Performance in an Introductory Computer Programming Course as a Predictor of Future Success for Engineering and Computer Science Majors

Authors:

Barbara T. Pioro, North Carolina A&T State University, Greensboro, NC, pioro@ncat.edu

Abstract — *In most schools, introductory computer programming courses are required for computer science as well as all engineering majors. It is generally believed that the programming courses are not just about programming per se, but that they provide a forum for teaching precise and logical thought processes. Computer programming courses constitute a necessary background for computer science majors by introducing basic concepts and techniques to be used and to be built upon in more advanced CS courses, and also make a valuable contribution to the foundation of engineering education. Programming courses are thought to provide a valuable framework for the development of problem solving and creative thinking skills for engineering as well as computer science majors. It is commonly acknowledged that many students experience problems with computer programming classes. However, it is also acknowledged that success in a major is not determined by a student's ability to code textbook problems in an introductory computer programming class. Until now, the relationship between performance in a programming course and success in an engineering major or computer science has not been well established. In particular, a relationship between performance on different types of computing tasks and a specific major has not been sufficiently examined. In general, engineering majors bring different sets of experiences and expectations from those brought in by computer science majors to the programming course; those factors have an effect on performance on tests and exams and manifest themselves in a variability of total scores and in a variability of scores on different types of computer programming problems. In this study we examined scores obtained from the final exam questions in a computer programming course offered in the Spring of 2002. The scores were then reviewed in the context of students' records at the end of the Spring 2004 semester. The Spring 2002 programming tasks involved ten multiple-choice questions and three programming problems. The problems were designed within the framework of the Rasmussen's skill-rule-knowledge model of human performance. Twenty-five engineering and computer science students took the final exam. The data obtained in Spring 2002 were examined in view of the students' ultimate performance and status. For each student, the analyzed data included scores obtained on individual exam problems, the student's self assessment scores for each exam problem, the GPAs from the Spring 2002 and Spring 2004 semesters, the major, as well as retention and progress in the program. This paper examines the predictive value of performance levels in introductory computer programming courses as indicators of overall student success in an engineering or computer science undergraduate program.*

Deleted:

Deleted: *the*

Deleted: *they*

Deleted: *skills*

Deleted: *The analyses of the Spring 2002 data identified a bimodal distribution of scores and showed significant differences in performance between the high-performing students and the low-performing students.*

Index Terms — *Learning computer programming, GPA, self-efficacy, gender differences.*

INTRODUCTION

In most schools, computer programming courses are required for all engineering, science and MIS/business majors. It has been widely recognized that many students, across various disciplines, find learning computer programming difficult. While research shows that many factors contribute to a student's performance in a computer programming class, it is still difficult to say what aspects of a student's background and personality determine his or her performance in the course. Factors such as GPA, SAT scores, time on computer, and performance in math classes have been examined to determine their relationship to performance in programming courses. However, none of them has proven to be a reliable predictor of programming performance [9].

Deleted:

In this study we wished to examine the factors that are known to be related to student performance in academic settings, as they exhibit themselves in various relationships in a sample of our population of students, in a particular computer programming course employing a specific design of test questions. The purpose of this study was to determine what challenges should be adressed in the classroom teaching techniques as well as in the design of the tutoring programs aimed at improving student performance in science and engineering courses. The grounds for this study was the belief that identifying and understanding factors that affect student performance in a computer programming course, which typically involves training in building semantic and syntactic knowledge in a very structured framework, would lead to not only to improvement of teaching and learning in programming courses, but also to greater student success rates in SMET courses in general. The skills related to building and using semantic and syntactic knowledge that are acquired in a programming course should be tranferable to other SMET courses where problem solving and derrivation of solution requires creativity, yet which must also be scientifically, conceptually, logicaly, and syntactically valid.

A cognitive model, based on the information processing approach to human memory and performance, proposes a syntactic / semantic distinction of the skills and knowledge required to account for successful computer programming performance [9]. In a programming course, a student is required to acquire knowledge and skills relevant to a particular programming language, as well as concepts and techniques common to many programming languages, in a continuous and sequential manner, in order to master basic programming skills. In the process of learning, the student must absorb and organize the newly acquired skills and knowledge in long term memory, and also must learn to use his or her short-term memory and working memory effectively.

The basic tasks that the student would be asked to perform in a programming course involve: generating a solution to a problem statement, understanding the existing program, and detecting errors in a program. To perform those tasks, the student must utilize his or her semantic and syntactic knowledge. Semantic knowledge involves programming concepts from the lowest through highest levels; this knowledge is acquired through learning from lectures, textbooks, problem solving, and programming experience – its acquisition is intellectually demanding. Syntactic knowledge is more related to learning the syntax and structures that are specific to a particular programming language; it is mostly acquired by rote and more easily forgotten than semantic knowledge [9]. Many researchers as well as computer programmers agree that successful programming performance requires the formulation of a mental representation of a problem that involves the semantic structures needed to generate a solution or to understand a presented program [1], [2], [9], [10].

Testing student knowledge in programming courses is cumbersome and difficult. The instructor must understand what is being tested in a particular test question and why. Grading computer programs requires a system that is consistent, precise, and replicable, yet which allows for individual differences in the derivation of the solution. Semantic knowledge tests involve generating a new programming solution and tracing an existing program. Research in psychology shows that the cognitive processes involved in generating a new solution are different from the processes involved in understanding a presented solution. Learning programming in a particular language requires learning of the language-specific syntax, thus grading the test must include evaluation of syntactic knowledge.

In this study, scores obtained on the final exam in a computer programming II with C++ course were examined in view of the factors that are associated with student performance in science, math, and engineering courses. The course was offered at the freshman level in the college of engineering. The relationships between the performance in the selected programming course and the GPA scores at the end of the semester in which the exam took place, Spring 2002, as well as the GPA scores at the end of the Spring 2004 semester, were examined in search of a predictor of academic performance. The exam scores were also examined in view of the relationship between the scores obtained from the General Purpose Self-Efficacy Scale, an instrument that has been tested in 14 cultures and shows, that among other things, high self-efficacy scores are related to higher achievent in academic settings [3], [7], [8]. The study also involved eliciting predicted scores just before the exam and estimated scores based on the just-completed exam. Detailed analyses of the relationships between actual performace and predicted and estimated scores on various parts of the exam was described in our previous paper [4].

There has been a persistent concern about underrepresentation of women in science, computer science, and engineering. To gain more insight into the education of women in science and engineering, in this study we also examined performance on the computer programming test separately for each gender and searched for gender-related differences in relationships between exam scores, GPAs, and General Purpose Self-Efficacy indices.

PROCEDURE

This study examined scores obtained on the final exam in the *Programming with C++* second-level course. Twenty-five students took the exam. The exam paper consisted of two parts. The first part contained ten multiple-choice questions. The questions showed multi-line fragments of C++ code and the students were required to trace the code and select output generated by the code in question. The second part of the exam consisted of three problem statements. For each problem, the students were required to write a short computer program.

The multiple-choice questions were selected from the data bank developed for the textbook used in the course. The programming problems were chosen from the short programming problem sections included in each chapter of the textbook [2]. The chosen problems were modified to fit the framework of Rasmussen's skill-, rule-, and knowledge-based levels model of human performance [5]. A solution to the first problem required the application of programming language, that is, syntax and structures of the C++ language, the skills that could be developed by repetition and memorizing. A solution to the second problem required application of the C++ rules in creating C++ structures such as functions. The third problem involved creating and manipulating C++ structures that in turn manipulated other structures. The solution to that problem required knowledge of the C++ language at a level that allowed manipulation of C++ structures as well as an understanding of the interactions between them.

Deleted: must be
Deleted:
Deleted:
Deleted:
Deleted:
Deleted:
Deleted: then,

The maximum attainable score for the multiple-choice part of the exam was 50 points. The skill-based, rule-based, and knowledge-based problems were assigned 10, 15, and 25 points respectively. The C++ code needed in the solutions to the programming problems was employed in the multiple choice question part of the exam. The exams were graded by the instructor. The scores received on multiple-choice questions and the three problem statements were used as the data examined in this study.

