A Study of Student Performance In Combined Courses

Letha H. Etzkorn
Mary Ellen Weisskop
Computer Science Department
The University of Alabama in Huntsville
Huntsville, AL, 35899 USA
letzkorn@cs.uah.edu weisskop@cs.uah.edu

Sampson Gholston
Industrial and Systems Engineering and Engineering Management Department
The University of Alabama in Huntsville
Huntsville, AL, 35899 USA
gholston@ise.uah.edu

ABSTRACT

Here at the University of Alabama in Huntsville, there are certain graduate courses that are open to both graduate students and upper-level undergraduate students. In the Computer Science Department, due to a lack of teaching faculty, for a period of time some of these courses became required courses for both undergraduate and certain graduate students, rather than having separate required courses for each. Similarly, for the same time period, most of the elective courses available to undergraduate students were those courses which were also open to graduate students, and thus also heavily populated by graduate students. This study investigates whether undergraduate students overall suffer by being placed in courses with graduate students. Similarly, it investigates whether the graduate students suffer by being placed in courses with undergraduate students. Both required and elective courses are examined. Variations such as additional preparation in the form of an extra prerequisite for undergraduates are investigated. The impact of student quality as indicated by ACT scores and GRE scores are also taken into account. The study found that undergraduate students perform about the same in courses with graduate students as they do in courses where normally only undergraduate students are present.

Keywords: graduate, undergraduate, students, performance

1. INTRODUCTION

In the Computer Science department at the University of Alabama in Huntsville (UAH), there is an active undergraduate program and an active graduate program. The undergraduate program is a fairly traditional program, except that many students work part- or full- time and therefore there are often a number of older students in a course. The graduate program includes part time students who work full time in local industry, and also a large number of full time students, most often foreign students who are attending the university on a student visa. Many of the full time graduate students work half time as teaching assistants or research assistants.

Undergraduate courses at UAH are numbered at the 100-to-400-levels. Graduate courses are numbered at the 500-level and 600-level (700-level courses are PhD courses and are not included in this discussion). 500-level courses are open to both graduate students and advanced undergraduate students. Graduate students can take 400-level courses but they do not count toward the graduate student's degree. Graduate courses at the 600-level are not normally open to undergraduate students. Content and organization of such courses is usually different depending on level (Wankat and Oreovicz, 1993).

In the past (in the 1980s), most undergraduate courses were filled primarily with undergraduate students, and most graduate courses (500-level and 600-level courses) were filled primarily with graduate students. However, beginning approximately in the middle 1990s, some of this differentiation of courses was lost, primarily due to a lack of the faculty required to cover all these separate courses.

For example, originally undergraduates took a 2-course architecture sequence, CS309 (a course on Digital Logic), and CS413 (Computer Architecture). Some entering graduate students who had a deficiency in the architecture area, typically due either to their undergraduate university (often a foreign university) stressing different areas, or due to their having had a BS degree in an area other than Computer Science, would take a remedial course, CS513 (Introduction to Computer Architecture). CS513 is a course that combines all the material from CS309 and CS413 into the same course.

For several years in the early to mid-90s, CS413 was taught by a reliable part-time instructor. However, when this part-time instructor ceased to teach the course, there were not enough reliable instructors available to ensure that CS309, CS413, and CS513 could all three be taught on a regular basis. The initial plan of the then-chair of the CS department was to have both undergraduate and graduate students take CS513, and to no longer teach CS309 or CS413. However, the instructors of those courses thought that CS513 by itself was too accelerated a course for many of the undergraduate students. Thus, a compromise was reached: those graduate students who needed it would continue to take CS513, as before. Undergraduate students would take CS309, and then take CS513 as the second course (instead of CS413).

Similarly, there were once two separate courses on Programming Languages, CS424 (Programming Languages for undergraduates), and CS624 (Programming Languages for graduate students). These were combined into a single course, CS524, which was a required course for all undergraduate students, and which was required for all graduate students who did not have a similar programming languages course in their undergraduate curriculum. For this course, on the first day of class a programming proficiency exam was administered. Students who failed this exam would normally be expected to remedy this lack of proficiency (usually by taking a remedial programming course) before being allowed into the course.

