RadGrad: Removing the "Extra" from Extracurricular to Improve Student Engagement, Diversity, and Retention

Philip M. Johnson
Carleton Moore
johnson@hawaii.edu
cmoore@hawaii.edu
Department of Information and Computer Sciences
University of Hawaii at Manoa
Honolulu, HI

Peter Leong
Seungoh Paek
peterleo@hawaii.edu
spaek@hawaii.edu
Department of Learning Design and Technology
University of Hawaii at Manoa
Honolulu, HI

ABSTRACT

RadGrad is a curriculum initiative implemented via a web-based application that combines features of social networks, degree planners, and serious games. RadGrad redefines the traditional meaning of "progress" and "success" in the undergraduate computer science degree program, with the ultimate goal of improving student engagement, diversity, and retention. In this paper, we relate Rad-Grad to other curriculum initiatives, overview its key functionality, present results from an evaluation conducted during its first year of deployment, and discuss our lessons learned and future directions.

CCS CONCEPTS

• **Social and professional topics** → **Informal education**; **Computing literacy**.

KEYWORDS

Curriculum Initiative; Diversity; Engagement; Retention

1 INTRODUCTION

Traditional curriculum initiatives for undergraduate computer science generally fall into one of three categories. The first involves the injection of a specific technology across the curriculum, such as initiatives for cloud computing [6]. The second involves the injection of a specific domain across the curriculum, such as initiatives for cybersecurity [15] or distributed computing [1]. The

third involves the integration of practices intended to address demographic problems including lack of access or diversity, such as initiatives at the University of Illinois [12] and University of Oklahoma [5].

One problem with these types of curriculum initiatives is that they all require changing, well, the curriculum. This is a problem because changing the curriculum is hard: any significant curriculum change will require the approval of committees at multiple levels of the educational institution, it will require buy-in across the faculty, and it will often require significant resources to implement and sustain.

One benefit of these types of curriculum initiatives is that if they are successfully implemented, then *student* buy-in is not necessarily required. Because the change is to the curriculum, if a student wants to graduate, then they need to follow the changed curriculum.

At our University, we have been experimenting with an initiative called "RadGrad", which both falls outside these typical categories of curriculum initiatives, and thus inverts the typical problems and benefits associated with traditional curriculum initiatives.

RadGrad originally rose from a recognition that our department did not have the resources to enable its curriculum to keep up with the rate of change in computer science technology and the ever widening domain of application areas, nor could we keep up with student demand for access to curriculum content. In some cases, the local high tech community or online educational services could fill in the gaps, but our department provided no support for student awareness of these extracurricular resources, or guidance as to which ones were worth spending time on, or how such activities might integrate with existing curricular offerings. Finally, since extracurricular activities, are, by definition, "extra", there is a structural incentive for students to avoid these activities and instead focus solely on curricular work which directly affects their GPA.

A second motivation for RadGrad was the recognition that an increase in demand for our undergraduate computer science degree programs was paradoxically leading to a decrease in the diversity of our student body. Increasingly few women and underrepresented groups were making it all the way through our program to graduation. Although approximately 40% of our first year students each year are women, only 15% of our graduating seniors in AY 2018 were women.

To address these issues, we created RadGrad, which is a web-based application that combines features of social networks, degree planners, and serious games. Students in our major can login to RadGrad and declare their interests and career goals. The system then recommends faculty-curated extracurricular opportunities matching their preferences. A degree planner tool enables students to lay out their "degree experience" as a combination of curricular activities (courses) and extracurricular activities (internships, meetups, online courses, hackathons, etc) for each upcoming semester until their planned graduation date. Finally, RadGrad enables students to learn about the broader ways in which computer science impacts upon local, national, and global society, as well as helping them to connect with and create communities of practice inside and outside of the department. As will be discussed below, related research suggests that making these connections can improve student engagement, diversity, and retention.

