Ready to Work: Evaluating the Role of Community Cultural Wealth during the Hiring Process in Computing

Stephanie Lunn
School of Computing and Information Sciences
Florida International University
Miami, United States
slunn002@fiu.edu

Monique Ross
School of Computing and Information Sciences
Florida International University
Miami, United States
moross@fiu.edu

***Abstract*—As computing jobs continue to proliferate, it is necessary to understand what traits and skills graduates need to succeed long term. This entails not only defining what employers expect from job applicants, but also what gaps they observe from recent hires. Moreover, it requires consideration of the cultural wealth that individuals contribute to the workplace, and how hiring practices may affect populations minoritized in computing — women, Blacks/African Americans, and Hispanics/Latinxs. This paper describes the results of a systematic literature review conducted to explore the hard (technical) and soft (non-technical) skills assessed during the hiring process, and to examine which are considered tantamount for professional success. In addition, we seek to understand how the inherent capital of different populations may be leveraged during the hiring process, and long term in the workplace. Our study presents an anti-deficit approach for the imperative of improving diversity in computing, and to bridge the gap between cultural backgrounds, academic curricula, and the needs of industry. These findings are intended to inform pedagogical practice, potential employers, and to encourage students to persevere despite the challenges inherent in the hiring process in computing.**

***Index Terms*—Hiring in computing, computing jobs, knowledge deficiency, diversity in computing, cultural wealth**

I. Introduction

Obtaining a position in computing requires job seekers to demonstrate a range of competencies, knowledge, and abilities. Not only are they required to possess technical prowess, but also to have a good personality, a propensity for critical thinking, an aptitude for learning, and strong written and oral communication capabilities [1], [2], [3], [4]. To determine if a computing candidate possesses the necessary skills, companies often use technical interviews within the hiring process.

Technical interviews are common for technology positions, and refers to the specialized and rigorous assessment of job candidates' capabilities, using live coding, logic, and problem solving [5]. Candidates are expected to not only demonstrate knowledge of computing fundamentals, but also to answer complex programming questions. Throughout the real-time coding, they are also encouraged to talk through their solution, justify their thinking, and to give consideration to algorithmic performance and testing. Unsurprisingly, students report that when trying to start their career, technical interviews are a major challenge [6]. Furthermore, these interviews rarely consider their own inherent bias, placing undue emphasis on certain skills while neglecting the cultural backgrounds and wealth that diverse students can contribute to a workplace.

Considering the rigors associated with obtaining a degree in computing, it is important that companies examine how current evaluation methods may deter groups that are already underrepresented— including women, Blacks/African Americans, and Hispanics/Latinxs. The computing workforce in the United States is, on average, 72.1% White, 15.94% Asian, 9.46% Black/African American, and 8.3% Hispanic/Latino [7]. Although men comprise 75% of those hired, when considering the intersectionality of gender and race/ethnicity in computing occupations, 56% are White women, 32% are Asian women, 7% are Black women, and 5% are Latinx women [8]. Furthermore, among 177 companies in Silicon Valley, Black women, Latinas, and Native American/Alaskan Native women comprise less then 2% of the total workers.

Companies and organizations are aware of the skewed representation in computing, and they are taking steps to ameliorate the situation [9], [10], [11]. Google established the Computer Science Summer Institute (CSSI), a "camp" to engage high school seniors in the discipline earlier [1]. They also offered a branch specifically for Historically Black Colleges and Universities (HBCUs) to help students build their support network and to increase retention in the field. In January 2021, Girls Who Code held their inaugural career fair event to connect the community with job opportunities [12]. In addition, companies may send recruiters to conferences intended to provide networking and professional development for minoritized populations such as the National Society of Black Engineers[2] (NSBE), the Society of Hispanic Professional Engineers[3] (SHPE), and the Grace Hopper Celebration[4].

[1]https://buildyourfuture.withgoogle.com/programs/
[2]https://www.nsbe.org/
[3]https://www.shpe.org/
[4]https://ghc.anitab.org/

978-1-6654-4905-2/21/$31.00 ©2021 IEEE

However, given the continuing discrepancies in representation, more needs to be done, and it is important to consider the role that the hiring process may play in perpetuating inequalities.

Given that diverse students may have unique experiences and backgrounds that make them more adept in a number of areas salient for long-term success in computing, we sought to understand what skills are valued during the hiring process, and which could be leveraged. Using the community cultural wealth model as a lens to explore the inherent capital that female, Black/African American, and Hispanic/Latinx students contribute, we worked to answer the following research questions (RQs): *1. Which hard skills do employers assess during hiring for computing roles?*; *2. Which soft skills do employers assess during hiring for computing roles?*; and *3. How could the hiring process be refined to better leverage community cultural wealth for groups underrepresented in computing?*

To address these questions, we performed a systematic literature review (SLR) of the hiring process in computing. Not only did we take an inventory of present publications describing hard and soft skills for computing professions, but we also present an analysis of which skills are evaluated during the hiring process. Then, we discuss the importance of diversity in hiring, and offer suggestions on ways to make the process more inclusive. One of the main goals in this work is to raise awareness of the assets that women, Blacks/African Americans, and Hispanics/Latinxs can contribute, and how approaches can be shifted to make hiring more inclusive.

In this document, we will discuss the background related to the work in Section II, and the theoretical framework driving this research in Section III. Then, we describe our review process in detail, in Section IV. We give an overview of the results from the SLR in Section V, and provide a discussion on what this may mean in Section VI. We address any remaining threats to the validity of this work in Section VII, and then present our conclusions in Section VIII.

II. Background

Employability is defined as a student's characteristics, in terms of their skills, comprehension, and personal attributes, which make them the most likely to obtain a job or to achieve career success [13]. Propelled by the needs of the workforce, as well as regional, national, and supranational agencies, universities are increasingly under pressure to find a balance between education and professional training, and employability is often linked to performance metrics [14], [15], [16], [17]. Typically hiring in computing involves seeking candidates with a range of hard and soft skills [18], [2], [3].