During the same time period, many undergraduate senior-level elective courses were seldom taught. Thus, most undergraduate students would take one of the 500-level electives that were open to both undergraduate and graduate students.

We should note that having undergraduates and graduate students in the same course was done differently than in many departments at many universities. Typically, when undergraduates and graduate students are in the same course, they sign up for different section numbers, and the graduate students typically have more work to do than the undergraduate students, in terms of additional course projects or extra tests, etc. During this time period, in the 500-level classes in the Computer Science department at UAH, undergraduates and graduate students signed up for the same course number, and were treated exactly the same within the course.

The idea for this study was brought to our attention for two reasons: first, treating undergraduate students and graduate students exactly the same in a course is atypical. Also, several of our undergraduate students complained because there were not sufficient undergraduate-only senior electives taught for them; it was clear that many of them would have preferred to be in a class only with undergraduates than to attend a class with graduate students where they thought they would be at a disadvantage.

This purpose of this study is to investigate whether undergraduate students overall suffer by being placed in courses with graduate students. Similarly, it investigates whether the graduate students suffer by being placed in courses with undergraduate students. The performance of undergraduates as compared to graduate students in the required courses, CS513 and CS524, is examined. The performance of undergraduate students as compared to graduate students in several 500-level elective courses, CS551, CS552, CS553, CS545, and CS590, are examined. Undergraduate student performance in 500-level courses is compared to their performance in CS317 and CS490, which are two required undergraduate courses. Graduate student performance in 500-level courses is compared to their performance in CS617, which is a required graduate course. The impact of student quality as indicated by ACT scores and GRE scores on the outcomes of these studies is evaluated. This study is performed over the time during which CS513 and CS524 were taught as described, approximately over the years 1999 through 2001; data for some classes is also included from 1997, 1998, and 2002.

Studies have been performed in the past measuring how various aspects of student learning affect their performance (Kell and Van Deusen, 2002; Bjornsen, 2000), however, there have been few previous studies of the performance of undergraduate and graduate students in the same courses. Kirlin and Hedstrom (Kirlin and Hedstrom, 1993) examined a random processes course that was taught to both undergraduate and graduate students; however, the focus of this paper was on the content of the random processes course rather than undergraduate versus graduate student performance. Similarly, Carroll and Kane (Carroll and Kane, 1976) discussed a computer graphics course that was taught to both graduate and undergraduate students; the emphasis in this paper was also on the content of the course rather than student performance. Gonzales (Gonzales, 1988) similarly discussed a microcomputer course that was taught to both undergraduates and graduate students; the course itself was the focus rather than student performance, although they did briefly discuss that undergraduate students had a different workload in the course than did graduate students.

Carroll et al. (Carroll et al., 1976) discussed the computer engineering curriculum at Auburn University back in the early 70s. They mention several courses open to both undergraduate and graduate students, but student performance in such courses is not discussed. Bernhard (Bernhard, 1999) discusses employing both undergraduate students and graduate students in a three student research team consisting of two undergraduates and one graduate student; the team was successful in completing its project, and the primary difference between undergraduate students and graduate students was that they had different types of school-related deadlines (master's thesis deadlines versus senior project deadlines) that affected the project.

Pendse and Johnson (Pendse and Johnson, 1996) discuss teaching methods for undergraduate courses, and compares them to teaching methods for graduate courses. They conclude that good teaching techniques will work in any class, graduate or undergraduate.