At the beginning of the exam and before receiving the exam papers, the students were asked to complete a questionnaire which asked them to provide predicted scores for each type of problem later given as the exam problem. Upon completion of the exam, the students were asked to examine their just-completed test work and provide an estimated score for each programming problem and for the multiple-choice problem section of the exam.

A questionnaire that contained, among other questions, the ten General Purpose Self-Efficacy questions was completed in the second week of the semester. The scores on those questions were extracted from the questionnaire and used in this investigation. The GPA scores for Spring 2002 and 2004 semesters were extracted from the students' records.

Among the twenty five students who took the final exam, there were seven women and eighteen men. Four women were from the Business School, one was an AS major, one was a computer science major, and one woman was 'undecided' but later became a computer science major. Six men were electrical engineering majors, four were computer science majors, one was from the Business School, one was an AS major, and six were 'undecided', of which four later transferred to computer science.

FINAL EXAM SCORES

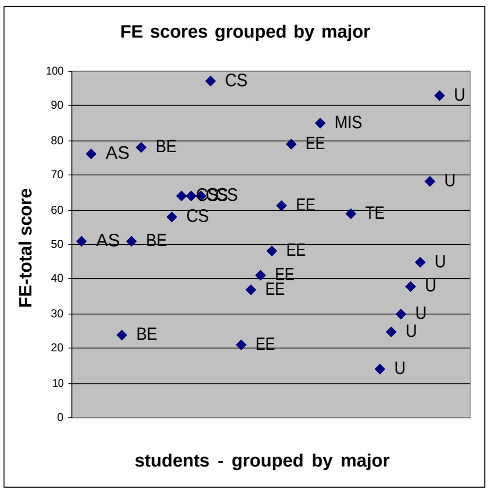

FIGURE 1
TOTAL EXAM SCORES GROUPED BY MAJOR.

Examination of the performance scores on the final exam confirms once more that programing skills vary greatly among the students in the class, and that some students have obvious difficulties in mastering the concepts and techniques required to perform programming tasks. Figure 1 illustrates the spread of scores for different majors. It is apparent that for different majors, the total scores vary from low to high. Figure 2, a histogram for the total scores obtaned on the final exam, shows distribution of grades for both the multiple-chice and problem solving / writing programs parts of the exam.

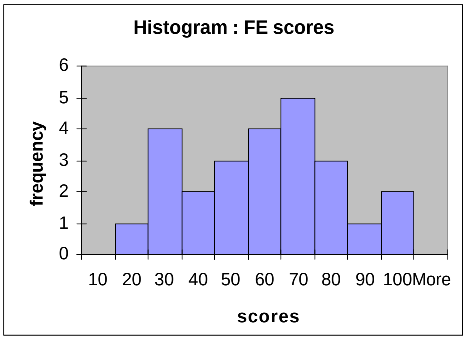

FIGURE 2
DISTRIBUTION OF TOTAL SCORES.

The distribution of the total scores obtained on the Final Exams obscures the difference between the distribution of scores obtained on the two distinct parts of the exam, - the multiple-choice question part and the programming problem part. Table 1 shows some descriptive statistics for the total scores, i.e., scores obtained on both parts of the exams, and the scores obtained on the multiple-choice question and programming problem parts of the exam.

Statistics	FETot	FEMCh	FEPrblm
Mean	54.8	33.9	20.9
Standard Error	4.6	1.9	2.9
Median	58.0	35.0	19.0
Mode	64.0	40.0	#N/A
Standard Deviation	22.8	9.5	14.6
Largest(1)	97.0	50.0	47.0
Smallest(1)	14.0	10.0	0.0

TABLE 1
DESCRIPTIVE STATISTICS FOR THE TOTAL EXAM, MULTIPLE-CHOICE QUESTIONS, AND PROGRAMMING PROBLEMS SCORES.

FINAL EXAM SCORES: MULTIPLE-CHOICE QUESTIONS AND PROGRAMMING PROBLEMS

One of the issues addresed in this investigation was the apparent difference in degree of difficulty that the students have in multiple-choice question exams and in writing programming solutions to the stated problems. Closer examination of the data indicates a large difference between the distribution of scores for the multiple-choice questions and the programming problems. The mean score on the multiple-choice question part is 34, and on the programming solutions part it is 21. In examining two paired sample test for means, t –test at $\alpha = 0.05$, d.f. =24, P(T<=t) = 4.783E-07, allows us to conclude that there is a significance diffrence between the means of the multiple-choice questions scores and the programming problems scores.

Deleted: a

The relationship between the multiple-choice question scores and the scores obtained on the programming problems is illustrated on Figure 5. The Pearson Correlation Coeficient, r = 0.77, indicates that there is a significant positive relationship between the scores obtained on the multiple-choice question part of the exam and the scores obtained on the programming problem part. The students who obtained high scores on the programming problems tended to obtain higher scores on the multiple-choice questions as well. Closer examination of this diagram shows that many students received high scores on the multiple-choice question part of the exam, but for the most of the students, performance on the programming problems was low. The scatterplot in Figure 5 indicates that among the students who received rather high scores on the multiple-choice questions, there is considerable variability in the scores received on the programming problems. However, the students who show high scores on the programming problems tend to have high scores on the multiple-choice questions.

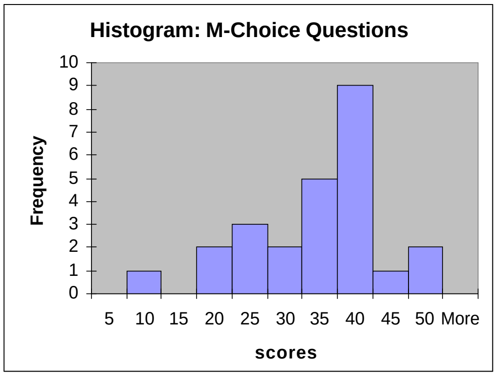

FIGURE 3
DISTRIBUTION OF SCORES FOR THE MULTIPLE-CHOICE QUESTIONS.

The results of this analysis support the cognitive model of human performance on cognitive tasks. Multiple-choice questions require a larger contribution of syntactic knowledge – recognition of syntax and structures which would provide a solution based on memory, or, recognition of sytax from which, incorporating semantic knowledge, a model of the solution can be constructed. Writing a program poses more demands on semantic knowledge, that is, it requires an understanding of the language that will allow a student to construct a cognitive model of the problem, which in turn would allow the student to choose appropriate conceptual languge structures and correctly apply them in the derivation of the programming solution. Moreover, generating a programming solution requires not only semantic knowledge, but also syntactic knowledge, since the relevant concepts retrieved from long-term memory must be utilized in a correct syntax acquired mostly by rote. The cognitive model of syntactic / semantic knowledge demands in computer tasks may be supported by an obsevation that students who achieve high performance levels on programming problems tend to obtain high scores on multiple choice problems. However, the reverse is not always the case, as can be seen on Figure 5 which shows the scatter diagram for the scores obtained on the multiple-choice questions and the scores obtained on the programming part of the exam.

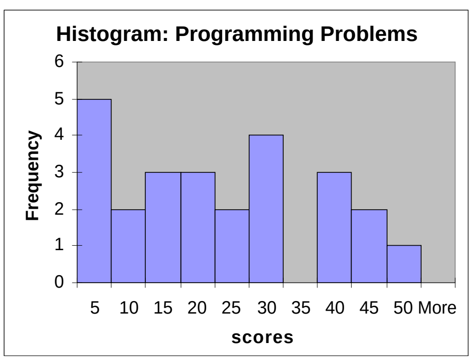

FIGURE 4
DISTRIBUTION OF SCORES FOR THE PROGRAMMING PROBLEMS.

Deleted: ¶
¶

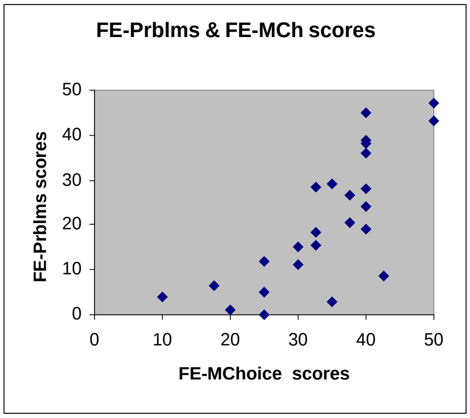

FIGURE 5
SCATTERGRAM OF PROGRAMMING PROBLEM SCORES VS MULTIPLE-CHOICE QUESTION SCORES.