Hard skills are considered the skills specific for a particular "work setting," and are learned through education or during the time on a job [19]. Although the exact categories may vary, some examples include software engineering methods, source code management, system administration, user interface design, and embedded, mobile, testing/quality, and web development [3]. Meanwhile, *soft skills* arise from an individual's personality, attitudes, behaviors, and ability to interpret social cues, to communicate, and to interact with others with

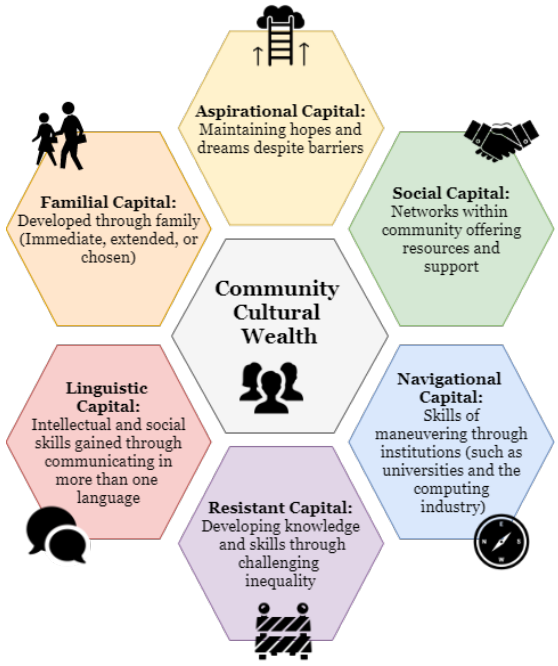

Fig. 1: Community cultural wealth model

emotional intelligence and empathy [19]. Some examples of soft skills include displaying an interest in learning, working cooperatively with others, possessing analytical skills, and being able to manage time effectively [20], [21], [4].

One of the more comprehensive skill categorizations identified was described by Calitz et al. for information and communications technology (ICT) positions. Their breakdown included necessary hard and soft skills requirements such as [22]: problem-solving skills (such as creativity, research skills, logical thinking, and working under pressure), interpersonal skills (such as conflict resolution and teamwork), work ethic, language skills, business processes, management skills, internationalization skills, project management, strategy skills, social network skills, software development, business applications, current languages (e.g., Java, C/C++, C#), legacy languages (e.g., Ada, COBOL), and mobile technologies.

III. Theoretical Framework: Community Cultural Wealth

To better understand our findings, we used the *Community Cultural Wealth (CCW) model*, illustrated in Figure 1. Preliminary versions of the CCW model were developed using critical race theory as a foundation by Sólorzano, Villalpando, an Oseguera [23] to discuss the oppressive practices that may create barriers for marginalized groups. The concept was additionally developed by Yosso [24] to further expand and define different types of capital that exist for people of color, as defined within their own community. Specifically, Aspirational, Linguistic, Familial, Social, Navigational, and Resistant capital are combined to produce the CCW framework.

Aspirational capital refers to the ability to maintain hopes and dreams despite barriers [24]. *Social capital* is described by the networks of people within the community, which

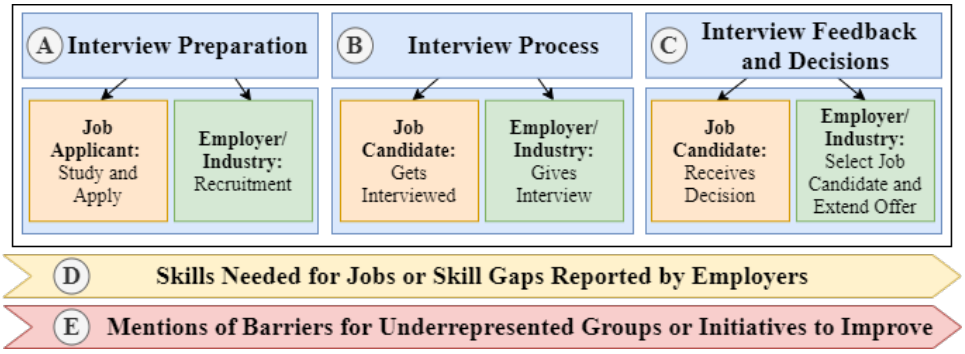

Fig. 2: Overall hiring process with consideration of skills needed

offer resources and support. *Navigational capital* considers the skills required to maneuver through social institutions (like universities, or the computing industry). *Resistant capital* refers to developing knowledge and skills through oppositional behavior challenging inequality. *Linguistic capital* refers to the intellectual and social skills gained through communication in more than one language and/or style. *Familial capital* refers to the cultural knowledge nurtured among family (either immediate, extended, or chosen) that carries a sense of community history, memory, and cultural intuition. It should be noted these are not discrete categorizations, and overlap can occur.

Previously, CCW has been demonstrated as an effective tool for considering the benefits of external contributors and extracurricular support systems to empower minoritized populations. In addition, work on students in STEM fields has shown it can also be a powerful approach for curricula development and teaching [25]. Faculty that encourage students to take advantage of their office hours to offer professional and academic advising, encouragement of internship and research, and other forms of social support can strengthen the social capital that establishes "supportive family-like relationships." This tends to also touch on familial capital, and may also provide a form of navigational capital, as the advising helps students to maneuver through their university. Additionally, instructors are recommended to apply an asset-based approach, like CCW, when considering the projects and examples that they give to students, to help foster additional capital. In this work, we consider how such ideas can extend to the hiring process in computing.

IV. Methods

Systematic literature reviews are used to parse through the existing work on a topic, to assess papers related to a phenomenon of interest, and to identify answers to specific questions, as well as areas where there may be gaps [26], [27]. We sought to examine existing publications to see what skills are valued and evaluated during hiring. Additionally, we explored how companies considered diverse skill sets during computing interviews, and initiatives to consider diversity or make the process more inclusive. In order to describe the hiring process in computing, we followed the procedure previously described by Petticrew and Roberts [26], and through application of the additional principles for SLR in software engineering described by Kitchenham and Charters [28].