To combat the view of extracurricular activities being perceived as "extra", RadGrad does not use GPA to represent student success or progress. Instead, RadGrad provides a three component metric called "ICE", an acronym for Innovation, Competency, and Experience. Students are awarded Competency points for successfully completing a course, and a varying amount of Innovation and/or Experience points for successfully completing an extracurricular activity defined within the system. For example, participating in a hackathon will earn some number of Innovation points, since the goal of hackathons is to create something new. Finishing a summer internship in a high tech company will earn Experience points since that activity provides students with a sense for the demands of a high tech workplace environment. (Depending upon the nature of the internship, it could also earn Innovation points.) To "win" at RadGrad, students must (among other things) earn at least 100 Innovation, Competency, and Experience points by completing some combination of curricular and extracurricular activities by the time they graduate.

Unlike traditional curriculum initiatives, RadGrad exists apart from the official curriculum requirements for our degree programs. As a result, we have been free to deploy and experiment with the approach without the multi-year approval process normally required for curriculum changes. On the down-side, participation in Rad-Grad is voluntary, and so students must opt in to the system and its philosophy to gain its benefits.

In this paper, we will present our findings from three years of design and development and our first year of deployment into the computer science department at our institution. Section 2 orients our research with respect to other curriculum initiatives and research on egagement, retention, and diversity. Section 3 briefly introduces the functionality of the application. Section 4 presents results from a mixed-methods evaluation of our first year of deployment, and Section 6 discusses ways we will build on our findings in the next several years.

2 RELATED WORK

There is a national need for undergraduate computer science degree programs to improve both retention (the percentage of students entering CS programs who finish the degree) and diversity (the percentage of graduates who are female and/or from an underrepresented minority group). We need to improve retention because the projected demand for skills in computer science far exceeds current production [4]. We need to improve diversity because a more diverse STEM population improves tech innovation at large. For example, mixed-sex teams filed 40% more information and technology patents than all-male teams [2], and management diversity leads to a $42M increase in S&P value of firms [7].

While the need is clear, solutions are complicated. Gender diversity in computer science has actually fallen in the last 20 years [8], with no well accepted explanation for its cause. Some diversity-related issues start in middle and high school: black students are less likely than white students to have computer science courses in middle and high school, and female students are less likely than male students to be told they would be good at computer science [9]. There is some research that provides evidence for a way forward: a study by Google [8] concludes that four factors primarily influence young womens' decision to pursue CS: (1) social encouragement (positive reinforcement of CS pursuits from family and peers); (2) self perception (an interest in problem solving and a belief that those skills can be translated to a successful career); (3) academic exposure (availability of curricular and extracurricular CS activities); and (4) career perception (view of CS as a career with diverse applications and a broad potential for positive societal impact). Stout and Camp [14] make similar points around social relevance, a sense of belonging, and cultural bias. RadGrad implements capabilities designed to help address isssues around self perception, academic exposure, career perception, and social relevance among its student users.

For those high school students who graduate and enter an undergraduate degree program in computer science, retention becomes a significant issue. More than half of the students who start out in science or engineering switch to other majors or do not finish college at all [10]. Initiatives to improve retention, such as the Threads undergraduate curriculum at Georgia Tech, emphasize giving students more control over their degree plan, a better understanding of how their studies relate to their career interests, and an increased emphasis on the importance of extracurricular activities [3]. RadGrad provides a technology platform, information system, and incentive structure with these emphases.

Communities of Practice (CoP) is a theory of learning first proposed in 1991 [11], more fully developed in 1998 [16], and extended to "landscapes of practice" in 2004 [17]. A loose definition of Communities of Practice is "groups of people who share a concern or a passion for something they do and learn how to do it better as they interact regularly." More specifically, three characteristics distinguish a community of practice from other kinds of communities: (1) There is at least one domain of interest shared by all members; (2) members engage in joint activities and discussions, help each other, and share information; and (3) members are practitioners in the domain, not just people with shared interests, and thus develop a shared repertoire of resources.

Communities of Practice show great promise for improving undergraduate retention and diversity, because participating students will find a new source for social encouragement, self-perception, academic exposure, and career perception. In undergraduate degree programs, CoPs are primarily found within disciplinary-specific

extracurricular activities: clubs, meetups, hackathons, and so forth. RadGrad makes these CoPs visible to students, and provides incentives to students for participating in them.