GPA AND FINAL EXAM SCORES

GPA is a major determinant for the student's retention in the chosen major, or, at the university at all. Success in a freshman level course, such as a computer programming course, is often taken as an indicator whether the student will remain in the program or not. In this part of our investigation we examined the final exam scores in the computer programming course in view of the student GPA scores. First at the end of the semester in which the students took the course, and then, by looking at the students' GPA scores two years later.

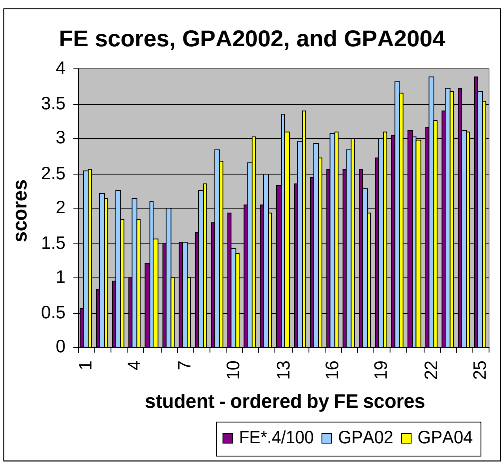

FIGURE 6
ORDERED FINAL EXAM SCORES, GPA 2002, AND PGA 2004.

Figure 6 shows the total Final Exam score, the GPA score from Spring 2002 semester, and GPA from the Spring 2004 semester for each student. As the diagram illustrates, in general, the increment in final exam scores is associated with the increase in the GPA scores for the 2002 as well as the 2004 semesters. The diagram also shows the drop in total exam scores is steeper than the drop in the GPA for 2002 as well as 2004. We had established that the distribution of scores for the multiple-choice part of the exam is different from the distribution of scores on the programming problem of the exam, so next we wished to examine the relationship between the scores on the two parts of the exam and the GPA scores for the 2002 and 2004 semesters. The strength of the relationships between those variables could be used as a predictor of a student's future success based on his or her current score in a computer programming course.

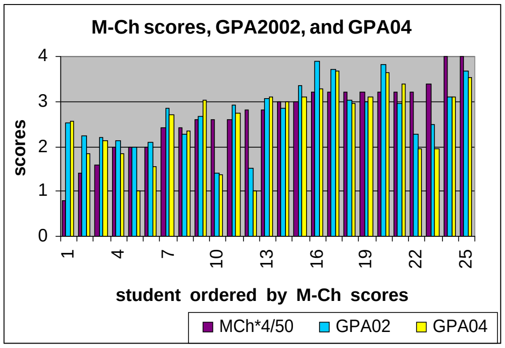

FIGURE 7
ORDERED MULTIPLE-CHOICE QUESTION SCORES AND GPA 2002 AND GPA 2004.

Figure 7 shows, for each student, ordered scores for the multiple-choice question part of the exam and the associated GPA 2002 and GPA 2004 scores. Figure 8 shows, for each student, ordered scores for the programming problem part of the exam and the associated GPA 2002 and GPA 2004 scores. As can be seen, the scaled multiple-choice question scores are closer to the GPA scores for both 2002 and 2004 semesters, and in some cases the multiple-choice question scores are higher than the GPA scores. That indicates that some students with low GPA scores could perform rather well on the multiple – choice question exams in a computer programming course. Figure 8 shows ordered scores from the programming problems and the GPAs scores from the Spring 2002 and Spring 2004 semesters associated with each exam score. Figures 7 and 8 show that the drop in performance is associated with the drop in GPA scores. Both figures show that the rate of decrement in the exam scores is much greater than the rate of decrement in the GPA scores associated with the students' overall grades. Moreover, in view of the GPA scores associated with the exam scores, the drop in exam scores is much more profound for the programming part of the exam than for the multiple-choice question part.

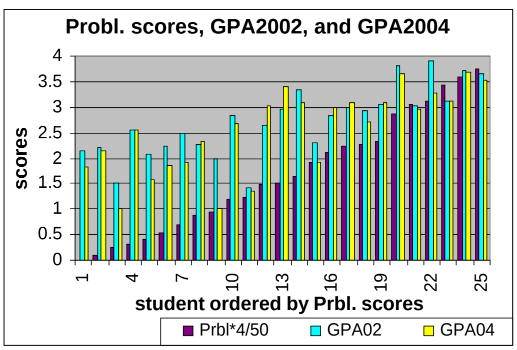

FIGURE 8
ORDERED PROGRAMMING PROBLEM SCORES AND GPA 2002 AND GPA 2004.

Figure 8 suggests that many students who do poorly on the programming part of the exam have lower GPA scores than the students who perform at the higher level. However, while their GPA two years later is, in general, lower than in 2002, it is not as dramatically affected as their programming problem scores would predict. This would suggest that, when students stay at school and take courses, even if they find problem solving difficult, they gradually develop problem solving skills as time goes on, since a passing grade in many other SMET courses would also require application of problem solving skills.

Relationships between 2002 GPAs and multiple-choice question scores and programming problem scores are illustrated on scatterplots shown in Figure 9 and Figure 10 respectively. The scatterplot showing the relationship between GPA 2002 scores and scores obtained on the multiple-choice questions indicates that the relationship is not very strong, r = 0.54. For the programming problems, Figure 10 shows a strong linear relationship between the GPA 2002 at the time when the exam took place and the scores obtained on the programming problems with r = 0.80. Moreover, the distribution of scores shows that in that linear relationship the exam scores increase rapidly with the increment in the GPA scores.

Deleted:

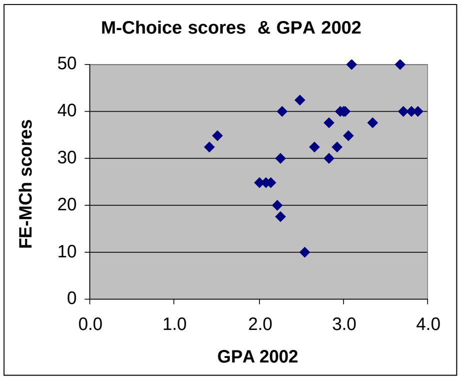

FIGURE 9
SCATTERGRAM OF M-CHOICE QUESTION SCORES VS. GPA 2002.

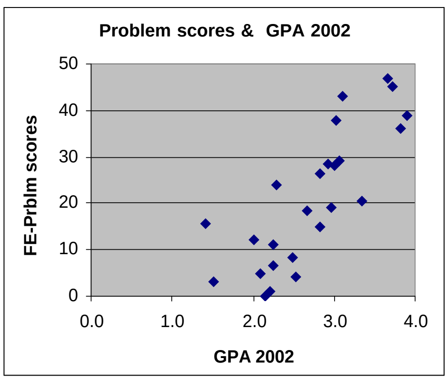

FIGURE 10
SCATTERGRAM OF PROGRAMMING PROBLEM SCORES VS. GPA 2002.

The relationships between GPA 2004, i.e, GPA from the semester two years after the exam, and the scores from the multiple-choice question part of the exam and the scores from the programming problem part are shown in Figure 11 and Figure 12 respectively. The question of interest was whether the degree of relationship between the scores and the GPA will hold two years after the test. The scattergram of exam scores and GPA 2004 shows that the multiple-choice question scores are related less to GPA 2004 than to GPA 2002, i.e., r = 0.51. Thus scores obtained on the multiple-choice problems are not strong predictors of the student success. The strength of relationship between the programming problem scores and GPA 2004 is also of a smaller degree but remains relatively strong with r = 0.76. Based on those observations, we may postulate that by knowing a student's score on the programming problem part of an exam in a computer programming course, we may estimate his his or her GPA two years later with a considerable degree of accuracy, and thus make predictions about the student' success in the future.

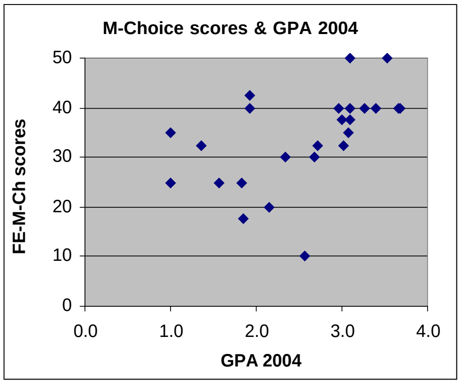

FIGURE 11
SCATTERGRAM OF M-CHOICE QUESTION SCORES VS. GPA 2004.

