics considered included gender, parental education and interest in computers, number of courses completed in computer science, mathematics, physics and chemistry, SAT math and verbal scores, and self-rated knowledge of BASIC and Pascal languages. A 10-minute exam on Pascal skills was also administered during the second class meeting.

At the end of Fall term, subject grades were obtained, and grade scale points were denoted as "no record (not passing)," "acceptable" and "distinction," respectively. The authors extended the grade scale to five points by asking course instructors to further rate subjects who failed the course among the following additional levels: (1) completed only a few assignments and examinations, (2) completed about half the assignments and examinations, and (3) completed almost all the assignments and examinations (Goodwin & Wilkes, 1986). From the data provided, a product-moment correlation coefficient was calculated and stepwise multiple-regression analysis was performed, resulting in an R -square of 0.25. "Knowledge of BASIC," self-rated by respondents on a four-point scale, was reported to be a strong predictor of student final grades ($r = .38$), whereas "knowledge of Pascal" and "Pascal skill test score" were said to be weak predictors ($r = .17$ and $r = .19$ respectively).

More than one-half (62%) of the subjects indicated they had fair- to expert-level knowledge of BASIC. This knowledge was also found to have a strong relationship to time spent on computers in high school ($r = .59$). Few subjects (10%) indicated they had much knowledge of Pascal. In fact, most subjects (82%) scored very low in the Pascal skills test (two or less of a possible score of seven). However, many of these Pascal novices were found to be able to "do well in the course" (Goodwin & Wilkes, 1986, p. 4).

The math SAT score was said to have significant influence upon prediction of beginning computer course success, but no correlation information was given. An unexpected finding was that the numbers of high school physics and chemistry courses taken were negatively associated with student achievement in Pascal ($r = -.22$ and $r = -.17$ respectively). The variable "knowledge of BASIC" was also negatively associated with the number of college courses taken in mathematics, physics, and chemistry, but the significance of the correlation between these variables was not noted (Goodwin & Wilkes, 1986).

For the course, no gender differences in achievement were determined. It was hypothesized that women entering a technical-oriented college might have more positive attitudes and stronger backgrounds in computer science than women at other colleges, and it was suggested that further research be developed to test the applicability of this finding. It was concluded that in predicting performance

for beginning Pascal, previous experience with computers was of great importance, followed by quantitative skills (Goodwin & Wilkes, 1986). As also asserted by Taylor and Mounfield (1991), lack of previous computer experience when entering beginning computer science courses was said to place such students at a disadvantage.

Oman (1986) was most interested in finding a simple and practical model (i.e., an advising tool that could be administered in less than 15 minutes) for the prediction of student success in computer science courses. For this study, 38 students (20 males, 18 females) enrolled in two introductory computer science courses, "Introduction to Computers" (CS 101) and "Introduction to Structured Programming" (CS 211), during 1982-1983 were included as subjects. Both courses were programming-oriented, with BASIC taught in CS 101 and Pascal taught in CS 211. Algebra was the prerequisite for both courses. Students in both courses were evaluated according to performances for two midterms, one final examination, and four programming assignments.

Subject grades (GRADE, encoded as four to zero for letter grades A to F) and weighted grade (GRADEPTS, computed by multiplying GRADE by course credit hours) were the dependent variables. The variables used as predictive factors included the number of computer systems previously used (TIMESHARE and MICROS), the number of programming languages previously used (LANGS), the number of years since

high school graduation (YEARS), and SAT scores (MATHSAT and VERBSAT). A questionnaire was administered to gather data, and subject SAT scores were obtained from the school Registrar. Pearson's product-moment correlation coefficient was calculated between all independent and dependent variables. MATHSAT, LANGS, VERBSAT, and TIMESHARE were all found to correlate highly with GRADEPTS ($r = .80, .65, .61$ and $.60$ respectively, all $p < .01$). The variable YEARS was not found to be significantly correlated to either GRADE or GRADEPTS ($r = .14$ and $.20$ respectively) (Oman, 1986).

A stepwise multiple-regression procedure was performed to generate the "best" predictive model, which was one that included all six independent variables for the prediction of GRADEPTS performance ($F_{(37)} = 23.83, p < .001$). A very high R-square (0.82) was reported, accounting for the variation of prediction. Mathematics proficiency was highly associated with GRADEPTS and was said to be the key predictor to success in computer science courses (Oman, 1986). The model was then tested for predictability power by application to CS 101 and CS 211 classes in the subsequent semester, predicting 70% of the student grades within one grade scale point. However, it was suggested that the model should be used only in conjunction with other advising tools (i.e., personal interviews) and should be continuously updated (Oman, 1986).

Kersteen et al. (1988) examined enrollment patterns and prior computing experience levels among students

enrolled in a college introductory computer science course, with a secondary interest in determining the possible interactions among gender, previous computing experience, and performance. A questionnaire focused upon level of prior computer experience was administered to students enrolled in "Introduction to Computer Science" during the Spring (176 subjects) and Fall (123) semesters of 1985 in paper-and-pencil format and via computer mail, respectively. Approximately one-fourth of the students enrolled were female, and freshmen and sophomores together comprised 76% and 66% of the classes for the Spring and Fall semesters, respectively.

"Introduction to Computer Science" was chosen as the source class since it had long served as a gateway to the computer science major. The course was described as extremely challenging, and included four hours of lecture and six hours of lab per week. The UNIX programming environment, Pascal and Logo programming skills, and problem-solving techniques were emphasized. Analyses of data from both semesters were completed separately, based upon concern for possible instructor effects and the existence of different levels of high school age microcomputer usage among the subjects (Kersteen et al., 1988).

The proportion of males and females who had taken computer courses in high school was found to be approximately equal. Five items regarding computer experience outside the classroom were asked. Males were found to have more

experience on "teaching yourself to program" and "working as a programmer." However, a higher proportion of females than males indicated that they had obtained computer experience from video games (31% versus 21% and 25% versus 20% for the Fall and Spring semesters, respectively), "educational software" (16% versus 14% and 10% versus 5% for each semester, respectively), and from "working with others" (34% versus 25% for the Fall semester). Final letter grades were coded numerically, ranging from zero (F) to four (A). A scale which consisted of 13 computer experience items was composed. From the results, alpha reliability of 0.83 and 0.81 were reported for the Fall and Spring semester experience scales, respectively. This experience scale was then used to predict final grades for the course by performance of a stepwise multiple-regression analysis (Kersteen et al., 1988).

Significant computer experience scale gender differences were found for both Spring ($F_{(1,175)} = 13.53$, $p < .001$) and Fall ($F_{(1,122)} = 17.87$, $p < .001$) semesters, using a two-tailed analysis of variance (ANOVA). The model showed no predictive power for females (R -squared less than .04 for both semesters). About 14% (Spring) and 25% (Fall) of the male prediction variance could be accounted for from this experience scale. It was stated that the "amount of prior self-initiated computer experience was highly predictive of university-level introductory computer science course performance" (Kersteen et al., 1988, p. 328).

Four subjects, including two males and two females, were interviewed in the effort to determine why a gender difference existed in the amount of prior computer experience. Each of the gender groups interviewed included two students who received either an A or a C in the course, otherwise little information about the interview procedure was provided (i.e., how the subjects were selected, how long and in what format they were interviewed, and the consistency of the interviews, etc.). Nancy received a C in the course, and support and encouragement from her father, a systems analyst, was reported to be the major reason she passed. Though she had a computer in her home, she was not interested in using it until late in high school. She had taken a BASIC course while in high school but had no prior experience with the UNIX operating system. Lack of experience with UNIX had created considerable difficulty for her when she learned both Pascal and UNIX at the same time. She also indicated that the learning process would have been easier if she had a stable working relationship with other students (Kersteen et al., 1988).

Mary also had a computer in her home which she never used, and she had no prior experience with the UNIX operating system. However, Mary received an A in the class. She attributed her course success to both taking advanced placement Pascal in high school and having formed a solid partnership in the lab with a student who was quite familiar with UNIX. Though unaware of UNIX, Tom had taken a

computer course in high school, had been "hacking" with his own computer, and had taught himself BASIC via unguided self-learning. He still received a C in the class and said that having to work part-time while going to school was to blame for his poor performance. John received an A in the class. Unlike his inexperienced peers, John had worked as a professional programmer and had a considerable working experience in UNIX. Though he did not have his own personal computer in high school, John had spent plenty of time teaching himself several programming languages as well as digital logic, using a computer in a nearby institute (Kersteen et al., 1988).

The result of the interviews did lend some support to the findings from the computer experience questionnaire. Both males indicated they had considerable self-initiated programming experience in addition to computer science courses. The findings from the questionnaire also revealed that males had more outside the classroom programming-related experience than had females. This self-initiated exploration of the computer was described to "arm one with more sophisticated tools for using the computer in an efficient manner" (Kersteen et al., 1988, p. 329). It is of interest that both females stressed the importance of support from their families and peers, whereas both males indicated they preferred to work on their own. The difference of this "working preference" provided some explanation

as to why more males experienced more hours of unguided "hacking" than did females.

Based on the findings from the interviews, though little evidence was provided to show gender differences in study habits or the use of additional resources, it was observed that lack of experience in UNIX might be the "stumbling block" in learning how to program in the course (Kersteen et al., 1988). Several educational implications were derived. First, prior computer experience was identified as a good predictor of success in beginning computer science courses for males, whereas it was not for females. For females, hours spent studying was considered as a better potential predictor of course performance. Second, support from parents and friends seemed to be influential for women in motivating them to pursue computer science courses. Third, a course focused on the mastery of an operating system, the acquisition of appropriate study skills, and providing strong peer support was suggested as possibly helpful to students who lacked prior computer experience.

The main concern of a study by Clarke and Chambers (1989) was to examine potential gender differences in computing achievement predictions. Two research questions were addressed: Are there gender differences on each of the identified factors? Do these factors affect student intentions to pursue further computing studies and/or are they associated with student performances in the course?

The sample included 222 (110 male and 112 female) students enrolled in a compulsory statistics and computing Concepts class at Deakin University, Australia. Questionnaires were distributed at the second class lecture (March, 1987) and were completed by the students. Students were asked to report their computing as well as mathematics experience in response to a set of at least 28 yes/no questions. Gender differences in previous computing experience were compared using the *Chi-square* method on percentages of positive responses to each item.

A significantly greater number of males than females had taken computer studies at the year twelve level (23% versus 6%, $X^2 = 12.60$, $p < .001$). However, no significant differences were found in the number of computer courses taken, nor in the average number of students enrolled in computer courses at other school levels. Men also reported they had taken more ($M = 2.7$) mathematics courses than women ($M = 2.0$) at the year twelve level. Men reported more experience using various types of computer systems (i.e., an average of 2.9 systems), except for the Apple-II where a similar percentage (62%) for both genders was found (Clarke & Chambers, 1989). On the other hand, women reported experience with only an average of 1.7 systems ($t_{(218)} = 5.05$, $p < .001$). Men were also found more likely to have some experience in all the languages commonly investigated, including BASIC, COBOL, FORTRAN, Pascal, Assembler and Logo. On average, men had used more than

twice as many types of computer languages as women (2.1 versus 1.0, $t_{(217)} = 6.14$, $p < .001$). The same results were also true for computing applications. Men had used more applications than women (2.4 versus 1.5 , $t_{(217)} = 4.17$, $p < .001$).

Forty-five percent of men and 30% of women reported that they had personal computers at home ($X^2 = 1.54$, $p > .05$). Significantly more men (67%) than women (16%) reported that they were the main users of their home computers ($X^2 = 20.81$, $p < .001$). Among those subjects who were not frequent users of home computers, more fathers and brothers (18%) than mothers and sisters (2%) were reported to be the main users (Clarke & Chambers, 1989).

Beliefs in abilities were assessed by using Likert items. More women than men stated that they enrolled in the course only because it was a required unit. Females were also found to have lower levels of confidence in their ability, lower expectations of success, and less positive attitudes toward the perceived difficulties of the course. The relative importance of each item which contributed to success or failure in the course was evaluated by a five-point scale ranging from zero to four, and similar rating patterns were found between men and women. Both men and women reported "hard work," "good teaching in class" and "personal help from lecturers/tutors" as the three most important factors to success. However, women rated all three factors significantly higher than men ($p < .05$).

Interestingly, men rated "own ability" as an attribute to their success higher than did women (1.7 for men, 1.3 for women, $t = 2.04$, $p < .05$). Men also gave a lower rating to "lack of ability" as a reason for their failure (1.4 for men, 2.0 for women, $t = 2.99$, $p < .01$). Moreover, significantly more women than men attributed their failure to the "difficulty of the course content" ($p < .001$) (Clarke & Chambers, 1989).

Subjects were also asked to predict their final grades. More men than women expected to gain higher grades ($X^2 = 16.81$, $p < .001$), agreeing with the finding that they were confident in their abilities. However, this optimistic expectation did not result in a higher level of academic performance. In fact, there were no significant gender differences in the average final scores for either computing (men, 55; women, 52; $t_{(218)} = 1.01$, ns) or statistics (men, 48; women, 53; $t_{(218)} = 1.84$, ns) component. The actual grades and the predicted grades were also not significantly correlated ($r = .08$ for men, $r = .15$ for women) (Clarke & Chambers, 1989).

A forward multiple-regression analysis was performed to assess the relative contributions of those variables measured in predicting student actual performances in the course. Previous computing experience was found to be a significant predictor, contributing 16% of the variance in the prediction. "Prior mathematics experience" and "university entrance score" were the remaining significant

predictors, both accounting for 2% of the variance. The total R -square achieved was 0.20, indicating that 80% of the variance in predicting final grades was unexplained (Clarke & Chambers, 1989). Using separate analyses for men and women, "prior computing experience" was reported to be a significant predictor for both men and women, whereas "previous mathematics experience" was found to better predict men's achievement than women's. The model was said to explain 27% of the variance for men, whereas for women, only 7% of the variance could be explained. For this low predictive result, it was proposed that future research attention be directed to other factors, such as "good instruction from experienced teachers with well developed curricula and effective class computer access" (p. 424).

Additional analyses of the separate variables used for measuring computing experience showed that "previous study of computing in year twelve at secondary school" was a significant predictor for both men and women. Experience with BASIC was a significant predictor only for men. Surprisingly, word processing, though more men than women reported experience for this application, only predicted achievement for women and not for men. Marked gender differences were found from the "intentions to take more computing courses." A significantly higher proportion of men than women intended to continue with their studies in computing (men, 88%; women, 34%; $X^2_{(1)} = 64.38$), to complete a computer studies major (men, 65%; women, 20%; $X^2_{(1)} = 45.42$), and to

continue into honors program (men, 30%; women, 7%; $X^2_{(1)} = 19.26$). A p -value less than 0.001 was reported for all three tests (Clarke & Chambers, 1989).

To further identify those factors contributing to the intention to pursue further computing studies, a forward regression analysis was performed, resulting in a regression model R -square of 0.42. "Computing attitudes" was found to be the most significant predictor, accounting for 31% of the variance. Gender (accounting for 7% of the variance) was the second variable entered into the model. "Sex-typing" and "statistics attitudes," both accounting for 2% of the variance, were the remaining two variables included in the model (Clarke & Chambers, 1989). Note that both "attitudes toward computing" and "attitudes toward statistics" were in the model, while experience and academic achievement were not.

Separate analyses of the data for men and women showed that "attitudes toward computing" were significant predictors of the intention to major in computing for both men and women. "Previous mathematics experience" and "attitudes toward statistics" were reported to be significant predictors for women, but not for men (Clarke & Chambers, 1989). Unlike results from other studies, a low attrition rate (slightly less than 6%) was reported. This result may be due to the fact that the course was compulsory. A total number of 13 students (seven men and six women) withdrew from the course during the semester.

The findings of the study did not reflect gender differences in course performance. However, existing gender differences in perceptions of personal ability and in attributing process were also found. Together, these findings were interpreted as the supporting evidence that it was the perceived difference, rather than the real ability difference, that should be attributed to the findings of gender differences in many computing-related research (Clarke & Chambers, 1989). It was hypothesized that by giving women greater opportunity to achieve minimal experience level, gender differences could possibly be eliminated. Further, to encourage the participation of women in computing studies, it was proposed that fostering the development of more positive attitudes to computing must be undertaken at the primary and secondary school levels. Moreover, it was also suggested that women were more likely to develop positive attitudes if computing learning were introduced by more cooperative group interaction. Yet, no citation supporting this statement could be found within the context of the study.

Similar weaknesses were found in the Clark and Chambers (1989) study as in many others reviewed. The questionnaire was designed specifically for the study. Almost all the questionnaire items were self-reported. Though items regarding psychological measures were involved, information on the validity and reliability assessment of the instrument was not provided. Furthermore,

failure to provide important statistical information was another reason that many of the findings could not be further verified.

Nowaczyk et al. (1986) sought to identify associations between student performance in introductory level computer science courses and certain background factors (i.e., high school performance, programming experience, and anticipated grade). Two studies were conducted, the first to develop a prediction equation and the second to test the predictive powers of the equation developed. From multiple-regression analysis, the results of an analysis of problem-solving ability was used in conjunction with the background factors to predict the final course grades.

A total of 413 students from three different courses, including Introductory Data Processing (DP), COBOL programming, and Introductory FORTRAN programming, at Clemson University participated in the first study. Among the subjects, 193 students (98 males and 95 females) were from the DP course; 90% of the students in the course were business or non-science majors; and another 92 subjects (57 males and 35 females) were students enrolled in the COBOL programming course. The COBOL course was the subsequent course for those who had completed DP and wished to gain more programming experience. The remaining 128 students (72 males and 56 females) were from the FORTRAN course, the initial course required for computer science majors. Most of the students in the FORTRAN course were computer

science, engineering, or mathematics majors (Nowaczyk et al., 1986).

All of the subjects were asked to provide information on majors, years in school, college GPA, previous programming experience, number of computer science courses taken at high school as well as college levels, and average grades in English, foreign languages, and mathematics (both in high school and in college) as background factor material. A set of seven problems, designed to test student problem-solving skills, was also administered during the first week of class meetings. Three of the problems regarded the ability to translate a word problem into an algebraic solution, two were about logical thinking, and the other two were designed to test for basic skills in understanding a computer program (Nowaczyk et al., 1986).

Problem-solving test performance (measured by the proportion of problems correctly solved) was first analyzed with respect to gender, course of enrollment, and final course grade. Subjects were categorized into grade levels A, B, C, D, and F, and those who withdrew from the course for further analysis. A 2 X 3 X 6 unequal - n ANOVA was then performed. It is of particular interest that a standardized procedure was used to permit comparisons among students within each course. The standardized mean value (M) was set to zero and the standard deviation was set to one. A positive mean value indicated above-average performance, whereas a negative mean value indicated below-

average performance. A significance level of 0.05 was used for all of the statistical analyses (Nowaczyk et al., 1986).

Males were found to perform significantly better ($F_{(1,380)} = 4.66$), but at only slightly higher levels ($M = .49$ versus $M = .45$) than females for the problem-solving test. Different performances were also discovered with respect to specific courses of enrollment ($F_{(2,380)} = 14.31$). A post-hoc Tukey test showed that among the three courses, students in the DP course performed the worst in the problem-solving test ($M = .42$ for DP, $M = .52$ for COBOL, and $M = .53$ for FORTRAN). Students with grade A ($M = .60$) in the courses significantly outperformed students with other grades (M ranges from 0.43 to 0.47, $F_{(5,380)} = 3.65$) (Nowaczyk et al., 1986). A similar trend was again found when series of analyses were carried out for individual problems in the problem-solving test. For the equation problem, "A" students performed significantly better than other students ($F_{(5,430)} = 3.85$) and students in the DP course performed worse than FORTRAN students on the recipe problem ($F_{(2,430)} = 3.83$).

The variables entered into the regression model were high school foreign language, previous programming experience, anticipated course grade, and performance in the problem-solving test. All other factors were nonsignificant for the prediction of the final course grades. A fairly low R -square of 0.21 was reported for the model.

Since values on all factors were standardized, the equation was said to provide "standardized scores which predict individual student's relative standing among other students in the course" (Nowaczyk et al., 1986, p. 274). A significant relationship was not found between course enrolled and course performance.

In the second study, the equation developed was used to predict student performances for the 24 students in the DP course and 18 students in an Introductory Psychology course, used as a control group. Students were asked to complete the same test form for problem-solving skill during the first week of class meetings. As in the first study, the values of all factors were standardized. A product-moment correlation coefficient was calculated for each course to determine the relationship between predicted and actual course performance. A significant relationship was found for the DP course ($r_{(22)} = .69$), but not for the psychology course ($r_{(16)} = .33$, $p = .18$). The model explained 47% of the variation in predicting performance for the DP course. It was stated that the hypothesis that the equation would effectively predict student performance in computer science courses, but not in other courses, was thus supported (Nowaczyk et al., 1986).

Based upon these findings, it was concluded that previous academic performance and general problem-solving ability were related to performance in computer science courses. Since the equation was developed with a variety

of courses, it was claimed that the generalizability of the findings was further supported by testing with different introductory level courses. However, since the predicting equation was developed for courses where programming skills were heavily emphasized, a caveat was added that similar findings may not be warranted if the equation were used for nonprogramming courses (Nowaczyk et al., 1986).

For his research, Renk (1986) intended to detect the potential relationships between student academic success (the dependent variable) in introductory computer programming courses and such factors as prior mathematics background, previous computer use, American College Test (ACT) math score, high school level GPA, sex, age, major, academic class, course expectations, and abstract reasoning ability. A student who ranked in the upper 50% in the class, denoted by a positive standardized score, was considered academically successful. The study was conducted at a small liberal arts college of 1,100 students. During the academic year of 1984-1985, 154 students enrolled in CS 131 (Introduction to Structured Programming), including nearly equal proportions of men and women, were included in the study. The BASIC programming language was taught in the course.

On the fourth class day, students were surveyed using a profile questionnaire. The instrument testing student abstract reasoning abilities (the Differential Aptitude Test, Bennett, 1974) was administered on the same day. The

test, consisting of 50 multiple-choice questions, was said to be capable of assessing student abilities to recognize patterns and to abstract the next step in a given sequence. A standardization process for test scores was derived by subtracting the test mean from the student actual test scores, then dividing by the test standard deviation. The standard final success score (STD), computed by adding the standardized scores for midterm and final examinations, then taking an average, was used to represent class rank for each student (Renk, 1986).

Variables with continuous data were stratified into a smaller number of layers for cross-tabulation analysis to examine potential group differences. The proportion of students, rather than individual students, who had positive STD scores (referred to as the success rate) was compared between different groups according to the factors examined. The significance level for all the statistical tests was set at .01 for the study, and each hypothesis was tested independently (Renk, 1986).

The Abstract Reasoning Test (ART) was administered to all subjects (five missing cases resulted in a valid number of cases of 149). One point was assigned to each of the questions correctly answered, making 50 the highest possible score that could be achieved. A mean score of 42.7 with a standard deviation of 5.1 was reported (Renk, 1986). Students with higher ART scores outperformed those with low ART scores. Among those who scored higher than 45, 78% (40

out of 51) were able to complete the course successfully. A much lower success rate, 46% (40 out of 87), was observed for students scoring between 36 and 45. Of those who scored less than 35, only 18% (2 out of 11) were able to succeed. A significant relationship ($r = .27$, $p < .001$) was found between ART scores and final success scores (STD).

Student ACT and SAT scores were obtained from the school Registrar. If ACT scores were not available, SAT scores were converted to equivalent ACT scores using conversion tables. The mean score for the ACT mathematics element (ACT-M) for 120 subjects was 23.3 ($SD = 4.96$), ranging from as low as 7 to as high as 36. The scores were divided into four groups (below 18, 19 to 23, 24 to 27, and over 27) for further analysis. An obvious progression in the percentage of success rate (from 19% raised to 49%, 63%, and 94%) was observed as the ACT-M scores increased. A significant relationship was also found between ACT-M and STD ($r = .47$, $p < .001$) (Renk, 1986).

Mathematical background was measured from the number of math courses subjects had taken both in high school and in college. On average, subjects (142 valid cases) had taken 3.8 math courses prior to enrolling in current computer programming courses ($SD = 1.0$), and 69% had taken four or more. The success rate in the computer science course was found to steadily increase from 36% for those who took three or fewer math courses, to 59% (39 out of 66)

for subjects who took exactly four math courses, and to 75% (24 out of 32) for those who took five or more math courses. A significant correlation between number of math courses and STD was reported ($r = .30$, $p < .001$). Based on the findings from subject math backgrounds and ACT-M scores, it was suggested that "strong math abilities may be a necessary, but not entirely sufficient, condition for high-level performance in introductory computer science classes" (Renk, 1986, p. 92).

Self-reported high school GPA were obtained from the profile questionnaires, averaging 3.27 ($SD = .46$) for 139 valid cases with a low range from 1.76 to a high of 4.0 for a 4.0 scale. Among these subjects, 77% had a high school GPA of 3.0 or better. The success rate in the computer science courses for those who had GPA 3.0 or lower was only 31% (15 out of 48). The success rate raised to 53% (21 out of 40) for those who had GPA between 3.01 and 3.49, and again increased to 82% (42 out of 51) for those with GPA 3.5 or higher. The correlation coefficient between STD and high school GPA was .41 ($p < .001$). This result led to the conclusion that student high school GPA were strongly correlated to student academic success in introductory programming courses (Renk, 1986).

A pattern of steady increase in success rate for various computer experience groups was again observed for "previous computer experience." For those with no computer experience, only 39% (23 out of 59) were successful in the

computer science courses. Success rates for students with some or extensive computer experience were 67% (53 out of 79) and 83% (five out of six), respectively. Both the finding of a strong correlation ($r = .31$, $p < .001$) and significant differences between groups ($F = 8.05$, $p < .01$) suggested a relationship between previous computer experience and academic success (Renk, 1986).