3 RADGRAD

The RadGrad project began in the Fall of 2015. For the first year, we worked on the underlying data model, user interface mockups, and possible game mechanics. Starting in Fall of 2016, we began work on an implementation using the Meteor application framework with Mongo as the back-end database. RadGrad is now a functional web-based application, implemented in approximately 30,000 lines of Javascript and 7,000 lines of HTML. It has extensive design and development documentation, unit and integration tests, and supports continuous integration. Instances can be deployed locally or using a cloud-based hosting service such as Galaxy. We have registered approximately 500 students with the system so far.

Figure 1 shows the RadGrad Degree Plan page which illustrates several capabilities of the system. The left side contains a four by three grid representing four academic years each containing three semesters (Fall, Spring, Summer). For each of the 12 semesters, this hypothetical student indicates not just their CS-related curricular activities (i.e. Courses) but also their CS-related extracurricular activites (called "Opportunities" in RadGrad).
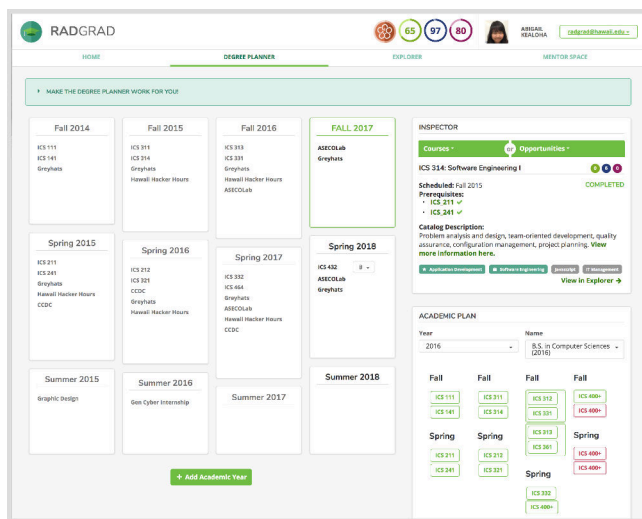

Figure 1: *RadGrad Degree Plan page*

The right hand side has two panes, one providing an "Inspector" and one providing an "Academic Plan" browser. The Inspector allows students to browse the currently available Courses and Opportunities and learn details about them including a description, links to external resources, the number of ICE points that can be earned by completing the Course or Opportunity, and which (if any) of the student's declared Interests are matched by the Course or Opportunity. The Academic Plan browser presents the set of academic plans available to that student (i.e. "B.S. Computer Science", "B.A. Computer Science", etc.) along with a canonical 8 semester plan to complete the coursework for the degree. The Academic Plan shows a course in green if it is present in the student's plan,

and red if it not yet added. A drag-and-drop UI allows students to easily take Courses from the Academic Plan pane and put them into their Degree Plan.

Each page has a documentation pane that is displayed with a light green background and is located just below the navbar. Clicking the triangle on the left side reveals documentation about the current page.

For students, the RadGrad UI is organized into four top-level pages as shown in the tabbed area underneath the navbar. The Home page provides a place for students to update their profile to indicate their current Career Goals and Interests, as well as personal information such as their picture and personal home page. The Explorer page allows students to browse Courses, Opportunities, Career Goals, Interests, and Academic Plans. The Degree Planner page (as show in Figure 1) allows students to design their degree experience as a combination of curricular and extracurricular activities. The Mentor Space page provides a mechanism for students to engage with alumni and local high tech professionals to learn more about career opportunities and other information about professional life.

Figure 2 zooms in on the portion of the navigation bar containing a student's Level (a green RadGrad icon) and ICE Points (three circles containing numbers inside). RadGrad provides a game mechanic consisting of six "Levels", each represented by a different colored RadGrad icon. As students earn ICE Points and carry out other RadGrad activities, such as writing reviews of courses and opportunities, they can progress through white, yellow, green, blue, brown, and black levels, similar to martial art belt colors. The rules for achieving Levels are designed such that a motivated student can achieve a new Level each semester, so a total of six semesters (typically three years) is required to get through all Levels. We print laptop stickers for each Level, and students can get a new laptop sticker free from the department with the color corresponding to each Level as they achieve it. This provides visibility for RadGrad and for the student's participation. The student in Figure 2 is currently at Level 3 (green).
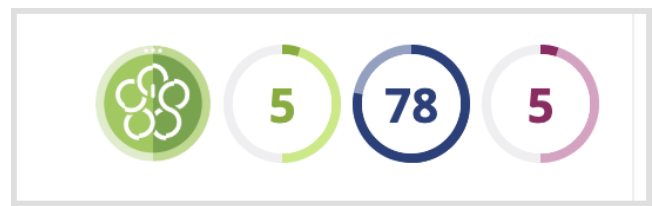