A. Source Selection and Search

An ad hoc review was first undertaken to assist with the development of search strings and to develop selection criteria [29]. Publications were expanded beyond journal articles and conference proceedings to also consider books since they proved the most thorough resource for both job applicants and employers conducting job interviews and included information gathered from multiple technology companies to inform what hiring looked like overall. Also, because the hiring process has changed over time and we only wanted to consider more recent practices, the timeframe for publications was limited to papers from the last decade (2010 or later). In addition, all research, books, journal and conference proceedings were examined manually to guarantee adherence to the inclusion/exclusion criteria. Specifically, we considered the areas pertaining to computing hiring, the skills needed for jobs, and attempts to make the process more inclusive as depicted in Figure 2.

As shown in this figure, we considered both the job applicant or candidate and the employer or industry perspectives. We looked into interview preparation (A), the interview process itself (B), and mentions of job feedback or offers (C). We also examined papers that mentioned interviews or hiring in reference to the desired skills of candidates (D). Additionally, we considered mentions of diversity issues in interviews or attempts to make the hiring process more inclusive (E).

In the spring of 2020, publications indexed in the following 4 databases were searched 1) Google Scholar (GS), 2) IEEE Xplore Digital Library, 3) ACM Digital Library, and 4) conference proceedings from ASEE. To keep the literature focused on the desired areas, search strings were created to query the selected sources. The following contained the relevant terms for the hiring process:

((Computing **OR** Technical **OR** Software Engineer* **OR** Software Develop*) **AND** (Interviews **OR** Hiring **OR** Occupations **OR** Jobs))
OR
(Diversity **OR** Inclusivity) **AND** (Computing Interviews **OR** Technical Interviews **OR** Software Engineer* Interviews **OR** Software Develop* Interviews)

Inclusion	Exclusion
Publications that address the RQs	Publications not in English
Publications displaying empirical evidence	Publications based solely on personal opinions
Publications that clearly define their methods/sources and books where the author had experiences or credentials	Publications without clear methods or books where the author's qualifications or credentials were not made explicit
Publications from journals, conference proceedings, books, or theses	Publications from corporate reports, blogs, patents, or opinion articles
Publications that discuss the hiring process in computing	Publications where computing fields are not the primary focus

TABLE I: SLR inclusion and exclusion criteria

One issue encountered was the stopping conditions in the databases queried. For example, GS often lists hundreds of thousands of results. Due to the manual nature of the review process, it was not possible to analyze each result. However, we used the databases' internal ranking algorithms to restrict the number of results to the most relevant. This is consistent with the heuristics used by others conducting SLRs [30].

B. Study Inclusion and Exclusion Criteria

The search strings generated a lengthy list of sources, that often were not related to the research questions. To filter from title to abstract to content, we applied a list of additional criteria, illustrated in Table I. Given that many accounts of the hiring process were published in books, we included them in the search so long as the author had firsthand experience with the hiring process, or their credentials as experts on the topic could be substantiated.

Technical interviews and computing positions tend to fall under a wide umbrella of topics and industries, and often roles exist within international companies. Given the evolving nature of the how interviews are conducted, and the range of fields that computing can include, it is necessary to continue to evaluate industrial needs and educational training, and to approach skills required with a worldwide perspective. For the purposes of our inquiry, we considered computing according to global descriptors as encompassing: software engineering (SE), computer science (CS), information technology (IT), information systems (IS), computer engineering (CE), or ICT. As such, any publication that discussed the hiring process in one of these fields was included for consideration. We also included publications that focused on computing sub-specializations such as cybersecurity or data science.

C. Study Execution

An overview of the SLR paper identification process is shown in Figure 3. After executing all search strings and manually searching through all selected sources, 6,890 publications were found (A). Then duplicates were removed, resulting in 5,640 publications (B). After sorting through the titles to identify relevant abstracts, 461 remained (C). Then the abstracts for these publications were read for applicability, leaving 237 (D). Each of the full text publications were read in their entirety, and after applying the inclusion and exclusion criteria, 64 remained (E). We also applied snowballing, examining the references from the publications as well as those that cited the publications, to maximize relevant sources on the topic [30]. We identified 41 additional publications from snowballing (F), resulting in a total publication count of 105 (G).

It should be noted that during the snowballing phase, it took three rounds of forward and backward snowballing to reach a point where there were no new relevant sources were identified. Also, the publications identified through snowballing were subject to the same process as the publications identified from the databases. Based on titles, we identified a total list of the 232 potential publications. Duplicates were removed, resulting in 144. Then the abstracts were read and the inclusion and exclusion criteria were applied, resulting in 110 examined in full. From these, only 41 remained applicable, and these were added to the 64 from the databases.

V. Results

The full list of publications is available at: https://bit.ly/3hVSFhq. Among these 105 publications, 16 were books, 40 were conference proceedings, 11 were theses, and 38 were journal articles. However, the majority of the papers focused on the skill gaps and knowledge deficiencies of job candidates or recent hires (93 in total), and far less focused on interviews themselves. When considering interview components, 51 publications touched on interview preparation, 31 on the interview process, and 18 on interview feedback or decisions. In addition, only 18 publications considered gender and 11 publications touched on race/gender in the hiring process, or discussed attempts to make the process more inclusive.

VI. Discussion

The breakdown of existing publications reveals that when evaluating the hiring process in computing, most often, scholars focus on skills employers want, and there is limited work conducted with consideration to diversity. Several papers did remark on gender, race, or ethnicity and offered suggestions to make the process more inclusive, or focused on the obstacles encountered/employer practices [31], [32], [33], [34], [35], [36], [37], [38], [4], [39]. Yet, relative to the larger pool of papers identified, only a few authors include women or racial/ethnic minorities as the primary populations studied [10], [6], [40], [41], [42], [43], [44], [45], [46], [47]. This represents a space that should be filled, and demonstrates a

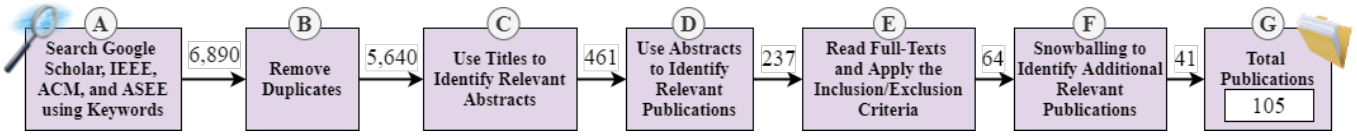

Fig. 3: SLR paper selection process

need for computer science education researchers to consider diverse populations when examining the hiring process.