Based on previous computer experience findings, Renk (1986) indicated that "previous computer experience seems to give students an advantage in introductory programming classes" (p. 93). However, the BASIC programming language considered in this study was generally the first language that students encountered in high school computer courses. Therefore, the researcher added that this finding may not truly reflect student abilities to comprehend new materials.

Gender was not found to be a significant factor for predicting academic success. About equal numbers of males ($n = 77$) and females ($n = 73$) participated in this study, and there was no great difference in success rates between different genders (i.e., 51% for men, 57% for women). Neither the correlation test ($r = .13$, $p < .05$) nor the ANOVA ($F = 2.55$, $p = .11$) reflected a significant relationship between gender and academic success (Renk, 1986).

The distribution of student academic classes indicated that more than 60% (86 out of 140) of the subjects were freshmen. The freshmen group also attained the highest

success rate (60%) among four academic classes. The negative correlation coefficient of -0.17 between academic class and academic success indicated a decreasing success rate in the upper classes ($p < .05$). Only 38% of the combined junior and senior groups were successful. The ANOVA results also did not indicate a statistically significant relationship between class and academic success ($F = 3.26$, $p = .024$). Generally speaking, subjects showed high expectations for academic success in the computer science courses. More than 50% (34 out of 67) of the students expected to receive an A, whereas 88% of these students actually achieved academic success. In contrast with this high success rate, only 30% of those with expectations of receiving a B and 20% of those expecting to receive a C were successful in the class (Renk, 1986).

When treating the data as continuous, a correlation coefficient of 0.37 ($p < .01$) was determined. Further evidence of a strong relationship between expectations and academic success was supported by the result of the ANOVA ($F = 20.44$, $p < .001$). It was suggested that the expectation of a higher grade might motivate some students to exert extra effort to achieve academic success (Renk, 1986). A stepwise multiple regression was also performed, with substitution of the mean for all missing values, to identify factors contributing the most to achievement prediction. ACT-M, expectations, GPA and experience were the four variables which accounted for 35% of the variance.

Note that reasoning skills (ART) and math background were completely dropped from the model. The explanation for this result suggested that "abstract reasoning skills may function as an important component of several key factors" (p. 97).

Taylor and Mounfield (1989) conducted the first of two studies during the academic year 1986, seeking to identify those factors that contributed to success in college computer science courses. This study was primarily directed at interest in student previous computer experience, rather than mathematics ability or previous academic achievement as previously investigated by a number of other researchers. Taylor and Mounfield hypothesized that prior experience could be an unwritten prerequisite to successful completion of college computer science courses, further theorizing, in the absence of empirical support, that there could be a link between the amount of time students spent in nonacademic works and success in computer science courses. In addition, the rapid decline in the number of female students in computer science programs was also a principal concern.

The subjects were 709 students enrolled in CSC 1250, Introduction to Computer Science I, at Louisiana State University during the academic year of 1986. All of the students were required to complete college algebra and trigonometry, or to enroll in a college calculus course, before enrolling in CSC 1250. Students majoring either in com-

puter science or in electrical engineering with a computing option comprised 77% of the sample. The ratio of male to female students enrolled in the course was reported to be three to one. In addition to gender and academic major, survey forms asked students to report prior high school and college computer science courses and the amount of time they were employed per week. At the end of the semester, student final course grades (the only measure of academic success in this study), along with the hypothesized factors, were analyzed to investigate success patterns (Taylor & Mounfield, 1991).

For a student to be proclaimed "successful" in CSC 1250, his or her final grade of the course had to be C or better. Only 373 students completed the course, however. The high attrition rate (approximately 47%) in the course implied that sample mortality might be a serious problem to this study. The proportion of the students who were considered successful in the course (referred as the "success rate"), rather than individual student performances, was compared according to the factors examined. Among the subjects who completed the course, 64% indicated that they had prior computing courses of some kind, either at high school or college levels. The success rate of these computing-experienced students was found to be significantly higher than those who had no prior experience (43% versus 24%, $p < .01$). Students who gained their computing experience in high school did slightly better, by a 2% margin, in

success rate than did those who took their first computing course in college. Both high school and college computing-experienced students were also found to outperform, with respect to success rate, their nonexperienced counterparts at the same level of significance (Taylor & Mounfield, 1991).

Employment (either part- or full-time) was not found to have a negative influence, as generally believed, upon computer science success. To the contrary, success rates for those who worked less than 20 hours per week (75%) was found to be significantly higher than for all other groups (all were about 67%), including those who did not work at all, at the confidence level of 0.10. Based upon this finding, Taylor and Mounfield (1991) suggested that the skills a student learned to manage both a job and schooling at the same time might be beneficial to his/her success in computer science.

There were three times as many males as females enrolled in the course. Of those who did not withdraw, 71% of the males were successful, compared to a female success rate of 62%, a significant difference at the level of 0.10. Female students were also found to perform at the extreme ends of the scale, that is, either as the very best or with little chance of success. From closer examination, it was determined that the ratio of males to females for computer science majors was 61% to 39%, or less than two to one, whereas the male-female ratio for electrical engineering

majors was 88% to 12%, or more than seven to one. Apparently, the gender ratio was largely skewed by academic majors. With this in mind, the interpretation of any finding for gender differences was provided with appropriate caution (Taylor & Mounfield, 1991).

The researchers hypothesized that freshmen students could have been subject to a disadvantaged status due to lack of academic maturity. Unexpectedly, 78% of the freshmen were reported to be successful in the computer science course, significantly higher than students for other class levels ($p < .05$), for example, 58% reported for juniors. Taylor and Mounfield (1991) attributed this phenomenon to the computer science course prerequisites. Those who enrolled in the freshmen year might have indicated stronger mathematics abilities, whereas enrollment in the course in a later class year could have implied some difficulties in math encountered in the earlier class years. It was concluded that prior computing experience, "whether in high school or college level, is a critical factor in success in computer science" (p. 196). Those with no prior computer experience were at a great disadvantage when competing with those who had prior computer experience. Employment was not a detrimental factor in students' performance in the college computer science courses. As a group, a larger percentage of male than female students were successful.

In a follow-up study, Taylor and Mounfield (1991) attempted a closer look into the nature of the high school

computing experience. The sample included in the study was described as "about one third the size of the group involved in the first study" (p. 242), and the male to female ratio was reported to be three to one, just as for the previous study. In the 1988-1989 academic year, students enrolled in CSC 1250 were surveyed at the beginning of the semester to collect information needed. To examine whether different types of computing experience resulted in various degrees of influence on success in college computer science course, some items were developed to separate those who had only application experience (e.g., word processing or database management) from those who had taken programming classes. Students were asked to state whether they owned a personal computer, whether they had taken any typing courses, and what grades they expected from the current computer science course. Those students with programming experience were asked to recall their high school programming class grades, and were also asked to complete a special subset of seven questions to determine if structured programming was taught in their high school programming class. At the end of the semester, student final grades were recorded.

Having a structured programming experience in high school was found to be one of the best predictors of success in college introductory computer science courses. Of those subjects with high school programming courses, 74% indicated they had been taught structured programming in a

high school BASIC class. Among these students, 77% were successful, in contrast to a 52% success rate for those who had not been taught structured programming. Thus, having application experience only (in the absence of programming experience) was not found to be a good indicator of college computer science success (Taylor & Mounfield, 1991).

Since keyboarding is a major method of entering commands, lack of typing ability was hypothesized to be a possible stumbling block in computer learning. However, students with no typing experience in high school were actually more successful than those who had typing experience, though the difference was not statistically significant. As for the issue of computer ownership, 47% of the subjects reported owning a home computer. These students were found to be only slightly more successful than nonowners. Since all of the CSC 1250 assignments were to be completed on terminals to a mainframe computer (and not on microcomputers), it was suspected that the inability to use their own computers for class work might reduce the advantage of owning a home computer (Taylor & Mounfield, 1991).

Of those who received an A in their high school programming classes, 77% were found to be successful in the college computer science courses as well. In contrast, only 20% of those who received a C in high school programming classes were able to succeed in their current computer science courses. As a result, Taylor and Mounfield (1991) suggested that grades received in high school programming

classes correlated well with college computer science success. Employment was again found not to negatively affect student performance in college computer courses. Seventy percent of the students were found to work part time to financially support themselves during college. As found in the first study, part-time workers did even better in the course than those who did not work at all. Amazingly, among those who worked 40 or more hours per week, 92% were successful in the introductory computer science courses. Consistent with the finding of Werth's (1986) study, employment was said not to be detrimental to the success in college computer science.

Based upon findings from both studies, Taylor and Mounfield (1991) concluded that high school computer science experience could itself have a positive effect on success in college computer science. It was suggested that it was definitely an advantage for students to have some type of programming course, especially in structured programming, prior to enrollment in their the first college computer science courses.

In place of examining the predictability of student performance for an introductory computer science course, Konvalina et al. (1983b) conducted a study to examine the differences in background factors and aptitudes toward computer science between withdrawers and nonwithdrawers in the beginning computer science courses. During Fall semester 1980, 382 students enrolled in Introduction to Computer

Science (CS 160) at the University of Nebraska in Omaha were surveyed and tested during the first week of class. The authors reported that 154 students subsequently withdrew from the course, resulting in an attrition rate of 41%. CS 160 was the first technical course for students who intended to major in computer science.

Student educational backgrounds, prior computer experience, mathematical abilities, and performances from a researcher-developed computer science aptitude test were compared. The variables investigated included age (AGE), estimated high school performance (HSP), hours worked per week for part-time employment (HW), prior computer education (PED), prior nonprogramming computer work (PWNP), prior programming work (PWP), years of high school mathematics (YRHSM), number of college mathematics courses (NUMC), and the total number of high school and college mathematics courses (TMATH). The TMATH factor was calculated by summing the number of years of high school mathematics (YRHSM) and the number of college mathematics courses taken (NUMC) (Konvalina et al., 1983b).

Following data collection, responses to the questions were recoded according to numerical values from one to five. The two-sample *t*-test was used to compare the means of recoded responses for all eight demographic factors and the mean scores for each component of the predictor test between withdrawers and nonwithdrawers. An aptitude test, frequently referred to by other researchers as the KSW

test, was used to predict potential academic success in computer science. It consisted of 25 multiple choice questions with five questions for each section in number and letter sequencing (SEQ), logical reasoning (LOGIC), calculator simulation (CALC), algorithms (ALG), and high school algebra word problems (WORDP). A K-R 20 measure of reliability of 0.76 and a predictive validity of 0.56 ($p < .001$) were reported. Thus, the reliability of the test was fairly good. However, the validity of the test was considered to be moderate for the prediction of student final examination achievements (Konvalina et al., 1983b).

A significant group performance difference for the predictor test was reported when all five sections were considered ($p < .001$). The scores for nonwithdrawers were significantly higher than for withdrawers in SEQ, LOGIC, CALC, and WORDP ($p < .001$). Nonwithdrawers were also found to be significantly older ($p < .01$), performed better in high school ($p < .05$), took more computer courses ($p < .05$), and had more substantial mathematics backgrounds (for both the NUMC and TMATH, $p < .01$) than those who withdrew from the course. Though not significantly different, nonwithdrawers were found to have more computing experience (for both the PWNP and PWP, $p > .05$) and more years of high school mathematics (YRHSM, $p > .10$) than the withdrawers. Nearly one-third of the students were employed and worked 40 or more hours per week. No significant difference was

found for number of hours worked between withdrawal and nonwithdrawal groups (Konvalina et al., 1983b).

It was concluded that the important role of mathematical reasoning ability and mathematical background for potential success in computer science courses had been confirmed by the results of the study. Based upon the finding that withdrawers and nonwithdrawers were significantly different with respect to the number of college mathematics courses taken (NUMC), taking more college mathematics courses as a remedy for those who scored low in the predictor test was proposed as a remedy. It was also concluded that the KSW test was an effective instrument for classification between withdrawers and nonwithdrawers. By implementing the predictor test as a placement tool for beginning computer science courses in the university, the withdrawal rate was reported to drop from 40% to 23% in the subsequent semester. Furthermore, only 11% of the nonwithdrawers failed (receiving D or F) (Konvalina et al., 1983b).

Ramberg and Caster (1986) were interested in examining possible performance differences between those who completed a course and those who eventually withdrew from the course. They also tried to correlate students final grades with performance on placement tests. The sample used for the study was about 800 students enrolled in entry-level computer science courses during 1984-1985 at the University of Wisconsin-Eau Claire. The courses included Elementary

Computer Concepts, Programming in the Basic Language, Programming in the FORTRAN Language, and Problem Solving in Pascal I. Students from an upper level course, Operating Systems, were used as a control group, but nothing further regarding this control group was noted and the actual purpose of the control group remains an unknown. The actual number of students enrolled in each course was not provided. The attrition rate for these courses was said to be 15% on the average (120 out of 800).

Background information on the subjects was obtained, but without indicating what instrument had been used. Eleven demographic variables were briefly listed on the table, but lacked further description. These included sex, age, high school performance, current performance, part-time employment, academic class, prior education, nonprogramming work, programming work, number of high school math courses, and number of college math courses. Since no further information was available, the meaning of some of the variables (i.e., prior education) could not be identified. Moreover, not all the findings of these variables were discussed. In fact, only findings regarding prior computer experience and math background were reported (Ramberg & Caster, 1986). A placement test developed by Konvalina et al. (1983a) was also administered on the first day of the class. No other information regarding the placement test was given. No test validity or reliability information was provided.

Students were first classified as finishers or nonfinishers. Scores on the placement test between finishers and nonfinishers were then compared, based on categories from the demographic data obtained. A *t*-test was performed to determine whether a significant difference existed between the test scores of finishers and nonfinishers. In addition, the Pearson's correlation coefficient was used to determine the relationship between scores on the placement test and the final grades in the courses. A significance level of 0.05 was employed for the statistical tests. Surprisingly, all the analyses were performed using a program developed by a computer science major. No further explanation was given regarding why commonly used statistical packages were not considered for analytical purpose (Ramberg & Caster, 1986). Thus, the accuracy of the statistical result could not be verified. The authors reported that data from each class as well as the entire student sample were analyzed. Yet, no statistical information about individual classes was given. A total number of 752 students was used for analysis. Since no explanation was given, it is unknown whether the remaining 48 subjects were students of the control group or simply a reflection of missing data.

Previous programming work, which was described as building a foundation from a logic background, was found to have a significant effect upon placement test scores. The difference on the mean test scores between finishers and

nonfinishers with "no," "some," or "considerable" experience in programming work were all significant (at $p = .01$, $p = .001$, and $p = .042$ respectively). Furthermore, among the finishers, student mean placement test scores were found to increase with the amount of experience in programming (with $M = 64.4$, $M = 66.4$, and $M = 71.9$, respectively). The extent of mathematics backgrounds was found to be positively associated with placement test scores among finishers. That is, the more math courses taken, the higher the student scored on the test. A positive correlation was also discovered between the finishers' performances on the placement test and their final grades earned ($r_{(642)} = .24$, $p = .016$). Why only 642 subjects were used for the correlation analysis was not explained (Ramberg & Caster, 1986).

Based on the findings presented, Ramberg and Caster (1986) concluded that prior exposure to computers, "whether that be a literacy/programming course in high school or college" (p. 37), was a key factor to success in computer science. However, the authors also indicated that the conclusions were valid only for entry-level programming courses. They found little correlation between scores on the placement test and final grades for nonprogramming courses (i.e., computer literacy course). Whether this finding was the result of the same study or the result of other research was not clear. Moreover, considering that no information was provided about the course content of the nonprogramming courses, the meaning of this assertion could

not be identified. It was also concluded that the placement test seemed to be a good predictor for the final grades earned as well as successful completion of the introductory computer science courses. Suggestions to place a prerequisite for entering computer science majors or to consider using the placement exam for advisory purposes were also provided.

In addition to the statements that contradicted each other, the validity of the conclusions was further threatened for several reasons. First, some conclusions were not supported by the statistical results. For instance, most of the demographic variables tested were classified into more than two categories. To test for differences between categories within that variable, an ANOVA should have been used. However, only the *t*-test was performed to determine the difference between the finishers and nonfinishers for each category of all the variables. No statistical tests were performed to examine the difference between various categories of the variable. Moreover, no correlation coefficient between those variables and scores on the placement test was given. Hence, the statistical results could not support the claim that "amount of math background was directly related to the exam score" (Ramberg & Caster, 1986, p. 36).

Obviously, most of the information was provided by subjects of the study, including their high school experience and current performances. Strangely, "no experience"

was included along with "A," "B," "C or below" as one of the responses for the variable "current performance" (Ramberg & Caster, 1986). The ambiguous questions included in the study, along with the factor of self-reported data, may decrease the accuracy of the collected information. Significant results were also reported for the t-test ($t = 4.44$, $p < .001$) between the mean test scores of finishers and nonfinishers, and on the product moment correlation between test scores and final grades of the finishers ($r = .24$, $p = .016$). Both of these findings tended to indicate that student final grades in the introductory computer science courses could be predicted by the performance on the placement test. Nevertheless, with this small correlation, the practical value of the predictive power claimed might be limited.

Greer (1986) focused attention on the potential relationship between experience gained through high school computer science courses and achievement in a college introductory computer science course. Discriminating between withdrawals and those who persisted in the course, the amount of high school computer science and the degree of emphasis upon structured programming instruction were the principal research interests. The sample consisted of computer science students registered in the introductory computer science course (CMPT 110.6) at the University of Saskatchewan during the academic year 1983-1984. Structured programming methodologies were emphasized in this course.

At the beginning of the course, all 285 registered students were pretested with the KSW computer science aptitude test (Konvalina et al., 1983a) and the Raven Advanced Progressive Matrices test (Buros, 1972). Though well-established validity was claimed for both tests, the results of the validity assessments were not given (Greer, 1986). A questionnaire was distributed to all the subjects to obtain background information. Due to concern for possible contamination, data from students who were from high schools that did not provide any computer courses as well as those who had completed mathematics or computer science courses at the college level were eliminated from the study. As a consequence, 117 students were included as subjects following this selection.

Among the selected subjects, 61 students were inexperienced in computer science, and 56 students had completed some high school computer science courses, ranging from one-half to three semesters. Students who had taken high school computer science courses were also asked to complete a Structured Programming Inventory (SPI) to determine the degree of emphasis, either low or high, on structured programming techniques in the courses taken. As designed by Greer (1986), the SPI was given to 194 high school students and eight teachers to assess its reliability. It was indicated that 84% of the student SPI ratings were accurately matched with their by the teachers, and it was claimed that the SPI provided high reliability in classifying students

into groups with various type of programming experience. However, no further information was provided regarding the validity of the instrument.

During the eight months of the course, records of the withdrawals and scores on four examinations (three midterms and a final) were collected and compared to pretest data. An average attrition rate of 42% was reported as the number of students in the class dropped from 117 to 68 by the end of the academic year. The students were then grouped according to the amount of high school computer science experience ("none," "some," or "much") and the degree of emphasis on structured programming methodology ("none," "low," or "high"). A correlation matrix for all variables was provided, and only the final examination score was found to be significantly related to the Raven (Buros, 1972) test ($r = .21$, $p < .05$) and the KSW test ($r = .37$, $p < .05$). However, the author indicated that the low correlation coefficient achieved "was considered too small to be useful for prediction of student achievement" (Greer, 1986, p. 218).

Raven (Buros, 1972) and KSW (Konvalina et al., 1983a) test achievements were compared among groups with varying amounts of computer experience and different degrees of emphasis upon structured programming methodologies. No significant group differences were found for both tests at 0.05 confidence level, and the F -ratio for the ANOVA were all less than 1.50. The same result was reported by

repeating the ANOVA for only the 68 nonwithdrawals. Again, the F -ratios for the ANOVA were all less than 1.50. Scores of the four examinations of those nonwithdrawals were also compared by performing a multivariate ANOVA among different groups, and the results again proved insignificant at F -values of 0.50 and 0.49 for comparisons based on the amount and the type of prior computer experience, respectively. The p -values for these statistical tests were not reported. Among the withdrawal subjects, 54.1% were inexperienced, 38.2% were moderately experienced, and 13.6% were experienced computer users. A significant relationship between withdrawals and the amount of high school computer experience was found ($\chi^2 = 11.1$). Students with more high school computer science experience were less likely to drop the college introductory computer science course. However, results of the statistical analysis for different degrees of programming experience was not discussed (Greer, 1986).

In an attempt to establish a pattern for the withdrawals, a discriminant function analysis was performed. Two pretests, along with the amount of computer experience and the amount of structured programming experience were used as variables. The discriminant function, using Wilk's $\lambda = 0.88$ and $\chi^2 = 14.2$, was said to correctly classify 61% of the students into withdrawal and nonwithdrawal groups. Univariate F -test results indicated that both the amount and the type of high school computer experience, with

F ratios of 11.89 and 6.03, respectively, contributed significantly to the predictive ability of the discriminant function. However, neither the KSW (Konvalina et al., 1983a) nor the Raven (Buros, 1972) tests were able to accurately discriminate withdrawals from nonwithdrawals ($F = 1.10$ and 1.99 , respectively).

Contrary to the claim by Konvalina et al. (1983a), it was concluded that neither of these two tests could be used to effectively predict student withdraw patterns. Greer (1986) further indicated that

students with lower ability who had high school computer experience were more likely to complete the course and achieve lower examination scores, while lower ability students with no high school computer experience were more likely to withdraw. (p. 223)

However, this claim was merely based on the observation that all three students who failed in the course had some high school computer science experience.

Several interpretations were provided in the attempt to explain the study findings. Due to the disproportionate number of withdrawals in various experience groups, it was suspected that there were possibly unmeasured achievement differences. In addition, it was hypothesized that a greater difference in achievement favoring computer-experienced students might have been found if the study had not been threatened by a mortality problem, which was believed to cause comparisons of achievement to be biased in favor of the nonexperienced group. Greer (1986) observed

that research would be needed with a more accurate measure of computer aptitudes to examine the role that withdrawing students served in the findings of the study. It was also suggested that a careful evaluation of the costs and benefits of high school computer science curricula be conducted.

The purpose of the study was clearly stated and most of the conclusions were drawn closely based on the findings. However, though Greer (1986) had noted its importance, information on the validity and reliability of the test instruments used for the study were not provided. Moreover, important statistical information, such as the p -values, was not provided. Without such information, the results of the statistical analysis reported in the article could not be verified. Nevertheless, in recognition that the study was seriously threatened by the high attrition rate, and from the viewpoint that the study was observational by nature, the researcher interpreted the results of the study with appropriate caution.

Konvalina et al. (1983a) sought to identify the factors that influenced both aptitudes for computer science and achievement in the computer science courses. A number of details regarding those influencing factors were provided, especially their relationship to the predictor test as well as to the final examination. The principal research interest was in determining the extent to which mathematics background, prior computer science education or

experience, age, hours worked as part-time employment, and high school performance influenced computer science aptitudes when compared to the achievements on a course final examination.

A Pearson's product-moment correlation coefficient of 0.56 was reported between the KSW (Konvalina et al., 1983b) scores and the scores of the final examination. However, lower correlation coefficients for the "item validity" of individual test sections was reported: $r = .41$ for SEQ; $r = .33$ for LOGIC; $r = .28$ for CALC; $r = .48$ for ALG; $r = .43$ for WORDP (the statistical significance of these correlation coefficients was not stated). Test reliability, measured by K-R 20 method, was specified as 0.76, and though the validity and reliability of the final examination was not provided, it was stated to be satisfactory. Correlation coefficients for all the variables were given in the form of a table. Although indicated as statistically significant, no exact p -values for the correlation coefficients were given (Konvalina et al., 1983a).

A stepwise-regression procedure was performed separately using the KSW test (Konvalina et al., 1983b) and the final examination as the dependent variables. The first factor entered into the regression model for both regression analyses, high school performance (HSP), was then said to be a good predictor for the computer science aptitude test ($R = .42$, $F = 34.96$, $p < .001$). A strong mathematics background was also found to be important in

the development of computer science aptitudes. Both the number of years of high school mathematics (YRHSM; $R = .48$, $F = 11.07$, $p < .001$) and the number of college mathematics courses (NUMC; $R = .51$, $F = 7.64$, $p < .01$) were included in the regression model for the aptitude test. Prior computer education experience (PED) and prior programming experience (PWP) were excluded from the regression model, since the model p -value became nonsignificant when the variables were included ($p = .11$ and $p = .44$, respectively) (Konvalina et al., 1983a).

HSP was also the first variable entered into the model for final examination ($R = .28$, $F = 13.57$, $p < .001$). PED was said to be significantly related to the achievement in entry level computer science courses ($R = .34$, $F = 6.50$, $p = .01$). The age of the student (AGE) was also found to be a significant predictor in achievement of the final examination ($R = .38$, $F = 5.26$, $p < .05$). Older students tended to achieve higher scores on the final examination than their younger peers. However, AGE alone showed little effect on achievement in the final examination ($r = .10$, $p > .05$). Note that no mathematics related factors (e.g., YRHSM, NUMC and TMATH) were included in the regression model for predicting achievement on the final examination (Konvalina et al., 1983a).

Based upon the results, it was concluded that the existence of a critical relationship between student high school performance and success in college computer science

courses was obvious. The importance of the mathematical background was again stressed. The advantage of having some prior computer education was also asserted. Although many of the correlation coefficients between the dependent and independent variables were statistically significant ($p < .05$), most of the correlation coefficients were lower than 0.30. Moreover, the regression models only accounted for 26% variance of the KSW predictor test (Konvalina et al., 1983b) and 14% variance of the final examination. Thus, the practical value of these findings were considered too low if prediction of student performances were to be based solely upon the model (Konvalina et al., 1983a).