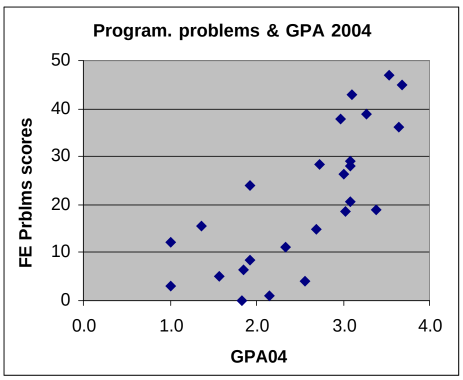

FIGURE 12
SCATTERGRAM OF PROGRAMMING PROBLEM SCORES VS. GPA 2004.

We have also examined the corelation between the total exam scores and the GPA 2002 and GPA 2004 scores. The correlation coeffient for the total exam scores and GPA 2002, r = 0.74, and the correlation coefficient for the total scores and GPA 2004, r = 0.70. The correllations between different scores and GPA 2002 and 2004 are displayed in Table 2. The table shows that the strongest relationships is between the scores obtained on the programming part of the exam and the GPA scores at the time the student took the exam with r = .80. Yet, overall performance on the final exam shows a strong relationship between the GPA 2002 and the GPA scores two years later with r = 0.74 and r = 0.70 respectively. The relationship between performance on multiple-choice questions and the GPA score in 2002 or 2004 doeas not show a strong relationship with r = 0.54 and r = 0.51 respectively, and thus it is a weaker predictor of the student's success in the program.

Our investigation revealed that for our group of students the strongest correllation, with r = 0.92, was between the GPA 2002 and GPA 2004; GPA showed itself to be a strong predictor of student success. Figure 13 illustrates the relationship between GPA 2002 and GPA 2004 for the students who took the final exam in the course. The GPA 2002 and GPA 2004 scores show a strong linear relationship.

Correlation Coefficients	r
GPA 2002 & FE total scores	0.74
GPA 2002 & FE M-Ch scores	0.54
GPA 2002 & Prblm scores	0.80
GPA 2004 & total scores	0.70
GPA 2004 & FE M-Ch scores	0.51
GPA 2004 & Prblm scores	0.76
GPA 2002 & GPA 2004	0.92

TABLE 2
SUMMARY OF THE CORRELATION COEFFICIENTS FOR THE EXAM SCORES AND GPA.

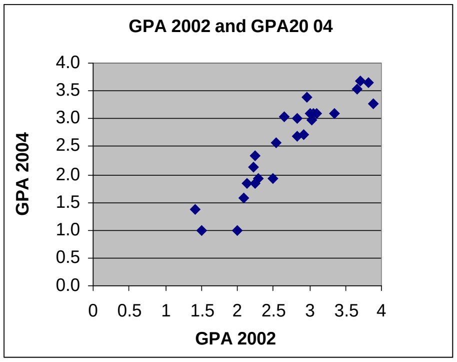

FIGURE 13
SCATTERGRAM OF GPA 2002 VS. GPA 2004.

Additional examination of the GPA indices uncovered a particular pattern of changes in the GPA scores. Figure 14 shows sorted GPA 2002 scores and the differences in GPA scores between GPA 2004 and GPA 2002 scores. On Figure 14, the decrement in the GPA scores appears to be more salient in the lower range of GPA 2002 scores than in the upper range. However, the correlation coefficient r = 0.23, does not indicate a strong relationship between the GPA scores and the values representing a decrement in the GPA scores over two years. However, the diagram does show that in the group of students participating in this study, in general, the students in the lower range of GPA scores in 2002, tend to have decrement in their GPA scores in 2004. However, the value of the decrement is not directly related to the value of the GPA score.

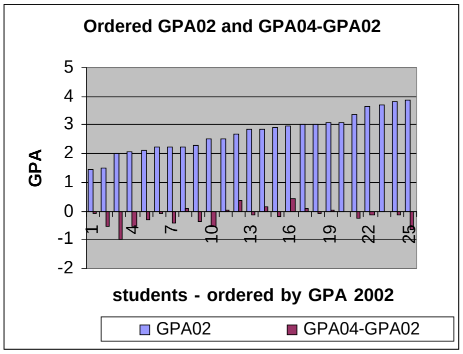

FIGURE 14
ORDERED GPA 2002 SCORES AND THE ACCOMPANYING DIFFERENCES BETWEEN GPA 2004 AND GPA 2002.

GENERAL PERCEIVED SELF-EFFICACY SCALE

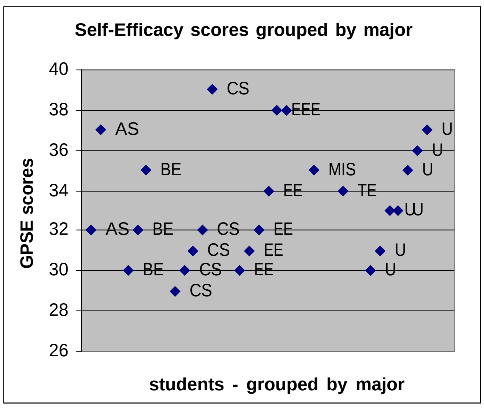

FIGURE 15
GENERAL PERCEIVED SELF-EFFICACY SCORES FOR STUDENTS GROUPED BY MAJOR.

The final exam scores were also examined in view of the scores obtained from the General Perceived Self-Efficacy questionnaire. The concept of Perceived Self-Efficacy reflects an optimistic self-belief – the belief that one can perform novel or difficult tasks in a variety of stressful situations, known as "can-do" cognition. The construct of self-efficacy introduced and developed by Bandura refers to personal action control. Research shows that self-efficacy is reflected in a strong sense of competence, which facilitates cognitive processes and performance in various situations, including academic achievement. High self-efficacy levels can enhance motivation. People with high self-efficacy set themselves higher goals, invest more effort, show more resilience, and persist longer than those with low self-efficacy [7]. However, the theory of self-efficacy does not imply that simply the belief in one's capabilities is sufficient to attain high performance; to perform at the satisfactory level a right combination of self-belief and level of skills and knowledge is required. Self-efficacy beliefs are significant determinants of how necessary skills and knowledge to accomplish a particular task are acquired and then how they are used. According to Bandura, beliefs of personal efficacy constitute the key factor of human agency [3].

Deleted: the

At the beginning of the semester, the students were asked to complete a questionnaire that included the General Perceived Self-Efficacy (GPSE) scale [9]. We wanted to examine the relationship between the scores obtained on the GPSE questionnaire with the scores obtained on the final exam in the Computer Programming II course, the GPA scores at the time the students took the exam, and the GPA scores two years after the exam took place, to see whether the self-efficacy theory postulates would bear themselves out in our study.

To determine whether we could state that the students enrolled in a particular major elicit higher scores on the GPSE, we grouped the students by major and constructed a diagram showing the total GPSE scores for different majors. The scores were sorted within the given major. Figure 15 shows the sorted GPSE scores sorted by major. As the diagram shows, for each major, the scores vary within the major, and based on the scores obtained within our sample of the student population, we cannot conclude that students in any particular major distinguish themselves by high a degree of self-efficacy, or, by beliefs in their capabilities and competence to control their environment in a wide range of novel and stressful situations.

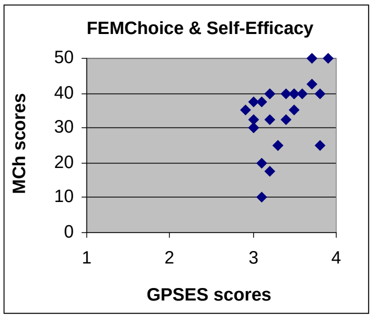

FIGURE 16
SCATTERPLOT FOR MULTIPLE-CHOICE QUESTION SCORES VS. GENERAL PERCEIVED SELF-EFFICACY SCORES

The GPSES scores and the total exam scores, multiple-choice question scores, and programming problem scores were examined to determine the strength of the relationships between the self-efficacy scores and performance on the exam. The relationship between total scores obtained on the final exam and the scores obtained on GPSES questionnaire for all students as represented by the coefficient of correlation between the two variables is positive, but not very strong with r = 0.45. Further examination of scores indicates a stronger relationship between the GPSES scores and multiple-choice question scores than between the GPSES scores and programming problem scores with r = 0.47 and r = 0.39, respectively.