A. Hiring Process in Computing

The hiring process in computing can vary based on the values, culture, and organization of the company; which may be classified as start-ups, mature consumer-facing companies, enterprise-oriented companies, government contractors, or embedded systems/chip design companies [48]. However, the process typically includes some combination of behavioral, technical, and problem-solving rounds. In addition, job candidates may have to complete a task or take home assignment. The exact number of questions, duration, and style are variable, but often includes assessment(s) via phone/video calls and on-site, before feedback is given or decisions are made [5].

When a technical challenge is given, programming questions are often expected to be completed on a whiteboard, with a paper and pencil, or through a simple text editor, and candidates are expected to also give consideration to algorithmic performance, and how their code can be tested [49], [5], [50], [34]. Also, pseudocode is insufficient, instead code is requested in a specific language [51]. Job candidates are also expected to effectively communicate while they solve the problem, as hiring managers want to see how candidates approach the question to evaluate their abilities [32], [52].

To prepare for these interviews, students are presumed to prepare weeks, or even years in advance [53], [5]. However, these expectations do not account for students' additional commitments, such as coursework, working other jobs, dealing with a personal recurring health issue, or caring for a family member. This can create an unequal divide between those with a greater availability to prepare, relative to those students which have limited time as a result of these other factors.

B. Hard and Soft Skills Desired in Computing

The skills and background preferred for technical roles differs widely by the position. For project manager jobs, as an example, apart from valuing leadership and communication, certain companies may favor candidates with backgrounds in business in addition to engineering and/or computer science [54]. For information technology positions, certification can be a positive way for candidates to demonstrate skills acquired [55]. However, in other branches of computing, certifications are treated neutrally, or even as a negative, reflecting that a candidate may be overly specialized [56], [13], [57], [5].

When considering software development roles, Agile is often considered the preferable model by industry, although recent graduates mention that it was infrequently covered, or not discussed at all during their education [58]. Being able to design and test software are also considered important for long term success [59], [2]; however, software testing was repeatedly mentioned by employers and recent graduates as a knowledge deficiency [3], [58], [60]. Interestingly, although testing may be valuable, companies in computing are roughly six times more likely to search for coders than they are for testers [59]. Furthermore, coding and programming are considered important to a computing career in the abstract, but to be successful in industry it is also important to think creatively, to write well organized code, and to be able to see both the broader pictures and more specific details [61], [62]. Accordingly, hiring personnel often look to soft skills as well.

In particular, teamwork and communication are considered tantamount to success [63], [21], [52], [64], [3], [58], [65], [4], [41]. Internationally, being proficient in written and verbal English is also considered highly important since it is the most widely used language in computing fields [63], [66]. Non-native speakers applying for computing positions are encouraged to practice everyday, either in front of a mirror or by watching television with English captioning [66].

Company-fit is also considered important during hiring and may include assessing items such as a candidate's preferences for [36]: the working environment, career growth (opportunities for upward mobility), if they prefer start-ups or established organizations, the products (do they believe in the product or have personal issues in opposition to them, e.g., tobacco), industry leadership, training policies (like educational reimbursement), competing local employers, the profitability of the company, cash flow, recent layoffs, recent mergers and acquisitions, and the overall corporate culture (dynamic and chaotic versus more relaxed). Unfortunately, many employers also utilize not "fitting into the culture" as a way of denying positions to more diverse candidates [67], [68].

Although it is evident that both hard or soft skills are considered during job interviews, there is a heated debate about which is more important to being successful in the workplace. A study by Exter et al. [69] used a mixed-methods approach to survey computing professionals, with anywhere from 1 to 40+ years of experience across industries and countries, and they observed that universally critical thinking, problem solving, and continued lifelong learning were all skills necessary for professional software development and engineering. Interestingly, they also found that although specific models and programming languages were considered important, it was the broader concepts relevant to many different languages and software engineering principles that were considered essential, such as understanding object-oriented design. Also, although universities covered assembly and specific algorithms, these were not especially relevant to the actual work completed in computing careers. These findings demonstrate that although academia does not need to be informed by the current trends and demands of industry, perhaps greater communication between them should be considered to enhance students' employability and to provide better preparation for the workplace.

C. Knowledge Deficiencies in Interviews

Literature frequently mentions a disconnect between what students learn in schools, and their performance in computing interviews and/or careers [70], [65]. From the perspective of academia, the best way to resolve the skill gap is to increase industry collaborations and internship opportunities for students [4]. However, students, recent graduates, and industry tend to have another view on these knowledge deficiencies [58], [20].

Typically, students and recent graduates attribute a skill gap to education placing a heavier focus on theory rather than practical experience [58], [20]. Meanwhile, 32% of employers believe that the skill gap is the result of education and/or changing technology [20]. In fact, when comparing graduates from universities to graduates from paid multi-week coding bootcamps some hiring managers reported that although the graduates from universities had a better grasp at the theoretical underpinnings, the graduates from bootcamps tended to have greater familiarity with current software programs and hardware, making them a greater asset [62]. Furthermore, although most of the bootcamp graduates had college degrees in fields other than computing, they managed to not only acquire savvy in code development, but also in the realm of communication, and an ability to better handle critical feedback.

Although academia is unable to modify its curriculum to support rapidly changing programs, languages, and technologies as fast as a shorter duration bootcamp, these findings hint at a great underlying issue that perhaps academia needs to reconsider its approach [62], [33]. While the way industry measures success may need refining, institutions need to also accept that they may not always offer adequate training. If the ultimate goal of universities is to prepare students for a career in the field, at the bare minimum schools need to consider offering opportunities and workshops for students to gain the type of exposure they need to be competitive in the job market.