Achievement Prediction From Results of Advanced Computer Science Courses

Campbell and McCabe (1984) examined factors influencing student success in a first-year computer science program. By evaluation of Registrar records, it was found that successful completion of a first-year computer science program was a useful indicator of success in the major. Success was measured in terms of three consecutive semesters of enrollment as a declared computer science, engineering, or other science major. Primary research interest was directed at determination of those factors that differentiated students who persisted in the major from those who changed their majors to disciplines other than computer

science, engineering, or other science after one full year of study in the university.

The sample included 256 first-semester freshman computer science majors who were enrolled in the first programming course for majors at a large midwestern university during the Fall semester of 1979 (Campbell & McCabe, 1984). Students with other courses in the university prior to enrollment in the programming course were excluded from the study. The factors examined included SAT scores, high school rank and size, and high school science and math background. No significant differences were found between students of computer science, engineering and other science majors when the dependent variables were compared. Hence, these three groups of students were combined into a CS+ group to compare with students who switched to other majors.

The results indicated that students in the CS+ group scored higher in both SAT math (621 versus 575, $p < .001$) and verbal (526 versus 486, $p < .001$) than those who switched to other majors. The CS+ group was also found to rank higher in high school (88.3 versus 85.8, $p = .03$), had taken more high school math (8.72 versus 8.25, $p = .001$) and science courses (6.29 versus 5.29, $p = .001$), and had received higher average grades in these courses. Men were found more likely to persist in the CS+ program (61% versus 39%) than were women. The results of discriminant analysis also identified SAT math scores, grades in high school math

and science, and sex as the most effective predictors that could be used to differentiate potential successful computer science students from those who eventually withdrew from the majors (Campbell & McCabe, 1984).

From analysis of Registrar records of 1,323 computer science majors enrolled at Purdue University from 1978 to 1981, Sorge and Wark (1984) sought to identify the factors that could be used to predict student success in the computer science major. These variables investigated included sex, verbal and math SAT scores, high school rank, and number of semesters and average grades in high school math, English, and science courses. Most of the students had taken six or more high school math courses with grades of B or above. Most of the sample ranked in the top one-third of their graduating class in high school and the sample male-to-female ratio was about two to one.

Among these subjects, 1,071 students started with a traditional beginning course for computer science majors, CS 230, Introduction to Structured Programming, using either Pascal or PL/I. Finding that students with SAT-math scores less than 540 usually did not do well in the beginning calculus course (the corequisite of CS 230), the Computer Science Department used a SAT math score of 560 as the safety factor for admission to CS 230. Students who scored less than 560 had to enroll in Introduction to the Computing System (CS 490A) before they could enroll in CS 230. Enrollment in four consecutive computer science

courses at a level higher than CS 230 was considered as the standard for success in the major (Sorge & Wark, 1984).

Regression analysis was used to determine the factors that effectively predicted if students achieved satisfactory progress in the computer science program. Semesters of high school math, English, and science as well as high school rank were dropped from the model due to their insignificant contribution to success prediction. The results of various models with a combination of different variables were compared. When students who scored 560 or higher in SAT-math and 500 or higher in SAT-verbal, and achieved grades of B or higher in CS 230 and a score of five or more on the trigonometry placement test were compared to the sole use of SAT scores as a selection criteria, retention rates for the former dramatically increased from around 50% to 79% (Sorge & Wark, 1984).

Although not the principal research interest, marked gender differences were also reported. More men than women were found to meet the SAT scores selection criteria (68% versus 32%), to have scored B or higher (76% versus 24%), and to have successfully completed four acceptable courses (63% versus 37%). However, 43% of those students who met the SAT scores selection criteria and earned grades of B or higher in CS 230, did not continue their studies in the computer science major. It was suggested that some factors other than academic ability might be involved which were

responsible for the high attrition rate among these capable students (Sorge & Wark, 1984).

Butcher and Muth (1985) were interested not only in student performance in a single course, but also the overall student success in college. They investigated the feasibility of predicting performance for precomputer science majors, using information available prior to enrollment in college, such as ACT scores and associated high school data. First-semester freshmen students who completed the introductory computer science course (CS 1) for computer science majors at West Virginia University during the academic year of 1981-1982 were the subjects of the study. Only data from 269 students (124 from the Fall, 1981 and 145 from the Fall, 1982) who completed the course were used.

Three variables were used to measure the success in the course, including the average examination scores (EXAM), average laboratory scores (LAB), and final course grades (GRADE). Overall college success was measured by first-semester college GPA (CGPA). The independent variables were high school data and ACT scores, including such high school-related variables as student class ranking (HS-RANK), class size (HS-SIZE), level of high school mathematics completed (HS-MATH), high school computer courses completed (HS-CS), number of physics and chemistry courses completed in high school (HS-PC), number of science courses completed in high school (HS-SCI), and high school GPA

(HS-GPA), all of which were obtained from student high school transcripts. Student percentile ranks (HS-PER) were calculated by dividing class size by class rank (Butcher & Muth, 1985).

Five ACT-related scores, obtained from standardized ACT examination reports, were used as remaining independent variables. These included ACT scores for mathematics (ACT-M), English (ACT-E), natural science (ACT-NS), social science (ACT-SS), and composite (ACT-C). The ACT-C was described as a linear function of the other ACT scores and student "self-reported" high school GPA. A questionnaire was used to obtain student background information, with all other information obtained from the Registrar's Office (Butcher & Muth, 1985).

The presence or absence of a high school computer course (HS-CS) was found to have no effect upon performance, either in introductory computer science ($p > .10$) or in the first-semester in college ($p > .10$). To test the effect of the HS-MATH variable on student academic performance, students were categorized into groups with no high school math course taken ($n = 17$), groups with completion of algebra ($n = 54$), and groups with completion of precalculus ($n = 198$). Significant mean differences between groups with various levels HS-MATH were found ($p < .01$). It was suggested that course work in high school mathematics did "improve student performance in college" (Butcher & Muth, 1985, p. 265). In addition, students who had com-

pleted greater numbers of physics and chemistry courses (HS-PC) in high school was a factor which exercised a positive effect on all four dependent variables ($p < .01$). A positive linear trend was also found between the number of science courses completed in high school (HS-SCI) and those variables examined ($p < .01$).

The variable HS-GPA appeared to have the highest correlation with LAB performance ($r = .45$, $p < .05$) and CGPA ($r = .60$, $p < .05$). The ACT-M was also found to have the highest linear relationship to EXAM ($r = .58$, $p < .05$) and GRADE ($r = .52$, $p < .05$). Though the correlation coefficient was significantly different from zero, it was concluded that using any single variable to predict student performance was limited in value. As for the "best" equation to predict performance for all four dependent variables (EXAM, LAB, GRADE, and CGPA), ACT-M and HS-GPA were again found to be the best predictors ($R^2 = .40$ for EXAM, $R^2 = .24$ for LAB, $R^2 = .37$ for GRADE, and $R^2 = .42$ for CGPA). Nonetheless, the predictability of student LAB performance was limited if only those independent variables examined in the study were used (Butcher & Muth, 1985).

To obtain a clear picture of the data collected, comparisons of student grades were also performed by classifying students into groups that satisfied current admission requirements and groups that did not. Among all subjects, 123 students would have been admitted into the program if current admission criteria were employed, including 92 stu-

dents (75%) who achieved a grade of A or B. Interestingly, 48 students (33%) among those who would not satisfy current admission criteria also earned a grade of B or higher in the course. However, only 68 of all 269 subjects (25%) eventually entered the computer science degree program. Among this 68 students, 20 would not have been admitted to the program if current admission requirements were met. Surprisingly, 75 out of 123 (more than 60%) students who satisfied current admission requirements decided not to pursue a career in computer science. It was suggested that reasons other than academic performance might be involved when students were choosing majors (Butcher & Muth, 1985).

Butcher and Muth concluded (1985) that performance in an introductory computer science course and in the first-semester in college could be predicted, based only upon information available prior to college enrollment, such as high school transcripts and ACT scores. The variables HS-GPA and ACT-M jointly provided the best predictive equation for both GRADE and CGPA. Nearly 37% of the variation in GRADE and 42% of CGPA could be explained by this relationship. However, exposure to high school computer courses did not contribute to the performance in a college computer science course or to first-semester college performance. It was suggested that the findings with respect to the relationship of high school computer courses might indicate the "failure of computer science departments to

assist secondary education with the development of meaningful high school computer science courses" (p. 268).

Some precautions for using the restrictive admission requirements were also made by Butcher and Muth (1985). First, more than 50% of the variation in GRADE and CGPA remained unexplained. Second, students without outstanding high school grade or standardized test scores such as from the ACT examination were still able to succeed. Finally, of those who had demonstrated their ability to handle college-level course work, it might be necessary to provide greater opportunities to them if they were to develop their full potentials.

Shoemaker (1986) was interested in finding which pre-admission measures could be used to predict college GPA for prospective engineering, and information and computer science (ICS) majors. The sample included 296 engineering students and 238 ICS majors, enrolled at the University of California at Irvine during Spring term 1982-1983. Predictor variables included high school GPA, SAT-math, SAT-verbal, and scores from College Board mathematics achievement (MATHACH) and English composition tests (ENGACH). The two dependent variables were sophomore cumulative GPA (CUMGPA) and sophomore major GPA (MAJGPA). The major GPA included grades from service courses required by the major departments.

The engineering and ICS majors had almost identical mean scores on many preadmission measures: high school GPA

(3.64 versus 3.62, $SD = .60$), SAT-math (598 versus 596) and SAT-verbal (454 for both majors). The ICS majors had both slightly higher cumulative GPA and major GPA than the engineering majors. Multiple-regression analysis was used to develop the prediction model. The "best" regression equation was defined as one in which the multiple-correlation coefficient was significantly different from zero, containing the fewest number of predictors. Analyses for engineering and ICS majors were performed separately (Shoemaker, 1986).

Two predictors, MATHACH and high school GPA, were included in the optimal prediction equations for the engineering majors for both cumulative ($R^2 = .38$, $SE = .44$ grade points) and major GPA ($R^2 = .38$, $SE = .52$). High school GPA and MATHACH were also the best predictors for ICS majors for both cumulative ($R^2 = .26$, $SE = .43$ grade points) and major GPA ($R^2 = .34$, $SE = .53$ grade points). It was concluded that the sample GPA were predictable from the scores of math achievement tests and high school GPA (Shoemaker, 1986). However, it was also indicated that some shrinkage in the size of the multiple correlation could be expected if the equation derived in the study was applied to subsequent samples.

Anyanwu (1989) conducted a study in Nigeria to determine the relationship between student achievement in computer science programs at the college level and the test scores of the Joint Admission and Matriculation Board

(JAMB), high school performance, and previous computer experiences. The JAMB test, an aptitude test in nature, was described as intended to "bring about a more even educational development as well as increase the likelihood that qualified students would be admitted into appropriate university programs of their choice" (p. 6). The study was intended to serve several purposes: (1) to identify reliable predictors of achievement in computer science; (2) to determine the predictive power of possible factors; (3) to find out the class levels which the predictive power of these factors were maximized; and (4) to evaluate the correlation between achievement in computer science program and achievement in math components of the program.

To ensure the representative nature of the sample population, five universities were randomly selected, based on the length of establishment and curriculum orientation. Due to missing data and incomplete/inconsistent information, several samples were excluded from the research. Eventually, 150 subjects (44 first-year, 62 second-year, and 44 third-year students) were as subjects. A questionnaire with 13 four-point scale items was used to gather student prior computer experience data. No validity or reliability information was provided. Test scores from the JAMB, high school records, and grades of university courses were all obtained from appropriate Registrar's offices (Anyanwu, 1989).

To analyze the data, the subjects were categorized into three cohorts. Cohort I included 150 subjects who had complete first-year records, cohort II comprised 106 students who had complete second-year records available, and cohort III included only 44 junior students who had complete third-year records. Pearson's product-moment correlation coefficient was used to examine the relationship between variables and multiple regression was employed to generate a prediction model. A significance level of 0.05 was set for all the statistical analyses, and a detailed table of descriptive statistics was provided.

Total JAMB test scores were found to significantly relate to achievement in computer science program math components (r range from .22 to .30). Total JAMB test scores were reported to significantly relate to achievement in nonmath components of only the computer science program at the first year level ($r = .17$, $p < .05$). Surprisingly, no significant relation was found between the JAMB math component test scores and computer science achievements at any of the three year levels (r range from 0.02 to 0.08) (Anyanwu, 1989). A strong relationship between achievement in overall computer science program and achievement in the math component of the computer science program was found for all three cohorts ($r = .71$, $.72$, and $.51$, respectively). The result was not surprising, given the fact that math courses comprised 40% of the courses required by the computer science program. A similar result was

reported between achievement in the overall computer science program and achievement in the nonmath components of the computer science program (r range from 0.52 to 0.69). It was suspected that the high correlation achieved was mainly due to self-correlation.

Only three variables (i.e., high school math GPA, previous computer experience, and high school GPA) were entered in the prediction model for the regression analysis of all three cohorts. A low predictive power was achieved (R^2 range from 0.12 to 0.17). Though not significant, the strength of the relationship between the joint effect of high school GPA and high school math GPA and achievement in the computer science program was reported to increase with length of time in the computer science program (Multiple R range from 0.33 to 0.42) (Anyanwu, 1989). Based on these findings, it was concluded that potentially high-achieving computer science majors could be predicted upon admission. Anyanwu also suggested that improvement in the math component of computer science programs would increase the overall computer science program achievement. Since achievement in the computer science program was highly correlated among the three class years ($r = .66$), it was also asserted that success in the freshmen year was also very likely an indicator of success in later year.

However, subject to careful observation, most of the correlation coefficients were too low to have an important value for education. Though the correlation coefficients

achieved were high in some of cases, such as between achievement in the math component of computer science courses and achievement in overall computer science program, those results were suspect for reason of self-correlation within the computer science courses tested. Furthermore, using the "number of time using computers" as a criteria of previous computer experience was also considered as an inappropriate reflection of the true computer experience of students.

Summary

During the past two decades, researchers have sought to identify those factors which can be used to predict success for college computer science majors. Several studies emphasized the effectiveness of using standardized test scores to predict potential success in college beginning computer science courses as well as overall success in complete computer science programs. The SAT or the ACT mathematics scores were found to correlate significantly with student course performance and were often included in the prediction models (Butcher & Muth, 1985; Campbell & McCabe, 1984; Dixon, 1987; Goodwin & Wilkes, 1986; Oman, 1986; Renk, 1986; Sorge & Wark, 1984). Similar results were also reported for SAT-Verbal and ACT-English scores (Butcher & Muth, 1985; Oman, 1986; Sorge & Wark, 1984). However, since the correlation coefficients were less than 0.60 in

almost all cases, the predictive power of standardized test scores used by themselves to predict student success in beginning computer science courses was recognized as limited.

Student mathematics backgrounds were also found to relate significantly to the performance of college beginning computer science courses and complete computer science programs. Significant results concerning the relationship between the number of high school mathematics courses taken and the final grades of the computer science courses were reported in a number of studies (Campbell & McCabe, 1984; Dey & Mand, 1986; Ramberg & Caster, 1986; Renk, 1986; Thronson, 1985). With the aforementioned findings regarding the predictive power of SAT or ACT mathematics scores, the role of a mathematics background in supporting college computer science majors seems to have been confirmed.

However, none of the studies found a significant relationship between the number of college mathematics courses taken and college computer science courses achievements (Dey & Mand, 1986; Konvalina et al., 1983b; Thronson, 1985). Moreover, Butcher and Muth (1985) found significant group differences between students who had taken various combinations of mathematics courses in high school. Combining these results suggests that it may be the content rather than the number of mathematics courses taken that contributes to student performance in college computer science courses.

Some researchers reported that prior exposure to computers had a significant effect upon success in the beginning computer science courses at the college level (Konvalina et al., 1983b; Taylor & Mounfield, 1991). However, the findings on the amount of prior computer experience were not conclusive. A number of other studies reported nonsignificant results (Butcher & Muth, 1985; Dixon, 1987; Goodwin & Wilkes, 1986; Nowaczyk et al., 1986; Ramberg & Caster, 1986). Although significant findings were reported, certain design weaknesses in some of the studies should be acknowledged. Anyanwu (1988) used the number of times using computers as the measure of previous computer experience. Oman's (1986) study included an insufficient sample from which to develop a prediction model, while Taylor and Mounfield (1989) used a "yes/no" question approach to collect prior computer experience data. Due to these weaknesses, the findings of these studies were considered to have contributed little knowledge to the relationship between student prior computer experience and success in the computer science programs.

Nonetheless, prior experience in structured programming was repeatedly reported to benefit student learning of other computer science courses. Dey and Mand (1986) reported that learning Pascal was beneficial for learning COBOL. Greer (1986) found a significant difference in structured programming experience between the withdrawers and the nonwithdrawers in beginning computer science

courses. Taylor and Mounfield (1991) also reported that the success rate (i.e., being classified into the upper 50% of the class rank) among subjects with structured programming experience was significantly higher than among those who lacked such experience. These findings indicate that it may be the structured programming experience, not general computer experience, which affected student learning and achievement within subsequent computer science courses. Nevertheless, most of the studies failed to establish the validity or reliability of the instrument, bringing their conclusions into question. Additional research will be required to determine if experience in structured programming methodology is beneficial to the learning of subsequent computer science courses.

Among the studies which investigated gender issues, several reported no significant gender differences in course performance. In fact, three of the studies found that females performed better than males in beginning computer science courses (Clarke & Chambers, 1989; Taylor & Mounfield, 1991; Thronson, 1985). Only Nowaczyk et al. (1986) found evidence to the contrary, when males outperformed females in a problem-solving test. Both Kersteen et al. (1988) and Clarke and Chambers (1989) reported that males had significantly more computer science experience, especially self-initiated programming experience. Clarke and Chambers (1989) also found significant gender differences in the perception of personal ability and the attri-

bution process of success/failure in a beginning computer science course. Their findings for gender differences tended to suggest that differences were actually those of self-perception rather than ability.

As for the issue of student prior academic performance, a number of researchers found that student high school GPA were significant academic success predictors for college beginning computer science courses (Butcher & Muth, 1985; Konvalina et al., 1983b; Renk, 1986; Thronson, 1985) as well as for first-year computer science programs (Anyanwu, 1988; Campbell & McCabe, 1984). Only Ramberg and Caster (1986) reported a nonsignificant difference for high school performance between the withdrawers and the nonwithdrawers from a beginning computer science course.

Moreover, Taylor and Mounfield (1991) found that students who performed well in high school computer science courses were more likely to be successful in college beginning computer science courses. Clarke and Chambers (1989) also reported that university entrance scores, though unspecified, were significant predictors of student final marks in beginning computer science courses. Though the predictive powers claimed was moderate, these findings suggested that prior academic performance, especially high school GPA, was useful as a predictor of student success in college computer science programs. Based upon these findings, the predictability of potential success, either in or beyond beginning computer science courses, appeared limited

if only individual factors were used. A higher predictive power was usually achieved from use of a combination of these factors. Among the research studies reviewed, most employed multiple-regression models to predict student potential success in the college beginning computer science courses. Nevertheless, only moderate predictive powers were achieved (i.e., an R^2 range from 0.20 to 0.40).

It is of particular note that studies achieving higher R^2 values were found to include "scores of standardized tests" as a success predictor for beginning computer science courses. Furthermore, the mathematics component of the standardized tests was included in the prediction model in all of the cases. This observation supports the predictability of student potential success in college beginning computer science courses from use of those factors that are available prior to college enrollment. The findings of the reviewed studies also suggested the important role of mathematics ability for success in the computer science programs.

However, several common weaknesses were detected in a number of the studies reviewed. Most used questionnaires specifically designed for that study to collect information. Since no other data source was applied to support the collected data, it is doubtful if the information collected actually reflects intended responses by the subjects. Furthermore, most of the researchers failed to establish the validity and reliability of the instruments

used in their research. Moreover, most of the studies either implicitly or explicitly considered success in a beginning computer science course as equivalent to success in a computer science program, but without providing empirical evidence to support such a hypothesis. Finally, since all of the studies used convenient samples (in many cases, from the university where the researchers were employed), rather than random samples from a target population (usually not clearly defined), the representative nature of the population samples was questionable. Hence, the generalizability of the results to subjects other than where sampled was quite limited.

Since most of the studies were conducted in the mid-1980s, a new look at the predictive factors is necessary, a need reinforced by the fact that computer science curricula and computer accessibility have dramatically changed during the last decade. Moreover, in responding to the weaknesses described above, additional research is needed to determine if the performance of college computer science majors can be predicted by a reliable prediction model.

CHAPTER III RESEARCH DESIGN AND METHODOLOGY

Introduction

The review of the literature demonstrates that the greater part of the research completed regarding academic success prediction in US computer science programs has been directed at the predictability of performance in the introductory computer science courses. Though not empirically supported by the evidence, these entry-level courses have nonetheless long served as gateways for entering computer science programs in many universities. However, few researchers have probed the long-term predictability of performance following experience in introductory computer science courses. Furthermore, little research of this type has been conducted in areas outside the US, where the problem of predicting academic success also needed careful consideration. Since most of the studies reviewed were primarily conducted in the mid-1980s, a new look at predictive factors may be necessary due solely to the fact that access to the personal computer has dramatically increased over the past decade.

The purpose of this research was to investigate the predictability of academic success for college computer science majors in the Republic of China (ROC) beyond the level of introductory computer science courses. Thus, several related research questions were posed:

- 1) Are College Entrance Examination (CEE) scores related to performance in college computer science programs?
- 2) Is math ability related to performance in college computer science programs?
- 3) Is prior computer science experience related to performance in college computer science programs?
- 4) Is overall high school performance related to performance in college computer science programs?
- 5) Is performance in introductory computer science courses related to overall performance in the computer science programs?
- 6) Can reliable models be developed to predict performance in (a) introductory computer science courses, and (b) complete computer science programs? If so, can the equivalency of the two models be demonstrated?
- 7) Are there gender differences in performance predictors for computer science majors?

Procedures for research design are discussed in the following sections, including a definition of the population, the subjects, and procedures for instrument development and data collection. For a clearer understanding of college computer science education in the ROC, computer science programs in the university as well as the admission process are also described.

Higher Education in the Republic of China

According to Ministry of Education (1993), higher education in the ROC is offered by junior colleges, technology institutes, four-year colleges and universities, and graduate schools. The educational goal of the junior colleges and technology institutes emphasizes the teaching of applied sciences, with the aim of training students as technicians. Five-year junior colleges admit junior high school graduates, while three-year and two-year junior colleges matriculate senior vocational school graduates. Rather than granting bachelors degrees, graduates of junior colleges grant a college diploma. The technology institutes admit junior college graduates who wish to further their education in relevant programs.

With different educational goals, four-year colleges and universities prepare students to become specialists in their chosen fields of study and provide opportunity for pursuing advanced study in graduate schools (Ministry of Education, 1993). Students are admitted to four-year colleges or universities based upon total CEE scores. However, only students with a high school diploma or the equivalent are allowed to take the examination. Quotas for individual departments are predetermined by the Ministry of Education (College Entrance Examination Board, 1994). Once admitted to a department, student decisions to change majors are restricted by positions available in the depart-

ment as well as their college academic performances. In general, changing college majors in the ROC is more difficult than in the US.

Due to the dissimilarity of academic backgrounds and admission criteria, certain constraints were employed in defining the population of this study. Computer science majors in a junior college, technology institute, or military academies and evening schools, who were matriculated by examination other than the CEE, were not included in the study population. For four-year colleges and universities, the CEE has served as a selection criteria for admission in the ROC for more than 30 years (Hsu & Lin, 1982). Following several revisions in response to suggestions by educators and researchers, the current CEE has evolved into a complicated matriculation procedure. Students first take a nation-wide examination held annually in July; then they must complete a choice-of-major form in a ranking order. Afterward, a centralized placement is conducted based upon student total CEE scores, specific restrictions set by individual departments, and the ranking of choice-of-major as indicated by the students (College Entrance Examination Board, 1994).

Currently, all academic disciplines in four-year colleges/universities are classified in the 10 categories of art, law, business, science-A, engineering-A, science-B, engineering-B, agriculture-B, medicine, and agriculture-A. These 10 categories are further grouped into four sections.

Section one includes art, law, and business categories; section two comprises science-A and engineering-A; science-B, engineering-B, agriculture-B, and medicine categories compose section three; and section four contains only the agriculture-A category.

The CEE covers 10 subject areas, including the Three People's Principles, Chinese (including a composition test), English, math-A (taken only by students interested in section one), math-B (for students majoring in programs in all other sections), history, geography, physics, chemistry, and biology. Among these subject areas, the first three must be taken by all examinees. Different subject areas of the remaining seven are required depending upon the academic program in which a student intends to major and to which section the program belongs (College Entrance Examination Board, 1994).

Computer-Related Programs in the Republic of China

According to the CEE Board (College Entrance Examination Board, 1994), there are three computer-related programs currently offered at the college level in the ROC. Twelve universities and colleges offer electrical/computer engineering (ECE), five offer computer science (CS), and 17 offer management information systems (MIS). These programs differ from each other dramatically in several ways. They are offered in different colleges. First, all of the ECE

programs are offered in the College of Engineering. The CS programs are offered in the College of Science (with one exception offered in the School of Business). The MIS programs are often offered either in the School of Business or the School of Management Science. Second, the programs are tested by different CEE subject areas. Though the required CEE subject area tests are the same for ECE and CS (math-B, physics, and chemistry), they vary for the MIS program (math-A, history, and geography). Finally, the programs differ in terms of the curricula designed for the individual disciplines. The required courses, as determined by the Ministry of Education, for ECE and CS are different from those required for the MIS program. Furthermore, core courses for ECE and CS also vary due to emphases within the disciplines.

Subjects

Due to the differing curricular requirements described above, the variables investigated would be difficult to analyze and compare with one another among the different programs, to the extent that all three computer-related programs are included. Since this study is the first investigation focused upon the predictability of performances among computer science majors in the ROC, the three programs were not compared. Moreover, for practical considerations, only computer science majors from within ROC

universities were chosen as the target population. The population for the study is thus defined as follows: All students admitted to and currently enrolled in computer science programs at a four-year university in the ROC.