Deleted:

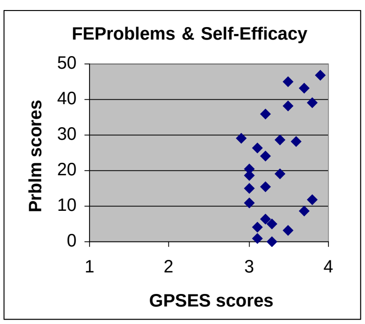

FIGURE 17
SCATTERGRAM FOR PROGRAMMING PROBLEM SCORES VS. GENERAL PERCEIVED SELF-EFFICACY SCORES

All students provided high scores on the self-efficacy questionnaire; the Pearson Corellation Coefficient confirms a positive relationship between beliefs in personal capabilities and the exam scores. However, that relationship is not very strong, and thus we cannot conclude with great confidence that the students with high scores on the General Perceived Self-Efficacy Scale deliver higher performances on any particular type of the test. The scatter diagrams in Figure 16 and Figure 17

Deleted: 's

show the relationships between the GPSES scores and multiple-choice question scores and programming problem scores, respectively.

GPA 2002, GPA 2004, AND GENERAL PURPOSE SELF-EFFICACY SCALE

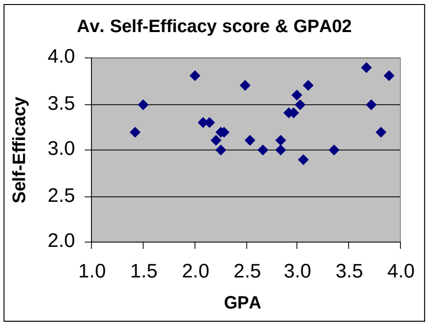

FIGURE 18
SCATTERPLOT OF AVERAGE OF GPSE SCORES VS. GPA 2002.

One of the questions of interest in this study was the relationship between General Purpose Self-Efficacy scores and the GPA scores. We expected that there would be a strong relationship between those two sets of scores. The relationship between the General Purpose Self-Efficacy (GPSE) scores and 2002 GPA scores is shown on Figure 18, and the relationship between GPSE scores and 2004 GPA scores is shown on Figure 19. The coefficient of corellation for GPSE and 2002 GP is r = 0.22, and for GPSE and 2004 GPA, it is r = 0.01. Therefore, we conclude that for our group of students, there is either a very weak or non-existent corellation between their beliefs in their own competence, their general beliefs in being able to cause an event, (including ‘can-do’ related to academic achievement as they expresed by elicited self-efficacy scores) and their academic performance as measured by the GPA index.

Deleted: e

Deleted:

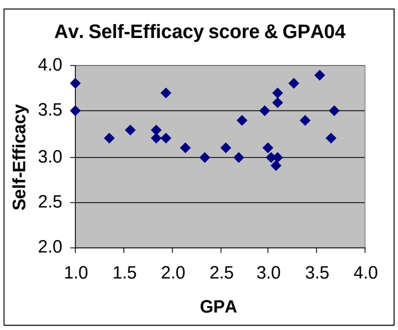

FIGURE 19
SCATTERPLOT OF AVERAGE OF GPSE SCORES VS. GPA 2004.

The self-efficacy concept would suggest that the students who scored highest on the General Purpose Self-Efficacy instrument would show more perseverance, tenacity, and control over their environment, and thus would show higher achievment levels as measured by the GPA index. Closer examination of the changes in GPA between 2002 Spring semester and 2004 Spring semester reveals that in the group of students participating in this study, the students with higher scores on the General Purpose Self-Efficacy Scale experienced greater decrease in their GPA index between the 2002 and 2004 semesters. Figure 20 shows the relationship between the average self-efficacy scores and the changes in GPA. The

Deleted:

relationship between the average score obtained on the self-efficacy questionnaire and the change in GPA between the Spring 2002 semester and the Spring 2004 semester as expressed by the correlation coefficient proves to be negative, with r = -0.44.

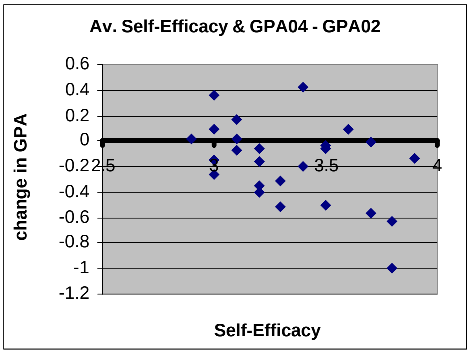

FIGURE 20
SCATTERPLOT FOR AVERAGE GENERAL PURPOSE SELF-EFFICACY SCORES VS. CHANGES IN GPA SCORES BETWEEN 2002 AND 2004.

GENDER DIFFERENCES IN PERFORMANCE, SELF-EFFICACY, AND GPA

In this part of the exploration of the relationships between performance on computer programming tests, self-efficacy, and GPA, the gender difference on those variables were examined. One of the first questions of interest was whether the self-efficacy levels were different for men and women students in this course. Comparision between the self-efficacy scores for men and women showed that the level of self-efficacy was slightly higher for men than women. Table 3 summarizes the gender specific information.

	Av. Self-Effic.	Prog. Problems	Multiple-Choice	GPA 2002	GPA 2004
Females	3.20	2.29	2.94	3.15	3.21
Males	3.40	1.44	2.62	2.55	2.30

TABLE 3
AVERAGE PGSE, PROGRAMMING PROBLEM, MULTIPLE-CHOICE QUESTION, GPA 2002, AND GPA 2004 SCORES FOR MEN AND WOMEN.

Figure 21 illustrates the information displayed in Table 3. In order to compare the values, the exam scores were converted to the scale 0 through 4.The average score on the General Perceived Self-Efficacy Scale was higher for men than women in this group, but the t-test indicates that we cannot reject the null hypothesis and must conclude that the GPSE means for men and women do not differ significantly, at $\alpha = 0.05$, $P(T <= t) = 0.18$. The average scores obtained on both programming problem and multiple-choice parts of the final exam were higher for women than for men. For the programming problem scores, the one-tail t-test for means of the two samples indicates that at $\alpha = 0.05$, $P(T <= t) = 0.03$. The scores obtained on programming problems are significantly higher for women than men. For the scores obtained on the multiple-choice questions, the one-tail t-test for means indicates that, at $\alpha = 0.05$, $P(T <= t) = 0.10$, and we conclude that the means of the scores obtained on multiple-choice questions are not significantly different for men and women.

Deleted: .

Deleted:

To test whether the differences between performance on multiple-choice questions and programming problems are significant for both men and women, we performed paired two sample t-tests for the means of the scores on the two parts of the exam, for both men and women. For women, we find that, one tail t-test at $\alpha = 0.05$, $P(T <= t) = 0.026$, allows us to conclude that the scores on the multiple-choice questions are significantly higher than the scores on the programming problems. For men, $P(T <= t) = 1.62E-06$, and thus we may conclude that the difference between the mean scores for multiple-choice questions and programming problems are highly significant.

The GPA scores were higher for women than men in both 2002 and 2004 semesters. To test whether the differences in the GPA indices were significantly different, we use the one-tail t-test for means and find that, at $\alpha = 0.05$, for GPA 2002, $P(T <= t) = 0.009$; and for GPA 2004, $P(T <= t) = 0.0003$, and thus we conclude that in both Spring 2002 and Spring 2004 semesters, the GPA scores were significantly higher for women than men.

