Getting Principled: Reflections on Teaching CS Principles at Two College Board University Pilots

Jeff Gray
University of Alabama
Box 870290
Tuscaloosa, AL 35487
gray@cs.ua.edu

Michele Roberts
IUPUI
723 W. Michigan St., SL 280
Indianapolis, IN 46202
microber@iupui.edu

Jonathan Corley
University of West Georgia
1601 Maple Street
Carrollton, GA 30118
JCorley@westga.edu

ABSTRACT

The College Board estimates that the new AP CS Principles (CSP) course will set a participation record for new course launches. With a large number of students across the USA enrolling in CSP at the high school level, CS departments at colleges and universities will need to begin considering their position for awarding AP credit. One possibility is the introduction of a new college-focused CSP course for non-majors that can serve as a mapping for AP credit. This paper summarizes the experiences of two faculty at different universities who were official CSP College Board Pilots for several years. An overview of each university's experience is provided in terms of student demographics, common evaluation measures, and individual course nuances, followed by a series of recommendations to faculty who are considering the creation of a CSP course within the curriculum of their own department.

Keywords

CS Principles; Non-Majors; Pre-service

1. INTRODUCTION

In an effort to address the lack of diversity observed among high school students in the enrollment of AP CS A (a Java-focused programming course) in the USA, the College Board began development of the AP CS Principles (CSP) curriculum in 2008 to provide a more engaging computing experience for high school students of all backgrounds. With funding from the National Science Foundation (NSF), 50 Pilot instructors were supported over several years to provide feedback during the development of a Curriculum Framework for a new introductory CS course, CSP. The Pilot instructors, from 38 high schools and 12 universities, shared in developing resources and participated in the evaluation of the CSP course as it evolved. Unlike a traditional CS1 course, CSP was intentionally designed to attract non-traditional computing students by focusing on what has been identified as the "Big Ideas" of computing (e.g., Creativity, Abstraction, Data, Algorithms, Programming, Internet, Global Impact), explored through cooperative, discovery-based activities. In addition to scoring on a final exam, CSP students are also required to submit two significant performance tasks (one that explores the global impact of a computing innovation and a second that creates an original programming artifact). Pilot CSP courses have been positively received. In fact, the College Board expects a record number of students to take the CSP exam in its first year with over 1,500 teachers registered through the audit process before the start of the 2016-2017 academic year. According to an interview with College Board officials, CSP will "be the largest debut of an AP ever in the history of the College Board, with nearly 50 percent more teachers launching the program than the previous crown-holder: the AP World History exam, which had 998 teachers in its first year" [4]. The high expectation for student participation in the first AP CSP exam in May 2017 suggests that many students will be seeking opportunities for AP credit at their future university. Thus far, over 100 universities have expressed interest and attested their commitment to awarding AP credit for CSP, but only a dozen universities currently offer an actual course that matches the CSP curriculum. Because of the volume of future students arriving on college campuses with CSP backgrounds, computer science faculty in higher education in the USA will need to consider their procedure for awarding AP CSP credit in terms of the course that maps best to a qualifying score. Solutions may include the introduction of a new CSP course in many CS departments, most likely to be offered to non-majors.

This paper describes experiences from two of the twelve official university College Board Pilot instructors: Jeff Gray at the University of Alabama ("UA") and Michele Roberts at IUPUI in Indiana. The experiences are informed with data collected by the College Board's external evaluator (Haynie Research and Evaluation) on several aspects of each course implementation. Experiences and data are shared to help other university faculty who may be interested in initiating a CSP course at their university. We contrast CSP student metrics to CS 1 student metrics for context only, as it has yet to be demonstrated how comparable the two student groups are composed. The summary findings of our study include the following core results:

- The enrollment at each university over three years indicates that there were twice as many women registered for CSP than the traditional CS 1 course within each department, but CSP has shown less enrollment impact for other URMs compared to the CS 1 course.
- A significant difference was observed in the DFW (Drop/Fail/Withdraw) rates for CSP compared to the CS 1 course. At both IUPUI and University of Alabama, the DFW rate for CSP is only 2% and 3%, respectively, but for CS 1 it ranges from 18% to 54% at IUPUI and 50% to 61% at the University of Alabama. CSP offers the potential for high growth enrollment with close to zero dropouts.