As described previously, there are five universities that provide computer science programs in the ROC. Among these institutions, three are named "national" universities which are fully-budgeted by the government and usually rank at the top of student choices of majors. The other two are private universities, operated principally from private funds.

The study was conducted primarily during the Fall semester of 1995. Due to the consideration that no information regarding college performances is available for entering college freshmen, freshmen students were excluded from the study. Consequently subjects consisted of only sophomore, junior, and senior year students.

Instrument Development

As noted previously, several weaknesses were uncovered within the research reviewed in Chapter II. To cope with the problem of using a single data source of self-reported information collected from the subjects, this study used data from a variety of sources. Furthermore, the research encompassed the establishment of both the validity and the

reliability of the instruments used for the study to enhance the creditability of the research findings.

To respond to the research questions, data from two different sources were used, including student records from institutional Registrar's offices, and a researcher-designed questionnaire. Informal interviews with a subset of the sample in the pilot test were also used to collect information of confusing or ill-phrased items for questionnaire revision.

Fourteen variables were identified, including gender (GENDER), high school average score for all course work (HS-AVG), high school average score for all math courses (HS-MATH), CEE total score (CEE-TOTAL), CEE math score (CEE-MATH), CEE English score (CEE-ENG), CEE physics score (CEE-PHY), CEE chemistry score (CEE-CHEM), number of computer courses taken (CS-COURSE), number of programming courses taken (CS-PROG), structured programming experience (CS-SP), average score of all the college math courses taken (C-MATH), future plan after graduation (PLAN), and the number of computer science core courses retaken due to poor performance (RETAKEN). Two indicator variables (UNIVERSITY, CLASS) were used for examining possible group differences between universities and between class years when conducting regression analysis. The dependent variables were scores achieved in introductory computer science courses (CS-INTRO) and average scores for computer science core courses (CS-MAJOR).

Validation Process and Reliability Establishment

Background information from each subject, to include gender, age, college class, average scores in high school math courses and for overall course work, CEE scores, and number of computer core courses retaken were collected from administration of a research questionnaire. Questions regarding the participants' prior computer experiences (including number of computer courses taken, number of programming courses taken, and information regarding those programming experience) were also included in the questionnaire. Subjects were also requested to indicate their future plans after graduating from the university while completing the questionnaire.

Items chosen to be included in the questionnaire were first generated by the researcher. Guidelines for asking appropriate questions (e.g., make each item clear and precise, avoid negative items, ask questions that are relevant to the sample, etc.) were followed, as suggested by Babbie (1986). Instructions for completing the questions were also provided in the beginning of each sections. A smaller interval range was employed to increase the accuracy of this self-reported information. For example, a five-point interval was provided for the question of high school achievement (in the form of average scores) and the scores of CEE. Opinions from a professional consultant from the Survey Research Center regarding the development of survey

questions were obtained to assure that questions were properly formatted.

The questionnaire (Appendix A) was then given to five experts in computer science or computer science education for review, along with the guidelines used for the questionnaire development (Appendix B). The review experts used this information to assess the content validity of the questionnaire, wherein 80% agreement upon the content of each item was considered as acceptable. For questions where agreement greater than 80% was not achieved, the items were revised in accordance with feedback from the experts. Following this assessment, the content validity of the revised questionnaire was then reassessed.

The questionnaire was then translated into Chinese and given to five experts in computer science or computer science education in the ROC for review. The same validation process was again followed. Following establishment of the validity, the questionnaire was pilot-tested using a group of 34 university computer science students from the sample of this study to establish reliability. A Cronbach's Coefficient Alpha of 0.91 was achieved for internal consistency of the programming experience questions.

Interviews

To establish comprehensive understanding of how the questions (especially items for prior computer experience)

were perceived, personal interviews were conducted with 17 randomly selected subsamples (4 females and 13 males) of the pilot test. Among these interviewed subjects, three were without any prior computer experience, four with some experience in applications, and 10 with programming experience. The purpose of the interviews was to collect additional data regarding the computer courses taken by the subjects. Specifically, the interviews intended to gather information about the content and length of prior computer courses and about subjects' experience with structured programming in the prior computer courses. Interview questions are included in Appendix C.

When pilot-testing the questionnaire, a written consent was first obtained from all 34 participants. After the questionnaires were collected, 17 students were randomly selected for interviews. The researcher again came to the classroom personally requesting cooperation in participating in the personal interviews. A sign-up sheet (with many 30-minute time slots available from 8 a.m. to 8 p.m. each day for three consecutive days) was provided for participants to select interview times.

All of the interviews were conducted by the researcher at the conference room in the computer science department of the pilot-tested university, and were completed within two days. The same format, with identical set of questions, was employed for all interviews. Data collected were recorded and analyzed by the researcher. Information

obtained from the interviews was used to verify the data gathered in the questionnaire and to identify confusing items. Some minor revisions on the questionnaire items were made, all associated with re-phrasing and provision of additional examples.

Data Collection

Before the questionnaire was administered, permission for conducting the research at the universities campus was granted. Contacts with sample universities were first made to obtain documents needed for permission of questionnaire administration and accessing students' academic records at the Registrar's offices. An official request for cooperation was sent to all five universities, along with a description of the research. At first, two of the candidate universities refused the request for accessing student transcripts in the Registrar's office. Fortunately, with the provision of additional information regarding anonymity and confidentiality of each participant and with the explanation in person regarding the significance of the research, permission from all five universities was obtained.

Questionnaire Administration

Following the granting of permission, contacts with the individual computer science departments were made.

Copies of the course schedule offered in Fall semester of 1995 was obtained from the individual departments. Scheduling of the dates and time for in-class questionnaire administration was made by contacting individual instructors of computer science core courses in each participating university.

Since the locations of the participating universities were scattered in three different cities, the data collection was completed in consecutive days if possible. However, more than two visits were unavoidable for two of the participating universities. Two days before the date of questionnaire administration, a confirmation call was made to avoid possible difficulties or delays for on-site data gathering.

On the day of questionnaire administration, verbal explanation of the purpose and the significance of the study was given. Anonymity and confidentiality for the information gathered were again stressed. Subjects were requested to complete the consent form and questionnaire by carefully following the instruction provided. The whole process of questionnaire administration was completed within 20 minutes.

Informed Consent

A written consent form (Appendix D), describing the purpose of the research and procedures to be used, was pro-

vided, assuring the confidentiality of the participants as well. Consent was also requested to review participants' academic records in Registrar's offices. Once consent was obtained, the researcher then collected the information from appropriate Registrar's offices, using the identification number provided by the participants to locate and review the students' transcripts.

Following data collection, a six-digit number was assigned to each of the subjects for purposes of data encoding. The informed consent form, containing the students' identification numbers, was removed from the questionnaire and stored separately to assure the anonymity of the subjects once the student records were located.

Registrar Records

In the ROC, each university receives a list of students who are admitted to individual departments along with their CEE scores from the CEE Board. Therefore, official reports of the student CEE scores were obtained from the Registrar's offices of each university. However, CEE scores were also collected from the questionnaire administration in case any of the participating universities refusing to release such information.

Dependent variable data (CS-INTRO and CS-MAJOR) were obtained from student transcripts in the Registrar's offices at each university. All the courses taken in the

university and their scores (including the average score for all math courses taken and scores from the introductory computer science course as well as other computer science core courses) were collected. However, only those computer courses offered and required by all the participating universities for computer science majors were considered as computer science core courses. These core courses included (1) calculus, (2) linear algebra, (3) discrete math, (4) probability, (5) numerical methods, (6) introduction to computer science, (7) programming, (8) programming languages, (9) data structures, (10) assemblers, (11) introduction to digital systems, (12) electric circuits, (13) system programs, (14) operating systems, (15) computer structures, (16) algorithms, and (17) projects. The average scores of these computer science core courses as well as math courses were then computed by the researcher based upon the information collected. The number of computer science core courses retaken was also calculated from the transcripts provided.

Data Encoding

When students returned the questionnaire, data were quickly reviewed to make certain that all sections of questionnaire were completed. A total of 958 questionnaires was collected. Following data collection, responses for each questionnaire item were manually checked to iden-

tify unusable data. Eighteen questionnaires answered by noncomputer science majors or graduate students were identified and were excluded. The exclusion of these questionnaires resulted with a sample size of 940, representing a 98% response rate, thus 81% of total population.

Questions with inconsistent or incomplete answers were treated as missing values. Responses of the questionnaire were then encoded and entered into a computer spreadsheet file. Data for each university were put into separate computer files for more efficient data entering. All five data files were merged into a single file for further analysis upon completion of data entry.

Data Analysis

Although it has been suggested that persistence in computer science programs may be a better measure of academic success, student scores in the computer science courses are typically used as the achievement measure in the ROC. As Butcher and Muth (1985) have indicated, more than 60% of those who satisfied the admission requirements of the computer science departments in their sample decided not to enter computer science programs. This result suggested that reasons other than academic performance are involved in choosing majors.

Furthermore, as suggested by Borg and Gall (1989), many students identify with a convenient or socially

acceptable reason regardless of the true reason for their withdrawal from the majors. Hence, persistence may not be a correct indication of future academic success. In addition, the drop-out rate for computer science programs in the ROC is considerably small (Hwang, 1990), suggesting persistence in a course or the program is not an effective predictor of academic success. For these reasons, student scores in computer science courses were used as an achievement measure.

The grading system used in the ROC is also different from that used in the US. Rather than using a letter grade for evaluating course performance, scores ranging from zero to 100 are used. A table for converting scores to letter grades is usually supplied upon request; however, criteria for the conversion may be different from school to school. For consideration of data consistency, the actual scores of courses, rather than converted letter grades, were used for all the analyses.

Research Hypotheses

The hypotheses used to test responses to the research questions include the following:

- H_0^1 There are no significant differences among the sample universities in terms of each of the variables investigated.

- H_o^2 There are no significant relationships between CEE scores and scores in the introductory computer science courses or average scores of computer science core courses.
- H_o^3 There are no significant relationships between math ability variables and scores in the introductory computer science courses or average scores of computer science core courses.
- H_o^4 There are no significant relationships between prior computer experience and scores in the introductory computer science courses or average scores of computer science core courses.
- H_o^5 There are no significant relationships between averages for all high school course work and scores in the introductory computer science courses or average scores of computer science core courses.
- H_o^6 There are no significant relationships between scores in the introductory computer science courses and average scores of computer science core courses.
- H_o^7 There is no significant linear predictive model for introductory computer science courses or complete computer science programs.
- H_o^8 There are no significant differences by gender for academic performance, prior computer experience, or success prediction.

The data collection procedures are summarized in Table 1.

Statistical Analyses

The data collected were carefully checked prior to analysis to assure correctness. Invalid responses were treated as missing values. However, questionnaires containing many missing values were considered as unusable and were discarded. All the responses were coded (and quantified if necessary) and entered into computer readable format. The data file was then printed out and manually checked to assure correct data entry. The data were analyzed by a PC version of *STATGRAPHICS* v7.0, a reliable and commonly used statistical software package.

To test each research hypotheses, several statistical analyses were completed (procedures are discussed in detail later). Generally, descriptive statistics (i.e., the number of subjects, mean values of variables and their standard deviations) for all the variables investigated were first calculated. An analysis of variance (ANOVA) was performed to test the first hypothesis (H_0^1) for group differences between sampled universities. Subsequent statistical analyses were completed separately by universities if significant group differences for the variables were found. Otherwise, the analysis combined all subjects from all universities. The analyses combining all subjects from the

Table 1. Research Hypotheses and Data Sources for Research Variables.

Variables	Data Sources	Use
GENDER	Questionnaire	H_0^7, H_0^8
HS-AVG	Questionnaire	$H_0^1, H_0^5, H_0^7, H_0^8$
HS-MATH	Questionnaire	$H_0^1, H_0^3, H_0^7, H_0^8$
CEE-TOTAL	1. Registrar 2. Questionnaire	$H_0^1, H_0^2, H_0^7, H_0^8$
CEE-MATH	1. Registrar 2. Questionnaire	$H_0^1, H_0^2, H_0^3, H_0^7, H_0^8$
CEE-ENG	1. Registrar 2. Questionnaire	$H_0^1, H_0^2, H_0^7, H_0^8$
CEE-PHY	1. Registrar 2. Questionnaire	$H_0^1, H_0^2, H_0^7, H_0^8$
CEE-CHEM	1. Registrar 2. Questionnaire	$H_0^1, H_0^2, H_0^7, H_0^8$
CS-COURSE	Questionnaire	$H_0^1, H_0^4, H_0^7, H_0^8$
CS-PROG	Questionnaire	$H_0^1, H_0^4, H_0^7, H_0^8$
CS-SP	Questionnaire	$H_0^1, H_0^4, H_0^7, H_0^8$
C-MATH	Registrar	$H_0^1, H_0^3, H_0^7, H_0^8$
CS-INTRO	Registrar	$H_{n7}^1, H_{n8}^2, H_n^3, H_n^4, H_n^5, H_n^6,$ H_0, H_0
CS-MAJOR	Registrar	$H_{n7}^1, H_{n8}^2, H_n^3, H_n^4, H_n^5, H_n^6,$ H_0, H_0

same class level as well as from an individual class of each university were also performed to further assess the data.

Pearson's product moment correlation coefficients (r) were calculated to examine relationships between investigated variables. The level of statistical significance was set at 0.05 for all statistical analyses. Since incomplete items were observed in several questionnaires, a pairwise deletion was used when dealing with missing values. While interpreting the results of the correlation coefficient analysis, statistical significance was not the only concern. The degree of the relationship was also examined for a possible indication of practical importance for education, as suggested by Borg and Gall (1989).

Concerning the identification of effective predictors for academic success in computer science programs at the college level, stepwise multiple-regression analysis was used. Manual control of which variables to be included in the model was also employed based on the knowledge of the importance in education of those variables. When a multiple-regression analysis is performed, the interpretation of an appropriate model achieved is defined as:

- 1) All the variables entering the prediction model, and the model itself, must achieve statistical significance, which was set at 0.05.
- 2) In determining whether or not to include a variable in the prediction model, the knowledge of

importance in education of the "candidate predictor" were also taken into consideration (based on the consulting results from statisticians).

- 3) With both the conditions met, the prediction model R-squared is maximized.

Specific statistical procedures required to verify individual research questions are described as follows:

- 1) Are college entrance examination scores (CEE) related to performance in college computer science programs?

The second hypothesis (H_0^2) was tested for this question. The correlation coefficient between student total CEE scores and average scores for computer science core courses were examined. As previously determined, significant relationships between student SAT scores (SAT-math in particular) and good academic performance in introductory computer science courses in the US have been reported. Thus, association between scores of specific CEE subject areas (particularly, the scores for math, English, chemistry, and physics) and student scores for the introductory course as well as average scores for computer science core courses were also of particular interest.

- 2) Is math ability related to performance in college computer science programs?

Variables to measure computer science performance included scores of introductory computer science courses (CS-INTRO), and average scores of computer science core

courses (CS-MAJOR). The average scores of college math courses taken by students as departmentally required courses (C-MATH) were first calculated. A correlation coefficient between the math ability variables (HS-MATH, CEE-MATH and C-MATH) and the computer science performance variables were examined, as described by the third hypothesis (H_0^3).

- 3) Is prior computer science experience related to performance in college computer science programs?

The fourth hypothesis (H_0^4) was tested for this research question, examining the correlation coefficient between the variables of prior computer experience and average scores as well as individual scores for computer science core courses (including introductory computer science courses). To understand student structured programming experience in more depth, interviews with a subsample of 17 students randomly selected were conducted. Information obtained from the interviews was used to verify the data gathered in the questionnaire, as well as to provide further information for interpreting the findings regarding prior computer experience.

- 4) Is overall high school performance related to performance in college computer science programs?

As noted previously, high school GPA has been a good predictor of performance in college introductory computer science courses in the US. However, high school performance has never been taken into consideration for college

admissions in the ROC. Hypothesis five (H_o^5) was tested by correlating HS-AVG to the scores of the introductory computer science courses and to the average scores of computer science core courses to determine if there is a significant relationship.

- 5) Is performance in introductory computer science courses related to overall performance in the computer science programs?

As previously stated, many universities have viewed student performance in introductory computer science courses as a predictor of future academic success in the complete computer science programs. However, the hypothesis for this relationship has never been verified empirically. The sixth hypothesis (H_o^6) was tested by correlating student scores in the introductory computer science courses to student average scores for computer science core courses.

- 6) Can reliable models be developed to predict performance in (a) introductory computer science courses, and (b) complete computer science programs? If so, can the equivalency of the two models be demonstrated?

Hypothesis seven (H_o^7) was tested in response to this research question. A correlation between variables and scores in introductory computer science courses and average scores for computer science core courses were examined. All the preadmission independent variables were considered

in the multiple-regression analysis, with student average scores for computer science core courses and scores in introductory computer science courses as the dependent variables for identifying the factors of academic success prediction for computer science majors. Both regression models were compared and factors entered into the predictive models were examined. If predictors related to performance in introductory computer science courses are not effective for predicting overall success in the computer science programs, then different predictive models may need to be employed.

- 7) Are there gender differences in performance predictors for computer science majors?

Hypothesis eight (H_0^8) was tested by performing a multivariate t-test on all CEE scores, high school performance variables (HS-AVG and HS-MATH), prior computer experience variables (CS-COURSE, CS-PROG, and CS-SP), and computer science performance variables (CS-INTRO and CS-MAJOR). In addition, the regression models for performance prediction were also compared between models with GENDER as the indicator variable.

CHAPTER IV RESULTS OF THE STUDY

Introduction

This study was designed to investigate predictive factors for academic achievement of college computer science majors in the Republic of China (ROC). A researcher-designed questionnaire was used to collect sample background information, including high school achievement factors. Eight questions were used to gather information regarding student computer experience prior to entering college. Scores from the College Entrance Examination (CEE) and college computer science courses were obtained from appropriate college registrar offices.

The study population consisted of 1,169 college computer science majors, including sophomore, junior and senior students currently enrolled at the participating universities. On the day the questionnaire was administered, 974 students were available and were surveyed. Since participation in the study was voluntary, 958 questionnaires were collected. Following careful and thorough examination, 18 questionnaires submitted by graduate students or noncomputer science majors were excluded. The actual sample size thus consisted of 940 subjects, including 796 males (85%) and 144 (15%) females, or a 5.5 to 1 male-female ratio. Consequently, the response rate to ques-

tionnaire administration was nearly 98%, assessing in excess of 81% of the selected population.

For reference purposes, the five universities were identified thereafter by the code letters A through E, where A, B, and C represented government-budgeted universities D and E were privately-funded universities. Since the computer science program at university C had been recently established, no senior level students were included from this program. Gender information for the subjects, by university and class level, is provided in Table 2.

Data collected were first entered into a spreadsheet for calculation of the average scores of all university math courses (C-MATH) and all core courses required by the computer science programs (CS-MAJOR). Responses to the prior computer experience items were quantified accordingly. Items regarding prior programming experience were summed to generate a score representing the amount of prior structured programming experience (CS-SP). The spreadsheet files for all five universities were then converted and combined into a STATGRAPHICS v7.0 file for statistical analysis. Hypotheses relating to specific research questions were tested. Results of the statistical analyses are described in the following section.

Table 2. Subjects by Gender, University and Class Participating in the Study.

	Male Subjects	Female Subjects	Total
<u>Combined</u>			
SO ¹	318	70	388
JU	261	46	307
SE	217	28	245
Subtotal:	796	144	940
<u>Univ. A</u>			
SO	50	8	58
JU	31	8	39
SE	38	8	46
Subtotal:	119	24	143
<u>Univ. B</u>			
SO	80	15	95
JU	62	9	71
SE	48	2	50
Subtotal:	190	26	216
<u>Univ. C²</u>			
SO	32	7	39
JU	30	2	32
Subtotal:	62	9	71
<u>Univ. D</u>			
SO	74	20	94
JU	77	13	90
SE	68	9	77
Subtotal:	219	42	261
<u>Univ. E</u>			
SO	82	20	102
JU	61	14	75
SE	63	9	72
Subtotal:	206	43	249

Notes:

¹ SO = Sophomore, JU = Junior, SE = Senior.

² University C program recently established; thus, no senior-level students are included.

Results of Statistical Analyses

Prior to the performance of other statistical analyses, the existence of significant differences in the variables investigated among the participating universities and among class levels was determined. That is, if significant differences were found to exist, then further analyses would need to be performed based on university attended and current class level. Therefore, the following hypothesis was first tested:

H_0^1 There are no significant differences among the sample universities in terms of each of the variables investigated.

For H_0^1 , a significant difference for total scores of the College Entrance Exam (CEE-TOTAL) was found between universities ($F = 4697.69$, $p < .001$) as all five universities differed from one another on CEE-TOTAL using a multiple-range test. Moreover, nearly all of the scores for CEE subject tests were also significantly different between universities and the various class levels (for all cases, $F > 43.63$, $p < .001$). Similar results were found for scores from introductory computer science courses (CS-INTRO) and CS-MAJOR. As a result, subsequent statistical analyses were performed according to current class level. Thus, analyses for all the students within the same class level as a group and for students within individual class levels at each university were computed separately. Hypo-

theses testing was then organized based upon the order in which the research questions have been listed.

CEE Scores

- 1) Are college entrance examination scores related to performance in college computer science programs?

To answer the question above, the following hypothesis was tested:

H_0^2 There are no significant relationships between CEE scores and scores in the introductory computer science courses or average scores of computer science core courses.

By separately computing correlation coefficients based upon university attended as well as separate class levels, the relationship between CEE scores and the variable academic achievement in college computer science programs was examined. The results are summarized in Tables 3 and 4.

A significant relationship between the scores for CEE English (CEE-ENG) and the scores for introductory computer science courses (CS-INTRO) was found for all sophomore ($r = .39, p < .001$), junior ($r = .19, p < .001$) and senior groups ($r = .26, p < .001$). However, when the relationships among individual classes within each university were examined, a significant relationship was found only for the sophomore ($r = .41, p < .005$) and senior levels ($r = .52,$

Table 3. Correlation between CEE Scores and Scores in the Introductory Computer Science Courses (CS-INTRO).¹

	ENG	MATH	PHY	CHEM	TOTAL ²
<u>Combined</u>					
SO ³	.39***	.10	.24***	.24***	.39***
JU	.19***	.06	.17**	.27***	.27***
SE	.26***	.22***	.17**	.29***	.31***
<u>Univ. A</u>					
SO	.41**	-.06	.13	-.05	-.10
JU	.08	-.27	.06	.20	.20
SE	.52**	-.35*	-.39**	.02	.10
<u>Univ. B</u>					
SO	.13	-.03	-.05	.13	.27*
JU	-.16	.03	.08	.21	-.28*
SE	.18	.16	.10	-.08	.23
<u>Univ. C</u>					
SO	.27	-.34*	-.01	-.12	.01
JU	.12	-.25	.16	-.05	.19
<u>Univ. D</u>					
SO	.32**	-.24*	-.24*	-.04	-.03
JU	.17	-.35***	-.14	.02	-.07
SE	.07	.12	-.05	.16	.11
<u>Univ. E</u>					
SO	.03	-.06	-.01	.12	.21*
JU	.01	.05	.06	.36**	.30*
SE	-.16	.06	.10	.13	-.01

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² ENG = English; MATH = Mathematics; PHY = Physics; CHEM = Chemistry.

³ SO = Sophomore; JU = Junior; SE = Senior.

$p < .005$) of university A and for the sophomore level of university D ($r = .32$, $p < .005$).

It was of interest that a similar pattern was also found between CEE-ENG and CS-MAJOR. Correlation coefficients of .23, .20, and .28 were observed for the respective class levels ($p < .001$ for all cases). A significant relationship was also found for the sophomore ($r = .36$,

Table 4. Correlation between CEE Scores and Overall Performance in Computer Science Programs (CS-MAJOR).¹

	ENG	MATH	PHY	CHEM	TOTAL ²
<u>Combined</u>					
SO ³	.23***	.11*	.21***	.14***	.26***
JU	.20***	.19***	.33***	.37***	.40***
SE	.28***	.31***	.27***	.30***	.44***
<u>Univ. A</u>					
SO	.36*	-.01	.13	-.06	-.22
JU	.05	-.19	.08	.20	.24
SE	.43**	-.25	-.46***	.01	-.01
<u>Univ. B</u>					
SO	-.04	-.01	-.01	.17	.21
JU	-.12	-.06	.23	.27*	-.10
SE	.16	.20	.11	-.09	.33*
<u>Univ. C</u>					
SO	.20	-.30	.05	.24	.06
JU	-.14	-.36*	.30	.23	.05
<u>Univ. D</u>					
SO	.15	-.04	-.15	-.09	-.01
JU	.03	-.11	.03	.09	-.00
SE	-.01	.10	.03	.20	.15
<u>Univ. E</u>					
SO	-.09	.01	-.03	.22*	.23*
JU	.02	.17	.03	.25*	.26*
SE	-.16	.06	.10	.13	-.01

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² ENG = English; MATH = Mathematics; PHY = Physics; CHEM = Chemistry.

³ SO = Sophomore; JU = Junior; SE = Senior.

$p < .05$) and senior levels ($r = .43$, $p < .005$) for university A, similar to the findings for CS-INTRO.

A significant relationship was found between CEE-MATH and CS-INTRO for the senior group ($r = .22$, $p < .001$). CEE-MATH was also found to correlate with CS-MAJOR significantly for all sophomore ($r = .11$, $p < .05$), junior ($r = .19$, $p < .001$) and senior levels ($r = .31$, $p < .001$). Thus, when correlation coefficients for the individual

classes of each university were examined, the result that CEE-MATH correlated negatively with both CS-INTRO and CS-MAJOR whenever a significant coefficient was observed was an unexpected result.

