ISSN: 2473-3857
v3 n4334

Be Careful What You Wish For: The Continued Limiting of Computing Majors

John H. Reynolds
john.reynolds@gvsu.edu

Roger C. Ferguson
roger.ferguson@gvsu.edu

School of CIS
Grand Valley State University
Allendale, Michigan 49401, USA

Diana Linville
dianar@nwmissiouri.edu
School of CSIS
Northwest Missouri State University
Maryville, Missouri 64468, USA

Abstract

At most institutions, the number of students attempting to obtain a computing degree has been increasing for the past several years. Student involvement in pre-college courses in computing has also increased. This paper examines the following: With this large pool of students attempting to obtain a computing degree, what type of student is successfully completing the degree? More specifically, this paper examines who successfully completes an introductory to computing course (e.g., Computer Science 1, Introduction to computing), because these courses are the gateway courses to graduation. The results of the research show that student success in these courses are based on two constraints: having a prior-programming experience, and/or having a very high level of academic skills. These constraints cause a narrowing of the number of students that are graduating; the consequences of this narrowing are the field of computing will (1) continue to struggle with attracting students from all backgrounds, (2) typically eliminate students from the pool of students if they switch to computing too late in the college career, and (3) cause some colleges to not be prepared for enrollment issues when student interest in computing decreases in the future. The study in this paper shows the result of these constraints causes the set of graduating computing students to be homogeneous.

Keywords: Computing, Computing Majors, Computer Science 1, Introductory Courses, Success Rates

1. INTRODUCTION

Enrollment changes in all universities has always fluctuated, and the field of computing is no exception. The number of computing majors at Grand Valley State University (GVSU), Northwest Missouri State University (NWMSU), and at most other institutions, has been increasing for the past several years (Zweben & Bizot, 2015). Student involvement in pre-college courses and/or events, such as, first robotics, Science Olympiad computing events, high school programming courses and other similar activities has also increased in student participation. Even

ISSN: 2473-3857
v3 n4334

social consciousness express through media (e.g., Facebook, online newspapers, etc.) has changed from the lack of support for computing careers, to a positive atmosphere for a career in computing (Heitin, 2014). Because of this interest in Computing, there are larger class sizes, number of sections offered has increased, and a potential desire to offer a new degree in Computing to increase revenue, etc. In other words, times are good to be in a department of computing.

With this increase in student population, this paper examines the type of computing graduates at GVSU and NWMSU, that is, who successfully completes a computing degree. To answer this question, data was collected from two unique institutions on who successfully completes a majors course in introductory to computing (e.g., Computer Science 1, Introduction to Computing, etc.). These courses (hereafter referred to as CS 1) are the gateway courses for most computing majors, whether the major is Computer Science (CS), Information Systems (IS), Information Technology (IT), etc. Students must succeed at this course if they are to get a degree in computing (Werth, 1986). The results of recent research show that student success in these courses is based on having some form of prior programming experience, such as a high school computing course (Reynolds, Adams, Ferguson, & Leidig, 2017).

In other words, a student that has had no prior experience in computing (such as, a high school programming class, a college pre-computing class, etc.) will typically not complete a degree in computing. This background constraint causes a narrowing of the pool of students currently graduating with a degree in Computing.

This limiting constraint of a pre-requisite background has significant consequences for institutions that offer computing degrees:

(1) The institutions are missing out on a large untapped pool of under-prepared students. The field of computing has traditionally struggled with attracting students who are not already interested in the field of computing as shown by the 57,937 or 2.2% of all students taking AP tests (College Board, 2016). In its first year, about 45,000 (1.7%) students took the new CS Principles course (Advances in AP, 2017). Added together, this represents nearly a 3-fold increase since 2013 in the percent of students interested in Computing, as reflected by those taking one of these AP tests (College Board, 2013). While these numbers have improved substantially due to the introduction of the CS Principles Course, the numbers are still low overall and remain very low within many subgroups.

(2) The pool of students that discover, or take an interest in computing, after starting college are typically locked out of the field. After learning about the opportunities of a computing career through social media or other informational sources, these students are eliminated because their lack of prior preparation leads to a lack of success in CS 1.

(3) Narrowing who succeeds in CS 1 courses may foster an indifference to inevitable enrollment fluctuations. Departments, schools, and colleges are potentially not addressing when student interest in computing decreases. Because of today's inflated enrollments, departments in the future may have professors willing to teach 300 and 400 level courses without any students in their classroom.