In part, it is difficult for universities to adequately prepare students if their own staff are unaware of what hiring in computing entails. Previous research has demonstrated that computer science professors at Historically Black Institutions may lack experience with technical interviews themselves [10]. Accordingly, they are unable to understand the components required nor the anxiety they can induce. While this work may have been conducted at a historically Black institution, the principles are true for professors at all universities. Offering faculty workshops, or encouraging professors themselves to undergo the interview process "in the name of research" could prove beneficial to understanding what their students go through, and could provide insight into areas they could improve their own course design to address shortcomings.

Furthermore, teaching interview skills and offering students opportunities to practice can be beneficial to improving communication and could enable them to test different strategies in a low pressure environment. Previous work integrating a practical course as part of the Bachelor of Information and Communications Technology (BICT) degree at a university in New Zealand yielded some positive insights [71]. The course itself offered students information about what to expect from the interview process, typical formats and questions, and other tips on presentations and non-verbal communication. It also included a role play session, simulating technical interviews. Although initially the students reported feeling anxious before the interview role play, as the mock interviews wore on, the students noted they began to apply the lessons learned from the preparatory process to feel more confident in their answers. As weeks passed, the students reflected on the simulation experience and they emphasized how beneficial it had been. Even though this is not a component typically offered with computing degrees, it provides a useful model to what such a course, or addendum to a capstone project could look like.

D. Valuing Diversity

Despite many reports that "hiring is broken," few papers address how to remedy the process, and even less discuss how the process may discourage women, Blacks/African Americans, and Hispanics/Latinxs. Reports note that even when these groups are represented at a company, they are often in roles other than software engineers [6]. To broaden participation in computing, it is necessary to recruit a diverse set of candidates and to consider inclusive practices supportive of marginalized groups. As we mentioned, companies have taken steps towards doing so, however, there is still more to be done to achieve representation more reflective of the general population. It is worth approaching the hiring process with a critical eye and asking, "How do current hiring practices overvalue certain, narrowly-defined ways of knowing as 'hard skills' vs. 'soft skills'?" Furthermore, consideration should be given to how this binary benefits and harms certain cultural identities, and limits the diversity of ways to demonstrate value to a company.

While this SLR emphasized that there is a lot variability in what companies may want, and which skills they value, several important considerations should be taken away. For example, despite differences in the processes and demands across industries and roles, communication and teamwork are widely considered necessary [63], [21], [52], [64], [3], [58], [65], [4], [41]. Given their value, it is imperative for educators and employers to think about how they can help students and employees develop these skills, and leverage their own capital for long term success in computing. Now we will tie the hiring process to opportunities to honor cultural wealth.

Social Capital: In professional settings pair programming has been shown to have several benefits [72]. Not only does it result in higher quality code with fewer bugs, but partners often learn from each other, leading to better understanding. Diversity in thought has also been shown to enhance the work developed by the team, as partners contribute unique perspectives to a given task. Likewise, universities can apply this approach to programming coursework, including pair-programming projects or assignments, and potentially rotating teammates to increase exposure to disparate approaches and thinking. Building these relationships with others in their classes, whom they may otherwise be uncomfortable approaching, could help students develop their ties within the computing community, which could improve social capital, and build support they lean on during the hiring process. In addition, assigning students to diverse groups for projects, rather than allowing students to choose their own teammates, is recommended to more accurately reflect scenarios encountered in the workplace [47].

Even before hiring begins, employers should seek to offer career mentoring or internships opportunities to underrepresented groups to encourage occupational trajectories in

computing [73], [74]. Alternatively, co-operative education programs (or "co-ops") are similar to internships, but typically last longer, and involve a partnership with an educational institution to offer academic credit for work completed with a company [75]. Although it may require a greater investment of time and resources, co-ops lead to job candidates with improved preparation and technical acumen.

Furthermore, purposeful leadership, and placing women and/or minorities as members on industry boards can also lead to more diverse mindsets, leveraging human capital, and could encourage representation of others within the company as well [76]. It is also important think about the staff representing the company. Mahmoudi mentioned that typically the human resources managers that greeted candidates for the on site visit were female. However, later interviews during the course of the on site were typically all White or Asian males, and usually more senior employees [45]. To demonstrate that a candidate may be able to find a community within a company, and develop their own social capital, companies should diversify those conducting the interviews.

Familial Capital: Prior literature has described how resumes can be used to perpetuate inequity in hiring across industries and occupations, and have highlighted how employers' inherent bias may result in neglecting qualified applicants based on presumptions about their identification with a particular gender, racial, or ethnic group [39]. To mitigate discrimination during application screening, programs have been developed (such as Blendoor[5]) to remove personal information from resumes with the intention of encouraging employers to focus on applicant's skills. However, since adaption of such technology may not be widespread, companies should consider other avenues for locating qualified candidates, and particularly those who may otherwise be overlooked based on human judgements about their name or the academic institution they attended. It has been noted that often the best candidates are obtained through peer referrals, where a friend, mentor, or colleague can vouch for the skills of a job applicant [51]. By tapping into the networks of trusted employees, companies may be able to locate adept applicants that have the capability to succeed in the role, but that may have been neglected while parsing through resumes as a result of human or algorithmic bias. Also, leveraging familial capital through hiring an employee's immediate, extended, or chosen family may lead to better workplace relationships, increased collaborations, and higher productivity. However, it can be a slippery slope to perpetuating nepotism and additional inequity, and while such channels may be beneficial for identifying applicants, all job candidates should be subject to the same interview process to assess hard and soft skills.

Navigational Capital: Institutional agents such as faculty have been shown to play an important role in recognizing students' inherent capital, and encouraging them to succeed [25]. To ensure optimal performance during interviews, hiring managers and recruiters should consider their own impact and should make an effort to provide positive support and encouragement. This is important since previously critical, harsh, or condescending interviewers have been shown to leave a lasting impact on candidates [33]. Interviewers must be cognizant of their influence, and rather than appearing stoic or overly trenchant, could offer support in the form of tips, and positive reinforcement to help bolster students' navigational capital as they solve difficult problems during technical interviews.