A similar pattern was observed in the relationship between the scores for CEE physics (CEE-PHY) and CS-INTRO, to scores for CS-MAJOR. For CS-INTRO, a significant correlation was found for all sophomore ($r = .24$, $p < .001$), junior ($r = .17$, $p < .005$), and senior class levels ($r = .17$, $p < .01$). CS-MAJOR was found to correlate significantly with CEE-PHY at 0.21, 0.33, and 0.27, respectively, for the three different classes ($p < .001$ for all cases). However, when results for individual classes were analyzed, significantly negative correlation coefficients were found between CEE-PHY and CS-INTRO with respect to CS-MAJOR.

The score for CEE chemistry (CEE-CHEM) was found to significantly associate with CS-INTRO for all sophomore ($r = .24$), junior ($r = .27$), and senior groups ($r = .29$). CEE-CHEM was also correlated to CS-MAJOR for all classes ($r = .21$, $.33$, and $.27$, respectively, $p < .001$ for all cases). A degree of negative correlation coefficient was observed between CEE-CHEM and CS-INTRO to CS-MAJOR when the individual classes from each university were analyzed. However, in contrast to results found for CEE-MATH and CEE-PHY, CEE-CHEM was associated with CS-INTRO for only the junior class ($r = .36$, $p < .005$) and with CS-MAJOR for only

the sophomore ($r = .22$, $p < .05$) and junior classes ($r = .25$, $p < .05$) at university E.

The CEE-TOTAL was also associated significantly with CS-INTRO for all class groups ($r = .39$, $.27$, and $.31$, respectively, for the sophomore, junior, and senior classes). For individual classes, four correlation coefficients reached the significance level. Correlation coefficients of 0.21 and 0.30 were found for the sophomore and junior levels at university E, respectively ($p < .05$ for each case). A positive correlation was also found for the sophomore level at university B ($r = .27$, $p < .05$). However, for the junior class at university B, CEE-TOTAL was negatively correlated to CS-INTRO ($r = -.28$, $p < .05$). No other observations reached levels of significance in relationship to individual classes.

For CS-MAJOR, a significant association was found with CEE-TOTAL for all sophomore, junior, and senior groups ($r = .26$, $.40$, and $.44$, respectively, $p < .001$ for all cases). CEE-TOTAL was significantly related to the sophomore and junior levels at university E, and to the senior level at university B ($r = .23$, $.26$ and $.33$, respectively, $p < .05$ for all cases). No other correlation coefficients between CEE-TOTAL and CS-MAJOR for the other classes reached levels of significance.

In summary, several significant relationships were determined to exist between the CEE variables and performance in college computer science programs. However, the

findings for individual classes were not consistent with the results found for the class level groups. Therefore, the findings for this question were not conclusive. Moreover, with correlation coefficients below 0.40 for most of the cases, it would be difficult to suggest that a strong relationship existed between student CEE performance and student academic achievements in college computer science programs.

Math Ability

- 2) Is math ability related to performance in college computer science programs?

To determine results for this question, the following hypothesis was tested:

H_0^3 There are no significant relationships between math ability variables and scores in the introductory computer science courses or average scores of computer science core courses.

To determine the relationship between math ability and college performance, hypothesis three (H_0^3) was tested. Scores of overall high school math courses (HS-MATH), overall college math courses (C-MATH), and CEE-MATH were correlated with student performance in college computer science programs (CS-INTRO and CS-NONMATH).

The results of an analysis of variance (ANOVA) in conjunction with findings from a multiple range test for the

two variables, CS-INTRO ($F = 11.76$, $p < .001$) and CS-MAJOR ($F = 18.05$, $p < .001$), showed that there was a significant difference between class levels. Correlation analyses were again completed by individual university and separate class levels. Table 5 presents a summary of the average scores for HS-MATH, C-MATH and the college performance variables (C-MATH, CS-INTRO and CS-MAJOR). The correlation coefficients relating to the H_0^3 testing are summarized in Table 6.

Table 5. Average Scores for Math-Related Variables and College-Performance Variables.

	CEE-MATH	HS-MATH	C-MATH	CS-INTRO	CS-MAJOR ¹
Combined	58.00	73.25	68.20	71.12	70.51
Univ. A	71.74	77.18	71.26	79.69	75.55
Univ. B	67.12	76.15	73.87	74.40	74.09
Univ. C	66.17	71.80	58.73	70.71	68.29
Univ. D	51.31	70.46	65.73	70.09	65.73
Univ. E	50.77	71.54	68.53	68.68	68.39

Notes:

¹ CEE-MATH = CEE mathematics; HS-MATH = high school mathematics; C-MATH = college mathematics; CS-INTRO = introductory computer science courses; CS-MAJOR = average scores of core courses for computer science majors.

Although HS-MATH was found to reach a level of significance level in correlation with CS-INTRO for all class groups ($r = .20$, $.12$ and $.20$, respectively, for sophomore, junior, and senior), all of the coefficients were at a level of $.20$ or lower. Such low coefficients, though

Table 6. Correlation between Math Ability and Performance in College Computer Science Programs.¹

	HS-MATH			C-MATH ²	
	CS-INTRO	C-MATH	CS-MAJOR	CS-INTRO	CS-NONMATH
<u>Combined</u>					
SO ³	.20***	.41***	.38***	.35***	.36***
JU	.12*	.45***	.41***	.37***	.58***
SE	.20**	.46***	.44***	.47***	.63***
<u>Univ. A</u>					
SO	.48***	.51***	.53***	.68***	.71***
JU	-.02	.56***	.37*	.58***	.72***
SE	.24	.51***	.47**	.59***	.76***
<u>Univ. B</u>					
SO	.17	.33**	.36***	.25*	.32**
JU	.12	.39***	.33**	.31*	.56***
SE	.09	.38**	.42**	.48***	.70***
<u>Univ. C</u>					
SO	.17	.27	.29	.35*	.24
JU	.12	.24	.25	.59***	.58***
<u>Univ. D</u>					
SO	.02	.35***	.21	.28**	.21*
JU	-.01	.47***	.30**	.22*	.58***
SE	.10	.37***	.33**	.45***	.70***
<u>Univ. E</u>					
SO	.01	.43***	.31**	.37***	.54***
JU	-.07	.43***	.36**	.35**	.54***
SE	.31*	.44***	.51***	.34*	.62***

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² HS-MATH = high school math; C-MATH = college math; CS-INTRO = introductory computer science courses; CS-MAJOR = average scores of computer science core courses; CS-NONMATH = average scores of computer science core courses with math courses excluded.

³ SO = Sophomore; JU = Junior; SE = Senior.

significant, provide little of practical value that may be concluded for educational purposes. Moreover, for the individual classes of each university, only the correlation coefficients for sophomores at university A ($r = .48$, $p < .005$) and for seniors at university E ($r = .31$, $p < .05$) were found to be significant. Therefore, the correlation coefficients achieved were too weak to provide

evidence for the existence of a significant relationship between HS-MATH and CS-INTRO.

To the contrary, HS-MATH was consistently found to associated significantly with CS-MAJOR for almost all cases, with the exception of sophomore and junior classes at university C ($r = .29$ and $.25$, respectively) and sophomores at university D ($r = .21$, $p = .05$). With respect to the correlation of HS-MATH with C-MATH, the coefficients for both sophomores and juniors at university C were not significant. However, for other classes as well as all class level groups, HS-MATH was significantly correlated to C-MATH. Moreover, there was an increasing level in correlation coefficients by class level between HS-MATH and C-MATH (r increased from 0.41 to 0.46) and between HS-MATH and CS-MAJOR (r increased from 0.38 to 0.44).

The findings for the relationship between C-MATH and CS-INTRO were not anticipated insofar as a significant correlation existed between C-MATH and CS-INTRO for all class level groups (r range from 0.35 to 0.47 , $p < .001$ for all cases) and for all individual classes. For the students of university A, the correlation coefficient for this relationship was in excess of 0.58 ($P < .001$).

To generate results for CS-MAJOR, scores for math courses were also included to calculate average scores for all computer science core courses. To determine the relationship between C-MATH and other nonmath computer science courses, a new variable, CS-NONMATH, was used to develop

the correlation analysis. Thus, a strong relationship was found between C-MATH and CS-NONMATH with a correlation coefficient in excess of 0.70 for several cases. The level of correlation increased by class level from the sophomore to the senior groups; that is, from 0.36 to 0.58 to 0.63, respectively ($p < .001$ for all three cases). The same pattern was also observed for almost all of the classes at the various universities included in the sample.

Though a significant relationship between HS-MATH and CS-INTRO could not be determined, the results obtained for H_0^3 supported the assumption that math ability can be correlated to performance in college computer science programs. Furthermore, the ascending pattern of relationships by class level between the math ability variables and college performance seems to suggest that as more computer science courses were taken, the importance of math ability became more evident.

It was also of interest to note that the students at university A obtained the highest coefficients by a substantial margin for all of the correlation between the math ability variables and the college performance variables in almost all cases. Moreover, the students from university A also had the highest HS-MATH scores from among subjects from all the universities. These findings imply that good math abilities can be of benefit to student performances in college computer science programs. It may also be hypothesized that university A employs a stronger, math-

oriented curriculum, especially for the introductory computer science courses, than do other universities for their respective computer science programs.

Prior Computer Experience

- 3) Is prior computer science experience related to performance in college computer science programs?

More than 60% (572 of 940 students) of the subjects had obtained some computer experience from a variety of sources prior to entering college computer science programs. Approximately 50% (463 of 940 students) of college computer science freshmen entered their programs with some degree of prior programming experience. The following hypothesis was tested in response to the research question addressed above.

H_0^4 There are no significant relationships between prior computer experience and scores in the introductory computer science courses or average scores of computer science core courses.

To test H_0^4 , computer courses taken (CS-COURSE), programming courses taken (CS-PROG), and experience in structured programming (CS-SP) prior to entering college were correlated with college performance as measured by CS-INTRO and CS-MAJOR. The variable CS-SP was calculated by adding all the scores for each of the items in the question set in which student experiences in structured programming were

assessed. The results of this analysis are summarized in Table 7.

Table 7. Correlation between Prior Computer Experience and College Performance.¹

	CS-INTRO			CS-MAJOR ²		
	CS-COURSE	CS-PROG	CS-SP	CS-COURSE	CS-PROG	CS-SP
<u>Combined</u>						
SO ³	.17**	.16**	.11	.14**	.11*	.02
JU	.20**	.19**	.12	.11	.07	.06
SE	.10	.17*	.12	.14**	.11	.19*
<u>Univ. A</u>						
SO	.26	.16	-.30	.38**	.26	-.23
JU	.19	.14	-.11	.17	.06	-.01
SE	-.09	-.05	-.09	.06	-.02	.07
<u>Univ. B</u>						
SO	.09	.19	.18	-.06	-.01	.03
JU	.26*	.32**	.35	-.03	.02	.45*
SE	.26	.33*	.32	.32*	.34*	.24
<u>Univ. C</u>						
SO	.32	.19	.11	.30	.21	.17
JU	.47*	.35	-.49	.37	.38*	-.09
<u>Univ. D</u>						
SO	.10	.11	.22	.14	.09	.17
JU	.08	.04	.14	.02	-.06	.10
SE	.24*	.24*	.17	.24*	.31**	.18
<u>Univ. E</u>						
SO	.22*	.18	.03	.07	.04	-.17
JU	.21	.27*	.09	.08	.11	-.21
SE	.15	.14	.23	.34**	.24	.50***

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² CS-INTRO = introductory computer science courses; CS-MAJOR = average scores of computer science core courses; CS-COURSE = number of computer courses taken prior to entering college; CS-PROG = number of programming courses taken prior to entering college; CS-SP = experience in structured programming.

³ SO = sophomore; JU = Junior; SE = Senior.

Though a significant correlation was found between CS-COURSE and CS-INTRO for the sophomore and junior groups, the coefficient obtained was less than .20 for both cases ($p < .005$). A significant correlation between CS-COURSE

and CS-INTRO was found for individual classes in four cases. However, the correlation coefficient for juniors at university C was 0.47 ($p < .05$), whereas those for the remaining classes were below 0.30.

CS-COURSE was also found to have a low though significant correlation with CS-MAJOR for the sophomore ($p < .01$) and senior ($p < .05$) class groups. In both cases, the correlation coefficient was less than 0.15. However, three of four individual classes obtained correlation coefficients greater than 0.30 for the relationship between CS-COURSE and CS-MAJOR. Results similar to those for CS-COURSE were found for CS-PROG. All class level groups were found to have significant but low relationships between CS-PROG and CS-INTRO ($r < .20$). However, among the significant correlation coefficients, only the juniors and seniors of university B had correlation coefficients greater than 0.30 ($p < .05$ for both cases). Even lower correlation coefficients were found between CS-PROG and CS-MAJOR for the class level groups, and only the correlation for the sophomore group reached the level of significance ($r = .11$). However, three of four classes where a significant relationship was found between CS-PROG and CS-MAJOR obtained correlation coefficients greater than 0.30.

With respect to the relationship between prior experience in structured programming (CS-SP) and college computer science performance (CS-INTRO and CS-MAJOR), none of the class groups nor individual classes had significant

relationships between the variables CS-SP and CS-INTRO. For the correlation between CS-SP and CS-MAJOR, only the senior group relationship was significant ($r = .19$, $p < .05$). In these cases, the two classes that reached the significance level achieved correlation coefficient of .45 ($p < .05$) and .50 ($p < .001$).

To examine more closely the relationship between prior computer experience and college performance in computer science programs, the sample was classified into three groups for further analysis. However, only the college performance variables (CS-INTRO and CS-MAJOR) for students who took more than two computer courses prior to entering computer science programs (formed as group A) and students without any computer experience prior to entering college (formed as group B) were compared. Two-tailed t -tests, with significance level set at 0.05, were used for the comparisons. A summary of the results obtained is given in Table 8.

When all the samples were included for comparison, a significant group difference for the CS-INTRO mean scores was obvious. The CS-INTRO mean score for group A was significantly higher than for group B on both the CS-COURSE (77.1 versus 70.7, $p < .001$) and CS-PROG (77.3 versus 70.6, $p < .05$) tests. Similar results were found on analysis of

Table 8. Average Score Differences for CS-INTRO and CS-MAJOR Between Students with Different Amounts of Prior Computer Experience.¹

	CS-INTRO			CS-MAJOR ²		
	A ³	B	Group Diff.	A	B	Group Diff.
<u>CS-COURSE</u> Combined	77.1 (n=38)	70.7 (n=350)	6.4***	74.2 (n=55)	69.8 (n=351)	4.4***
SO ⁴	77.6 (n=22)	70.0 (n=137)	7.6***	73.2 (n=22)	68.6 (n=137)	4.6*
JU	77.3 (n=10)	69.9 (n=124)	7.4	73.4 (n=12)	69.8 (n=124)	3.6
SE	74.4 (n=5)	73.1 (n=85)	1.3	75.7 (n=19)	72.0 (n=86)	3.7
Univ. E	73.8 (n=6)	67.7 (n=107)	6.1	74.0 (n=21)	67.5 (n=107)	6.6***
SE	72.0 (n=2)	67.8 (n=121)	4.2	74.7 (n=15)	67.3 (n=21)	7.4**
<u>CS-PROG</u> Combined	77.3 (n=11)	70.6 (n=460)	6.7*	74.9 (n=26)	69.9 (n=462)	4.9**
SO	78.4 (n=6)	69.9 (n=199)	8.5*	74.6 (n=6)	68.7 (n=199)	5.9
JU	74.5 (n=2)	70.0 (n=150)	4.5	68.1 (n=4)	69.9 (n=150)	-1.8
SE	77.0 (n=3)	73.3 (n=106)	3.7	76.7 (n=16)	72.4 (n=108)	4.3*
Univ. E	-	-	-	74.2 (n=14)	67.67 (n=134)	6.6**
SE	-	-	-	75.2 (n=13)	68.4 (n=26)	6.8**

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² CS-INTRO = introductory computer science courses; CS-MAJOR = average scores of computer science core courses.

³ A = students with more than two computer or programming courses taken prior to entering college; B = students with no prior computer experience.

⁴ SO = Sophomore; JU = Junior; SE = Senior.

CS-MAJOR, where group A students were found to have significantly higher CS-MAJOR mean scores for both CS-COURSE (74.2 versus 69.8, $p < .001$) and CS-PROG (74.9 versus 69.9, $p < .005$) tests.

From the results it was also consistently found that the mean scores of group A were higher than the mean scores of group B for the different class groups for almost all tests. However, only the sophomore group achieved CS-COURSE significance level for both CS-INTRO (77.6 versus 70.0, $p < .001$) and CS-MAJOR (73.2 versus 68.6, $p < .05$), as well as CS-PROG significance for CS-INTRO (78.4 versus 69.9, $p < .05$). The CS-PROG group difference for seniors for CS-MAJOR was also significant (76.7 versus 72.4, $p < .05$).

On average, students enrolled at university E had more prior computer experience than had students from other universities. Therefore, the university E sample was analyzed to determine if similar results could be found. Again, the students in group A had significantly higher CS-MAJOR mean scores than did those in group B for both CS-COURSE (74.1 versus 67.5, $p < .001$) and CS-PROG (74.2 versus 67.6, $p < .005$). Even when the seniors of groups A and B from university E were compared, the same results were obtained for both CS-COURSE (74.7 versus 67.3, $p < .005$) and CS-PROG (75.2 versus 68.4, $p < .01$).

Though several significant correlation coefficients were found, results for the relationship between prior computer experience and college performance in computer science programs were not conclusive. However, a close correlation could not be determined due to the low coefficients obtained in most of the cases. Nevertheless, the mean

scores of students with more than two computer courses taken prior to entering college were found to be consistently higher than those of students with no prior computer experience. This finding suggests that having computer experience prior to entering college computer science programs may exercise a positive benefit upon future performance.

Overall High School Performance

- 4) Is overall high school performance related to performance in college computer science programs?

Hypothesis five (H_o^5) was tested in response to the above research question:

H_o^5 There are no significant relationships between averages for all high school course work and scores in the introductory computer science courses or average scores of computer science core courses.

Table 9 summarizes the results of correlation analyses between high school achievement and college performance in computer science programs. Significant correlations were found between high school overall performance (HS-AVG) and CS-INTRO for the sophomore ($r = .22, p < .001$) and senior groups ($r = .21, p < .005$). However, only sophomores from

Table 9. Correlation Between Overall High School Performance (HS-AVG) and CEE-TOTAL and College Performance (CS-INTRO and CS-MAJOR).¹

	CEE-TOTAL	CS-INTRO	CS-MAJOR ²
<u>Combined</u>			
SO ³	.31***	.22***	.36***
JU	.35***	.09	.41***
SE	.38***	.21**	.48***
<u>Univ. A</u>			
SO	.10	.48**	.50***
JU	.04	.09	.42**
SE	.11	.22	.46**
<u>Univ. B</u>			
SO	.08	.19	.37***
JU	.29*	-.13	.26*
SE	.29	.03	.33*
<u>Univ. C</u>			
SO	.26	.04	.14
JU	.12	.19	.26
<u>Univ. D</u>			
SO	.01	.06	.24*
JU	.08	-.01	.30**
SE	.04	.08	.39***
<u>Univ. E</u>			
SO	.14	.01	.21*
JU	.05	-.11	.30*
SE	.08	.25	.51***

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² CS-INTRO = introductory computer science courses; CS-MAJOR = average scores of computer science core courses.

³ SO = Sophomore; JU = Junior; SE = Senior.

university A achieved a significance level for the correlation coefficient between HS-AVG and CS-INTRO ($r = .48$, $p < .005$). For some classes, a negative correlation resulted. Given the low scores for most of the correlation coefficients, the findings did not support a significant relationship between HS-AVG and CS-INTRO.

Contrary to the findings for CS-INTRO, a positive relationship between HS-AVG and CS-MAJOR was determined

from an increasing level of correlation coefficients (in the range 0.36 to 0.48, $p < .001$ for all cases) for the different class groups. With the exception of university C, consistently significant correlations between HS-AVG and CS-MAJOR were also found for almost all of the individual classes from the different universities. Therefore, the hypothesis that there are no significant relationships between overall high school performance and average scores in college computer science core courses was rejected. Although only limited variance can be accounted for when HS-AVG was considered by itself, a significant relationship between HS-AVG and CS-MAJOR was supported.

Introductory Computer Science Courses

- 5) Is performance in introductory computer science courses related to overall performance in the computer science programs?

The following hypothesis was tested in response to the question listed above:

- H_0^6 There are no significant relationships between scores in the introductory computer science courses and average scores of computer science core courses.

Results for this correlation analysis are provided in Table 10. A very high correlation was found between CS-INTRO and CS-MAJOR in all the cases (r ranged from 0.54

Table 10. Correlation Between CS-INTRO and Overall Course Performance (CS-MAJOR and CS-NOBCC).¹

	CS-MAJOR	CS-NOBCC ²
<u>Combined</u>		
SO ³	.78***	.50***
JU	.66***	.50***
SE	.69***	.61***
<u>Univ. A</u>		
SO	.85***	.74***
JU	.83***	.77***
SE	.77***	.70***
<u>Univ. B</u>		
SO	.58***	.40***
JU	.66***	.44***
SE	.70***	.60*
<u>Univ. C</u>		
SO	.88***	.63***
JU	.82***	.71***
<u>Univ. D</u>		
SO	.81***	.40***
JU	.54***	.32***
SE	.63***	.51***
<u>Univ. E</u>		
SO	.72***	.50***
JU	.64***	.52***
SE	.64***	.56***

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² CS-INTRO = introductory computer science courses; CS-MAJOR = average scores of computer science core courses; CS-NOBCC = average scores of computer science core courses with CS-INTRO excluded.

³ SO = Sophomore; JU = Junior; SE = Senior.

to 0.88). It was reasonable to conclude that the strength of this relationship may have been due in part to the factor of self-correlation. Therefore, a new variable, CS-NOBCC, wherein the scores of CS-INTRO were excluded from the calculation of CS-MAJOR, was generated to retest H_0 .⁶

Following this change in data analysis, a significantly positive though somewhat lower correlation was again

found between CS-INTRO and CS-NOBCC. For all three class groups, CS-INTRO was found to correlate significantly with CS-NOBCC ($r = .50, .50, \text{ and } .61$, respectively). With respect to the individual classes from the different universities, CS-INTRO was still found to correlate strongly with CS-NOBCC for all cases (r ranging from 0.32 to 0.74). In view of the consistent nature of these findings, a close relationship between performance in introductory computer science courses and success in complete computer science programs was confirmed.

Prediction Models

- 6) Can reliable models be developed to predict performance in (a) introductory computer science courses, and (b) complete computer science programs? If so, can the equivalency of the two models be demonstrated?

To answer the above question, multiple regression analysis was performed and the following hypothesis was tested:

- H_0 ⁷ There is no significant linear predictive model for introductory computer science courses or complete computer science programs.

With all the subjects included in the regression analysis, a model with the selected variables HS-AVG, CEE-TOTAL, CS-PROG and GENDER was generated ($R^2 = .24$).

Further model analyses for GENDER, CLASS and UNIVERSITY as indicator variables were used to identify gender differences and differences due to university and class level.

Basically, different CEE question sets on each subject tests are used each year, resulting in CEE score differences for each of the different class groups. Moreover, the long-term prediction of academic achievement for college computer science majors was the primary concern of this research. Therefore, the regression models for this prediction purpose were focused upon models for senior level students. Models for the senior classes from the individual universities were also generated for in-depth examinations of possible differences due to university of enrollment. A regression model was not generated for university C since no seniors were enrolled in computer science in this university at the time of testing. Results of the regression analyses are summarized in Table 11.

High school performance variables, either HS-AVG or HS-MATH, were selected into the models for all the class groups when the highest R^2 was obtained. Scores of various CEE subject tests were selected into the prediction models for different class level groups. Several combinations of variables selected into the model were examined to determine which models could be used to effectively predict student CS-MAJOR. Only a slight change in R^2 was found when HS-AVG was used to substitute for HS-MATH, and when CEE-TOTAL was used to substitute for different scores of

Table 11. Regression Models for CS-MAJOR Prediction.^a

	Original Model ^b		Modified Model		CS-INTRO Model	
	Predict.	R ²	Predict.	R ²	Predict.	R ²
<u>Combined</u>						
SO ^c	HS-MATH CEE-ENG CEE-PHY GENDER	.19***	HS-AVG CEE-TOTAL	.17***	HS-AVG CEE-TOTAL CS-INTRO	.63***
JU	HS-AVG CEE-PHY CEE-CHEM	.28***	HS-AVG CEE-TOTAL	.27***	HS-AVG CEE-TOTAL CS-INTRO	.59***
SE	HS-AVG CEE-TOTAL	.30***	-	-	HS-AVG CEE-TOTAL CS-INTRO	.59***
<u>Univ. A</u>						
SE	HS-MATH CEE-PHY CEE-MATH	.43***	HS-AVG CEE-TOTAL	.27***	HS-AVG CEE-TOTAL CS-INTRO	.73***
<u>Univ. B</u>						
SE	HS-MATH	.09***	HS-AVG CEE-TOTAL	.10***	HS-AVG CEE-TOTAL CS-INTRO	.50***
<u>Univ. D</u>						
SE	HS-AVG CS-PROG	.20***	HS-AVG CEE-TOTAL	.12***	HS-AVG CEE-TOTAL CS-INTRO	.46***
<u>Univ. E</u>						
SE	HS-AVG	.23***	HS-AVG CEE-TOTAL	.25***	HS-AVG CEE-TOTAL CS-INTRO	.44***

Notes:

^a * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

^b Original model = original fitted model with the highest R² achieved; modified model = model refitted with only HS-AVG and CEE-TOTAL included; CS-INTRO model = modified model refitted with CS-INTRO added as a predictor.