SIGCSE'17, March 8–11, 2017, Seattle, WA, USA.
© 2017 ACM. ISBN 978-1-4503-4698-6/17/03…$15.00.
http://dx.doi.org/10.1145/3017680.3017758.

Year	2013-2014		2014-2015		2015-2016	
	CSP	CS 1	CSP	CS 1	CSP	CS 1
Women	37%	13%	58%	12%	63%	22%
URM	8%	12%	18%	9%	19%	20%

Table 1 Enrollment demographics for IUPUI CSP and CS1

Year	2013-2014		2014-2015		2015-2016	
	CSP	CS 1	CSP	CS 1	CSP	CS 1
Women	52%	27%	55%	26%	70%	28%
URM	30%	16%	21%	21%	13%	21%

Table 2 Enrollment demographics for University of Alabama CSP and CS1

- Common evaluation questions given to students at both universities suggest positive or stable affect changes concerning confidence and interest in CS.

The remaining sections provide an overview of the CSP course at IUPUI and UA, followed by a discussion of lessons learned in implementing the course over several years. Our goal is to help higher education faculty understand the benefits and challenges in introducing a new CSP course into their own curriculum.

2. OVERVIEW OF EACH COURSE

This section summarizes the specific details of the CSP course offered at both IUPUI and UA.

2.1 Snapshot: IUPUI

IUPUI is an urban public research university located in downtown Indianapolis. IUPUI has an approximate total enrollment of 30,000, making it the third largest university in Indiana. Over 90% of the student population are state residents. The CSP course was offered by the Computer and Information Science Department ("CSCI"), one of many departments located within the School of Science. The CSCI department serves approximately 2,100 students a year, including both majors and non-majors. The CSP course is offered as a 3-credit, 200-level course for non-majors.

At IUPUI, CSP was launched Fall 2013 and offered every Fall and Spring semester thereafter. Enrollment growth has been steady and steep. Course launch had a single section with 22 students enrolled. Fall 2016 has an enrollment total of 175 students across five full and waitlisted sections. The course was marketed by targeting campus advisors: CSP faculty met with advisors from each department unit on campus and provided course information, sample lab descriptions, student project displays, and testimonials by previous students. In Fall 2015, school and university administration were successful in adding the CSP course to the Analytical Reasoning category of state core educational requirements. This promoted the CSP course to be on a list of approved courses that can substitute for 3 of the 9 mandatory mathematics course credits required for graduation from a public university in Indiana. The IUPUI CSP course has maintained strong fidelity with the CSP framework, including content across the "Big Ideas" explored in predominantly collaborative settings and, through Spring 2016, requiring completion of the two College Board required Performance Tasks.

Enrollment in the CSP course has grown with increasingly broadened participation, reflecting the intent of the CSP initiative. Female participation has risen from 37% to 63%, while URM participation (Latino and African American) has grown from 8% to 19%. A school-level minority outreach program presumably benefitted URM diversity in the CSP course; whereas student end-of-semester survey responses repeatedly identify the collaborative CSP curriculum as particularly attractive for female students. Student demographics in the CSP course and the major CS 1 course are compared in Table 1.

The CSCI department awards incoming credit on the two CS courses available from the College Board (CS A and the new CSP course) with the qualifying scores mapping shown in Table 3.

Course	Score	Awarded
CS A	**3**	**Undistributed credit (3 credits)**
	4, 5	**Credit for CS 1 Course for majors (4 credits)**
CSP	**3, 4, 5**	**Credit for 200 level non-major CSP course (3 credits) (if a score of 5, student is encouraged to test out of CS 1 course for majors)**

Table 3 AP CS Qualifying Scores at IUPUI

2.2 Snapshot: University of Alabama

The University of Alabama ("UA") is a suburban public research university located in a college town. UA has an approximate total enrollment of 37,000, which is the largest university in its state; however, only 46% of University of Alabama students are in-state residents. The CSP course at UA is offered as a 3-credit course within the Department of Computer Science in the College of Engineering. The course is primarily enrolled by non-majors, specifically Secondary Math Education (SME) and Math majors.