In the interview itself, it may be important to consider practices that ensure a candidate has the capability of performing the job, without requiring intensive practice to prepare. Whether a candidate is fresh out of school or more advanced, they should be treated the same and the approach should be equal. Rather than offering a rigorous programming test to assess their abilities, a better approach would be to ask questions or problems that employees at the company recently solved. This method ensures that irrespective of the candidate's experience level, they can demonstrate their approach and problem solving acumen in a concrete way for the types of things they would need to do in the role [77]. After it has been established that the candidate does possess the foundations to perform the job, there are other ways to gauge skill that could be more equitable to assess hard and soft skills like creativity, critical thinking, and communication.

Another option both for universities and companies is to consider collaborating to offer opportunities for mock interviews, which can help to reduce job applicants' anxiety, strengthen communication skills, and can help them navigate through the hiring process [10], [71]. It has been suggested that since students often encounter "friendly teams" during meetings at school, mock interviews can help them gain insight into how meetings are conducted in an industry setting [47]. They can also help students utilize learning transfer as they apply theories taught into practice.

In addition, since students report an abundance of focus on theory, educators should ruminate on providing increased opportunities for hands on examples and problem solving. Project work and simulations that tackle real life problems or role playing, have been shown to be a valuable aspect of university studies, and can be beneficial when students complete technical interviews [47]. While universities are not meant to serve as vocational institutions, increased familiarity with how algorithms can be applied, and in which contexts, may reinforce the concepts covered. Also, embedding the mindset of testing throughout design and development could help to improve the quality of the work submitted on exams and assignments, as students become stronger at walking through their solutions and correcting their own errors [58].

Resistant Capital: As a first step, companies should provide inclusivity training to all employees. Interviewers and hiring managers should consider how their demeanor and disposition can impact (and potentially discourage) job candidates. Many students report that during the hiring process they feel the recruiters are poor at conveying information, and they are often "ghosted" [33]. Providing feedback about the performance

[5]https://blendoor.com/

should be considered carefully, to encourage aspirational capital for job candidates. Offering concrete suggestions like specific data structures to review, or ways the student could improve could be given along with encouragement to reapply once they have mastered whatever area they fell short in, so that despite the present failure, the job seeker can learn from the experience and move forward in a positive way. Rather than crushing their hopes, proper framing could help strengthen their resistant capital, encouraging them to prepare more in support of their goals, despite the temporary setbacks.

Companies should also consider their job postings, the way they discuss the corporate "culture" during recruitment sessions, and how technical complexity or certain references (i.e., geek culture, gendered, or racial/ethnic) may discourage applicants [35]. Job descriptions should be reviewed by multiple team members, perhaps anonymously, to avoid alienating any individual. In addition, the programs employed to sort through resumes and identify candidates, should be analyzed and checked to mitigate potential algorithmic bias.

To ensure more equitable development in the future, educators can also include course material to train students on algorithmic bias. It is important that students can recognize and detect "intentional discrimination, statistical and classification bias, as well as data errors and absences that may perpetuate structural disadvantage" to shift the mindset towards one of inclusivity [39]. Such training is vital towards students' development of knowledge and skills that will yield consideration of fair-minded practices and implementation.

Linguistic Capital: Although non-native speakers were encouraged to conform for interview preparation, we recommend that employers consider the benefits of hiring multilingual individuals. Speaking another language (or multiple languages) should be considered an asset that may make them more adept at sharing their work and explaining their code during the hiring process in the short term [46], [21]. In the long term, it could also translate into a potential employee that can effectively communicate not only with their boss and coworkers, but also with clients when trying to elicit specifications for software or describing a product. Rather than placing such a premium on live coding during hiring, interviewers could offer candidates more opportunities to talk through prior projects, or to explain take home assignments.

Aspirational Capital: There are several other things that employers could do to encourage hiring and retention of employees. Blincoe et al. [43] demonstrated that women often feel they have to work harder to prove their value, and that there is an implicit bias that disadvantages women towards receiving promotions; however, this is likely true of many minorities. Promoting work life balance, offering mentoring opportunities for new employees, and ensuring promotions are based on measurable achievements could help to improve diverse talent and create a more equitable environment.

Within the university setting, there are also steps educators can take to encourage persistence and engagement in computing. Previously, scholars have argued it is important to develop students' disciplinary identities in STEM, in addition to their cognitive skills [78], [79], [80]. Departments should consider that diverse faculty may serve as mentors and role models [81], [82], and that improving parity in representation "may help mitigate the educational climate, which our participants described as isolating and insensitive to their needs" [82, p. 174]. Also, instructors should be judicious of the way the speak with students, and the feedback they give on exams and assignments. When delivered in the right way, constructive feedback can help improve pedagogy and can encourage maintaining hopes and dreams despite setbacks [83].

VII. Threats to Validity

Some limitations exist in this work, which should be discussed. First, source selection was conducted primarily by the first author, which may lead to some subjective bias. Also, although the search terms were based on a preliminary assessment and seemed to cover the subject adequately, additional terms might have identified further sources.

It should also be noted that each company may have their own hiring procedures, and differences may exist depending on the role. As such, there may be others that are unique, which were not examined here simply because there were no formal publications about their process. Also, while our findings enabled us to establish a general consensus on the overall process in computing, there are many sub-disciplines and specializations which future work should examine to gain a more comprehensive look at hiring across computing fields.

In addition, although our publications gathered span many countries, and multiple continents, our sources were limited to papers in English. We would like to acknowledge that there may be additional work that was filtered out in our review which may be applicable, and we suggest future studies consider inclusion of papers from other databases or written in other languages. Furthermore, we have collected works that consider various perspectives on the process, including not only the companies and hiring managers, but also from job applicants, students, and industry experts. We argue that based on the literature collected, we have obtained a diverse range that comprehensively covers the larger process of computing hiring. Accordingly, there are limited issues with external validity that should be considered.

VIII. Conclusions

Ultimately this work serves to inform students and educators what the hiring process in computing entails, and how to prepare. Moreover, this research is intended to provide industry and academia with evidence of the problems inherent with the existing systems. Going forward, academia should consider revisions to its curricula, and the computing industry should reconsider their current hiring practices and try to refine hiring to make it more inclusive. Knowing that "hiring is broken" in computing is not enough. Instead, it is important to consider what the issues are, to find ways to fix them, and to make the process more equitable to broaden participation of women, Blacks/African Americans, and Hispanics/Latinxs.