Figure 2: *UI for student Level and ICE Points*

We developed a circular representation for ICE Points to help students quickly understand their progress using RadGrad. Achieving a successful undergraduate degree experience according to RadGrad's game mechanics means that a student must first plan out a sequence of curricular and extracurricular activities that will eventually earn them at least 100 Innovation, Competency, and Experience points. Once they have made that plan (which they are free to change at any time), they must then actually earn the points by completing the curricular and extracurricular activities. To verify completion of a curricular activity, RadGrad imports data from

the University's course management system indicating the coursework the student has taken and the grade they received. To verify completion of an extracurricular activity, the student must first "request verification" of the Opportunity. This request sends a message to the RadGrad administrator or faculty member who has responsibility for checking that the student participated. The admin or faculty member can accept or decline the verification request, and if accepted, the system will award the points associated with the Opportunity to the student.

RadGrad represents ICE by three circles: a green circle for Innovation, a blue circle for Competency, and a red circle for Experience. For each of the three components of ICE, the circle is partially or fully colored in a light shade to represent the number of *planned* points, based upon what the student has put in their degree plan. In Figure 2, the Innovation circle is colored only half way in a light green color, which indicates that this student has only planned to earn 50 Innovation points so far. The blue circle is completely filled in, indicating that the student has planned to achieve 100 Competency points. Finally, the Experience circle is also only half filled, indicating that only 50 Experience points are currently in the degree plan.

RadGrad uses a dark shade on the circle to represent the number of points actually earned so far, and also indicates that number in the center of the circle. So, the student in Figure 2 has so far earned 5 Innovation, 78 Competency, and 5 Experience points.

Finally, RadGrad defines four other roles for users in addition to Student: Admin, Faculty, Advisor, and Mentor. The user interface to RadGrad varies significantly in functionality depending upon the role of the user. For example, users with the Advisor role have the ability to add new students to the system, and see existing student degree plans. Due to space limitations, only a brief overview of functions and capabilities associated with the Student role are described in this section. For more detailed information about the system, please go to https://radgrad.org.

4 EVALUATION

We began beta-testing RadGrad with selected students in 2017, and that uncovered a variety of usability and performance issues. Widespread rollout of RadGrad into our department started in Fall 2018. To assess this roll-out, we conducted a mixed-methods evaluation consisting of the following types of data collection: (1) A user survey administered as part of the RadGrad onboarding session in order to assess student attitudes toward the CS degree program; (2) custom instrumentation within RadGrad to track user behaviors; (3) An end-of-year survey administered to students, and (4) Peer-to-peer interviews led by undergraduate members of the RadGrad project.

Onboarding survey. We introduce RadGrad to students as part of their second semester programming course via a 30 minute demonstration and onboarding during a programming lab period where they all have access to a computer. During the onboarding, we introduce them to the goals and motivation for the system, after which they login and specify their desired career goals and interests, then set up an initial degree plan with courses and opportunities, and finally obtain their first RadGrad laptop stickers depending upon the Level they achieved. At the start of this session, we

request that they fill out an onboarding survey. So far, we have collected 141 responses, which provides useful data on the "baseline" state of students prior to their use of RadGrad. Our first question asked them how they evaluate their progress in the degree program, and students could select one or more of over a dozen possible responses. Only three responses were selected by over 50% of students: Looking at the GPA, having a clear career goal, and having a clear academic plan. Another question asked what activities are important for being a successful computer science graduate, and only two responses were selected by more than 50% of the students: doing internships and participating in professional organizations. Finally, when we asked what factors get in the way of doing the activities that you think are important, the responses that were selected by over 50% of the students were: Lack of time, lack of financial support, lack of information about opportunities, and lack of direction/advice from program.