The UA CSP course was officially launched as a College Board Pilot in Fall 2011 and has been offered every subsequent Fall, but not Spring. To coincide with the first offering of CSP at IUPUI, only the courses taught from 2013 to 2015 are included in the UA evaluation reported in this paper. Enrollment growth of the UA CSP course has been artificially capped at 40 students, but has been waitlisted during each offering. A Spring offering of the course will begin in 2017 and there are additional plans to expand

Year	2013-2014		2014-2015		2015-2016	
	CSP	CS 1	CSP	CS 1	CSP	CS 1
IUPUI-DFW	2%	54%	2%	23%	2%	18%
UA-DFW	3%	52%	3%	60%	3%	51%

Table 4 CSP and CS 1 Drop/Fail/Withdraw Rates at IUPUI and UA

the course further to accommodate growing demand. The 2011 initial marketing of the course focused on contacts to advisors in the College of Arts and Sciences.

A collaboration with the College of Education at UA began with the Fall 2013 offering as a way to initiate pre-service CS training among SME students. Prior to 2013, the SME students were required to take the UA equivalent CS 1 course, which was not popular with SME students due to the deep focus only on programming. The win-win departmental collaboration allowed the SME students to take a CSP course that was more broad and addressed topics closer to their interest (e.g., data investigation topics in CSP using mathematics), while also providing initial exposure to these future high school math teachers who may be encouraged to introduce CSP when they begin a new teaching position. Since 2013, only Math and SME students have been allowed to enroll in the CSP course at UA, but the future Spring 2017 offering will open the enrollment to all majors again. Like the IUPUI CSP course, the UA CSP offering has also maintained strong fidelity with the CSP framework, exploring the Big Ideas within a collaborative curriculum, and requiring the final scored exam and two Performance Tasks that are stipulated in the College Board CSP curriculum.

Enrollment in the UA CSP course has been dominated by women, reflecting the intent of the CSP initiative to broaden CS interest. The single Fall offering and the capped enrollment will be expanded in the future with additional sections and more open enrollment to other majors. The past enrollment demographics (e.g., large percentage of women) may be influenced by the collaboration with SME students, who are predominantly female. Student demographics in the UA versions of the CSP course and the major CS 1 course are compared in Table 2.

UA was one of the earliest universities to officially recognize CSP and provide a formal statement of mapping from qualifying scores to credit for a university course. Table 5 shows the UA qualifying scores for both CS A and CSP, which are also documented on the UA Registrar's website.

Course	Score	Awarded
CS A	**4, 5**	**Credit for CS 1 Course for majors (4 credits)**
CSP	**3, 4, 5**	**Credit for 100 level non-major CSP course (3 credits)**

Table 5 AP CS Qualifying Scores at University of Alabama

3. Common Evaluation Measures

In this section, we provide a brief evaluation of the two courses focusing on three key concerns: student success, student interest in CS, and student confidence in CS.

3.1 Student Success

In addition to a broadened demographic, we also observed a significantly higher course success rate in the CSP pilots than in the more traditional CS 1 courses. In fact, hardly any of the CSP students withdrew from any CSP course section at either university (most semesters, the withdraw figure was a single student). Looking more broadly at student success by calculating DWF rates ("Drop, Withdraw or Fail"), the CSP courses compare favorably to CS 1 success values, with consistent low numbers recorded each year (two percent at IUPUI and three percent at UA.). Table 4 shows the comparable DFW rates over three years for all sections of CSP and CS 1 at each university.