2. BACKGROUND

Previous research has illustrated that prior experience plays a substantial role in the success of a student within an introductory programming course (Alvarado, Lee, & Gillespie, 2014; Konvalina, Wileman, & Stephens, 1983; Rountree, Rountree, & Robbins, 2002). In addition, academic achievement in high school, including mathematical knowledge, has been shown to be a factor in the success of a student within a STEM field (Alvarado et al., 2014; Konvalina et al., 1983, Campbell & McCabe, 1984). These factors indicate the type of student who has the highest probability of success in a CS1 course.

Alvarado et al. (2014) looked at how the traditional factors of prior experience and confidence still influence success for students in regards to the new pedagogies that have emerged in CS. Their study included 3 sections of a CS1 course where students were surveyed about their prior CS experience and confidence in doing well in the course. They found that prior experience and confidence are still factors in the success of students in a CS1 course. Another study showed that students who claimed to have knowledge of a programming language had a higher success rate in a CS1 course but that prior knowledge did not guarantee success (Rountree et al., 2002). Wilson and Shrock (2001) supported this theory that prior knowledge does not guarantee success with their research. They

ISSN: 2473-3857
v3 n4334

concluded that comfort level within an introductory CS course was the best indicator.

Other studies have looked at how students and university faculty in STEM fields perceive the importance of prior experience. While students were found to think that prior experience is a factor for success in a CS1 course (Tafliovich, Campbell, & Peterson, 2013), faculty did not rank prior knowledge as an important factor for success in STEM courses (Gandhi-Lee, Skaza, Marti, Shrader, & Orgill, 2015). Gandhi-Lee's (2015) study did indicate though that faculty felt that the key factor for success in STEM fields was mathematical knowledge and implied that students should take more math in high school. This leads to students gaining more experience in the area and becoming more comfortable with the content. This lends support to the perception that prior experience in high school to programming will in turn allow students to not only be exposed to the content, but to also become more comfortable with it, therefore resulting in a higher probability for success in a CS1 course.

Student achievement within high school has emerged as another indicator that can determine student's success. In the research completed by Konvalina et al. (1983), high school performance and mathematics background were found to be influences in students who withdrew from a CS1 course and those that did not. Students who had significantly higher SAT math scores and completed more semesters of math in high school persisted in the CS major (Campbell & McCabe, 1984) and taking advanced science and mathematics courses increased the student's interest in a STEM field as well (Sadler, Sonnert, Harzan, &Thi, 2014).

Wang and Degol (2013) completed a literature review that looked at interest value and self-concept ability when choosing a STEM career. Student's interest in math and science is associated with the number of math and science courses taken in high school. Lewis, Yashuhara, & Anderson (2011) conducted a qualitative study in regards to student's perceptions that influence their attraction to a computing-related major. They conducted interviews at two large public universities of thirty-one CS1 and CS2 students. They found that ability in regards to previous experience was one factor that students consider when deciding to major in CS.

When correlating the literature provided, a pattern emerges as to the academic background that a student possesses to succeed within a CS1 course.

This paper examines both the high school academic credentials and any prior college courses of students in different majors that complete CS1 successfully. As Campbell and McCabe (1984) found in their research, 75% of students who continued within the CS major their sophomore year completed the CS degree. Indicating that the first year for a CS student is critical to completion of the program; therefore, this study will be looking at data of the CS1 course that students typically take within their first year of college.

3. THE PROBLEM

Research has shown that there is a correlation between prior preparation and success in the first computing course. NWMSU is a university with both college students and Academy students within the CS 1 course. Academy students are high school students who live on campus that are obtaining an associate's degree along with their high school diploma. To be eligible to attend, high school students must have a cumulative GPA of 3.5 or higher, be ranked in the top 10% of their class, and have minimum ACT scores of 24 math and 23 composite.

The study in this paper first surveyed all students in a one semester CS 1 course using the methodology from a previous study (Reynolds, 2017). Specifically, proportions of successful (B or Above) Academy students vs college students were evaluated on prior preparation attributes (i.e., Self-taught, High School experience).