The onboarding survey provides evidence that RadGrad is useful: it can help students estalish a clear career goal and academic plan, it can help students find internships and professional organizations to participate in, and most importantly, it can address two of the four obstacles: lack of information about opportunities, and lack of direction/advice from the program.

Custom instrumentation. It is clearly important to understand how students use the system, and to support this process we built instrumentation to would monitor user interactions and classify them according to a set of key "behaviors". For example, we wanted to know how many students were completing their degree plan (i.e. planning to obtain 100 Innovation, Competency, and Experience points), how many were Leveling Up, how many were engaged in Planning, and so forth. Once we establish a baseline for these behaviors, we can determine if any behaviors are not occurring appropriately, plan experimental interventions, and measure the results. Figure 3 shows an example of user behaviors for the Spring, 2019 semester:
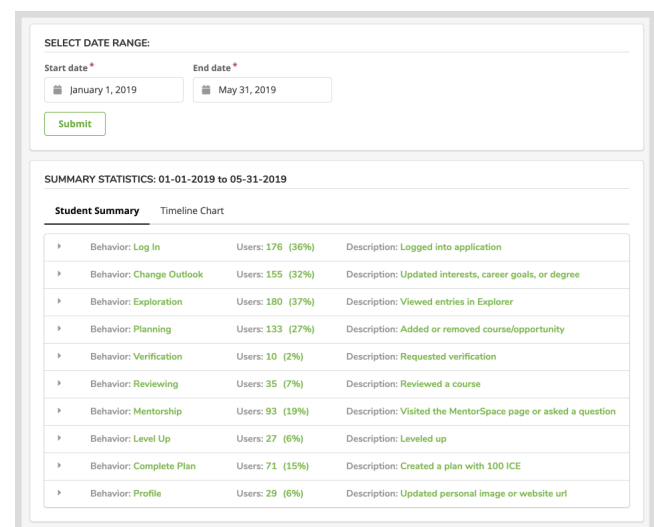

Figure 3: *Behavioral Analytics, Spring 2019 Semester*

As an example of how we use this instrumentation, the low level of "verification" behaviors during the Spring 2019 semester jumps out as a significant issue. If students don't verify Opportunities, they don't earn the associated points, which means they don't Level Up, which means they aren't prompted to explore more sophisticated uses of RadGrad. As a result of this analysis, we are planning a special emphasis on verification for the Fall 2019 semester.

End-of-Year and Peer-to-Peer surveys. Analysis of instrumentation data helps us to understand *what* students do in RadGrad, but it doesn't help us understand *why* students do what they do in RadGrad. And, even more important, it doesn't give us any help in understanding why some students don't use RadGrad after the initial onboarding process. To gain insight into these issues, we have instituted an ongoing mixed-methods data collection process with two formats. The first is an online end-of-year survey that we ask students to fill out, and the second is a "Peer-to-Peer" survey in which undergraduates participating in the RadGrad research project conduct face-to-face interviews with fellow ICS students to gather information on their issues with RadGrad. We plan to conduct these surveys each year. For AY 2018-2019, 28 students participated in the end-of-year survey, and 13 students participated in the Peer-to-Peer survey, for a total of 41 student responses.

In both forms of surveys, we asked students if they found the Opportunities available in RadGrad to be useful. 72% of the students who used RadGrad found them to be useful, and most said they learned about new Opportunities from the system. One student said, "There are a lot of internships and programs out there that I had never heard of", and another said, "the sustainability opportunities really jumped out at me".

The Peer-to-Peer survey revealed significant issues related to diversity and inclusion. Approximately 75% of students responded in the affirmative to the question of whether or not the Department has diversity and/or inclusion issues. Comments included: "There is a lack of female representation. There is a lack of appealing to interests outside of the typical programming experience", "There is a definite bias for males", "If you sit in on a 400-level class, there is a noticeably smaller percentage of female students as opposed to the 100 and 200 level classes.", "The atmosphere was very non-inclusive. Even when I tried to join the existing clubs, I felt that they were cliquey."

Finally, both surveys provided useful new insight into ways to make RadGrad more useful and easy to use. Students suggested new Opportunities that they had experienced and wanted to communicate to other students through the system. System enhancements included sorting and filtering options for Opportunities, messaging to other students, and language localization.