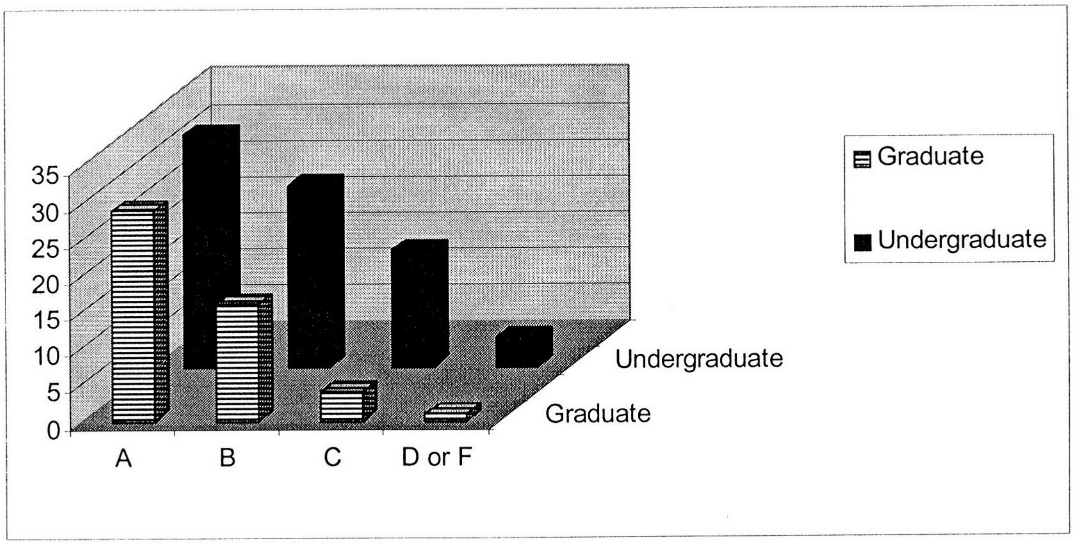


Figure 1. Grade comparison for CS513 for graduate and undergraduate students

Classification	D or F	C	B	A	Total
Graduate	1	4	16	29	50
Undergraduate	4	16	25	32	77
Total	5	20	41	61	127

Table 1. Contingency table for CS513 for graduate and undergraduate students.

2. STUDIES

The first study we performed compared the performance of undergraduates and graduate students in CS513. The second study compared the performance of undergraduates and graduate students in CS524.

Figure 1 shows the grade distribution for graduate and undergraduate students in CS513. The grades are on a 4 point scale where an A = 4, B = 3, C = 2, D = 1, and F = 0. Table 1 was used to test the hypothesis that the grades are different for graduate and undergraduate students in CS513. The Chi-squared test was used to test this hypothesis. A p-value of 0.1306 was calculated, thus the null hypothesis above cannot be rejected.

The Chi-squared distribution is one of several statistical distributions based on degrees of freedom. The degrees of freedom are the number of values that are free to vary after a sample statistic has been computed. The Chi-squared statistic, which is used in the Chi-squared test, is a random variable based on the Chi-squared distribution. The p-value is the actual area under the distribution curve that represents the probability of a particular sample mean occurring if the null hypothesis is true (Lapin, 1990). Therefore there is no significant difference in the graduate and undergraduate student grades in CS513.

Figure 2 shows the grade distribution for graduate and undergraduate students in CS524. The hypothesis that the grades are different for graduate and undergraduate students in CS524 was tested employing the information in Table 2 using the Chi-squared test. A p-value < 0.001 was calculated, indicating that there is a significant difference in the grades of graduate and undergraduate students in CS524.