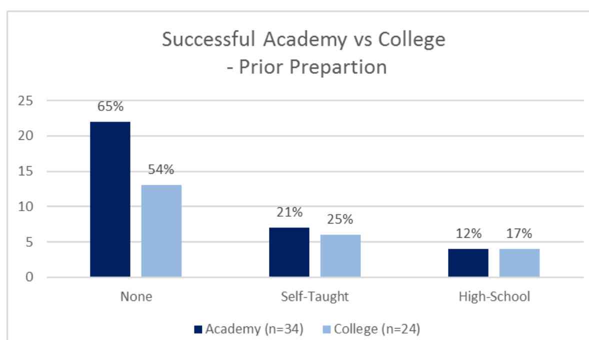


Chart 1.0 – Academy vs College at NWMSU

The results showed no significant difference for any category between Academy students and their college peers. These results raised an interesting question: Is prior preparation the only significant variable? In addition to whether students took a previous programming course,

ISSN: 2473-3857
v3 n4334

this paper looks at high school academic standing, as evidenced by GPA, Rank, and ACT scores, to try to identify attributes of those who succeed.
The methodology of this study is additionally to evaluate all students at GVSU who have taken the first computing course for computing majors over a two-year period (F15-W17). The following attributes were individually compared with student success: repetition of this course, taking a previous college programming course, declared major, High School GPA and Class Rank, ACT Composite scores and individual ACT section scores in English, Math, Reading, and Science. For the purposes of this study, success is defined using final course grades where a B- or above is considered successful and the sample mean of each academic attribute is the cutoff for above average.

The de-identified data was provided by the Office of Institutional Analysis. As is often the case with large datasets, not all students had all of the attributes recorded. 860 students were in the dataset, but only 722 had ACT scores (84%), therefore the sample size was reduced.

To maintain consistency with the previous study (Reynolds, 2017), the students were again divided into three groups (Computer Science majors, Information Systems majors, and non-Computing majors) with each group individually being compared to Computer Science majors. The following major hypotheses reflect this grouping and for each hypothesis, there are individual tests evaluating each of the attributes listed above.

1) Are CS majors more successful than IS majors or non-Computing majors in the first programming course?

> *H_1: The percentage of CS majors with a grade of B- or higher in the first programming course will be different when compared to IS majors.*

> *H_2: The percentage of CS majors with a grade of B- or higher in the first programming course will be different compared to non-Computing majors.*

Some may argue that successful CS majors have higher grades and ACT scores, thus the following hypotheses:

2) Do successful CS majors in the first programming course have better academic credentials than successful IS majors or non-Computing majors?

> *H_3: The percentage of CS majors will be different, regardless of High School academic credentials, when compared to IS majors.*

> *H_4: The percentage of CS majors will be different, regardless of High School academic credentials, when compared to non-Computing majors*

Some might suggest that all students who succeed in the first programming course have a significantly higher level of academic preparation:

3) Do successful students in the first programming course, regardless of major, have better academic credentials?

> *H_5: Based on prior preparation, the percentage of students with a grade of B- or higher in the first programming course will be different from those with a grade below B-.*

Two final categories of students that are often overlooked as individual groups in these types of studies are those who withdraw from a class and those who transfer from another institution:

4) Do those students who withdraw from a course have differing academic credentials from those complete the course?

> *H_6: Students who withdraw from a course will have different High School academic credentials from those who do not withdraw.*

5) Do those successful students who transfer from another institution have differing academic credentials from those who did not transfer?

> *H_7: Successful students who transfer from another institution will have different High School academic credentials from those students who did not transfer.*

Statistical Methodology
For each of the seven hypotheses, the null hypothesis will be accepted or rejected using the significance level of .05. To compare two independent groups based on binary variables, most statistics guidelines suggest using the chi-square test of independence as long as the sample sizes are large enough. Sauro and Lewis

 ISSN: 2473-3857
v3 n4334

(2008) contend, however, that the "latest research suggests that a slight adjustment to the standard chi-square test, and equivalently to the two-proportion test, generates the best results for almost all sample sizes" (p. 75).

To determine whether a sample size is adequate for the chi-square test, calculate the expected cell counts in the 2x2 table to determine if they are greater than 5. When the values in this study met this test, the chi-square test results were used. When the values of one or the other of the subgroups did not meet this test, the N-1 chi-square test was used. The formula for the N-1 chi-square test (Sauro and Lewis, 2008) is below in formula 1 using the standard terminology from the 2x2 table:

$$\chi^2 = \frac{(ad - bc)^2(N-1)}{mnrs} \qquad (1)$$

When the values for both groups in the study failed to meet the threshold, the more conservative Fisher Exact Test was used. The formula for this test is also given by Sauro and Lewis and is formula 2 below:

$$\rho = \frac{m!\,n!\,r!\,s!}{a!\,b!\,c!\,d!\,N!} \qquad (2)$$

Test Results

Hypotheses are supported when the null hypothesis is rejected. In this study, the null hypothesis is rejected when there is a statistically significant difference between the proportions represented by p<.05. On this basis, the first hypothesis (H_1) is accepted and the second (H_2) is rejected. There is a significantly higher percentage of CS majors taking the first programming course who earn a B- or higher vs IS majors.

