Establishing the Impact of a Computer Science/Mathematics Anti-Symbiotic Stereotype in CS Students

James A. Jerkins †
jajerkins@una.edu

Cynthia L. Stenger ‡
clstenger@una.edu

Jessica Stovall ‡
jstovall@una.edu

Janet T. Jenkins †
jltruitt@una.edu

† Department of Computer Science and Information Systems
‡ Department of Mathematics
University of North Alabama
One Harrison Plaza
Florence, Alabama 35632

1. ABSTRACT

In this paper we explore CS students' perception of the relationship between computer science and mathematics. It is widely accepted by educators and researchers that the study of mathematics is an essential part of the CS curriculum. We were surprised to discover that many of our CS students believed that no relationship or dependence exists between computer science and mathematics. We establish that many of our CS undergraduates do not believe that mathematical reasoning and problem solving skills are necessary for success in computer science. Furthermore, we will show that this *anti-symbiotic stereotype* is harmful to student's success in computer science.

2. INTRODUCTION

As colleagues in a regional university's department of mathematics and computer science, we designed an instructional treatment to push students to develop a mental framework for abstraction and generalization. Our interdisciplinary approach was designed to build students' mathematical reasoning skills through strategically placed computer programming exercises [5]. Our initial professional development workshop with high school math teachers was followed by pilots with our college-level students. During these sessions with undergraduate STEM majors, we discovered a foundational flaw in the students' understanding of the symbiotic relationship between mathematics and CS. Our pre-treatment assessments revealed that our students were under the influence of strong stereotypical beliefs. Some of these were consistent with existing personality stereotypes in the literature, particularly those concerning women in CS [4]. But overarching the male-female roles, we observed another adverse assumption that challenged the fundamental premise of our collaborative work. We found that students did not share our conviction that there is a strong symbiotic tie between computer science and mathematics. Moreover, they saw their math and CS courses as fundamentally unrelated. Many CS students felt that they did not need mathematical thinking or problem solving in order to do well in programming. They viewed mathematics as a purely computational endeavor. This attitude was summarized in the following comment from a CS major:

"Math majors probably know how to work the equation... do the equation, while CS majors can put it in the computer and do it a heck of a lot easier."

Thus, even though it is a widely held belief among CS educators and researchers that a symbiotic relationship exists between mathematics and the computing sciences, we found our students did not embrace this belief. In fact, their perceptions of the relationship would be better described as antagonistic, incompatible or anti-symbiotic. We were surprised to find our CS students harboring this negative stereotype. In this paper we establish that the *anti-symbiotic stereotype* exists and is harmful to our computer science students.

The remainder of the paper is organized as follows. In Section 3 we review the literature on the relationship between the disciplines. We describe our methodology in Section 4. Findings and results are presented in Section 5. Sections 6 and 7 give our conclusion and plans for future work.

3. BACKGROUND

A discrete structures course is listed as *foundational material* in the 2008 ACM Computer Science Curriculum [1]. The discussion of the role of calculus and statistics in the curriculum has been debated since the early 1970s [7]. According to Ronald Douglas the study of Calculus "is the key to understanding systems that change in the social, the biological, or the physical sciences" [2]. In addition, he stated that "Computer Scientists will need to learn Calculus if they are to understand many of the most exciting applications of large scale computing" [2].

CCSC-MS April 5-6, 2013, Fort Smith, AR, USA

It is a well established fact that educators and researchers in CS believe that the study of mathematics is an essential part of the CS curriculum. Influential CS practitioners and educators have stated that mathematics is the root of algorithm development and where students learn the essential skill of abstraction [3]. Bill Marion says "the mathematical viewpoint can help deepen the students' understanding of abstraction as it relates to computational objects" [6]. Tony Ralston notes that "the crucial but unprovable reason for teaching any mathematics at any level of education is its effect on the mind" [8].

Clearly, educators and researchers in CS agree that a symbiotic relationship does exist between advanced mathematical reasoning and computer science. A 2001 study concluded that mathematics background was a predictor of success in an introductory computer science course [9]. However, our literature review indicated no studies have investigated the students' perception of the relationship between mathematics and computer science and the effect of that perception.

4. METHODOLOGY

We designed an experiment to uncover student perceptions of this relationship and to investigate the effects. We surveyed students in computing classes, conducted videotaped interviews, and collected academic records.

4.1 Participant description

Our Phase I participants included 46 students at a regional four year liberal arts public university from our CS1, CS2, and Artificial Intelligence courses. The number of student participants in each course were 16, 12, and 14 respectively.

The Phase II participants were composed exclusively of CS majors enrolled in the AI course. These students were in their final three semesters of undergraduate work. The participants included one female and thirteen male students. Four of the males were international students. The overall GPAs ranged from 1.79 to 3.93 with an average of 3.04. The CS GPAs ranged from 0.5 to 3.8, with an average of 2.88. The math GPAs for courses required in the CS major ranged from 0 to 4.0, with an average of 2.49.