REFERENCES

[1] C. Litecky, A. J. Igou, and A. Aken, "Skills in the management oriented IS and enterprise system job markets," in *Proceedings of the 50th annual conference on Computers and People Research*, 2012, pp. 35–44, https://doi.org/10.1145/2214091.2214104.

[2] R. Florea and V. Stray, "A global view on the hard skills and testing tools in software testing," in *2019 ACM/IEEE 14th International Conference on Global Software Engineering (ICGSE)*. IEEE, 2019, pp. 143–151, https://doi.org/10.1109/icgse.2019.00035.

[3] C. Scaffidi, "Employers' needs for computer science, information technology and software engineering skills among new graduates," *International Journal of Computer Science, Engineering and Information Technology*, vol. 8, no. 1, pp. 1–12, 2018, https://doi.org/10.5121/ijcseit.2018.8101.

[4] D. Oguz and K. Oguz, "Perspectives on the gap between the software industry and the software engineering education," *IEEE Access*, vol. 7, pp. 117 527–117 543, 2019, https://doi.org/10.1109/access.2019.2936660.

[5] G. L. McDowell, *Cracking the coding interview: 189 Programming Questions and Solutions*. CareerCup, 2015, ISBN: 9780984782857.

[6] M. Lara, K. Cunningham, and B. Su, "Breaking into Tech: Job Placement Experience and Perceptions of Alumni from a Three-year Computer Science Program," in *2019 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT)*. IEEE, 2019, pp. 1–5, https://doi.org/10.1109/respect46404.2019.8985777.

[7] U. B. of Labor Statistics, "Labor force characteristics by race and ethnicity, 2018," Oct 2019. [Online]. Available: https://www.bls.gov/opub/reports/race-and-ethnicity/2018/pdf/home.pdf

[8] W. of Color in Computing Collaborative, "Women and girls of color in computing," 2018. [Online]. Available: https://www.wocincomputing.org/wp-content/uploads/2018/08/WOCinComputingDataBrief.pdf

[9] W. Aspray, *Women and Underrepresented Minorities in Computing: A Historical and Social Study*. Springer, 2016, https://doi.org/10.1007/978-3-319-24811-0.

[10] P. Hall Jr and K. Gosha, "The Effects of Anxiety and Preparation on Performance in Technical Interviews for HBCU Computer Science Majors," in *Proceedings of the 2018 ACM SIGMIS Conference on Computers and People Research*. ACM, 2018, pp. 64–69, https://doi.org/10.1145/3209626.3209707.

[11] Google, "2020 diversity annual report," 2020. [Online]. Available: https://diversity.google/annual-report/

[12] Girls Who Code, "Girls Who Code to Host Inaugural Virtual Hiring Summit, Serving Hundreds of Women Seeking Technical Roles in Computer Science," 2021. [Online]. Available: https://girlswhocode.com/news/girls-who-code-to-host-inaugural-virtual-hiring-summit-serving-hundreds-of-women-seeking-technical-roles-in-computer-science

[13] M. Jumabayeva, "Employer satisfaction with IT graduate's skills," Ph.D. dissertation, Nazarbayev University Graduate School of Education, 2014.

[14] R. Boden and M. Nedeva, "Employing discourse: universities and graduate 'employability'," *Journal of Education Policy*, vol. 25, no. 1, pp. 37–54, 2010, https://doi.org/10.1080/02680930903349489.

[15] T. McCowan, "Should universities promote employability?" *Theory and Research in Education*, vol. 13, no. 3, pp. 267–285, 2015, https://doi.org/10.1177/1477878515598060.

[16] C. A. Matherly and M. J. Tillman, "Higher Education and the Employability Agenda," in *The Palgrave international handbook of higher education policy and governance*. Springer, 2015, pp. 281–299, https://doi.org/10.1007/978-1-137-45617-5_16.

[17] D. Bennett, E. Knight, A. Divan, L. Kuchel, J. Horn, D. van Reyk, and K. Burke da Silva, "How do research-intensive universities portray employability strategies? A review of their websites," *Australian Journal of Career Development*, vol. 26, no. 2, pp. 52–61, 2017, https://doi.org/10.1177/1038416217714475.

[18] S. Fincher and J. Finlay, "Computing graduate employability: sharing practice," *Council of Professors and Heads of Computing*, 2016.

[19] F. Ahmed, L. F. Capretz, S. Bouktif, and P. Campbell, "Soft skills requirements in software development jobs: A cross-cultural empirical study," *Journal of systems and information technology*, 2012, https://doi.org/10.1108/13287261211221137.

[20] T. M. Heffernan, "Alignment of information technology hiring manager requirements with bachelor degree learning outcomes: Qualitative inquiry," Ph.D. dissertation, Capella University, 2014.

[21] M. Stevens and R. Norman, "Industry expectations of soft skills in IT graduates: a regional survey," in *Proceedings of the Australasian Computer Science Week Multiconference*, 2016, pp. 1–9, https://doi.org/10.1145/2843043.2843068.

[22] A. P. Calitz, J. H. Greyling, and M. D. Cullen, "South african industry ict graduate skills requirements," *Southern African Computer Lecturers' Association (SACLA)*, vol. 1, pp. 25–26, 2014.

[23] D. G. Sólorzano, O. Villalpando, and L. Oseguera, "Educational Inequities and Latina/o Undergraduate Students in the United States: A Critical Race Analysis of their Educational Progress," *Journal of Hispanic Higher Education*, vol. 4, no. 3, pp. 272–294, 2005, https://doi.org/10.1177/1538192705276550.

[24] T. J. Yosso*, "Whose culture has capital? A critical race theory discussion of community cultural wealth," *Race ethnicity and education*, vol. 8, no. 1, pp. 69–91, 2005, https://doi.org/10.1080/1361332052000341006.