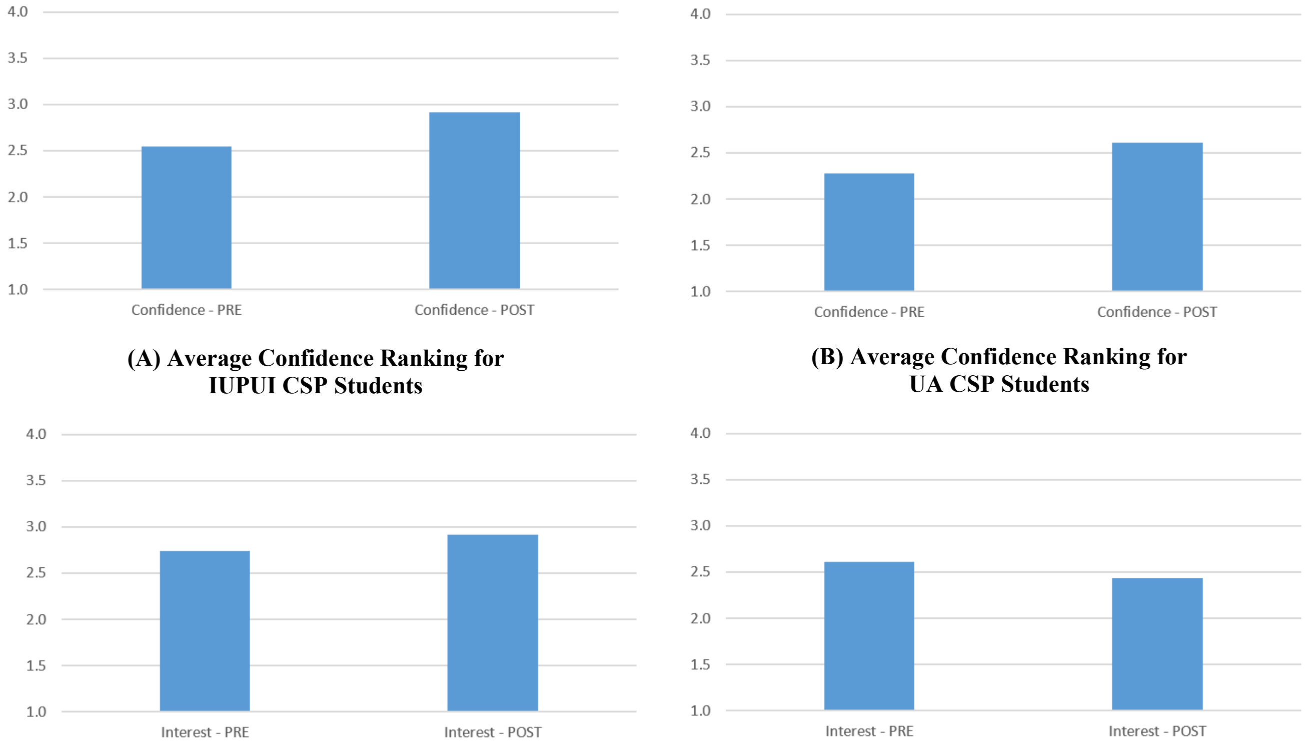
Rephrasing the DFW rates as Pass rates, the three-year average Pass rates for IUPUI were 98% (CSP) and 68% (CS 1); with UA realizing a 97% Pass rate for CSP and 46% for CS 1. The positive CSP Pass rate for both of our three-year averages was a pleasant surprise. With respect to CS 1, two separate studies of hundreds of universities reported that the average pass rate for CS 1 is approximately 67% [1, 5]. A more recent study among students at just one university observed a 71% Pass rate for CS 1 [3]. The much higher CSP Pass rates that we observed, across many sections of the course over three years, suggests the ability of CSP courses to keep students enrolled in the course while earning a passing grade.

3.2 Student Interest and Confidence in CS

As College Board Pilot programs, students at both IUPUI and UA participated in a common set of evaluations to measure student affect toward the CSP course specifically, and CS in general. In particular, the same student survey was given to each of our classes every semester and we had access to a portion of the results for this study. As part of the survey, students were asked to rank their agreement with a variety of statements. The students ranked their agreement on a 4 point Likert scale range: 1-Strongly Disagree, 2-Disagree, 3-Agree, 4-Strongly Agree. The student survey and evaluation measures provided us with a rich data source for common comparison. The following subsections summarize our findings with respect to student confidence and interest in CS both before and after taking our CSP courses.