After just one year of deployment, it is too soon to empirically evaluate the precised degree to which RadGrad improves engagement, diversity, and retention, but we believe our evaluation does provide evidence that RadGrad can play an important role in achieving these goals.

5 LESSONS LEARNED

Based upon the above data and other experiences, we believe it is useful to present several "lessons learned" from this first year of use that might apply to any group developing initiatives similar to RadGrad:

FERPA compliance limits social network capabilities. The Family Educational Rights and Privacy Act of 1974 (FERPA) [13] is a federal law that protects the privacy of student education records. In general, FERPA prohibits release of student data beyond "directory information", such as a student's name, address, telephone number, date of birth, and dates of attendance. Early versions of RadGrad provided a variety of social networking features intended to support the development of "communities of practice", including automatically allowing students to see the Courses and Opportunities associated with each other's degree plans. The current version of RadGrad implements a more restrictive approach, in which students must explicitly opt-in to sharing FERPA-restricted content such as the coursework that they've completed (and, furthermore, they can later opt-out and the system will stop displaying their data to others.) We are still exploring the impact of an opt-in policy on the mechanisms for community building in RadGrad.

Integration with formal academic advising is complicated. During the development and beta-testing of RadGrad, we received strong encouragement from our department academic advisors, who saw RadGrad as fulfilling an important need in the advising process. In fact, support was so strong that the advising staff recommended that RadGrad be integrated into the mandatory department advising process. We now realize that, at least at our institution, this is more difficult than anticipated due to time and resource constraints. Our advisors are required to meet with all 500+ CS majors (and prospective majors) every semester. This means that these meetings are generally less than 15 minutes, and all of that time can easily be taken up with University general degree requirements and other issues. In the coming year, we are exploring alternatives, including peer advising services and integration of RadGrad into computer science club meetings.

Engagement, diversity, and retention are cultural, not technological. We want to be clear: no "app" can single handedly solve the problems of engagement, diversity, and retention in computer science. Instead, these problems can only be addressed by a constellation of behavioral and attitudinal changes from students, faculty, and staff, and that these changes cannot happen in a few months. That said, we are finding that RadGrad provides a useful starting point for such conversations around culture by providing a different framework for evaluating the undergraduate degree experience.

6 FUTURE DIRECTIONS

In the next two years, we plan to make progress along the following fronts of interest to the SIGCSE community. We welcome collaborators: please contact us if you wish to become involved!

RadGrad 2: a generic framework to tailor RadGrad to other departments and institutions. The current RadGrad platform is open source and available for download, but the implementation contains shortcuts that make it difficult to adapt to other institutions. We are committed to removing these dependencies in the next major release of the system, so that other institutions can more easily evaluate the system. We have already made significant

progress, and expect RadGrad2 to be available in the first quarter of 2020.

CSExplore: expanding the pipeline by focusing on high school and first year students. One of the most exciting outcomes of our first year of use is the recognition that the underlying data in RadGrad, built and managed by the actions of undergraduate CS majors and CS faculty, could be repurposed in a form that can provide novel insight into computer science degree programs for high school and undeclared college students. This has the potential to have a significant impact on diversity, since high school and the early, pre-major college years are the most effective times to "expand the pipeline". We are currently in the early design stages of this new system, which we are calling "CSExplore". The idea is that instead of focusing on helping a CS major obtain a more well-rounded CS degree experience, CSExplore would focus on helping younger students gain insight into the wide spectrum of applications of computer science, the range of activities they could participate in as a CS major, and the activities of current students in our degree program. To be blunt: too many high school students think computer science is just for video game design. CSExplore can help them to see that this skillset can apply to so much more.

Longitudinal data on engagement, diversity, and retention. We are collecting data on the demographics of students in our program, as well as whether or not they use RadGrad and how they use it. After the next two years, we expect to have enough data to start making empirically supported conclusions regarding the impact of RadGrad on engagement, diversity, and retention.