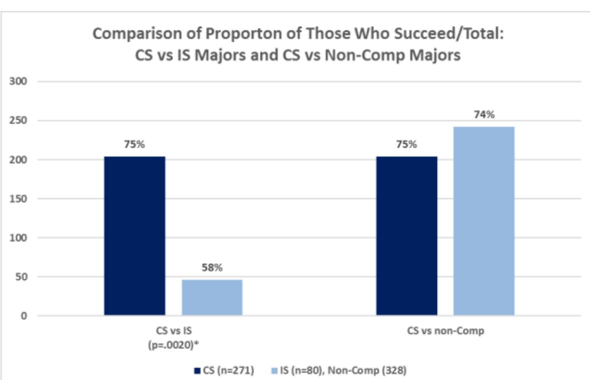


Chart 2.0 – Successful CS vs IS Majors and Successful CS vs Non-Computing Majors

When comparing successful CS majors to successful IS majors, the third hypothesis (H_3) is accepted for all ACT scores.

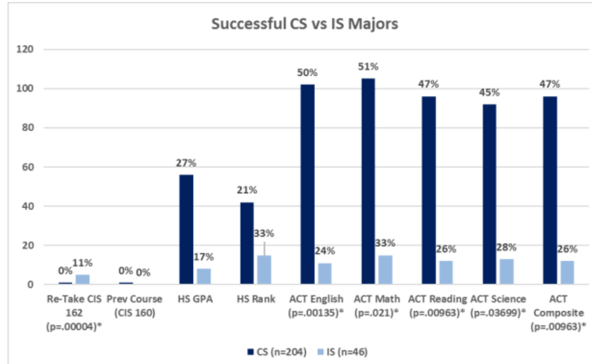


Chart 3.0 – Successful CS vs IS

While the proportions are reversed, the fourth hypothesis (H_4) is also accepted for the following categories of academic preparation: Retaking this course, taking another college programming course, HS GPA, ACT Math and ACT Science.

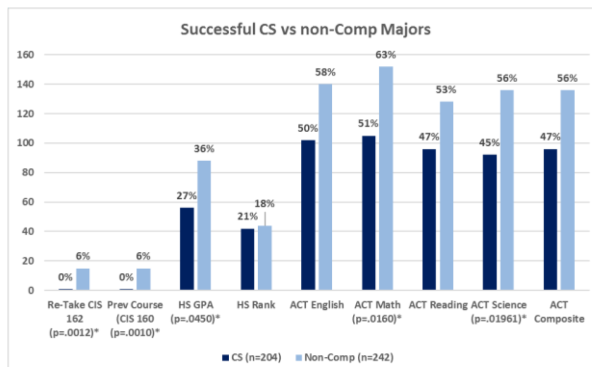
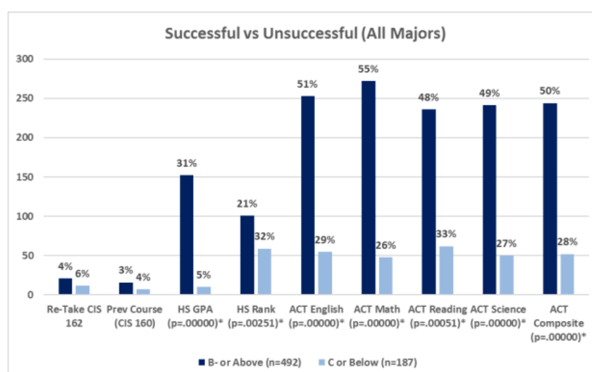


Chart 4.0 – Successful CS vs non-Computing

When combining all students who earned a B- or above, the fifth hypothesis is accepted for all categories of academic attributes, but neither of the previous college courses.



ISSN: 2473-3857
v3 n4334

Chart 5.0 – Academic Attributes of Successful vs Unsuccessful Students Across All Majors

ISSN: 2473-3857
v3 n4334

For students who withdraw from the first programming course, the sixth hypothesis is accepted for HS GPA, HS Rank, ACT Math and ACT Composite Scores (there is not enough data to analyze previous college course, except to say no student who withdrew had taken a previous course).