4.2 Data collection

We administered a survey in Phase I to gather baseline information from a broad range of students in our CS1, CS2, and Artificial Intelligence courses. The survey elicited student comments on the relationship between CS and mathematics. We asked exploratory questions such as "Do you enjoy mathematics?", "Are you good at mathematics?", "Do you believe mathematics and computer science share common concepts?", and "Why does our CS degree require mathematics courses?". Students were also asked to indicate which math classes they had completed and how each class had helped with their computer science coursework. The survey was administered by the course instructor and the data was compiled by the authors.

In Phase II, we interviewed CS majors enrolled in a 400-level computing course. We recorded their opinions concerning mathematics in general and the specific mathematics they encountered in a recent programming assignment requiring a genetic search algorithm. Interviews were conducted in a classroom setting by two members of the mathematics faculty with the instructor of record not present.

Question	Yes Response
"Do you enjoy mathematics?"	75.9%
"Are you good at mathematics?"	61.25%
"Do you believe mathematics and computer science share common concepts?"	92.86%

Table 1: Summary of initial survey results, questions 1-3

Rater	ρ
A	-0.7472
B	-0.7344

Table 4: Correlation Coefficient - Anti-symbiotic stereotype vs. success in CS

The interviews were videotaped and then transcribed independently by two of the authors. Transcriptions were compared and where discrepancies occurred the original video tape was referenced to resolve the discrepancies.

Finally, we obtained the participants academic transcripts. Each student's overall GPA, CS GPA, math GPA, and ACT scores were recorded for analysis.

5. RESULTS

The results from the survey data were initially confounding. The majority of the students reported they enjoyed math, were good at math, and believed that CS and math shared common concepts. Table 1 describes these results.

Based on the overwhelmingly positive response to the question about common concepts, we asked students to explain how their mathematics courses helped them in CS. We expected students to describe the connections between the disciplines in terms of reasoning, logic, and precision. Their positive reactions turned out to be a shallow, purely computational conception of mathematics. Table 2 shows typical student comments that demonstrate their superficial understanding of why studying math is beneficial for CS majors. In fact, 63.45% of the survey responses indicated that the students believed their math courses had not helped them in CS. They failed to recognize the common core between the disciplines, advanced reasoning and problem solving.

Moreover, when students attempted to explain the common concepts, negative comments began to surface. This antagonistic attitude was also evident in their responses to the question "Why does your computer science degree require mathematics courses?". We found that our students did not share our conviction that there is a strong symbiotic tie between computer science and mathematics. Their perceptions would be better described as antagonistic, incompatible, or anti-symbiotic. We define this *anti-symbiotic stereotype* as the belief that mathematical reasoning and problem solving skills are not necessary for success in computer science. Table 3 is a representative sample of anti-symbiotic comments observed in our survey results.

Clearly, many of our students harbor the anti-symbiotic stereotype. They believe that mathematical reasoning and problem solving skills are not necessary for success in computer science. In Phase II interviews we focused our attention on uncovering evidence of the harmful effects of the

• "'Everything is about calculation and algorithm." • "'[Algebra] is working with variables to find the unknown." • "'When your [sic] writing programs you have to use numbers and if you don't know how math works then you'll never be able to use it in your program." • "Understanding of area calculation." • "'You use equations."

Table 2: Student perceptions of how math helps in CS courses

• "I don't really feel like I'm using anything from my math courses that I really need that I couldn't look up online and reuse." • "After four years of classes in the CS field I have yet to have any teacher prove a relationship between math and computer science." • "I haven't needed any math, except discrete. Other classes seem unnecessary." • "Because without the extra 30 hours of math courses CS majors would be short hours to graduate. PS, I do believe math courses go to [sic] far and to [sic] deep into mathematics." • "I do not understand, however why such difficult math classes (the higher calculuses, for example) are required for a CS major." • "I think **they** think math is important to CS" (emphasis added) • "Because it's just the standard for all colleges and universities?"

Table 3: Evidence of an anti-symbiotic stereotype among computing students

anti-symbiotic stereotype. We analyzed video transcriptions and survey data from the CS majors. Based on the frequency and strength of anti-symbiotic responses, we assigned an anti-symbiotic score to each student. The scores ranged in value from one to five, no evidence to strong evidence respectively. Each student's data was judged independently by two of the authors. The rankings were tabulated and analyzed for inter-observer reliability ($\kappa = 0.7219$).