ACKNOWLEDGMENTS

REFERENCES

[1] Michael Abebe, Brad Glasbergen, and Khuzaima Daudjee. 2019. WatDFS: A Project for Understanding Distributed Systems in the Undergraduate Curriculum. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*. ACM, New York, NY, USA, 920–926. https://doi.org/10.1145/3287324.3287473 event-place: Minneapolis, MN, USA.

[2] Catherine Ashcraft and Anthony Breitzman. 2012. *Who invents it? Women's participation in information technology patenting.* Technical Report. National Center for Women and Information Technology. http://www.ncwit.org/sites/default/files/resources/2012whoinventsit_web_1.pdf

[3] Jerri Barrett. 2017. Expanding the Pipeline: Key Learnings on Retaining Underrepresented Minorities and Students with Disabilities in Computer Science. https://goo.gl/CQfy2K

[4] Tracy Camp. 2017. *Generation CS: Computer Science Undergraduate Enrollments Surge Since 2006.* Technical Report. Computing Research Association. https://cra.org/wp-content/uploads/2017/02/Generation-CS.pdf

[5] Mathilde Collain and Deborah Trytten. 2019. "You Don'T Have to Be a White Male That Was Learning How to Program Since He Was Five". In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*. ACM, New York, NY, USA, 968–974. https://doi.org/10.1145/3287324.3287383 event-place: Minneapolis, MN, USA.

[6] Debzani Deb, Muztaba Fuad, and Keith Irwin. 2019. A Module-based Approach to Teaching Big Data and Cloud Computing Topics at CS Undergraduate Level. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*. ACM, New York, NY, USA, 2–8. https://doi.org/10.1145/3287324.3287494 event-place: Minneapolis, MN, USA.

[7] Cristian Dezso and David Ross. 2007. *Girl Power: Female participation in top management and firm performance.* Technical Report. University of Maryland. http://bit.ly/2WZH1fa

[8] Hai Hong and Abby Bouchon. 2014. *Women who choose computer science: what really matters.* Technical Report. Google, Inc. http://bit.ly/2x8r8nC https://static.googleusercontent.com/media/edu.google.com/en//pdfs/women-who-choose-what-really.pdf.

[9] Google Inc. 2016. *Diversity Gaps in Computer Science: Exploring the Underrepresentation of Girls, Blacks and Hispanics.* Technical Report. Google, Inc. http://bit.ly/2KvGYkC

[10] Nancy Kober. 2015. *Reaching Students: What Research Says About Effective Instruction in Undergraduate Science and Engineering.* National Academies Press. https://doi.org/10.17226/18687

[11] Jean Lave and Etienne Wenger. 1991. *Situated Learning: Legitimate peripheral participation.* Cambridge University Press.

[12] Heather E. Metcalf, Tanya L. Crenshaw, Erin Wolf Chambers, and Cinda Heeren. 2018. Diversity Across a Decade: A Case Study on Undergraduate Computing Culture at the University of Illinois. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education (SIGCSE '18)*. ACM, New York, NY, USA, 610–615. https://doi.org/10.1145/3159450.3159497 event-place: Baltimore, Maryland, USA.

[13] Family Policy Compliance Office. 1974. Family Educational Rights and Privacy Act. https://www2.ed.gov/policy/gen/guid/fpco/ferpa/index.html

[14] Jane Stout and Tracy Camp. 2014. Now what? Action items from social science research to bridge the gender gap in computing research. *SIGCAS Computers and Society* 44, 4 (Nov. 2014). https://cra.org/cerp/wp-content/uploads/sites/4/2015/09/Stout-Camp-2015.pdf

[15] Cara Tang, Cindy S. Tucker, Christian Servin, Markus Geissler, and Melissa Stange. 2019. Shaping Curricular Guidelines for Associate-Degree Cybersecurity Programs. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE '19)*. ACM, New York, NY, USA, 334–335. https://doi.org/10.1145/3287324.3287516 event-place: Minneapolis, MN, USA.

[16] Etienne Wenger. 1998. *Communities of Practice: Learning, Meaning, and Identity.* Cambridge University Press. Google-Books-ID: Jb8mAAAAQBAJ.

[17] Etienne Wenger, Mark O'Creevy, Steven Hutchinson, Chris Kubiak, and Beverly Wenger-Traynor. 2004. *Learning in landscapes of practice.* Routledge.