^c SO = Sophomore; JU = Junior; SE = Senior.

specific CEE subject tests. The results of these substitutions indicated that if only HS-AVG and CEE-TOTAL were used to predict student performance in college computer science programs, similar results would be obtained.

A model predicting CS-MAJOR, with HS-AVG, CEE-TOTAL, CS-PROG, and GENDER selected, was generated for the senior groups ($R^2 = .31$, $F = 24.13$). However, when only HS-AVG and CEE-TOTAL were selected for the model, the model R^2 decreased slightly to 0.30 and the GENDER difference found in the full model previously described was nonsignificant. When the variables CS-PROG and CEE-TOTAL were included in the model to predict CS-INTRO for the senior groups, a considerably lower R^2 was obtained ($R^2 = .11$, $F = 13.22$).

A similar result was also found for the senior classes from individual university. Variables for high school achievement were included in the prediction model for CS-MAJOR in all cases. It was of interest to note that while HS-MATH was selected by the models for universities A and B, HS-AVG was included in the models for universities D and E, both of which were privately-funded universities. The model R^2 (0.43) obtained for university A was higher than those found for the other universities. However, insofar as the R^2 levels were less than 0.30 in most cases, the practical value of using these models for performance prediction was limited.

Model findings for predicting CS-INTRO differed insofar as they appeared to be university-dependent. An R^2 of 0.38 was observed in the model for the seniors of university A. However, CS-INTRO performance did not prove to be so predictable for the other universities (i.e., for university B, $R^2 = 0.07$), and no prediction model was gener-

ated for either university D or E. The results found for the CS-INTRO prediction indicated that predicting student performance in introductory computer science courses using the variables investigated in this study was not appropriate. Moreover, the prediction models for CS-INTRO were not equivalent to the models generated for CS-MAJOR prediction.

Otherwise, CS-INTRO was closely related to CS-MAJOR. In search of an improved prediction model for CS-MAJOR, the regression models previously generated for the class groups and for the individual classes were all reanalyzed with the addition of the variable CS-INTRO. An unanticipated result was that R^2 values of 0.60 or higher were found for the different class groups. For individual senior classes, $R^2 = 0.44$ or higher was observed in all models. For university A, the model R^2 reached a high value of 0.73. These findings suggest that the prediction of CS-MAJOR could be achieved more effectively if CS-INTRO was included in the prediction model. Figure 1 presents the graphical results of CS-MAJOR prediction for seniors of individual university in relation to overall model for combined senior group. Figures 2 to 4 exhibit the prediction models of individual class level groups with different lines for male and female participants.

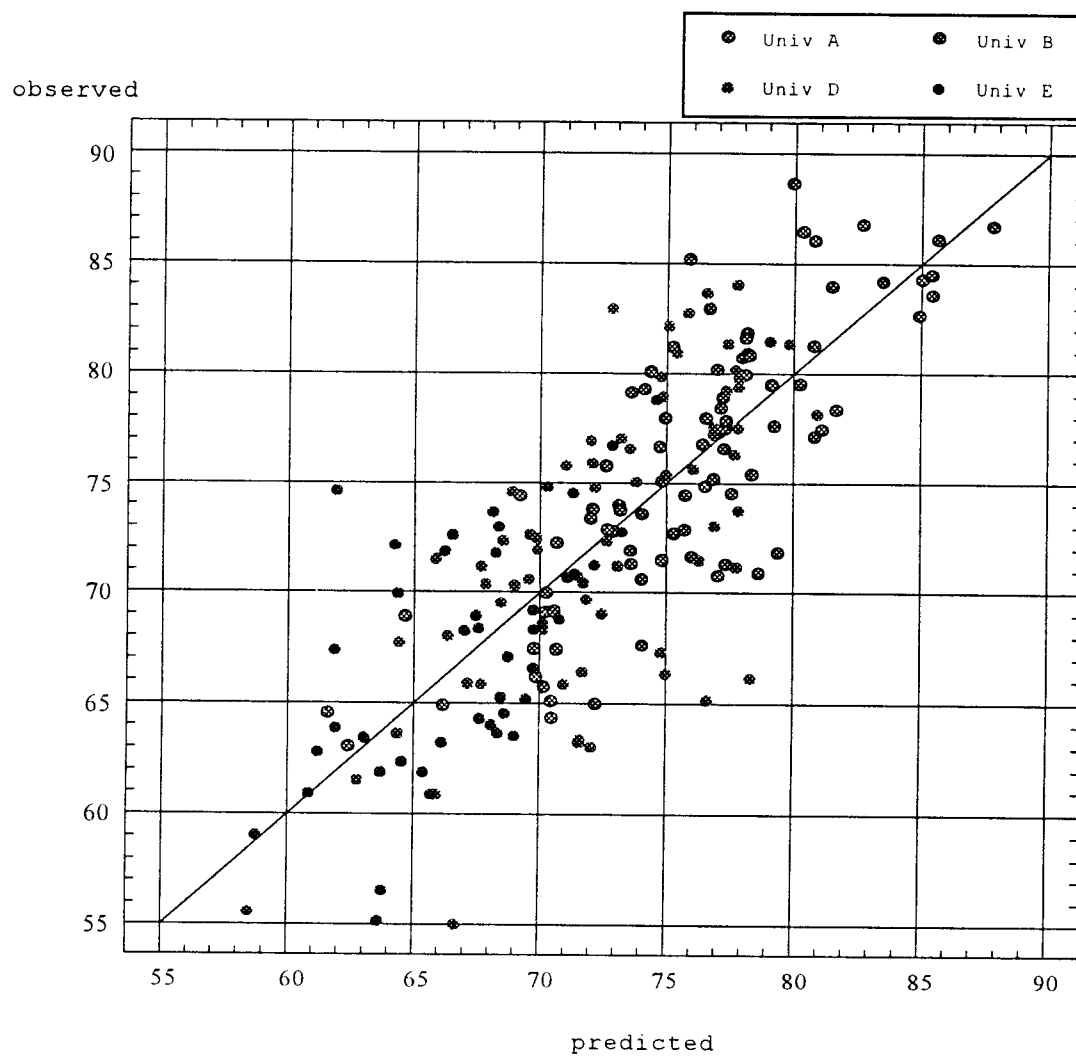


Figure 1. Plot of CS-INTRO Model for CS-MAJOR Prediction for Senior Classes by University Enrolled.

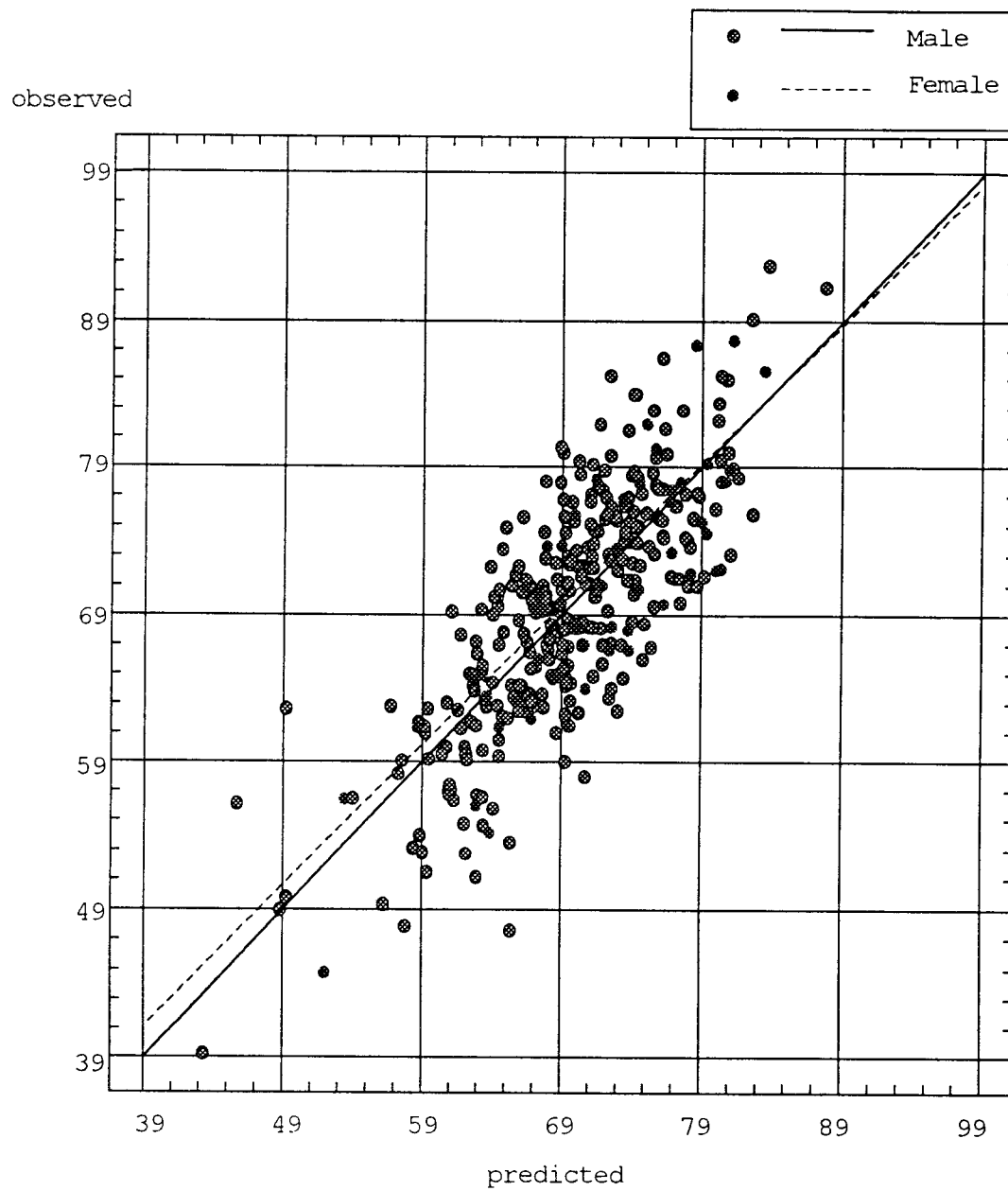


Figure 2. Plot of CS-INTRO Model for CS-MAJOR Prediction for Combined Sophomore Group.

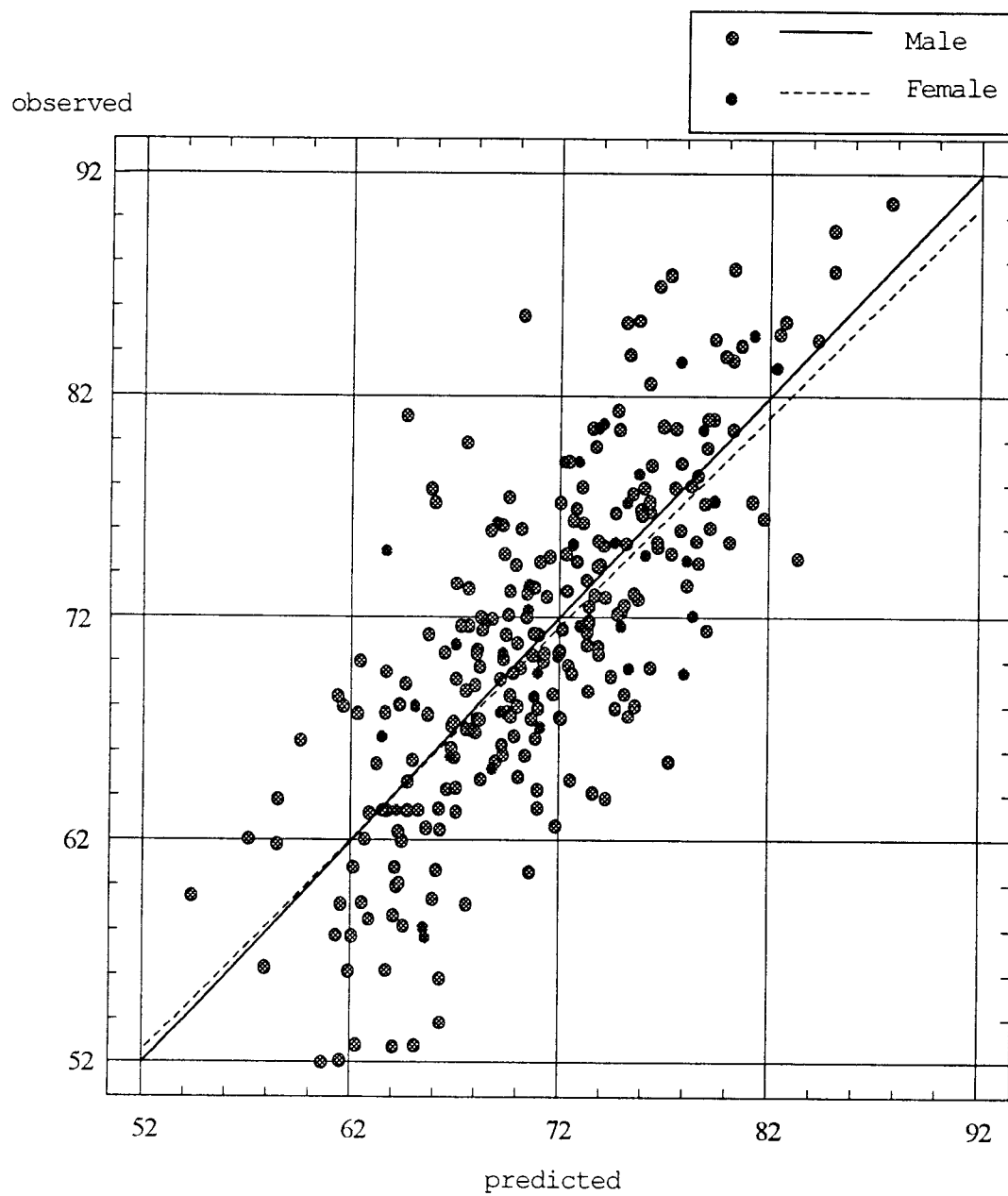


Figure 3. Plot of CS-INTRO Model for CS-MAJOR Prediction for Combined Junior Group.

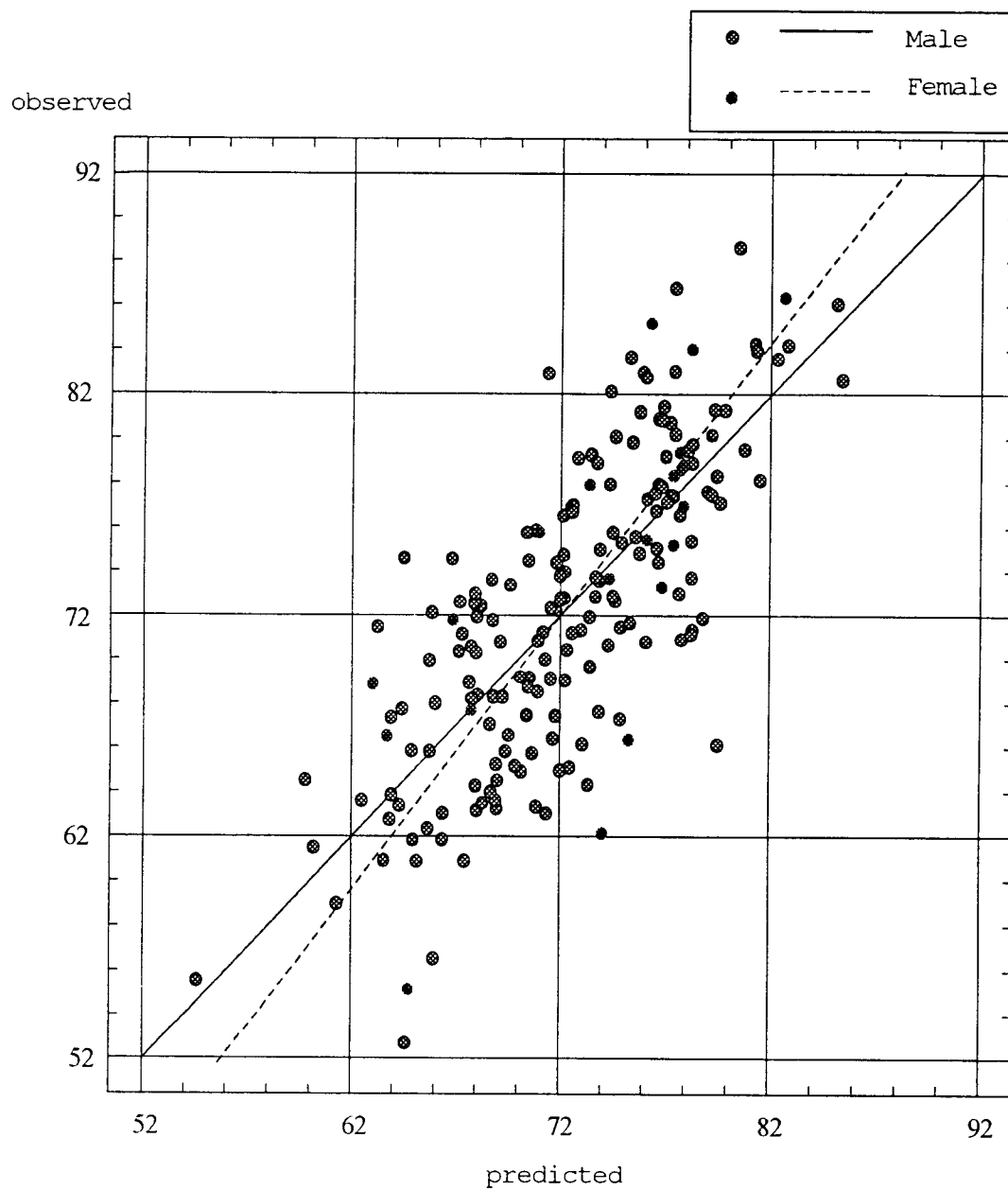


Figure 4. Plot of CS-INTRO Model for CS-MAJOR Prediction for Combined Senior Group.

Gender Differences

- 7) Are there gender differences in performance predictors for computer science majors?

To investigate possible gender differences between college computer science majors, the following hypothesis was tested:

H_0^8 There are no significant differences by gender for academic performance, prior computer experience, or success prediction.

Gender differences were examined for all the variables related to academic performance and prior computer experience as well as the achievement predictors. The results of the analysis for gender differences are presented in Table 12.

Since students admitted to the same university were selected based upon CEE scores, no significant gender differences for CEE scores were detected when males and females from the identical class level at the same university were compared. No significant gender differences were found in CEE-TOTAL, CEE-CHEM, and CEE-MATH for students in the same class level. However, males within the same class level were found to have higher but nonsignificant scores than those of females for CEE-CHEM and CEE-MATH. A significant gender difference was found in CEE-PHY for the junior group (69.3 for males versus 65.9 for females, $p < .05$). In contrast, males obtained significantly lower

scores than females in CEE-ENG for both the junior (63.0 for males versus 67.0 for females, $p < .05$) and senior groups (55.1 for males versus 63.2 for females, $p < .005$).

Table 12. Results of Analysis for Gender Differences.¹

	Males	Females	Group Difference
CEE-PHY ²			
JU	69.30	65.90	3.40*
CEE-ENG			
JU	63.00	67.00	4.00*
SE	55.10	63.20	8.10*
HS-MATH			
SO	73.20	75.70	2.50*
JU	72.10	76.20	4.10**
SE	73.00	78.80	5.80**
HS-AVG			
SO	73.60	75.70	2.10*
JU	72.90	75.80	2.90*
SE	74.10	79.10	5.00**
C-MATH			
SO	65.30	71.80	6.50***
SE	70.60	75.70	5.10**
CS-INTRO			
SO	70.00	72.80	2.80*
CS-MAJOR			
SO	68.40	72.00	3.60**
SE	72.60	76.80	4.20**

Notes:

¹ * = $p < .05$; ** = $p < .01$; *** = $p < .001$.

² CEE-PHY = CEE physics; CEE-ENG = CEE English; HS-MATH = average score of high school math; HS-AVG = average score of high school course performance; C-MATH = average score of college math; CS-INTRO = score of introductory computer science courses; CS-MAJOR = average scores of computer science core courses.

³ SO = Sophomore; JU = Junior; SE = Senior.

Significant gender differences were not found for any of the computer experience variables. However, males achieved significantly lower scores than did females for

both HS-MATH and HS-AVG in all class level groups. Differences greater than five points in average scores were detected for the senior group in both HS-MATH (73.0 for males versus 78.8 for females, $p < .005$) and HS-AVG (74.1 for males versus 79.1 for females, $p < .005$).

As concerns college level performance, the same pattern as the results for high school achievement was found in that females outperformed males in CS-INTRO, C-MATH, and CS-MAJOR. It was an unanticipated result that male students obtained substantially lower C-MATH scores than did females in both the sophomore (65.3 versus 71.8, $p < .001$) and senior groups (70.6 versus 75.7, $p < .005$). Males were also found to have significantly lower CS-MAJOR scores for both the sophomore (68.4 versus 72.0, $p < .005$) and senior groups (72.6 versus 76.8, $p < .005$), and male students were found to achieve lower scores than their female counterparts in CS-INTRO for the sophomore group (70.0 versus 72.8, $p < .05$).

GENDER was used as an indicator variable to generate a prediction model for CS-MAJOR. It was of interest to observe that a significant gender difference was found only for the sophomore group, and not for the junior or senior groups. When prediction models for individual senior classes were generated, significant gender differences were not detected. Even when the prediction models were modified to include only HS-AVG and CEE-TOTAL, significant gender differences were not found in any case.

In summary, significant gender differences were not detected for most CEE scores. Though males obtained higher scores than did females for CEE-CHEM, CEE-PHY, and CEE-MATH, female students achieved significantly higher scores than males for CEE-ENG. Moreover, females outperformed males for academic achievement at both the high school and college levels. However, if prediction models are to be used for overall performance prediction in college computer science programs, it will not be necessary to develop different models for males and for females.

Summary

In summary of the results found in this study, significant correlations were found between all test scores except the math component for the College Entrance Examination (CEE) and course performances in college computer science programs in combined class levels. However, findings for classes from individual universities were not consistent with the results found for the combined class groups. For the present study, due to the low levels of correlation coefficients obtained, the predictive power of CEE scores for predicting college performance is apparently of limited accuracy.

The CEE math component was found to negatively correlate to course performance for college computer science programs for individual classes. Significant correlations

were not found between average high school math course scores (HS-MATH) and course performances in introductory computer science courses (CS-INTRO). However, HS-MATH was correlated significantly with performance in college math courses (C-MATH) and computer science core courses (CS-MAJOR). Moreover, significant relationships were also found between C-MATH and CS-INTRO and overall course performance in nonmath components in computer science programs (CS-NONMATH). The importance of math ability with respect to academic achievement in computer science programs was confirmed by this study.

Both overall performance of high school course work (HS-AVG) as well as HS-MATH were identified as effective predictors for CS-MAJOR, but not for CS-INTRO. However, this relationship between high school achievement and overall college performance did not extend to students from university C. It was suspected that their low average C-MATH scores was a factor of importance in the nonsignificance of this relationship.

The findings for the beneficial relationship of prior computer experience to subsequent performance in college computer science programs were not conclusive. Significant correlation coefficients were seldom found between the variables measuring student prior computer experience (CS-COURSE, CS-PROG, and CS-SP) and subsequent performance in college computer science programs. However, students who took more than two computer or programming courses

prior to entering college were consistently found to outperform those who had not taken any computer or programming courses prior to entering college.

The close relationship between performance in beginning computer science courses and overall course performance in computer science programs was validated. Significant correlation coefficients were found for all combined class groups and individual classes. This close relationship was further supported when the R^2 value of the prediction model for CS-MAJOR was dramatically increased when CS-INTRO was entered into the model.

Significant linear prediction models for overall college performance, but not for performance in introductory computer science course, were generated. The R^2 value decreased slightly (i.e., by less than 0.02) when only CEE-TOTAL and HS-AVG were included in the prediction models for all the combined class groups. However, the predictive effectiveness of these models was limited, subject to significant improvement of model predictive powers when CS-INTRO was entered into the models.

Significant gender differences were not found for most of the CEE scores, for prior computer experience, or for the prediction models. On average, males achieved higher but nonsignificant CEE-PHY, CEE-CHEM and CEE-MATH scores. However, females outperformed their male counterparts in course performance both at high school and college levels.

CHAPTER V DISCUSSION AND CONCLUSIONS

Introduction

During the past two decades, researchers in the United States (US) have sought to identify factors which can be used to predict academic achievement in college computer science programs. Some researchers reported the effectiveness of using scores from standardized aptitude tests such as the Scholastic Aptitude Test (SAT) to predict college performance in computer science courses (Butcher & Muth, 1985; Dixon, 1987; Goodwin & Wilkes, 1986; Oman, 1986; Renk 1986). Others suggested that mathematics background related significantly to student performance in college computer science courses (Butcher & Muth, 1985; Dey & Mand, 1986; Renk, 1986; Thronson, 1985). Still others indicated that prior computer science experience was beneficial to student performance in college introductory computer science courses (Clark & Chambers, 1989; Dey & Mand, 1986; Greer, 1986; Nowaczyke et al., 1986; Oman, 1986; Taylor & Mounfield, 1989).

The principal purpose of the present study was to determine whether student academic achievement in college computer science programs in the Republic of China (ROC) could be predicted by factors reported to have been effective in many of the US studies cited. This study focused primarily upon the prediction of overall performance in

computer science programs, rather than achievement within a single computer science course. Moreover, by examining the relationship between performance in beginning computer science courses and performance in complete computer science programs, the study was designed to verify a hypothesized relationship between performance in introductory computer science courses and overall performance in complete computer science programs. Interrogating possible gender differences with respect to predictors was also a principal research interest.