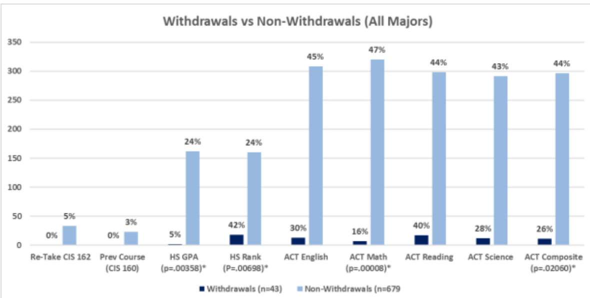


Chart 6.0 – Withdrawals vs Non-Withdrawals

For students who transfer from another institution and then take the first programming course, the seventh hypothesis is accepted for all HS academic credentials, but not for taking a previous college course or retaking this course.

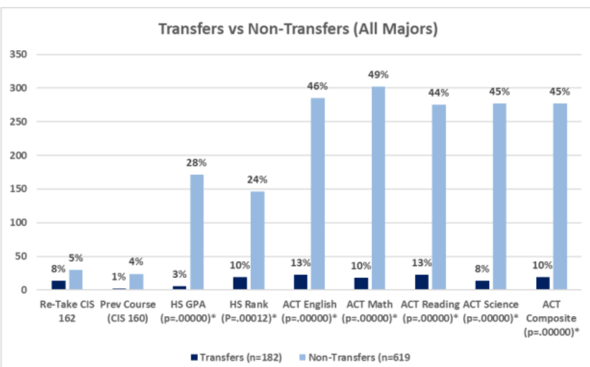


Chart 7.0 – Transfers vs Non-Transfers

4. THE DISCUSSION AND CONCLUSION

Normally, one would assume that CS majors do better in CS 1 than other majors. This is the case between CS and IS majors, but not between CS and non-Computing majors.

Compared to IS majors, this study found that there is a significantly higher percentage of CS majors who have higher than average ACT scores in all areas. Not surprisingly, then, is the significantly larger proportion of successful IS majors who are re-taking this course. Non-Computing majors, too, show a significantly higher proportion who are retaking this course or have taken a previous college programming course, when compared to successful CS majors. In contrast, though, these students also have a significantly higher average HS GPA, ACT Math, and ACT Science compared to CS majors!
When all majors are combined and the successful vs unsuccessful students are compared, there is a significantly higher proportion of students who have above average HS GPA, HS Rank, and ACT scores in all categories. A significantly lower proportion of students who withdraw from this course have an above average HS GPA, HS Rank, ACT Math Score and ACT Composite Score, though none of the students who withdrew had taken this class or another college programming course.

A significantly larger proportion of students who transfer to this institution and succeed in this course, regardless of major, have above average HS academic credentials across all categories, but not in the categories of previous college courses.

While it is expected that students with higher than average HS academic credentials will succeed in CS 1 (see Chart 1), what about those who succeed and are not in this group? The evidence suggests that the only factor that might offset the advantage of high HS academic credentials is prior preparation as evidenced by either taking a programming class twice or taking another college programming course first.

5. THE FUTURE.

If classroom instruction/course design and curriculum pathways stay the same, this pattern will continue.

Historically, majors in the traditional liberal arts and sciences can assume availability and opportunity of coursework in high school to prepare a student for college work in the field. This is not true for the field of computing; many schools do not have room in their state-mandated curriculum or have the financial resources to offer elective courses in Computing.

A program to increase computing options in high schools would address a small portion of this under-prepared population, such as the new AP CS Principles course and the increased focus on STEM in high schools. Unfortunately, those who, for whatever reason, did not have the opportunity or did not choose to take a computing class in high school and now have the desire to enter the discipline after they get to college, are left out, given the status quo.

ISSN: 2473-3857
v3 n4334

Higher education is not meeting the growing demand for graduating enough students qualified to enter the computing field (Ferguson, 2015). Given the constraints on public high school curriculum and budget, pushing the problem of under-preparedness in the computing discipline on them is futile.

College curriculum design and course pedagogy must address this issue in order to meet the growing demand and to insulate computing programs from fluctuations in enrollment. Adding one course to a curriculum would appear to be a benefit to any computing program thereby increasing the success rate of the first programming course (CS 1).

Finally, research has shown that these limiting constraints on who graduates affect a disproportionate number of certain demographic subgroups (O'Neill, 2017). Further research is needed on the effects on various demographic sub-groups of these trends as well as curriculum design and course pedagogy strategies to address these issues.