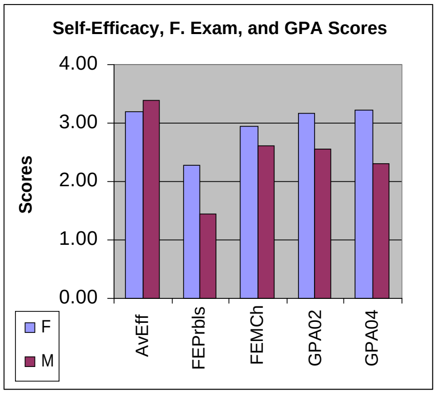

FIGURE 21
GENERAL PERCEIVED SELF-EFFICACY, PROGRAM. PROBLEM, MULTIPLE-CHOICE, GPA 2002, AND GPA 2004 SCORES.

FIGURE 22
GPA 2002 AND GPA 2004 FOR WOMEN.

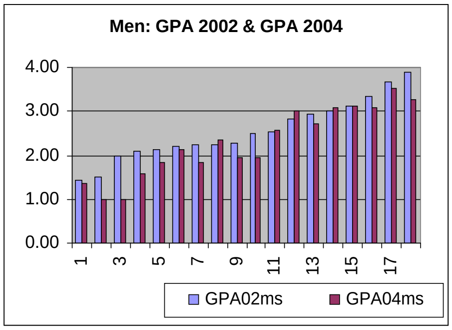

FIGURE 23
GPA 2002 AND GPA 2004 FOR MEN.

The stability of the GPA scores, or knowing the direction of changes in the GPA scores, should allow for greater accuracy in prediction of student success in a given program. Two sample, two tail t-test for means indicates that, at α = 0.05, P(T <= t) = 0.016 , and we conclude that the change in the GPA scores between 2002 and 2004 for women was not the same as the change in the GPA index for men. Further investigation reveals that the paired two sample two-tail t-test, at α = 0.05, P(T <= t) = 0.54 show that for women, there is no significant difference between their 2002 GPA and 2004 GPA scores. For men, the t-test indicates that difference between the means of 2002 GPA and 2004 GPA is significant, at α = 0.05, P(T <= t) = 0.0023. For the women who took part in our studies, between the Spring 2002 and the Spring 2004 semesters, the mean GPA increased from 3.15 to 3.21, but the change was not statistically significant. For men, between Spring 2002 and Spring 2004 semesters, the mean of the GPA scores decreased from 2.55 to 2.30, and the difference in the means has shown to be statistically significant. Figure 22 and Figure 23 show sorted GPA 2002 and associated GPA 2004 for women and men respectively. Figure 24 and Figure 25 illustrate the relationship between the GPA scores in the 2002 semester and the difference between 2002 and 2004 GPA scores for both women and men.

Deleted: mean

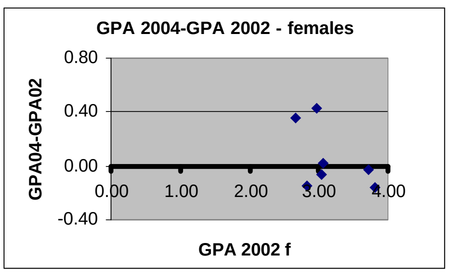

FIGURE 24
SCATTERPLOT OF GPA 2002 SCORES VS. DIFFERENCES IN GPA BETWEEN 2002 AND 2004 FOR WOMEN.

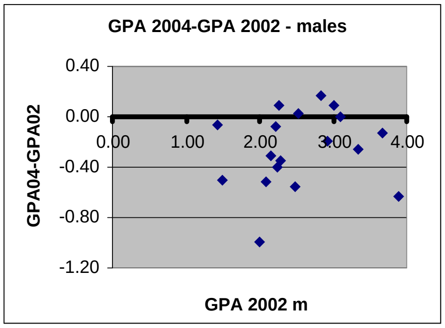

FIGURE 25
SCATTERPLOT OF GPA 2002 SCORES VS. DIFFERENCES IN GPA BETWEEN 2002 AND 2004 FOR MEN

To better understand the possible gender-related differences in an individual student's performance in a computer programming course as they are related to self-efficacy, we calculated coefficients of correllation between the sets of variables under consideration. We derived correlation coefficients between scores elicited on the General Perceived Self-Efficacy Scale (GPSES) and the total scores obtained on the final exam, the scores from the problem solving part of the exam, and the scores from the multiple-choice question part of the exam. Correlation coefficients were also calculated for GPSE scores and the GPA indices. The strength of the relationship between the GPA indices and the exam scores was examined. were The summary of the results is shown in Table 3.

Variables	women	men
Self-Effic & FE TotSc.	0.70	0.56
Self-Effic & M-Choice	0.82	0.54
Self-Effic & Problems	0.60	0.51
Self-Effic & GPA2002	0.42	0.36
Self-Effic & GPA2004	0.50	0.13
Problems & GPA2002	0.80	0.78
Problems & GPA2004	0.62	0.75
M-Choice & GPA2002	0.66	0.51
M-Choice & GPA2004	0.77	0.47

TABLE 3
COEFFICIENTS OF CORRELATION BETWEEN SELF-EFFICACY, EXAM SCORES, AND GPA FOR MEN AND WOMEN

As the results presented in Table 3 show, there is a strong relationship between the GPSES (self-efficacy) scores and multiple-choice exam scores for women, r = 0.82, but that relationship is much weaker for men, r = 0.54. The coefficient of correlation is not very high for the GPSES scores and the scores obtained on the programming problem part of the exam, r = 0.60 for men, and r = 0.51 for men; the GPSES scores were high but the programming performance varied greatly.

The relationship between the GPSES scores and the GPA 2002 index for both men and women is not very strong, 0.36 and 0.42 respectively; all students rated themselves rather high on the self-efficacy scale, but the GPA 2002 indices show considerable variability. The relationship between GPSES scores and GPA scores becomes stronger with time for women, from 0.42 to 0.50, but decreases for men, from 0.36 to 0.13. On the bases of these results, it could be concluded that the strong self-efficacy beliefs as expressed by the students in the Spring 2002 semester did not appear to have a marked impact on their academic performance as measured by the GPA index.

As could be expected, there is a strong linear relationship between the scores obtained on the problem solving part of the exam and the GPA scores at the time the students took the test for both men, r = 0.78, and women, r = 0.80. In 2004, the coefficient of correlation decreases slightly for men, to r = 0.75, but there is a large dropin the strength of that relationship for women, to r = 0.62.

Deleted: 3
Deleted: ,
Deleted: average
Deleted: scaled
Deleted: on
Deleted: s
Deleted:
Deleted: as
Deleted: GPA scores in 2002 and 2004 semesters. The correlation coefficients were also calculated for exam scores and GPA scores.
Deleted: ¶
Deleted: scores for the
Deleted: since
Deleted: in the results on the problem solving part of the exam varied greatly
Deleted: for women
Deleted: in time
Deleted: for both
Deleted: , and women, to r = 0.62,
Deleted:

There is a strong relationship, r = 0.77, between women's performance on multiple-choice part of the exam and their GPA scores two years later, i.e., 2004, a marked increment from 0.66 in 2002. The relationship between men's performance on the multiple-choice questions and their GPA scores is not very strong in 2002, r = 0.51, and it decreases furtherfor the GPA scores in 2004, r = 0.47. In general, except for those related to solving problems with programming in C++ and GPA scores two years later, the coefficient of correlation values show stronger relationships between examined variables for women than for men, and thus we may attempt to predict success in engineering and computer science with grater confidence for women than for men.

Female students showed lower, but not significantly different scores, on the General Purpose Self-Efficacy Scale, yet they had greater GPAs than male students in both Spring 2002 and Spring 2004 semesters. Women also showed greater stability in their GPAs along their academic careers than men did. Women achieved higher scores than men did on both types of the exam problems, multiple-choice as well as programming. When the exam scores were scaled in order to compare them with the GPA levels, it could be seen that while both men and women performed poorly on the programming problem part of the exam, on average, women performed at a level of 0.89 below below their 2002 GPA and 0.92 their 2004 GPA, and men performed at a level of 1.12 and 0.86 below their 2002 GPA and 2004 GPA respectively.