We examined the relationship between each student's anti-symbiotic score and CS GPA. Each rater computed a parametric measure of correlation between the intensity of the student's anti-symbiotic attitude and CS GPA. Table 4 shows the correlation coefficient computed by each rater. Our research revealed that the strength of the student's anti-symbiotic rating is negatively correlated to their success in CS as measured by the computer science GPA. Figure 1 shows that as the strength of the student's anti-symbiotic attitude increases their success in CS decreases. This correspondence demonstrates that a strong negative relationship exists between the anti-symbiotic belief and success in computer science.

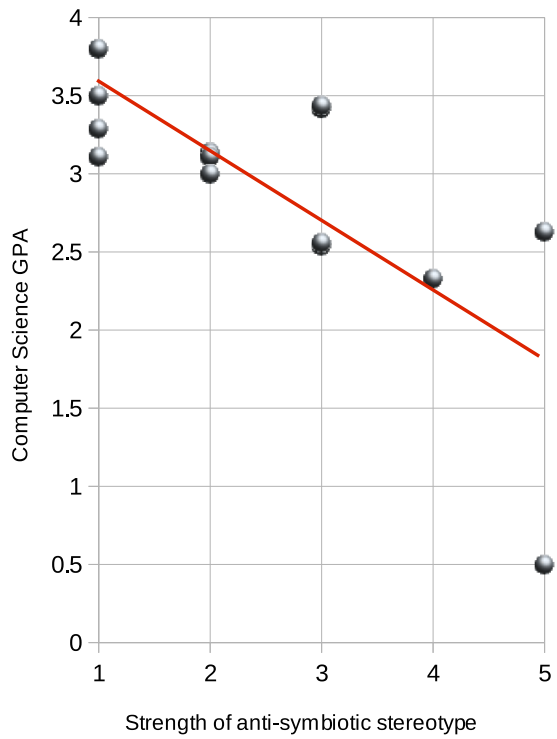

Figure 1: The anti-symbiotic stereotype is harmful

6. CONCLUSION

In our research, we found that many of our students did not believe advanced mathematics coursework was beneficial for computer scientists. In addition, we discovered students perceived the required mathematics coursework as simply a hoop to jump through to complete a CS degree. A significant number of students viewed their math courses as a catalog of procedures for calculation. The notion that mathematics is the language of problem solving and algorithmic thinking was rarely indicated in our data.

We have defined the anti-symbiotic stereotype as the belief that mathematical reasoning and problem solving skills are not necessary for success in computer science. Our results indicate that the anti-symbiotic stereotype exists and the degree to which a student possesses this belief is negatively correlated to student achievement in computer science.

7. FUTURE WORK

We plan to investigate if our instructional treatment is one way to defuse the anti-symbiotic stereotype among CS majors. Our design utilizes strategically designed computer experiences to solve mathematical problems [5]. We believe this approach demonstrates to students that a symbiotic relationship exists between mathematical reasoning and computer science.

8. REFERENCES

[1] Association for Computing Machinery. Computer Science 2008, An Interim Revision of CS2001. http://www.acm.org/education/curricula/ComputerScience2008.pdf, 2008.

[2] Douglas, R., and Douglas, R. The importance of calculus in core mathematics. *Toward a lean and lively calculus* (1985), 3–5.

[3] Henderson, P. B., Hu, C., Talburt, J., and Tomer, D. S. Panel discussion: mathematical thinking in computer science. In *Proceedings of the 2nd annual conference on Mid-south college computing* (Little Rock, Arkansas, United States, 2004), MSCCC '04, Mid-South College Computing Conference, pp. 210–212.

[4] Jenkins, J., Stenger, C., Jerkins, J. A., Stovall, J., Davis, J., and Lloyd, E. Level the Playing Field for Women with Immediate Immersion into Beginning Programming in a Problem Based Learning Environment. In *Tennessee Women in Computing* (October 2011).

[5] Jenkins, J. T., Jerkins, J. A., and Stenger, C. L. A Plan for Immediate Immersion of Computational Thinking into the High School Math Classroom through a Partnership with the Alabama Math, Science, and Technology Initiative. In *Proceedings of the 50th Annual Southeast Regional Conference* (New York, NY, USA, March 2012), ACM-SE '12, ACM, pp. 148–152.

[6] Marion, B. Discrete mathematics: support of and preparation for the study of computer science. *J. Comput. Sci. Coll. 16*, 1 (Oct. 2000), 190–199.

[7] Power, J., Whelan, T., and Bergin, S. Teaching discrete structures: a systematic review of the literature. In *Proceedings of the 42nd ACM technical symposium on Computer science education* (2011), ACM, pp. 275–280.

[8] Ralston, A. Do we need any mathematics in computer science curricula? *SIGCSE Bull. 37*, 2 (June 2005), 6–9.

[9] Wilson, B. C., and Shrock, S. Contributing to success in an introductory computer science course: a study of twelve factors. In *Proceedings of the thirty-second SIGCSE technical symposium on Computer Science Education* (New York, NY, USA, 2001), SIGCSE '01, ACM, pp. 184–188.