6. REFERENCES

Advances in AP (2017). *AP Computer Science Principles.* Retrieved June 5, 2017 from https://advancesinap.collegeboard.org/stem/computer-science-principles

Alvarado, C., Lee, C. B., & Gillespie, G. (2014). New CS1 pedagogies and curriculum, the same success factors? *SIGCSE '14*, 379-384.

Campbell, P. F., & McCabe, G. P. (1984). Predicting the success of freshmen in a computer science major. *Communications of the ACM*, 27(11), 1108-1113.

College Board (2013). Student Score Distributions 2013. Retrieved June 5, 2017 from http://media.collegeboard.com/digitalServices/pdf/research/2013/STUDENT-SCORE-DISTRIBUTIONS-2013.pdf

College Board (2016). Program Summary Report 2016. Retrieved June 5, 2017 from https://secure-media.collegeboard.org/digitalServices/pdf/research/2016/Program-Summary-Report-2016.pdf

Ferguson, R., Leidig, P. M., & Reynolds, J. H. (2015). Including a programming course in general education: Are we doing enough? *Information Systems Education Journal*, Volume 13(No. 3), 34-42. (A preliminary version appears in the Proceedings of ISECON 2014).

Gandhi-Lee, E., Skaza, H., Marti, E., Schrader, P. G., & Orgill, M. (2015). Faculty perceptions of the factors influencing success in STEM fields. *Journal of Research in STEM Education*, 1(1), 30-44.

Konvalina, J., Wileman, S. A., & Stephens, L. J. (1983). Math proficiency: a key to success for computer science students. *Communications of the ACM*, 26(5), 377-382.

Lewis, C. M., Yasuhara, K., & Anderson, R. E. (2011). Deciding to major in computer science: a grounded theory of students' self-assessment of ability. *ICER '11*, 3-10.

Heitin, L. (2014), Computer Science: Not Just an Elective Anymore, *Education Week*, http://www.edweek.org/ew/articles/2014/02/26/22computer_ep.h33.html, February 25, 2014

Maggie O'Neill (2017), Computer Science Before College, Retrieved June 5, 2017 from http://www.computerscienceonline.org/cs-programs-before-college/

Mohr-Schroeder, M. J., Jackson, C., Miller, M., Walcott, B., Little, D. L., Speler, L., & Schroeder, D. C. (2014). Developing middle school students' interests in STEM via summer learning experiences: See Blue STEM camp. *School Science and Mathematics*, 114(6), 291-301.

Reynolds, John. H., Adams, D. Robert, Ferguson, Roger, & Leidig, Paul. M. (2017). Programming in the IS curriculum: Are requirements changing for the right reason? *Information Systems Education Journal*, Volume 15(No. 1), 80-85. (A preliminary version appears in the Proceedings of EDSIG 2016)

Rountree, N., Rountree, J., & Robins, A. (2002). Predictors of success and failure in a CS1 course. *ACM SIGCSE Bulletin*, 34(4), 121-124.

ISSN: 2473-3857
v3 n4334

Sadler, P. M., Sonnert, G., Hazari, Z., & Thi, R. (2014). The role of advanced high school coursework in increasing STEM career interest. *Science Educator*, 23(1), 1.

Zweben, S., & Bizot, B. (2015), Continued Booming Undergraduate CS Enrollment http://cra.org/wp-content/uploads/2016/05/2015-Taulbee-Survey.pdf, From *Cra Taulbee Survey, Computing Research News*, May 2016 Vol. 28 / No. 5

Tafliovich, A., Campbell, J., & Petersen, A. (2013, March). A student perspective on prior experience in CS1. *SIGCSE '13*, 239-244.

Wang, M. T., & Degol, J. (2013). Motivational pathways to STEM career choices: Using expectancy–value perspective to understand individual and gender differences in STEM fields. *Developmental Review*, 33(4), 304-340.

Werth, L. H. (1986). Predicting Student Performance in a Beginning Computer Science Class. In *SIGCSE '86 Proceedings of the Seventeenth SIGCSE Technical Symposium on Computer Science Education* (Vol. 18, No. 1, pp. 138-143).

Wilson, B. C., & Shrock, S. (2001). Contributing to success in an introductory computer science course: a study of twelve factors. In ACM SIGCSE Bulletin (Vol. 33, No. 1, pp. 184-188). ACM.