Research was conducted in Taiwan, ROC during the Fall academic term, 1995. Students enrolled in universities offering computer science programs were surveyed for subjects. Following selection, a researcher-designed questionnaire was used to collect background information from subjects who volunteered to participate, each of whom completed a written, voluntary consent form. A total of 940 questionnaires were collected, representing more than 81% of the population. Scores from subject College Entrance Examination (CEE) and college computer science courses were collected through access to student academic records at appropriate college registrar's offices.

The validity and reliability of the questionnaire were carefully considered and the following were completed: validity assessment, questionnaire pilot testing, and interviews conducted with subjects selected from the pilot test sample. Data were checked and entered into a computer

readable format for further analysis following collection. Appropriate statistical analyses were performed. Significant differences for most of the variables were detected between subject class levels and university of enrollment. All of the statistical analyses were completed by grouping students within the same class level across universities as well as students in the same class level who were enrolled at the same university.

Results of the analyses are discussed in the following sections. The limitations of the study are presented following discussion of the results. Recommendations for future research as well as implications for computer science education are also addressed.

Discussion of the Results

Based upon interpretation of the statistical analyses, discussion of the results is presented according to specific research questions posed in previous chapters. Conclusions for the study are based upon finding from the results of the statistical analyses.

CEE Scores

Standardized test scores, such as the SAT and the American College Test (ACT), have been reported to be effective performance predictors for college computer sci-

ence programs by numerous studies conducted in the US. Test scores of math component have often been found to relate significantly to student course performance in computer science programs and have been included frequently in prediction models (Butcher & Muth, 1985; Campbell & McCabe, 1984; Dixon, 1987; Goodwin & Wilkes, 1986; Oman, 1986; Renk, 1986; Sorge & Wark, 1984). Similar results were also found for SAT-Verbal and ACT-English scores (Butcher & Muth, 1985; Oman, 1986; Sorge & Wark, 1984).

For the current study, test scores from the CEE math component (CEE-MATH) were found to have a significant relationship to overall performance in college computer science programs (CS-MAJOR) for all class levels, but only for the senior level group in relationship to performance in introductory computer science courses (CS-INTRO). Significant correlation coefficients were seldom found between CEE-MATH and CS-MAJOR for individual classes. Moreover, when the correlations were significant, they nonetheless did not account for an acceptable level of variance. Rather, CEE-MATH correlated negatively with both CS-INTRO and CS-MAJOR for most (65% or greater) individual classes. It was also observed that all of the significant correlation coefficients between CEE-MATH and college performance were negative.

This finding of negative relationships is a contradiction of results reported in a number of research studies conducted in the US, wherein math scores were found to

correlate highly with course performance for college computer science programs (Butcher & Muth, 1985; Dixon, 1987; Goodwin & Wilkes, 1986; Oman, 1986; Renk, 1986). However, in a Nigerian study by Anyanwu (1988), a nonsignificant relationship between test scores for the math component of the Joint Admission and Matriculation Board (JAMB) and overall achievement in college computer science programs was reported.

In addition to the negative correlation, whenever CEE-MATH was selected into the prediction models (discussed below in greater detail), negative coefficients were again detected. Nonetheless, a significant correlation between math ability and college performance in computer science programs was confirmed (discussed in the following section). These findings seemingly indicate that CEE-MATH may not be a valid instrument for the measurement of student math ability, suggesting that the use of CEE-MATH to predict student future achievement in college computer science programs is not appropriate in the ROC.

There were significant relationships between scores for the CEE English component (CEE-ENG) and both CS-INTRO and CS-MAJOR for all class level groups. This finding is consistent with results reported by Butcher and Muth (1985), Oman (1986), and Sorge and Wark (1984) in the US. However, only 2 of the 14 classes considered were found to have significant correlations between CEE-ENG and CS-MAJOR. Therefore, with less than 20% of the variance for CS-MAJOR

explained, the predictive power of using CEE-ENG by itself to predict student college performance in computer science programs also is apparently limited.

Results similar to those for CEE-ENG were found for the science components (CEE-PHY and CEE-CHEM). Significant correlations between CEE-PHY and CS-INTRO and CS-MAJOR, and between CEE-CHEM and CS-INTRO and CS-MAJOR, were identified for all class groups. However, relatively few classes reflected significant correlations for the relationship between college performance and CEE-PHY or CEE-CHEM. In addition, CEE-PHY was also negatively correlated with both CS-INTRO and CS-MAJOR for individual class in the instances that significant correlation coefficients were observed.

Since no physics or chemistry component was tested either for the SAT or the ACT, corresponding results in the US could not be compared to findings for CEE-PHY and CEE-CHEM. However, Goodwin and Wilkes (1986) reported a negative correlation between the number of physics courses taken in high school and performance in an introductory computer science course. But since no further information was given, it would be unwise to assert any hypothesized explanation for the negative correlation between CEE-PHY and performance in college computer science programs at this time. Nonetheless, due to the low levels of correlations obtained, the use of CEE-PHY or CEE-CHEM to predict student future achievement in computer science programs is also not recommended for the ROC.

Relationships between the total scores of the standardized test and college performance were seldom investigated. Only Anyanwu (1988) reported a significant relationship between total scores for the JAMB and achievement in the math components of computer science programs. From the present study, significant correlations were detected for all class groups for the relationship between CEE-TOTAL and CS-INTRO and CS-MAJOR. The strength of the relationship between CEE-TOTAL and CS-MAJOR appeared to increase with length of time enrolled in computer science programs. In the knowledge that sophomores took from only four to six computer science related courses during their freshman year, or possibly too few in number to result in a significant relationship between the CEE-TOTAL and CS-MAJOR, the finding of this relationship pattern was not surprising.

Significant correlation coefficients for the relationship between CEE-TOTAL and college performance were not often found for individual classes. Yet, several negative correlations were found for individual classes with respect to the relationship between CEE-TOTAL and CS-INTRO and CS-MAJOR. One possible explanation for this low correlation may be that the CEE-TOTAL variation for students within the same class level of the same university was too small to distinguish differences in college academic achievement. Thus, the findings of this study with respect to the relationship between CEE-TOTAL and college performance were not conclusive.

In general, CEE scores were correlated significantly with college performance for computer science program students when the data analyses were completed by class level groups. However, these coefficients, for the greater part less than 0.40, were too low to constitute an important educational value from which appropriate conclusions could be drawn. Consequently, the predictive power of CEE scores, as the sole means to predict student performance in college computer science programs is apparently limited due to the fact that less than 15% of the variance in college performance was accounted for in most of the correlated cases.

Math Ability

Significant results concerning the relationship between the number of high school math courses taken and final grades earned in introductory computer science courses have been reported in a number of studies completed in the US (Butcher & Muth, 1985; Dey & Mand, 1986; Ramberg & Caster, 1986; Renk, 1986; Thronson, 1985). Dey and Mand (1986) further indicated that the average grade of high school math courses taken related significantly to performance in college introductory computer science courses. A similar result was also reported by Campbell and McCabe (1984), although the first year grade-point average (GPA), was used for the correlation analysis

instead of the grade for an individual course. A significant relationship between HS-MATH and overall performance in college computer science programs was also found in a Nigerian study (Anyanwu, 1988).

Since a uniform curriculum as determined by the ROC Ministry of Education (1990, 1993) was used, all high school students took the same number of math courses. Therefore, the average scores of all high school math courses (HS-MATH), rather than the number of math courses taken, were used to verify the significance of relationships between math background and college performance in computer science programs. HS-MATH was found to correlate significantly with both C-MATH (r range from 0.35 to 0.56) and CS-MAJOR (r range from 0.30 to 0.53), but not to CS-INTRO, for almost all the combined class groups as well as classes in individual universities. This finding supports results obtained by a number of studies conducted in the US, to the effect that math background related significantly to performance in college computer science programs. To summarize, course work in high school math seems to improve student college performance in the ROC, just as was indicated in the study conducted in the US by Butcher and Muth (1985).

In addition, average scores of college math courses (C-MATH) were also found to relate significantly to performance in college computer science programs, both with respect to CS-INTRO and the nonmath components of the com-

puter science core courses. However, this finding was not consistent with results reported in the US by Dey and Mand (1986), Konvalina et al. (1983b), and Thronson (1985). One possible reason for this incompatibility is that average scores were used for the present study, while the number of math courses taken in college was used in the other cited studies.

It was also of interest to note that a nonsignificant relationship between HS-MATH and college performance was found only for the students at university C. Upon closer examination of C-MATH by university, it was found that the students of university C obtained lower scores than those students from other universities. This low average score in C-MATH might be attributed to the low correlations obtained between C-MATH and CS-NONMATH, and the nonsignificant correlation found between HS-MATH and C-MATH, for the students at university C.

From all of the findings regarding significant relationships between HS-MATH and college performance, and between performance in C-MATH and other nonmath components, the role of math ability in supporting academic achievement in college computer science programs is seemingly confirmed. However, the use of HS-MATH by itself to predict college computer science program performance is recognized as inappropriate due to the reason that only less than 30% of the variance can be explained by the correlations.

Prior Computer Experience

The benefits of prior computer experience to course performance in college computer science programs was not conclusive. A number of studies found no significant link between the number of computer courses taken prior to entering computer science programs and course performance in introductory computer science courses (Butcher & Muth, 1985; Dixon, 1987; Goodwin & Wilkes, 1986; Nowaczyk et al., 1986; Ramberg & Caster, 1986). Yet, some researchers indicated significant results for this relationship (Greer, 1986; Oman, 1986; Taylor & Mounfield, 1989). The findings from the present study, though several significant correlations were found, did not provide consistently strong evidence of a relationship between CS-COURSE and CS-INTRO as well as CS-MAJOR.

Ramberg and Caster (1986) as well as Nowaczyk et al. (1986) asserted that prior programming experience correlated significantly to performance in introductory computer science courses. Significant correlations were found between the number of programming courses taken prior to entering college computer science programs and performance for all three combined classes. However, given the fact that the correlation coefficients obtained were less than 0.20, the strength of the relationship appeared to be too weak to support the assertion that prior programming

courses taken was related to performance in college computer science programs.

Some researchers concluded that it was experience in structured programming methodology, rather than general computer experience, that benefited learning in subsequent college level computer science courses (Dey & Mand, 1986; Greer, 1986; Taylor & Mounfield, 1991). However, the results of the present study show no evidence in support of such an assertion.

Nevertheless, students who took more than two computer or programming courses prior to entering college computer science programs (group A) were consistently found to outperform students with no such experience (group B) for both CS-INTRO and CS-MAJOR. Though some of these group differences were not significant, it was suspected that the small number of subjects in group A (i.e., in some cases the group consisted of only two samples) accounted for the non-significance of the results in tests of group differences for CS-INTRO and CS-MAJOR.

Moreover, it was also of interest to note that most of the significant correlations between the relationship of CS-MAJOR and CS-COURSE and CS-PROG were found at the senior level, and that no similar patterns were detected for the relationship between CS-INTRO and CS-COURSE and CS-PROG. Therefore, additional research may be required to verify the hypothesized benefits of prior computer experience to course performance in college computer science programs.

Overall High School Performance

High school grade point average (GPA) was repeatedly reported to correlate significantly to course performance in college computer science programs and was included in the prediction models of numerous studies (Anyanwu, 1988; Butcher & Muth, 1985; Konvalina et al., 1983a; Renk, 1986; Shoemaker, 1986; Thronson, 1985). Ramberg and Caster (1986) reported a nonsignificant difference for overall high school performance between withdrawers and nonwithdrawers from a beginning computer science course.

A significant but low correlation between high school overall performance (HS-AVG) and CS-INTRO was demonstrated for the sophomore and senior groups. However, significant correlations were seldom found for individual classes. The absence of a significant relationship between HS-AVG and CS-INTRO contradicted findings reported in the US (Konvalina et al., 1983a; Renk, 1986; Thronson, 1985).

On the other hand, similar to results found in US and other research (Anyanwu, 1988; Butcher & Muth, 1985; Shoemaker, 1986), HS-AVG was consistently found to correlate significantly with CS-MAJOR for all combined classes and almost all the university-specific class levels. The strength of this relationship appeared to increase with time in computer science programs (i.e., the r increased from 0.36 for sophomores to 0.48 for seniors). With consistent findings within all class levels for different uni-

versities, the close relationship between HS-AVG and overall performance in college computer science programs was seemingly validated. Nevertheless, the predictive power of using HS-AVG by itself to predict overall performance in college computer science programs is still limited, given the fact that correlations obtain in most cases were less than .50.

Introductory Computer Science Courses

It has long been accepted that good performance in beginning computer science courses is a good indicator for future success in the computer science programs. However, this hypothesized relationship has never been empirically proven. In the present study, the relationship between CS-INTRO and CS-MAJOR was investigated. Moreover, to avoid obscuring the relationship between CS-INTRO and overall computer science performance, the correlation between CS-INTRO and CS-NOBCC (a subset of CS-MAJOR, with CS-INTRO excluded from the calculation of CS-MAJOR) was also examined.

Significantly high correlation coefficients were found for all class groups as well as all individual classes in the relation between CS-INTRO and CS-MAJOR (r range from .54 to .88) and between CS-INTRO and CS-NOBCC (r range from 0.32 to 0.77). Findings for this close relationship between CS-INTRO and CS-MAJOR or CS-NOBCC supported the

common hypothesis that good performance in the first computer science course taken is an indicator of future academic success in the computer science programs. Therefore, the adequacy of using course performance in introductory computer science courses as indicators of overall performance in college computer science programs was validated for college students in the ROC.

Prediction Models

Student performance in beginning computer science courses was claimed to be reliably predicted by preadmission variables in studies conducted in the US (Butcher & Muth, 1985; Clarke & Chambers, 1989; Goodwin & Wilkes, 1986; Kersteen et al., 1988; Nowaczyk et al., 1986; Oman, 1986; Renk, 1986). However, R^2 values in excess of 0.30 were seldom found in these studies.

In the present study, similar results were obtained for the prediction models of CS-INTRO. Although significant linear models were generated for combined class groups, the R^2 obtained was 0.12 or lower for the junior and senior groups. An R^2 at this level is the equivalent of a random guess prediction. Thus, the practical value of these prediction models is limited. Moreover, no significant linear models were generated for the senior class of universities D and E. Hence, when these findings are considered in combination, they are inadequate to conclude

that performance in the introductory computer science courses can be predicted by the variables investigated in the present study.

Only limited research has dealt with performance prediction beyond the level of introductory computer science courses. Among these studies, Butcher & Muth (1985) reported an R^2 of 0.42 for a prediction model for first semester GPA, with high school GPA and ACT-MATH included in the model. Shoemaker (1986) found that high school GPA and College Board math achievement were the best predictors for major GPA for college computer science students ($R^2 = .34$). In the Nigerian study, Anyanwu (1989) indicated that the overall college performance of computer science majors could be predicted by high school GPA, prior computer experience, and GPA for high school math. However, a low model R^2 was observed (R^2 range from 0.12 to 0.17).

In the present study, R^2 values of 0.30 or lower were obtained for models predicting CS-MAJOR for different combined class groups. Different variable combinations of high school performance and CEE scores were selected into the prediction models, consistent with the findings previously reported (Butcher & Muth, 1985; Shoemaker, 1986). Other than GENDER, no other preadmission variables were entered in the prediction models, indicating the lack of predictive power for these variables. Similar to results from other studies, the overall predictive power of these models was also apparently limited.

Considering that CEE-TOTAL is likely to continue to be used as the primary selection criteria for college admission in the ROC, all subject CEE scores selected into the prediction models were replaced by CEE-TOTAL and the models were then reanalyzed. Similar results were obtained, with the model R^2 slightly decreasing (i.e., 0.02 less than in the original model). This result indicated that a similar predictive power would be obtained if only CEE-TOTAL and HS-AVG were used for the prediction of CS-MAJOR.

As described previously, CS-INTRO was found to closely relate to CS-MAJOR. The R^2 for the models also significantly increased when CS-INTRO was entered into the CS-MAJOR prediction models for all the combined class levels and individual classes. This finding further supported the close relationship between CS-INTRO and CS-MAJOR. Hence, it is suggested that CS-INTRO be included in prediction models of overall performance to select successful students for computer science programs in the ROC.

Gender Differences

Although gender differences have been an important issue in computer science education and have frequently been subject to investigation, significant gender differences in course performance have seldom been found (Clarke & Chambers, 1989; Goodwin & Wilkes, 1986; Nowaczyk et al., 1986; Renk, 1986; Taylor & Mounfield, 1989). Nonetheless,

several researchers have reported that females tended to achieve better course grade performances in beginning computer science courses (Clarke & Chambers, 1989; Thronson, 1985; Taylor & Mounfield, 1991).

The present study found no significant gender differences for CEE-TOTAL, CEE-CHEM, or CEE-MATH performance. Male students achieved relatively higher scores than their female counterparts, though the score differences did not achieve required significance levels. However, females achieved significantly higher scores in HS-MATH, HS-AVG, CEE-ENG, C-MATH, CS-INTRO and CS-MAJOR. Therefore, though the gender differences in CEE scores were not significantly obvious, female students apparently outperformed males with respect to academic achievements at both the high school and college levels.

No significant gender differences were found in prior computer experience for all combined class groups, a finding that is incompatible with those reported by Kersteen et al. (1988) and Clarke and Chambers (1989), wherein males were reported to have significantly greater computer experience than females. This result may not be surprising, given the fact that more than 60% of the subjects had taken at least one computer course prior to entering college computer science programs.

GENDER, when used as an indicator variable, was found to be an effective predictor for the CS-MAJOR prediction model for sophomore groups. However, GENDER was not

selected into the CS-MAJOR prediction models for any of the senior classes. Furthermore, when CS-INTRO was entered into the models, GENDER became nonsignificant for all the prediction models generated for CS-MAJOR. As a result, different prediction models for males and females should not be a necessity for the prediction of CS-MAJOR using models generated from this study.

Limitations of the Research

Several limitations of the present study were recognized. The primary limitation can be directly linked to the voluntary nature of participation in the survey. Though the sample represented in excess of 81% of the defined population, some students were absent from classes during administration of the questionnaire and were not contacted. Written consent to participate was not obtained from these students. As a result, questionnaire information as well as registrar's records for these students were not available. Therefore, generalization of the findings from this study to the entire population of computer science programs must be approached with caution.

Because high school performance is not considered as a selection criteria for college admission in the ROC, high school transcripts for the sample were not available in registrars' office. Consequently, all information regarding high school performance was self-reported by the parti-

cipants during questionnaire administration. Data concerning prior computer experience were also self-reported using the same source.

As described in Chapter III, including all students in computer-related programs in the ROC in the investigation was not encompassed within the present study. As a result, comparisons of differences between various programs (i.e., computer engineering, computer science and management information systems) were not undertaken. Hence, the ability to generalize the findings of this research to populations other than those within computer science programs is limited.

The restricted number of female students in computer science programs is also recognized as a limitation of this study with respect to the examination of gender differences. As shown in Table 2, for all five participating universities, fewer than 10 female subjects were observed from the senior classes. With this substantial difference between the numbers of male and female subjects within the same class, the findings for gender differences should be approached with caution when results from the comparisons for senior classes are used.

A similar limitation regarding the small number of subjects was also identified when comparing the college academic performances of students with various prior programming experience. As indicated previously, students with more than two programming courses taken prior to

entering college (group A) were consistently found to outperform students without such experience (group B) for both the CS-INTRO and CS-MAJOR tests. However, it was suspected that the small number of subjects included in group A might be attributed to the result of nonsignificant group differences. If more subjects had been included in the comparisons, the results indicating significant group differences in course performance could have been more persuasive.

Implications for Computer Science Education

The findings of this study indicated that CEE-MATH was not closely correlated to either CS-INTRO or CS-MAJOR. In addition, negative coefficients were observed whenever CEE-MATH was selected into the prediction models for CS-MAJOR. These results failed to demonstrate the close relationship between CEE-MATH and college performance in computer science programs, as reported in a number investigations reported in the US (Butcher & Muth, 1985; Campbell & McCabe, 1984; Dixon, 1987; Goodwin & Wilkes, 1986; Oman, 1986; Renk, 1986; Sorge & Wark, 1984).

However, a strong relationship was determined to exist between course performance in college computer science programs and math ability. Combining these findings, the effectiveness of the continued use of CEE-MATH to measure student math ability is questionable. Due to the lack of

power for predicting student future achievement in computer science programs, as well as its incapability in measuring student math ability, the use of CEE-MATH as a major selection criteria for entering college computer science programs is considered to be inappropriate.

A limitation of the predictive power of CEE-TOTAL for predicting CS-MAJOR was found. Similar results in predicting college performance were reported by other researchers (Hsu & Lin, 1982; Tsong et al., 1977), though students in other than computer science programs were used. Currently, college admission in the ROC is principally determined by CEE-TOTAL. A score within the upper 50% percentile for certain CEE subjects is required as a corequirement for specific program admissions at some universities. High school performance has never been used for this selection purpose. However, high school performance in math courses as well as overall course work were consistently found to correlate well with college performance in computer science programs. In consideration of this result, if a different admission process is employed in the future, it is suggested that high school performance be included as one of the admission criteria for the selection of potentially successful students for computer science programs.

No significant gender differences were found for the prior computer experience variables (CS-COURSE, CS-PROG and CS-SP), results that were inconsistent with those reported by Clarke and Chambers (1989) and Kersteen et al. (1988) in

the US. One possible explanation for the inconsistency of these findings may be that in excess of 60% of freshmen enter college with some computer experience. Results from the interviews during pilot testing indicated that taking at least one semester of computer courses was required in some high schools. This may also explain the nonsignificance of the results for gender differences in prior computer experience.

It was not surprising that no gender differences were found for the scores of CEE-TOTAL, given that students were admitted to specific universities based primarily on their CEE-TOTAL scores. Though males achieved higher relative scores, significant gender differences were not found for CEE-MATH, CEE-CHEM, and CEE-PHY (with the exception of the combined junior class levels). However, female students in computer science programs achieved significantly higher scores in CEE language component (CEE-ENG). Furthermore, female students were also found to outperform males in academic achievement at both the high school and college levels. Several studies conducted in the US reported similar results (Clarke & Chambers, 1989; Taylor & Mounfield, 1991; Thronson, 1985). The results of the present study seem to suggest that gender difference is a perceived difference, rather than an ability difference, as indicated by Clarke and Chambers (1989).

When the findings for gender differences are considered in combination, the results imply that females enrol-

led in the computer science programs in the ROC may be more confident in their ability to compete with males in this male-dominated field. From these findings, if lack of computer experience is an obstacle for females in the ROC when choosing computer science as a major, this barrier may be eliminated by obtaining more computer experience prior to college entry, as was also suggested by Kersteen et al. (1988).

The use of beginning computer science courses as a gateway for entering a computer science major has long been practiced in the US. However, college majors are determined based solely upon the total CEE scores when students are admitted to a university in the ROC. Course performance in the beginning computer science course is not taken into account for admission purposes.

Findings on the relationship between CS-INTRO and overall course performance in computer science programs supported the common hypothesis that good performance in the first computer science course may indicate future academic success in computer science programs. Predictive powers were significantly increased when CS-INTRO was included in the prediction models. If CS-INTRO could be used with other predictors (such as overall high school performance), a more satisfactory selection outcome may be expected than when using CEE-TOTAL as the sole basis for admission to a college computer science program.

Some of the inconclusive results of this research, in combination with findings from previously conducted studies, suggest that performance prediction findings should be viewed cautiously. Butcher and Muth (1985) pointed out that studies using standardized test scores all identify high school grades or GPA as important parameters, but nonetheless leave more than 50% of the variance unexplained. Chin and Zecker (1985) warned that the use of a mathematics pretest as the only success predictor for computer science courses was inappropriate. Since test scores tend to improve with practice, Sharma (1987) questioned the usage of test scores as the sole screening tool for college admissions. Sorge and Wark (1984) also suggested that factors other than academic ability were involved in succeeding in computer science programs. Therefore, as indicated by Oman (1986), the prediction model developed using preadmission variables should be supplemented with other methods (e.g., personal interviews) if academic advice or selection for successful computer science majors is the principal purpose of a process.

One of the major purposes for identifying effective predictors of college performance is the intention to make better use of limited resources by helping students reach reasonable decisions for choices of college major. However, as Butcher and Muth (1985) indicated, some of those who have been classified as unlikely to succeed in beginning computer science courses have been found eventually to

perform well in subsequent courses. Therefore, the practice of individual success prediction for academic performance should not be used to discourage students with high motivation in computer science studies. Instead, it would be more appropriate to use information from performance prediction as a means to better advise high school graduates in the ROC in the process of choosing college majors.

Recommendations for Future Research

In this study, significant correlations were found between CEE scores and performance in college computer science. However, the predictive powers of these scores considered in the absence of supplementary information were found to be limited (accounting for less than .20 of the variance). Moreover, CEE-MATH was negatively correlated to performance, both in introductory computer science courses and overall course work, for the computer science programs of many classes. These results suggest that reassessing the predictive validity of CEE scores, especially CEE-MATH, may be necessary.

The close relationship between math ability and overall performance in college computer science programs was confirmed by the present study. Clarke and Chambers (1989) reported that males took a greater number of high school mathematics courses than did females, also pointing out that the number of mathematics courses taken by females in

the 12th grade level was highly correlated with intention to further computer science studies. Therefore, in view of a disproportionate male-female ratio in college computer science programs, it was hypothesized that the lack of certain mathematical knowledge could have been a "stumbling block" for women in their intention to enroll in computer science courses.

However, students in the ROC take the same number of math courses at the high school level and it would be meaningless to examine the relationship between high school math and performance in college computer science using the number of high school math courses taken as a variable. Rather, the identification of certain mathematics courses as prerequisites or corequisites to beginning computer science courses may be necessary. Moreover, the specific mathematical knowledge that contributes to successful learning in subsequent computer science courses also needs to be identified through additional research.