The increased difference between the problem solving scores in 2002 and 2004 GPA indicates that, in spite of poor problem solving skills on programming tasks, women improved their overall performace over the two following years. On the other hand, for men, poor performance scores on programming tasks may have a relationship with the decline of overall academic performance during the two years following the exam in 2002. On the multiple-choice part of the exam, women performed at a level of 0.21 below their 2002 GPA, but men performed at a level 0.07 above their 2002 GPA. High self-efficacy scores appear to have stronger relationships with exam performance for women than for men. For the students who took the computer programming course in Spring 2002, it could be said that, in accordance to the self-efficacy theory, women seemed to exercise stronger control over the outcome of their actions related to learning and performance than men – the outcome of that exercise of control was better academic overall performance for women than men. For men who took the course, on average, the high self-efficacy score that would indicate strong beliefs in their capabilities, apparently was not match with sufficient levels of computer programming skills and knowledge or, stated in terms of the cognitive theory model, with sufficient levels of semantic and syntactic knowledge, to result in a high level performance. Thus the self-efficacy score was not a good predictor of academic success.

Forty students were registered in the course at the beginning of the semester, ten women and thirty men. Twenty-five students took the final exam, seven women and eighteen men. Fifteen students withdrew from the course during the semester, three women and twelve men. The attrition rate thus could be stated as 3/10 = 0.30 or 30% for women, and 12/30 = 0.40 or 40% for men. The attrition rate for the whole course is 15/40 = 0. 375 or 37.5%. The GPA indices, performance in the course until the student withdrew, the General Purpose Self-Efficacy score, or reasons for which the student left the course were neither examined nor considered in any way in this study.

PREDICTED AND ESTIMATED SCORES

Student self-evaluation has been recognized as one of the major factors that enhances student learning. Moreover, it has been argued that the ability to evaluate one's own work constitutes one of the factors necessary in the development of lifelong learning skills. In this study we used students' self-assessment ability first to predict the scores on presented exam problems based on their beliefs in their capability to provide solutions , and then to obtain expected scores on a just-completed exam. Later we examined the relationship between the expected scores derived through the self-assessment process, and the actual scores assigned by the instructor. The level of accuracy of the predicted and estimated scores was then examined in view of the level of the actual performance. The relationship between predicted, actual, and estimated scores in this exam was discussed in our previous paper [4]. The first question of interest in this part of our study was how a student's ability to estimate his or her skills and knowledge level relevant to exam problems before a particular exam was related to his or her 2002 and 2004 GPA scores. The other question was related to a student's ability to evaluate his or her completed work and their overall performance as expressed in GPA 2002 and GPA 2004.

As discussed in the previous paper, an examination of different types of scores indicated that there was a significant difference between the predicted scores and the actual scores. We also found that there was a significant difference between the actual scores and the elicited estimated scores. An examination of the relationship between the estimated total score and the difference between the estimated total and actual total scores from the final exam revealed a strong negative linear relationship with the correlation coefficient, r = -0.84. The multiple-choice questions scores showed a strong negative relationship between the actual scores and the difference between the estimated and actual scores with r = -0.81. The correlation coefficient for actual scores on all three programming problems and the difference between the actual and estimated scores was shown to be, r = -0.73. The analysis showed that the lower the actual total score, the greater the difference between the actual and the estimated score. For all students, the lower actual scores showed larger differences in estimates for both parts of the exam. The estimated scores on multiple-choice questions were closer to the actual scores than the estimated scores on the programming part of the exam. For the multiple-choice questions the average of the actual scores was 33.9 and the average of the estimated scores was 39.7. For the programming problems the average of actual scores was 21.0 and the average of estimated scores was 32.0.

Deleted: ,

Deleted: An examination of the relationship between the differences between the actual and estimated scores, and the GPA 2002 and GPA 2004 scores, shows positive relationships with r = 0.68 and r = 0.69 respectively. The higher between the differences between estimated scores and actual scores, and the GPA scores, for the programming problem part of the exam, are, r = 0.67 and 0.65 for 2002 and 2004 respectively. Based on those results, it can be concluded that the ability to evaluate one's performance has a positive relationship with one's academic success as measured by the GPA index. Similar results were obtained for the difference between the predicted and the actual scores. The abilty of the student to asses his or her state of skills and knowledge is has a positive relationship with the GPA index.

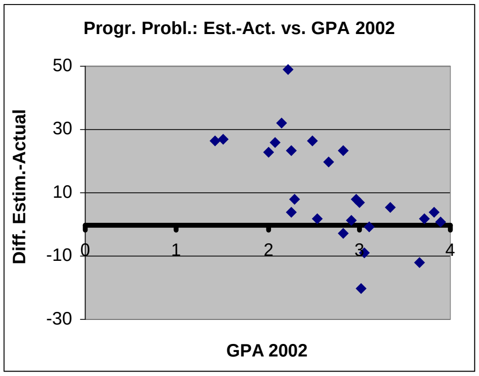

FIGURE 26

Deleted: 4

PROGRAMMING PROBLEMS: SCATTERPLOT OF DIFFERENCES BETWEEN ESTIMATED AND ACTUAL SCORES VS. GPA 2002

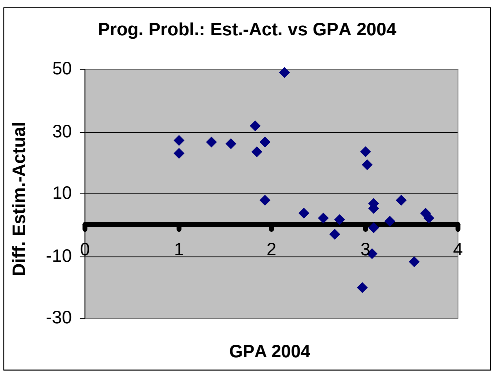

FIGURE 27

Deleted: 5

PROGRAMMING PROBLEMS: SCATTERPLOT OF DIFFERENCES BETWEEN ESTIMATED AND ACTUAL SCORES VS. GPA 2004

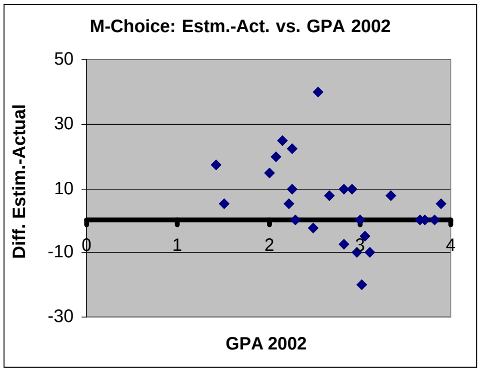

FIGURE 28
MULTIPLE-CHOICE QUESTIONS: SCATTERPLOT OF DIFFERENCES BETWEEN ESTIMATED AND ACTUAL SCORES VS. GPA 2002

Deleted: 6

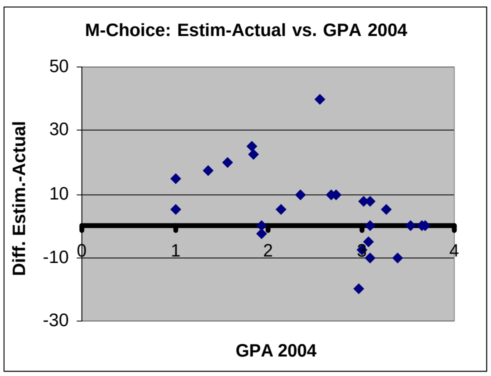

FIGURE 29
MULTIPLE-CHOICE QUESTIONS: SCATTERPLOT OF DIFFERENCES BETWEEN ESTIMATED AND ACTUAL SCORES VS. GPA 2004

Deleted: 7

For the total exam scores, an examination of the relationship between the differences between the estimated and actual scores, and the GPA 2002 and GPA 2004 scores, shows negative relationships with r = -0.68 and r = -0.69 respectively. The higher the student's GPA score, the better is his or her estimate of the work just completed. More detailed analyses of scores show that for the multiple-choice questions, the ceofficients of correlation between the differences between the estimated and actual scores, and the GPA scores are r = -0.45 and r = -0.49, for 2002 and 2004 respectively. The strengths of relationship between the differences between the estimated and actual scores, and the GPA scores, for the programming problem part of the exam, are r = -0.67 and -0.65 for 2002 and 2004 respectively. Based on those results, it can be concluded that the ability to evaluate one's performance has a positive relationship with one's academic success as measured by the GPA index. Similar results were obtained for the differences between the predicted and the actual scores. The abilty of a student to asses his or her state of skills and knowledge has a positive relationship with his or her GPA index. Figures 26 through 29 show the relationships between differences in estimated and actual scores and the GPA indices for the multiple-choice questions and the programming problems. Figure 30 and Figure 31 show ordered differences between the estimated and actual scores and the associated GPA scores for the programming problem part of the exam and for the multiple-choice question part of the exam respectively.