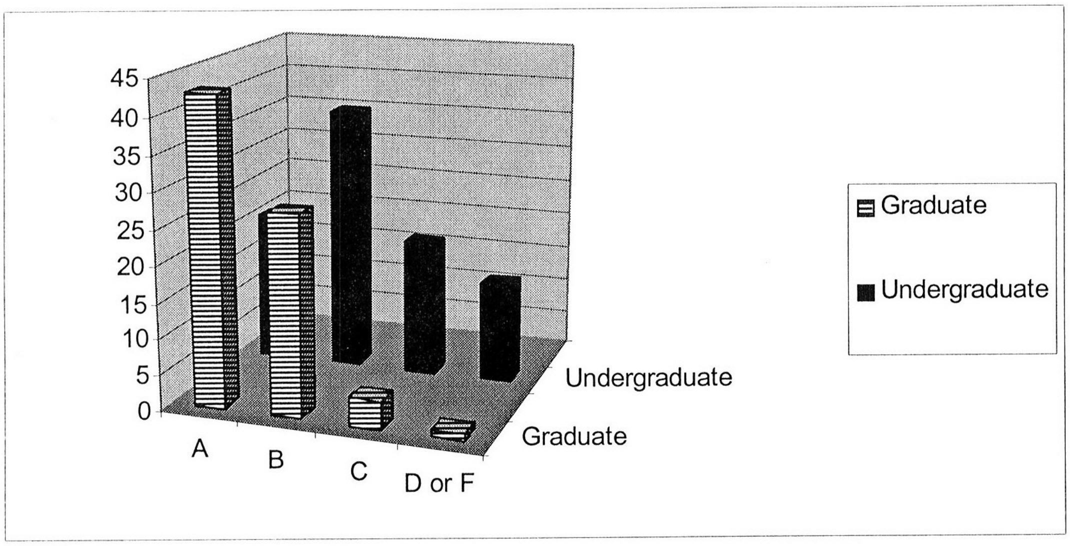


Figure 2. Grade comparison for CS524 for graduate and undergraduate students.

Classification	D or F	C	B	A	Total
Graduate	1	4	28	43	76
Undergraduate	14	19	37	21	91
Total	15	23	65	64	167

Table 2. Contingency table for CS524 for graduate and undergraduate students.

As you can see, undergraduates perform better compared to graduate students in CS513 than they do in CS524. Our assumption is that this is because of the strong prerequisite required of undergraduates in CS513. Before we make this assumption, however, it is necessary to check that the quality of students in each course is approximately the same; for example, if the undergraduate students in CS524 were poorer students in general than those undergraduate students in CS513, then the assumption that the additional prerequisite was the cause for undergraduate students performing better in CS513 is not necessarily a good assumption. Thus, we tested to see whether the students in CS513 and CS524 were different quality students.

To ensure that the quality of students entering each class was similar, a t-test was performed using GRE scores, ACT scores, and SAT scores. The results in Table 3 indicate in each case that there is no significant difference in the test scores for students entering CS513 and CS524. Thus, the results are not biased by the quality of student entering each class. So our assumption that the additional prerequisite required for undergraduate students in CS513 is the cause of their better performance is still a good assumption.

In the next study, we did an analysis of undergraduate performance in undergraduate-only courses, compared to undergraduate performance in all the 500-level classes for which we have data. Here, we found that undergraduates perform about the same in 500-level classes, including both graduate and undergraduate students, as they do in classes by themselves (undergraduate students only). The p - value is 0.34 which indicates no significant difference in the undergraduate grades for 500-level classes and classes below the 500-level.

We then performed the same study for graduate students; that is, we analyzed their performance in a graduate-student-only course, CS617, and compared it to their performance in 500-level classes, which include both graduate and undergraduate students. The p-value for the data in Table 5 is 0.0009, which indicates that the grades are significantly different for graduate student in 500 level courses and graduate students in a 600 level course. In fact, more as were awarded in the 500-level courses.

GRE	t = 0.919	p - value = 0.36
SAT	t = 0.36	p - value = 0.70
ACT	t = -0.96	p – value = 0.34

Table 3. T-tests of tests scores of students entering CS513 and CS524.

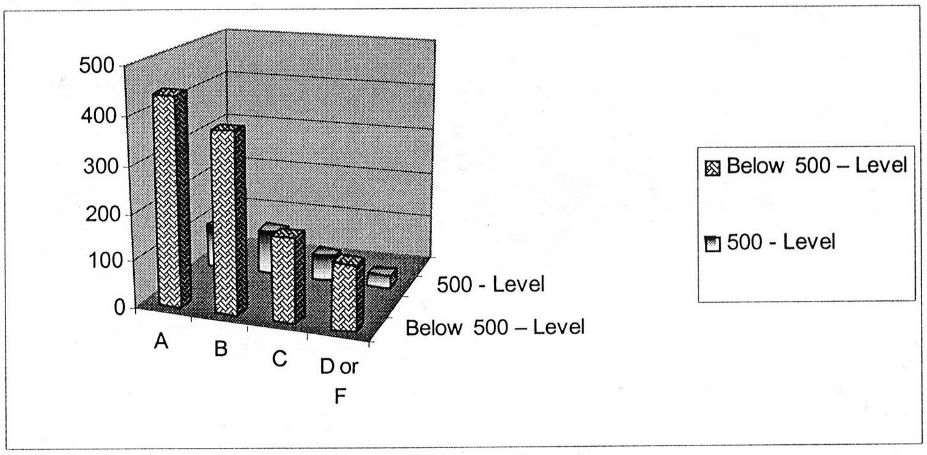


Figure 3. Comparison of undergraduate performance in 500 and below 500-level courses.