Several researchers reported that prior exposure to computers demonstrated a significant effect upon performance in the computer science courses at the college level (Anyanwu, 1988; Greer, 1986; Konvalina et al., 1983b; Oman, 1986; Taylor & Mounfield, 1989). Other research indicated that the number of computer courses taken prior to entering college did not relate to performance in introductory computer science courses (Butchers & Muth, 1985; Dixon, 1987; Goodwin & Wilkes, 1986; Nowaczyk et al., 1986; Ramberg &

Caster, 1986). The findings from this study on the effect of prior computer experience were inconclusive. Correlations, though significant, between variables assessing student prior computer experience (CS-COURSE, CS-PROG and CS-SP) and college performance (CS-INTRO and CS-MAJOR) were considered too low to be of importance in reaching educational conclusion. However, students taking in excess of two computer courses prior to college entry were consistently found to outperform students without such experience for both CS-INTRO and CS-MAJOR. When these findings are considered in combination, it may be hypothesized that experience from only one computer course may not be sufficient to reveal the benefit of prior computer experience to subsequent learning in college computer science courses. Additional research will be required to determine if prior computer experience is linked to performance in college computer science programs.

Only computer science majors currently enrolled in a university in the ROC, and not students in other computer-related programs, were included within the present study. Generalizing the results from this study to populations other than computer science majors is thus inappropriate. Therefore, future research should be conducted to determine if similar results can be found for students enrolled in other computer-related programs. Clarification of the predictability of student success in computer-related programs may be obtained if the results of such research can be com-

bined or compared to the findings reported in this study. In addition, conducting research to verify if the prediction models developed for the sophomore and junior level students surveyed for the present study remain valid may be necessary.

REFERENCES

- Anyanwu, L. O. (1989). A study of selected predictors of achievement in the computer science programs in the Nigerian universities. Unpublished doctoral dissertation, Morgan State University.
- Babbie, E. (1986). The practice of social research (4th ed.). Belmont, CA: Wadsworth Publishing Co.
- Bennett, G. K. (1974). Differential aptitude test. New York, NY: Psychological Corporation.
- Borg, W. R., & Gall, M. D. (1989). Educational research: An introduction (5th ed.). White Plains, NY: Longman Inc.
- Buros, O. K. (1972). The seventh mental measurements year-book. Highland Park, NJ: Gryphon Press.
- Butcher, D. F., & Muth, W. A. (1985). Predicting performance in an introductory computer science course. Communications of the ACM, 28(3), 263-268.
- Campbell, P. F., & McCabe, G. P. (1984). Predicting the success of freshman in a computer science major. Communications of the ACM, 27(11), 1108-1113.
- Chen, R. H. (1975). What does the scores of the College Entrance Examination predict? Journal of Chinese Measurement Association, 22, 95-98.
- Chen, S. C. (1988). A Study of predicting the supply and demand of information professionals in the ROC at the year of 2000. Unpublished doctoral dissertation, National Sun Yat-Sen University, Kaoshung, Taiwan.
- Chin, J. P., & Zecker, S. G. (1985). Personality and cognitive factors influencing computer programming performance. Paper presented at the Annual Meeting of the Eastern Psychological Association, Boston, MA.
- Clarke, V. A., & Chambers, S. M. (1989). Gender-based factors in computing enrollments and achievement: Evidence from study of tertiary students. Journal of Educational Computing Research, 5(4), 409-429.
- College Entrance Examination Board. (1994, March). Guidelines of college entrance examination for the year of 1994. Taipei, Taiwan: Author.

- Dey, S., & Mand, L. R. (1986). Effects of mathematics preparation and prior language exposure on perceived performance in introductory computer science courses. SIGSCE Bulletin, 18(1), 144-148.
- Dixon, V. A. (1987, June). An investigation of prior source of difficulties in learning university computer science. Paper presented at the National Educational Computer Conference. Philadelphia, PA. (ERIC Documentation Reproduction Service No. ED 295 596)
- Goodwin, L., & Wilkes, J. M. (1986). The psychological and background characteristics influencing students' success in computer programming. AEDS Journal, 19(1), 1-9.
- Greer, J. (1986). High school experience and university achievement in computer science. AEDS Journal, 19(2-3), 216-225.
- Hsu, P. L., & Lin, P. C. (1982). Predicting academic achievement of college students in the R.O.C. Journal of Chinese Measurement Association, 29, 51-62.
- Humphreys, L. G. (1968). The fleeting nature of the prediction of college academic success. Journal of Educational Psychology, 59(5), 375-380.
- Humphreys, L. G., & Taber, T. (1973). Postdiction study of the GRE and eight semesters of college grades. Journal of Educational Measurement, 10, 179-184.
- Hunt, A. W. (1977). A decision rule for predicting academic success. Decision Science, 8, 270-286.
- Hwang, B. W. (1990). A comparative study of computer information systems curricula in the United States and the Republic of China. Doctoral dissertation, University of Texas at Austin. Dissertation Abstracts International, 52(1), 68-A.
- 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.
- Kersteen, Z. A., Linn, M. C., Clancy, 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), 321-333.

- Konvalina, J., Stephens, L., & Wileman, S. (1983a). Identifying factors influencing computer science aptitude and achievement. AEDS journal, 16(2), 106-112.
- Konvalina, J., Wileman, S. A., & Stephens, L. J. (1983b). Math proficiency: A key to success for computer science students. Communications of ACM, 26(5), 377-382.
- Lips, H. M., & Temple L. (1990). Majoring in computer science: Causal models for women and men. Research in Higher Education, 31(1), 99-113.
- Lu, C. M., & Jien, M. F. (1976). A study of the College Entrance Examination: It's format, scoring methods and predictive validity. Journal of Chinese Measurement Association, 23, 28-45.
- Lunneborg, C. E., & Lunneborg, P. W. (1970). Relations between aptitude changes and academic success during college. Journal of Educational Psychology, 61, 169-173.
- Ministry of Education, Republic of China. (1990). Required courses for colleges and universities. Taipei, Taiwan: Author.
- Ministry of Education, Republic of China. (1993). Educational statistics of the Republic of China. Taipei, Taiwan: Author.
- Ministry of Education, Republic of China. (1993, May). Statistics of college and university graduates of the Republic of China. Taipei, Taiwan: Ministry of Education, Bureau of Statistics.
- Ministry of Education, Republic of China. (1994). Index of educational statistics of the Republic of China. Taipei, Taiwan: Author.
- Ministry of Education, Republic of China. (1994, February). Statistics of colleges and universities of the Republic of China. Taipei, Taiwan: Ministry of Education, Bureau of Statistics.
- Nowaczyk, R. H., Connor, A. C., Stevenson, D. E., & Hare, E. O. (1986). Developing a prediction equation for success in introductory level computer science course: A first attempt. Proceedings of National Educational Computing Conference '86, 271-276.

- Oman, P. W. (1986). Identifying student characteristics influencing success in introductory computer science courses. AEDS journal, 19(2-3), 226-233.
- Ralston, A., & Shaw, M. (1980). Curriculum '78: Is computer science really that unmathematical? Communications of the ACM, 23(2), 67-70.
- Ramberg, P. & Caster, S. V. (1986). A new look at an old problem: Keys to success fro computer science students. SIGCSE Bulletin, 18(3), 36-39.
- Renk, S. C. (1986). Factors affecting academic success in introductory computer programming. Doctoral dissertation, University of Iowa. Dissertation Abstracts International, 48(3), 579-A.
- Sharma, S. (1987). Learners' cognitive styles and psychological types as intervening variables influencing performance in computer science courses. Journal of Educational Technology System, 15(4), 391-399.
- Shoemaker, J. S. (1986, April). Predicting cumulative and major GPA of UCI engineering and computer science majors. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA. (ERIC Document Reproduction Service No. 270 468).
- Sorge, D. H., & Wark, L. K. (1984). Factors for success as a computer science major. AEDS Journal, 17(4), 36-45.
- Sproull, L, Zubrow, D., & Kiesler, S. (1986). Cultural socialization to computing in college. Computers in Human Behavior, 2(4), 257-275.
- Stephens, L. J., Wileman, S., & Konvalina, J. (1981). Group differences in computer science aptitude. AEDS Journal, 16(4), 228-233.
- Taylor, H. G., & 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.
- Teague, J., & Clarke, V. (1991). Fiction and fact: Students' and professionals' perceptions of women in computer science. In A. Lehto and I. Erickson (Eds.), Proceedings of the Conference on Women, Work and Computerization (pp. 377-390). Helsinki, Finland: Ministry of Social Affairs and Health.

- Thronson, R. M. (1985). Achievement as a function of learning style preference in beginning computer programming courses. Doctoral dissertation, Montana State University. Dissertation Abstracts International, 45(10), 3100-A.
- Tsai, S. E. (1984). Relationship between high school students' computer attitude and their achievement in computer science courses. Unpublished doctoral dissertation, National Chenchi University, Taipei, Taiwan.
- Tsong, T. L., Chen, F. B., Lu, C. M., Lin, P. C., Lee, N. C., & Wang, S. P. (1977). A study of relationship between students' score in college entrance examination and students' performance in high school and college. Taipei, Taiwan: College Entrance Examination Board.
- Wang, S. Y. (1989, July). The shortage of information systems professionals cannot be ignored. In Institute for Information Industry, Essay collection on education for information professionals (pp. 105-129). Taipei, Taiwan: Institute for Information Industry.
- Ware, N. C. Steckler, N. A., & Leserman, J. (1985). Undergraduate women: Who chooses a science major? Journal of Higher Education, 56(1), 73-84.
- Werth, L. H. (1986). Predicting student performance in a beginning computer science class. SIGCSE Bulletin, 18(1), 138-143.
- Windall, S. E. (1988). AAAS presidential lecture: Voices from the pipeline. Science, 241, 1740-1745.

APPENDICES

APPENDIX A Research Questionnaire

Assigned code #: _____

STUDENT QUESTIONNAIRE

INSTRUCTIONS: The following questionnaire is part of a research regarding computer science education in this country. It comprises two sections: 1) prior computer experience, and 2) general background information. Please response to the questions as indicated. Thank you!

PART I: PRIOR COMPUTER EXPERIENCE

A. Number of computer courses taken prior to entering the university.

1. Please identify the number of computer science courses you had taken at each of the following levels **BEFORE** you were admitted to this university? (Computer course is defined as any course that focuses upon computer knowledge or application and meet for **at least 20 hours** of total instructional time. For example, basic computer concepts, computer literacy, programming, word processing, or other computer application)

*** If you had not taken any computer courses prior to entering this university, please enter "0" in this space: _____, then you may omit the rest items in **PART I** and **CONTINUE** your responses starting from item 7 in **PART II**, thank you!!

NUMBER OF COMPUTER COURSES (please fill in the number)

ex: _____ 1 Middle school (grade 7 - 9)
 _____ Middle school (grade 7 - 9)
 _____ High school (grade 10 - 12)
 _____ Junior college, if any
 _____ Other (please specify : _____)

2. Among those computer courses, how many (if any) were programming related courses (For example, BASIC, Logo, COBOL, Pascal, FORTRAN, C, C++ or other programming languages)?

*** If you had not taken any computer courses prior to entering this university, please enter "0" in this space: _____, then you may omit the rest items in **PART I** and **CONTINUE** your responses starting from item 7 in **PART II**, thank you!!

NUMBER OF COMPUTER COURSES (please fill in the number)

ex: _____ 1 Middle school (grade 7 - 9)
 _____ Middle school (grade 7 - 9)
 _____ High school (grade 10 - 12)
 _____ Junior college, if any
 _____ Other (please specify : _____)

(Please Go On To The Next Page)

B. Experiences in Structured Programming

3. The idea of **top-down design** is to divide the main task into several independent subtasks (**modules**), if needed, in a hierarchical structures, as illustrated in the following Figure 1.

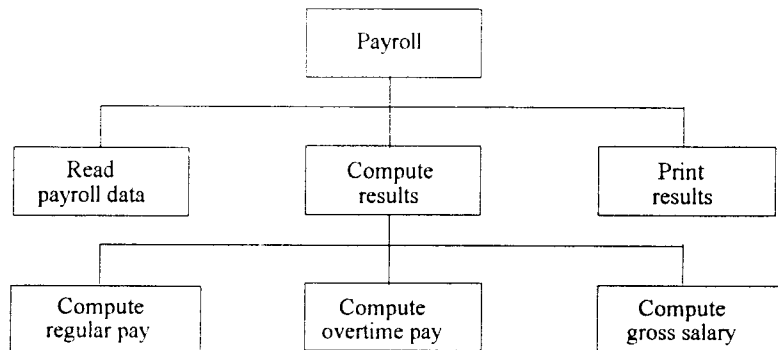


Figure 1. Modularity in a Top-down design

- a. Was top-down design stressed in your programming courses? (**CIRCLE one number**)
- 0 DON'T REMEMBER
 - 1 IN NONE OF THEM
 - 2 AT LEAST ONE, BUT LESS THAN THREE, OF THOSE COURSES
 - 3 IN MORE THAN THREE OF THOSE COURSES
- b. In general, to what extent was top-down design stressed in your programming courses? (**CIRCLE one number**)
- 0 DON'T REMEMBER
 - 1 NOT AT ALL STRESSED
 - 2 NOT TOO STRESSED
 - 3 SOMEWHAT STRESSED
 - 4 STRONGLY STRESSED

(Please Go On To The Next Page)

4. The main idea of **modularity** emphasizes that dependency between modules is minimized so that modifications made in one module will not cause dramatically subsequent changes in other modules (refer to Figure 1 for an example).
- Was **modularity** emphasized in your programming courses? (**CIRCLE one number**)
 - DON'T REMEMBER
 - IN NONE OF THEM
 - AT LEAST ONE, BUT LESS THAN THREE, OF THOSE COURSES
 - IN MORE THAN THREE OF THOSE COURSES
 - In general, to what extent was **modularity** emphasized in your programming courses? (**CIRCLE one number**)
 - DON'T REMEMBER
 - NOT AT ALL EMPHASIZED
 - NOT TOO EMPHASIZED
 - SOMEWHAT EMPHASIZED
 - STRONGLY EMPHASIZED
5. In structured programming, programs are efficiently constructed with **sequence**, **selection** (such as IF-THEN-ELSE, SELECT, and CASE), and **repetition** (LOOP) structures.

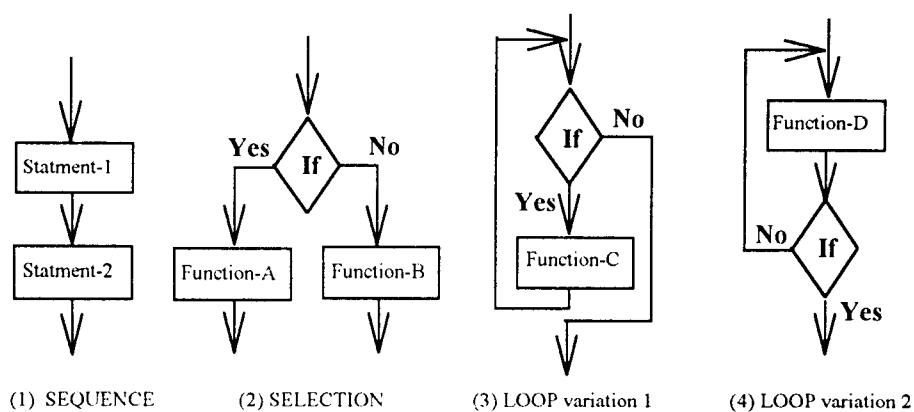


Figure 2. Command structures in structured programming design

(Please Go On To The Next Page)

- a. Was the **GOTO statement** frequently used in the completion of the assignments in your programming courses? (**CIRCLE one number**)
- 0 DON'T REMEMBER
 - 3 IN NONE OF THEM
 - 2 AT LEAST ONE, BUT LESS THAN THREE, OF THOSE COURSES
 - 1 IN MORE THAN THREE OF THOSE COURSES
- b. In general, how often was the **GOTO statement** used in the completion of the assignments in your programming courses? (**CIRCLE one number**)
- 0 DON'T REMEMBER
 - 1 VERY OFTEN (in more than 80% of the assignments)
 - 2 OFTEN (in more then 60% of the assignments)
 - 3 OCCASIONALLY (in about 30% of the assignments)
 - 4 HARDLY (in less than 10% of the assignments)
- 6a. Were **LOOP** structures frequently used in completion of the assignments in your programming courses? (**CIRCLE one number**)
- 0 DON'T REMEMBER
 - 1 IN NONE OF THEM
 - 2 AT LEAST ONE, BUT LESS THAN THREE, OF THOSE COURSES
 - 3 IN MORE THAN THREE OF THOSE COURSES
- b. In general, how often were **LOOP** structures used in completion of the assignments in your programming courses? (**CIRCLE one number**)
- 0 DON'T REMEMBER
 - 4 VERY OFTEN (in more than 80% of the assignments)
 - 3 OFTEN (in more then 60% of the assignments)
 - 2 OCCASIONALLY (in about 30% of the assignments)
 - 1 HARDLY (in less than 10% of the assignments)

(Please Go On To The Next Page)

PART II: BACKGROUND INFORMATION

INSTRUCTION: For questions 7 and 8 below, use the categories provided to indicate the range which most clearly fits the score you earned. Take item a in question 7 as an example, if you earned an average score of 67 in high school math courses, the correct response to the question would be enter the letter "N" in the space provided.

- | | | | | |
|--------------|--------------|--------------|--------------|---------------|
| (A) 0 to 5 | (B) 6 to 10 | (C) 11 to 15 | (D) 16 to 20 | (E) 21 to 25 |
| (F) 26 to 30 | (G) 31 to 35 | (H) 36 to 40 | (I) 41 to 45 | (J) 46 to 50 |
| (K) 51 to 55 | (L) 56 to 60 | (M) 61 to 65 | (N) 66 to 70 | (O) 71 to 75 |
| (P) 75 to 80 | (Q) 81 to 85 | (R) 86 to 90 | (S) 91 to 95 | (T) 96 to 100 |

7. Identify your performance in high school:

ENTER LETTER CATEGORY

- a. Average score of all math courses _____
- b. Average score of all your high school courses work _____

8. Identify the scores you earned in the College Entrance Examination:

ENTER LETTER CATEGORY

- a. Math _____
- b. Physics _____
- c. Chemistry _____
- d. English _____
- e. Total _____

(ENTER ACTUAL SCORE)

9. What is your gender? (**CIRCLE one number**)

- 1 MALE
- 2 FEMALE

10. What is your age?

_____ YEARS OLD

11. What is your class year?

_____ CLASS YEAR

(Please Go On To The Next Page)

12. What is your future plan after graduation from this university? (**CIRCLE one number**)

- 1 Find a job in a traditional position for computer science majors, such as PROGRAMMER, SYSTEM ANALYST, COMPUTING CONSULTANT, and the like
- 2 Find a job as a SALES REPRESENTATIVE IN A COMPUTER HARDWARE COMPANY
- 3 Find a job other than computer science related fields
- 4 Pursue graduate study in the computer science related field, such as COMPUTER SCIENCE, INFORMATION SCIENCE, MANAGEMENT INFORMATION SYSTEMS, OR COMPUTER ENGINEERING
- 5 Pursue graduate study other than in computer science related fields
- 6 Other (please specify: _____)
- 7 Don't know yet

13. How many computer science courses have you retaken in this university due to **NOT PASS**?

_____ COURSES

***** Thank you for your patience and the provision of valuable information! !**

=====

APPENDIX B

Evaluation for Content Validity

The purpose of this research is to determine if the academic achievement of college computer science majors in the ROC can be predicted by student scores on the College Entrance Examination (CEE), by high school performance, and from prior computer experience, particularly experience in structured programming.

To fulfill the goal described above, a questionnaire and interviews will be used to collect information required to complete this research. However, student scores for course work in university will be obtained from Registrar's office, and high school data as well as scores of the CEE will also be collected from the Registrar's office whenever they are available.

Based on the "validity, " "suitability," and "necessity" of the questions, with respect to the degree to which they match the purpose of the study, please indicate your opinion according to the categories given:

Code	Description
=====	=====
A	Accept as it appears
R	Acceptable, but requires rewording
D	Delete from the questionnaire
O	Other suggestion (please indicate)

Moreover, if rewording or more questions are recommended, please also indicate your suggestions in the space provided or directly adjacent to the individual questions.

A. QUESTIONNAIRE

The questionnaire comprises two sections:

I. Prior computer experience: number of computer courses taken prior to entering the university (question one), number of programming courses taken (question two), and the degree of emphasis upon structured programming methodology in programming courses (questions three to six).

II. Background information: includes student achievement in high school (question seven) and scores in College Entrance Examination (questions eight), future plan (question nine), gender (question 10), age (question 11), class year (question 12) and computer courses retaken in college (question 13).

Question =====	Purpose of the Question =====	Your Opinions =====
1	Obtain number of computer courses subjects have taken prior to entering university	1.
2	Obtain number of programming courses subjects have taken prior to entering university	2.
3a	Obtain number of programming courses top-down design was taught	3a.
3b	Obtain the extent top-down design was stressed	3b.
4a	Obtain number of programming courses modularity was taught	4a.
4b	Obtain the extent top-down design was stressed	4b.
5a	Obtain number of programming courses in which GOTO statement was often used	5a.
5b	Obtain frequency that GOTO statement was used	5b.
6a	Obtain number of programming courses in which LOOP statement was often used	6a.
6b	Obtain frequency that LOOP statement was used	6b.
7a	Obtain subject average scores for all high school math courses	7a.
7b	Obtain subject average scores for overall high school course work	7b.
8a	Obtain subject scores for CEE math	8a.

8b	Obtain subject scores for CEE physics	8b.
8c	Obtain subject scores for CEE chemistry	8c.
8d	Obtain subject scores for CEE English	8d.
8e	Obtain subject total CEE scores	8e.
9	Obtain subject gender	9.
10	Obtain subject age	10.
11	Obtain subject class year	11.
12	Obtain subject future plans after graduation	12.
13	Obtain number of computer courses retaken due to poor performance	13.

B. INTERVIEWS

The purpose of the interview is to collect additional data regarding the content and length of prior computer courses, as well as subject experience with structured programming in the prior programming courses. Information obtained from the interviews will be used to verify the data collected from administration of the questionnaire as well as to provide further information for interpreting the findings regarding prior computer experience.

<u>Question</u>	<u>Purpose of the Question</u>	<u>Your Opinions</u>
1	Obtain computer courses subject has taken prior to entering the university to verify subject responses to survey question 1	1.
2	Obtain a description of the content of prior computer courses to verify subject responses to survey questions 1 and 2	2.

- | | | |
|---|---|----|
| 3 | Obtain more information on prior computer courses to verify subject responses to survey questions 1 and 2 | 3. |
| 4 | Obtain information regarding the length of instruction of prior computer courses to verify subject responses to survey question 1 | 4. |
| 5 | Obtain information to examine subject knowledge in structured programming design to verify subject responses to survey questions 3 to 6 | 5. |
| 6 | Obtain information to examine which principles of structured programming design were taught in prior programming courses to verify subject responses to survey questions 3 to 6 | 6. |
| 7 | Obtain information to examine if principles of structured programming design were used in completing program assignments for prior programming courses to verify subject responses to survey questions 3 to 6 | 7. |
| 8 | Obtain information to determine if subject believed experience in structured programming was beneficial to his/her work in college computer science courses to provide more insight into such experience | 8. |
| 9 | Obtain information to identify confusing survey questions | 9. |

APPENDIX C
Interview Questions

1. What computer courses have you taken prior to entering this university?
2. What topics or concepts were taught in each of these class?
3. Where and when were these computer courses taken?
4. What was the length (in hours) of the instruction period of your computer courses (specify individually)? How many times did the class meet each week? For how many weeks did the class meet?
5. How would you define structured programming? Described your understanding on the principles of structured programming methodology (If the subjects is not able to clearly described those principles, the examples in the questionnaire will be shown).
6. Which of these principles, if any, were taught in your prior programming courses? Please specify. To what extent were those principles emphasized in these programming courses?
7. Did you use any of these principles in completion of the assignments in your programming courses --- in high school? --- in college? Please specify. If none, why not?
8. Do you believe your prior computer experience in programming was beneficial to your work in the college computer science courses? To which courses? In what way?
9. Are there items, if any, in the questionnaire that you consider confusing or ambiguous? Please specify.

APPENDIX D
Informed Consent Form

The following questionnaire is a part of a research study. The study attempts to find the relationship between a student's academic performance in a computer science program and his/her College Entrance Examination scores, high school academic achievement in math, prior computer experience, and factors which have been reported as influential upon student academic performance in college level computer science courses. Your help on this study will provide valuable information for improvement of admission processes, computer science curriculum development, and for more effective advice to students in selecting college majors.

As a participant, you will need to provide some background information by spending **15 minutes** completing the attached questionnaire. The researcher will collect your College Entrance Examination scores and scores of all the courses you have taken in this university by accessing your academic records in the Registrar's office. If you are willing to help the researcher complete the study, please sign your name in the space provided in the bottom of this form.

The confidentiality of each participant will be strictly maintained. All the data will be stored in a locked metal cabinet in the researcher's office and only the researchers will have access to the data collected. A four digit number will be assigned to each questionnaire and the information you provide will be combined with data from others subjects for analysis. The results of this research will be reported anonymously in the researcher's dissertation.

Participation is voluntary, refusal to participate will involve **no penalty in any form**. For questions about this research, please contact Dr. Maggie Niess at (503) 737-1818 (email niessm@ucs.orst.edu) or Mr. Allen Fan at (08) 744-0269 (email fant@ucs.orst.edu).

I, _____, understand the above information and
(please PRINT your name)

will participate in this study. I further give my consent to Mr. Fan to review my academic records in the Registrar's office.

Student ID #: _____ **Signature:** _____
(please PRINT your ID no.) (please SIGN your name)