3.2.1 Confidence

To evaluate student confidence in CS, we identified a key survey question present in both pre- and post-surveys to indicate the change of the students' confidence as a result of the course. The students ranked their agreement to the statement, "I have self-confidence when it comes to CS." Figures 1 and 2 present the average ranking provided by students in the pre/post surveys for IUPUI and UA, respectively. We observed an increase in confidence between pre- and post-surveys. To evaluate the significance of the differences, we performed a pair-wise t-test with target of $p < 0.05$. The results found a significant difference was present. Thus, the survey responses indicate that students taking the CSP course at both IUPUI and UA had an increase in



(A) Average Confidence Ranking for IUPUI CSP Students

(B) Average Confidence Ranking for UA CSP Students

(C) Average Pre/Post Interest Ranking for IUPUI CSP Students

(D) Average Pre/Post Interest Ranking for UA CSP Students

Figure 3 Pre/Post Survey Results for CSP Students

confidence after taking the CSP course. This is an exciting finding because it indicates that the CSP course was able to improve confidence in CS for students outside of the field (i.e., non-majors).

3.2.2 Interest

To evaluate student interest in CS, we identified a key survey question present in both pre- and post-surveys that had the students rank their agreement to the statement, "I would take additional CS courses if given the opportunity." The average pre- and post- responses observed for students at IUPUI and UA are presented in Figures 3 and 4, respectively. A slight increase was observed at IUPUI, and a slight decrease was observed at UA. The observed changes were slight enough that a pairwise t-test performed with target $p < 0.05$ did not find a significant difference for either university. These results indicate that students are not discouraged from completing the CSP course, but also not more likely to pursue CS further. To better understand these findings, both universities intend to add end of course survey questions that more deeply tap student reasoning around interest in additional CS coursework, as well as add content to the CSP course that provides students with information on subsequent coursework available for both majors and non-majors. Though we had hoped to find an increase in student interest in CS, the overall measures of student success in CSP compare quite favorably with similar measures in respective CS 1 courses.

4. KEY LESSONS LEARNED

In this section, we briefly overview several lessons that were learned from the shared and individual experiences across the two university Pilot courses.

4.1 Shared Lessons

(A) Both Pilot instructors note the strong infrastructure support for the CSP course, particularly the wide range of resources available from a growing number of curriculum providers [2]. The various communities of practice, endorsed curriculum providers, and CSP portals have made available a wealth of resources to aid instructors in building and maintaining a CSP course, including complete course content freely accessible in the public domain. Prospective and existing CSP faculty can adopt a complete course or utilize content at the level of individual lesson plans and assignments; join an online community; participate in face to face or online training, and more. The emerging CS K-12 pipeline owes a tremendous debt to the infrastructure investment made by the NSF, academia, non-profits, College Board, and passionate teachers.

(B) As experienced by both faculty at IUPUI and UA, the CSP curriculum is noticeably different than traditional CS 1 content. Curriculum engagement helped in attracting CSP students, and word of mouth "advertising" is likely to continue to grow enrollment, even without policy leverage or audience focus. Anonymous student end-of-course feedback at both universities was very positive, soliciting comments such as "hard but so much fun," "relevant to my life," "I loved the group work." Because of the curriculum engagement, both universities expect demand for

the course to continue to increase. Additionally, both faculty agree that student engagement in CSP is a key contributor to lower DFW rates observed at both universities. Future research thrusts will explore student success impact from infusing CS 1 with some of the engagement techniques and content of CSP.

(C) The CSP curriculum is both rigorous and rich – instructors at the university level will find it challenging to cover all the proposed content in a single semester (given the reduced contact hours compared to the time a high school teacher has to cover the curriculum over a full year). The base course as designed for high school covers a wide range of material across a set of 7 Big Ideas. Thoughtful decisions had to be made to identify what work was best completed collaboratively in class, and what work could be completed effectively in less collaborative arrangements outside of class. The faculty across both universities identified key areas where concepts could be introduced and reinforced in parallel with other concepts. For example, Abstraction can be included within units devoted to other Big Ideas, such as Programming or Data. A concept can be introduced in a variety of contexts to aid students in understanding abstraction, and to allow a semester long university level course to cover content more efficiently. Many of the Big Ideas are crosscutting in nature and can be introduced in combination.

(D) Both faculty noted that the recommended learning culture is collaborative and more discovery-based than more traditional CS classroom pedagogy. Without some orientation, not all CS faculty will be comfortable with the recommended CSP classroom culture. In addition to faculty orientation, Teaching Assistants will also need training to support the shift from a more lecture delivered approach with individual student work to a more discovery-based approach supported through more collaborative student work. With faculty commitment, however, CSP course success can provide opportunity to explore new classroom delivery techniques that may then be fruitfully mapped back to more traditional introductory CS courses.