Classification	D or F	C	B	A	Total
Below 500 – Level courses	133	175	378	440	1126
500 - Level courses	27	55	97	97	276
Total	160	230	475	537	1402

Table 4. Comparison of undergraduate performance in 500 and below 500-level courses.

This is reasonable, since we would expect that courses at the 600-level (true graduate courses including only graduate students) would be more rigorous than lower level courses. Also, the 500-level courses tend to be more application-related and less theory-related than higher level graduate courses, which would tend to make the 500-level courses easier courses, resulting in higher grades. For this reason, a master's student in Computer Science is allowed to have no more than 50% of his or her coursework at the 500-level.

Table 6 shows the grade distributions for all the classes examined. The grade distribution for the classes is significantly different (p-value < 0.001).

3. CONCLUSION

In summary, we found that in CS524, graduate students overall perform much better than undergraduate students. In CS513, where undergraduates had an additional strong prerequisite, the performance of graduate students and undergraduate students was equivalent.

Overall, however, undergraduate students perform about the same in 500-level courses as they do in courses (CS317 and CS490) where normally only undergraduates are present. Graduate students, however, perform better overall in 500-level courses than they do in CS617, where only graduate students are present. We note that CS617 is normally considered a "tough course," but then CS513, CS524, as well as CS317 and CS490 are also normally considered "tough courses." CS617 is a course is a course on the Design and Analysis of Algorithms. As a highly mathematical course, many of our students find it difficult. The titles of each of these courses are provided in Table 6.

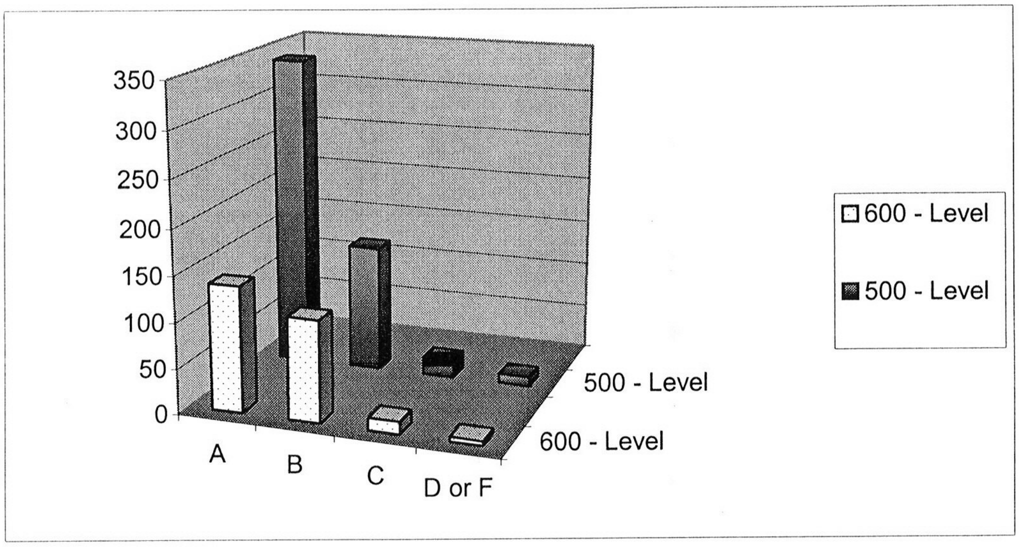


Figure 4. Comparison of graduate performance in 500 and 600-level courses.

Classification	D or F	C	B	A	Total
600 - Level courses	5	14	111	138	268
500 - Level courses	10	19	139	341	509
	15	33	250	479	777

Table 5. Comparison of graduate performance in 500 and 600-level courses.