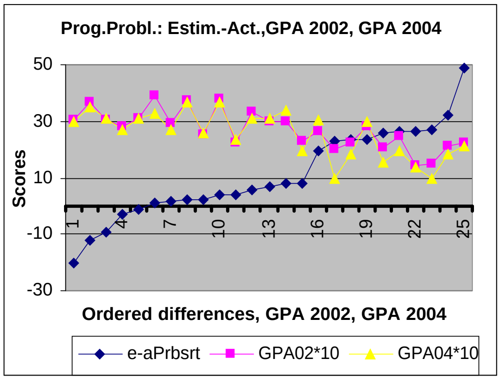

FIGURE 30
PROGRAMMING PROBLEMS: DIFFERENCES BETWEEN ESTIMATED AND ACTUAL SCORES, AND ASSOCIATED GPA 2002 AND GPA 2004

Deleted: 28

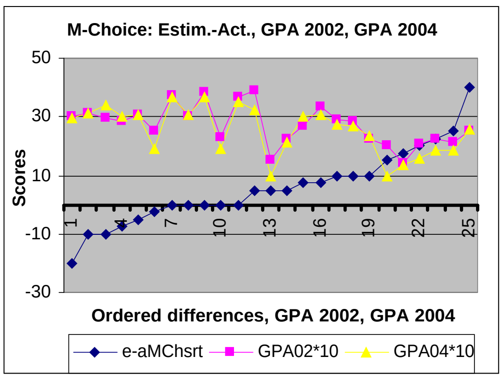

FIGURE 31
MULTIPLE-CHOICE QUESTIONS: DIFFERENCES BETWEEN ESTIMATED AND ACTUAL SCORES, AND ASSOCIATED GPA 2002 AND GPA 2004

Deleted: 28

Deleted: PROGRAMMING PROBLEMS

SUMMARY AND DISCUSSION

In this study, we considered those factors that in other studies had proven to have relationship to student success. We examined their relationships to student performance in an introductory computer programming course taken by engineering, computer science, and business students. The purpose of this examination was the search for verifiable results that could be applied to improving chances for student success in our educational settings. We anticipated that the insights derived from this examination would be utilized in classroom teaching techniques, in individual learning methods in structured tutorials. In this study, we analysed the scores obtained on the final exam, the scores elicited from the students before and after the exam, the GPA indices, and the scores obtained on the General Purpose Self-Efficacy Scale. More specifically, we wanted to establish the relationships that exist between student performance on a computer programming exam and various variables that may have predictive value related to student success. We also hoped to identify areas that would suggest directed intervention in order to optimize student success.

Deleted: factors that have proven to have relationships to student success

Deleted: ,

Deleted: , as well as in search of indicators that can serve as predictors of student success

Deleted: in an introductory computer programming course

One of the most salient factors identified in this study was the significant difference in scores obtained on multiple-choice questions and programming problems. It was found that the performance levels on the programming part of the exam

Deleted: higher

have stronger relationship with GPA scores than performance levels on multiple-choice questions. The strength of that relationship indicates that performance levels on programming problems have a strong predictive value.

In this study, the ability to evaluate one's state of skills and knowlegde relevant to the task at hand, as well as the ability to estimate one's performance on a just-completed task, were shown to be correlated with performance on the exam, particularly, the performance on the programming problem part of the exam. The self-evaluation ability was shown to be strongly related to student success as measured by the GPA index at the time the student took the exam, and the GPA index two years later.

For the group of students who took part in this study, the General Purpose Self-Efficacy Scale, which was used to elicit responses from the students indicating their beliefs in their competency and control of novel and demanding tasks, was not shown to be correlated with the students' success as measured by their performance on the exam or the GPA index. In general, the students provided high scores on the self-efficacy scale, but those scores were not always matched with a corresponding performance at the time of the exam, or in the future. Moreover, we found that the students who elicited higher self-efficacy scores tended to show larger decrement in their GPA index over the next two years.

Examining differences in performance between men and women, we found that men show significantly greater variability in the scores obtained on the exam, as well as greater decrements in their GPA indexes, in the two years after the exam. Women participating in this study tended to maintain their GPA levels over the two years after the exam.

Based on the analyses presented here, we wish to suggest that student performance on the computer programming tasks could be improved by applying intervention techniques that address the area that were shown to create the most difficulties, i.e., generating programming solutions. In this study, the ability to evaluate one's state of skills and knowledge before the exam, and the ability to evaluate one's performance level on just-completed tasks, was shown to be related to the actual performance level on the final exam, as well as to the general performance index as shown by the GPA scores at the time of the exam and two years later. We suggest, that in the development of classroom teaching techniques or tutorials, the methods of acquisition of semantic knowledge and the role of mental cognitive models should be considered. An additional area that should be addressed in efforts to improve student performance is the ability to evaluate one's skills and knowledge as well as one's performance on completed tasks. Recent studies in brain and human performance suggest that the brain plasticity concepts, as related to eliciting superior performance, should be considered for applications in academic settings. Recent studies in cognition suggest that, by following brain research guidelines, anyone can achieve superior performance in academic endevours. Particularly, the role of practice in the development of a high level of performance on a domain-specific tasks should be examined and utilized to maximize a student's true academic potential [6].

The analyses presented here involed 25 students, and, for the derivation of information used, for the most part, descriptive statistics. Although the strength of relationships between variables examined here suggests that the results would apply for larger populations, more studies are needed to ascertain the findings from this study.

REFERENCES

[1] Adams, W.S., "The use of APL in Teaching Programming", *Programming Teaching Techniques, W.M. Turski, (Eds.),* Elsevier , New York, 1973.

[2] Dale, N., C. Weems, M. Headington, *Programming and Problem Solving with C++*, Jones and Bartlett, Boston, 2002.

[3] Pajares, F., "Self-Efficacy Beliefs in Academic Contexts: An Outline", www.emory.edu/EDUCATION/mfp/efftalk.html, 2004.

[4] Pioro, B.T., "Estimated and Actual Performance Scores on Computer Programming Tasks", *Proceedings of IEEE SouteastCon2004*, IEEE, 2004.

[5] Rasmussen, J., "Cognitive Control and Human Error Mechanisms", *New Technology and Human Error, J. Rasmussen, K. Duncan, and J. Leplat (Eds.)*, John Wiley & Sons, New York, N.Y., 1987.

[6] Restak, R., *The new brain : how the modern age is rewiring your mind*, Rodale, 2003.

[7] Schwarzer, R., "General Perceived Self-Efficacy in 14 Cultures", userpage.fu-berlin.de/~health/selfscal.htm, 2004.

[8] Schwarzer, R., "General Perceived Self-Efficacy", userpage.fu-berlin.de/~health/selfscal.htm, 2004.

[9] Shneiderman, B., *Software Psychology: Human Factors in Computer and Information Systems*, Little, Brown and Company, Boston, 1980.

[10] Weinberg, G.M., *Psychology of Computer Programming*, Van Nostrand Reinhold Company, New York, 1971.

Deleted: a

Deleted: a stronger

Deleted: s

Deleted:

Deleted:

Deleted: .

Deleted: should not be underestimated

Deleted: ,

Deleted: *Estern XXXX*

Deleted: **1 line¶**
TABLE 1¶
Point Sizes and Type Styles¶
1 line¶
<sp> ¶
¶
1 line¶
¶
<sp>¶
1 line¶
Figure 1¶
Logo of the International Conference for Engineering Education.¶
1 line¶
Headers and Footers¶
1 line¶
Please use the following format guidelines for the Header and Footer.¶
No headers are allowed: as shown on the top of this document. ¶
Footer text: should be 10 pt Times New Roman. The only text of the footer should say the same as shown on the bottom of this document. Please copy and paste this information into your document exactly as shown on this page. ¶
1 line¶
Acknowledgement¶
1 line¶
Use the singular heading even if you have many acknowledgments. Put sponsor acknowledgments also in this section.¶
1 line¶
References¶
1 line¶