(E) Because of the general collaborative culture of CSP, additional and differently trained infrastructure support may be required to implement a successful CSP classroom. Managing collaborative student work, potentially supporting a wide variety of technical platforms, and maintaining student engagement in full classrooms will typically require additional TAs. At IUPUI, a classroom strategy of Peer Led Team Learning ("PLTL") was particularly helpful in maintaining student engagement [6]. In this implementation, students who were very successful in the CSP course in a previous semester were trained in PLTL strategies and hired to lead small groups in curriculum discovery and discussion. At both universities, TAs selected from a traditional pool of CS majors had no experience with supporting discovery-based learning, collaborative strategies, or diversity. In fact, additional technical training was required so that traditional TAs could effectively help CSP students in block-language platforms (e.g., Snap!, App Inventor, Scratch), as well as provide help to CSP students in user interface design.

(F) Not all classroom practices were received equally by CSP students. Some practices that were perceived well in the course, as reported by students at both universities, included the following: group projects, team learning and showing off their programming work from a cell phone. Particularly, providing students the ability in a non-majors course to develop applications for their phones was informally observed to aid student engagement. Students were also asked to journal their reflections on CSP coursework (e.g., reviewing their process for completing the projects, both individual and collaborative, recording instances of abstraction, etc.), which had a positive outcome in terms of improved curriculum understanding, but was not perceived favorably by students as a general practice.

(G) The results of both pilot experiences provide interesting questions for future investigations in CS education. The CSP framework is built around some of the key attributes identified as successful strategies for CS 1 students (e.g., curriculum engagement, cooperative learning, on-ramping with a visual language) [7]. Do these attributes benefit non-traditional and traditional CS students uniformly, or is impact amplified for URM's? Are the most optimal learning strategies for URM non-majors different from those of URM majors? Why did the CSP course, at both universities, fail to significantly increase interest in CS?

4.2 Lessons from IUPUI

At IUPUI, one of the earliest lessons learned was the need to better thread the "Big Ideas" of the curriculum. In other words, treating the Big Ideas as separate modules to be completed sequentially was ineffective: there were not enough contact hours to complete all the course materials, and final exam scores were a direct reflection of module sequencing (e.g., students performed best on those Big Ideas introduced latest in the course). By revising the curriculum to reflect more of the interrelated nature of the Big Ideas, more curriculum could be covered and exam scores improved. In particular, the Big Ideas of Global Impact and Internet offered significant threading efficiencies, with no apparent exam impact from modularizing the content and distributing content sections across the curriculum.

A second significant lesson learned at IUPUI was how to best manage and scaffold the Programming thread. The non-STEM students successfully acquired fundamental programming skills, but responded best to a programming thread that began early, included on-ramping, and was sustained throughout the semester. Student feedback suggested that the acquisition of programming fluencies had a "run-rate"; in comparison to other threads, students needed more reflection time and hands on experience to become both confident and competent in programming fundamentals.

4.3 Lessons from UA

At UA, a significant concern during the first two years of the course offering was how best to cover programming fundamentals including: decision constructs, looping constructs, basic collection construct, functions, and basic data types along with variables. The Fall 2011 course adopted Snap!, but a change was made to App Inventor in 2012-2013 to benefit from the potential opportunity for global impact and engagement provided by leveraging smart phone development for programming projects. However, due to time constraints and for the sake of consistency, we moved back to Snap! because we found it was more natural to include the various programming constructs within samples and student projects in Snap! In particular, it was often difficult to include lists and looping constructs naturally within the simple App Inventor projects that were typical of students in our CSP course, most of whom had no prior exposure to programming. With App Inventor, our students spent more time building a GUI and creating event handlers, compared to the opportunity in Snap! to jump straight to the core programming concepts without any deep setup of context. However, students seemed more engaged overall when using App Inventor, perhaps due to the excellent opportunity in App Inventor for students to demonstrate completed projects on their phone to friends. Platform selection may seem trivial, but it is not: success in the CSP philosophy requires both engagement and rigor, and additional consideration is required at the university level in particular to adequately provide course coverage to non-traditional students in a single semester.