Class	D or F	C	B	A	Total
CS 390 **Unix Programming**	14	12	58	106	190
CS 317 Intro. to Design and Analysis of Algorithms	9	115	196	217	619
CS 490 Intro. to Operating Systems	27	48	124	117	316
CS 513 Intro. to Computer Architecture	5	19	42	60	126
CS 524 Programming Languages	15	24	64	64	167
CS 545 Intro. to Computer Graphics	14	28	68	113	223
CS 551 Object-Oriented Software Modeling	3	1	33	87	124
CS 553 Client/Server Computing	0	0	9	46	55
CS 590 Programming Environments with Unix	0	4	20	63	87
CS 617 Design and Analysis of Algorithms	5	14	111	138	268
Total	174	265	725	1011	2175

Table 6. Grade distribution for all classes.

The question arises as to whether the graduate students perform better than undergraduate students in the 500-level courses due to quality of student (presumably most graduate students, when undergraduates, were in the higher categories of undergraduate students based on GPA and ACT/SAT scores), or due to having more background in those areas than the undergraduate students.

Based on this study, we find that having undergraduate students and graduate students in the same course, with exactly the same requirements, is acceptable. Neither undergraduate nor graduate students suffer from this situation. However, we note that many undergraduate students seem to have the perception that they are at a disadvantage in these courses compared to undergraduate-only courses, even though our data shows that they are not.

We note that, in line with UAH university policy, CS513 and CS524 have recently been split back into separate graduate and undergraduate courses, so the situation we studied is no longer present.

4. REFERENCES

Bernhard, J.T. (1999), "Undergraduate and Graduate Students in a Research/Design Team: Strategies for Success," Proceedings of the Frontiers in Education Conference, Nov. 10-13, pp. 13B9.

Bjornsen, C.J. (2000), "Undergraduate Student Perceptions of the Impact of Faculty Activities on Education," Teaching of Psychology, Vol. 27, No. 3, Summer, pp. 205-208.

Carroll, B.D., and G.R. Kane (1976), "A Graduate/Undergraduate Graphics Course and Laboratory," Proceedings of the 14th Annual Southeast Regional ACM Conference, April, pp. 347-348.

Carroll, B.D., H.T. Nagle, G.R. Kane., and J.D. Irwin. (1976), "Computer Engineering at Auburn University," Proceedings of the 14th Annual Southeast Regional ACM Conference, April, pp. 312-318.

Gonzales, R.L. (1988), "A Microcomputer Course for Sophomore and Graduate Students," Proceedings of the Frontiers in Education Conference, October 22-25, pp. 462-465.

Kell, C., R. Van Deusen (2002), "Student Learning Preferences Reflect Curricular Change," Medical Teacher; Vol. 24, No.1, Jan.uary, pp. 32-40.

Kirlin, R.L., and B.A. Hedstrom (1993)., "A Computer-Oriented Random Processes Course for a Mix of Graduates and Undergraduates," Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, April 27-30, pp. 24-27.

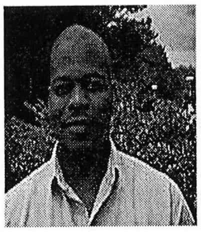
Lapin, L. (1990), Probability and Statistics for Modern Engineering, PWS-KENT Publishing Company, Belmont, CA.

Pendse, R., and E. Johnson (1996), "Teaching an Undergraduate Class Vs. Graduate Class: Is There a Difference?" Proceedings of the Frontiers in Education Conference, November 6-9, pp. 59-62.

Wankat, P. C. and F.S. Oreovicz (1993), Teaching Engineering, McGraw Hill, Inc. New York.

AUTHOR BIOGRAPHIES

Letha H. Etzkorn is an Assistant Professor at the University of Alabama in Huntsville. Her research interests include software engineering (especially metrics and program understanding), and mobile and intelligent agents. She has a BSEE and an MSEE from the Georgia Institute of Technology, and a PhD in Computer Science from the University of Alabama in Huntsville



Sampson Gholston is an Assistant Professor at the University of Alabama in Huntsville. His research interests include quality engineering, design for quality, supplier management, quality management, and applied statistics. He has a Masters from the University of Alabama in Industrial Engineering, and a PhD from the University of Alabama in Huntsville in Industrial and Systems Engineering.

Mary Ellen Weisskopf is an Assistant Professor at the University of Alabama in Huntsville. Her research interests include operating systems and distributed computing. She has a BS in Mathematics from Vanderbilt University, and an MS and PhD in Computer Science from the University of Alabama in Huntsville.