The UA collaboration with Math Education (SME) students provided several unique opportunities. A handful of the top performing SME students were invited to participate on NSF and Google grants associated with professional development of CSP for in-service teachers who had little CS experience. The SME students who completed CSP helped with summer camps for high school students learning CSP and also in-service teacher training. Thus, the roles were reversed because it is more typical for in-service teachers to train pre-service teachers. In this case, the pre-service teachers who completed the CSP course had much more content knowledge than in-service teachers who were participating in a CSP professional development workshop.

5. CONCLUSION

The College Board's development of a new AP CSP curriculum was motivated by a need to create a larger and more diverse pipeline of computing students. The record number of teachers who have registered CSP courses in the inaugural 2016-2017 course launch suggests that universities will soon begin seeing incoming students with CSP experience, and may need to develop departmental mapping policies for AP CSP scores. One consideration is to introduce a college-focused version of CS Principles, typically to be offered as a 3 credit, one semester course for non-majors. This paper summarized the experiences of two university CSP pilots with the purpose of helping other university faculty make decisions about CSP support. In both pilots, CSP course experience has been overwhelmingly positive, with the observation of steep enrollment growth, diversity gains, and stable to positive affect gains in confidence and interest. These results were achieved with some challenges, including the one semester pacing decisions for non-majors, as well as additional and supplementary trained TA's. In summary, as well as building a more diverse K-12 pipeline, the CSP course appears to offer equally exciting promise at the university level: an opportunity to expand mission and enrollment, as well as suggest future CS educational investigative thrusts into broadening participation, creating equitable classroom cultures, and developing engaging curriculum content.

6. ACKNOWLEDGMENTS

We would like to thank the College Board, through support from the National Science Foundation, for selecting us as CS Principles Pilot instructors. In particular, we are grateful to Lien Diaz at the College Board, Owen Astrachan and Amy Briggs (NSF PIs), and Jan Cuny (NSF). Special thanks to Kathy Haynie and Sheryl Packman for their help with evaluation data. Part of the project at the University of Alabama was supported in part by NSF CE21 (1240944). The CS104 course at Alabama also benefitted from the support of Amber Wagner and Mokter Hossain.

7. REFERENCES

[1] Jens Bennedsen and Michael E. Caspersen, "Failure Rates in Introductory Programming," *SIGCSE Bulletin,* June 2007, pp. 32-36.

[2] Marie desJardins, "Creating AP CS Principles: Let many Flowers Bloom," *ACM Inroads*, vol. 6, no. 4, pp. 60-66.

[3] Diane Horton and Michelle Craig, "Drop, Fail, Pass, Continue: Persistence in CS1 and Beyond in Traditional and Inverted Delivery," In *Proceedings of the 46th ACM Technical Symposium on Computer Science Education* (SIGCSE '15), Kansas City, KS, March 2015, pp. 235-240.

[4] Mary Jo Madda, "College Board Hopes to Broaden Access to Coding With New 'AP Computer Science Principles' Course," *EdSurge*, July 25, 2016, URL: http://bit.ly/edsurge-csp .

[5] Christopher Watson and Frederick W.B. Li, "Failure Rates in Introductory Programming Revisited," In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (ITiCSE '14). Uppsala, Sweden, June 2014, pp. 39-44.

[6] Susan Horwitz, Susan H. Rodger, Maureen Biggers, David Binkley, C. Kolin Frantz, Dawn Gundermann, Susanne Hambrusch, Steven Huss-Lederman, Ethan Munson, Barbara Ryder, and Monica Sweat, "Using Peer-led Team Learning to Increase Participation and Success of Under-represented Groups in Introductory Computer Science," In *Proceedings of the 40th ACM Technical Symposium on Computer Science Education* (SIGCSE '09), March 2009, pp. 163-167.

[7] Arto Vihavainen, Jonne Airaksinen, and Christopher Watson. "A Systematic Review of Approaches for Teaching Introductory Programming and their Influence on Success," In *Proceedings of the tenth annual conference on International computing education research* (ICER '14), Glasgow, UK, August 2014, pp. 19-26.