[25] M. Denton, M. Borrego, and A. Boklage, "Community cultural wealth in science, technology, engineering, and mathematics education: A systematic review," *Journal of Engineering Education*, 2020, https://doi.org/10.1002/jee.20322.

[26] M. Petticrew and H. Roberts, *Systematic reviews in the social sciences: A practical guide*. John Wiley & Sons, 2006, https://doi.org/10.1002/9780470754887.

[27] D. Verdin, A. Godwin, and B. Capobianco, "Systematic review of the funds of knowledge framework in STEM education," in *ASEE 123^{rd} Annual Conference and Exposition, Conference Proceedings*, 2016, https://doi.org/10.18260/p.25999.

[28] B. Kitchenham and S. Charters, "Guidelines for performing systematic literature reviews in software engineering," 2007.

[29] A. Radermacher, "Evaluating the gap between the skills and abilities of senior undergraduate computer science students and the expectations of industry," Ph.D. dissertation, North Dakota State University, 2012.

[30] V. Garousi, G. Giray, E. Tüzün, C. Catal, and M. Felderer, "Aligning software engineering education with industrial needs: A meta-analysis," *Journal of Systems and Software*, vol. 156, pp. 65–83, 2019, https://doi.org/10.1016/j.jss.2019.06.044.

[31] C. Evans, "Recruitment initiatives aimed at increasing the gender diversity within ITEC employment: Not so "gender neutral"?" *Equality, Diversity and Inclusion: An International Journal*, 2012, https://doi.org/10.1108/02610151211277608.

[32] K. Grossman, *Tech Job Hunt Handbook: Career Management for Technical Professionals*. Apress, 2012, https://doi.org/10.1007/978-1-4302-4549-0.

[33] M. Behroozi, S. Shirolkar, T. Barik, and C. Parnin, "Debugging Hiring: What Went Right and What Went Wrong in the Technical Interview Process," in *2020 IEEE/ACM International Conference on Software Engineering, Software Engineering In Society (ICSE-SEIS)*. IEEE, 2020, https://doi.org/10.1145/3377815.3381372.

[34] M. Behroozi, C. Parnin, and T. Barik, "Hiring is Broken: What do Developers Say About Technical Interviews?" in *2019 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. IEEE, 2019, pp. 1–9, https://doi.org/10.1109/vlhcc.2019.8818836.

[35] A. T. Wynn and S. J. Correll, "Puncturing the pipeline: Do technology companies alienate women in recruiting sessions?" *Social studies of science*, vol. 48, no. 1, pp. 149–164, 2018, https://doi.org/10.1177/0306312718756766.

[36] J. Rothman, *Hiring Geeks That Fit*. Rothman Consulting Group, Inc., 2013.

[37] T. Whitney, D. Gammal, B. Gee, J. Mahoney, and C. Simard, "Priming the Pipeline: Addressing Gender-Based Barriers in Computing," *Computer*, vol. 46, no. 3, pp. 30–36, 2013, https://doi.org/10.1109/mc.2013.40.

[38] R. Rawlings-Goss, *Data Science Careers, Training, and Hiring: A Comprehensive Guide to the Data Ecosystem: How to Build a Successful Data Science Career, Program, or Unit*. Springer, 2019, https://doi.org/10.1007/978-3-030-22407-3.

[39] L. Yarger, F. C. Payton, and B. Neupane, "Algorithmic equity in the hiring of underrepresented IT job candidates," *Online Information Review*, 2019, https://doi.org/10.1108/oir-10-2018-0334.

[40] A. Kapoor and C. Gardner-McCune, "Exploring the participation of cs undergraduate students in industry internships," in *SIGCSE 2020-Proceedings of the 51st ACM Technical Symposium on Computer Science Education*, 2020, https://doi.org/10.1145/3328778.3366844.

[41] R. S. Dubey and V. Tiwari, "Operationalisation of soft skill attributes and determining the existing gap in novice ICT professionals," *International Journal of Information Management*, vol. 50, pp. 375–386, 2020, https://doi.org/10.1016/j.ijinfomgt.2019.09.006.

[42] Y. Wang and D. Redmiles, "Implicit Gender Biases in Professional Software Development: An Empirical Study," in *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Society (ICSE-SEIS)*. IEEE, 2019, pp. 1–10, https://doi.org/10.1109/icse-seis.2019.00009.

[43] K. Blincoe, O. Springer, and M. R. Wrobel, "Perceptions of Gender Diversity's Impact on Mood in Software Development Teams," *IEEE Software*, vol. 36, no. 5, pp. 51–56, 2019, https://doi.org/10.1109/ms.2019.2917428.

[44] P. L. Li, A. J. Ko, and A. Begel, "What distinguishes great software engineers?" *Empirical Software Engineering*, vol. 25, no. 1, pp. 322–352, 2020, https://doi.org/10.1007/s10664-019-09773-y.

[45] D. Mahmoudi, "Making Software, Making Regions: Labor Market Dualization, Segmentation, and Feminization in Austin, Portland and Seattle," Ph.D. dissertation, Portland State University, 2017, https://doi.org/10.15760/etd.3458.

[46] E. M. Trauth, C. C. Cain, K. D. Joshi, L. Kvasny, and K. Booth, "Embracing intersectionality in gender and IT career choice research," in *Proceedings of the 50th annual conference on Computers and People Research*, 2012, pp. 199–212, https://doi.org/10.1145/2214091.2214141.

[47] S. V. Nagarajan, "Professional work experiences of recent Australian Information Technology graduates," Ph.D. dissertation, University of Technology, Sydney, 2011.

[48] A. Aziz, T.-H. Lee, and A. Prakash, *Elements of Programming Interviews: The Insiders' Guide*. EPI, 2012, ISBN: 978-1479274833.

[49] M. Wyrich, D. Graziotin, and S. Wagner, "A theory on individual characteristics of successful coding challenge solvers," *PeerJ Computer Science*, vol. 5, p. e173, 2019, https://doi.org/10.7717/peerj-cs.173.

[50] W. Gant, "Why Software Development Interviews Are Hard," in *Surviving the Whiteboard Interview*. Springer, 2019, pp. 1–6, https://doi.org/10.1007/978-1-4842-5007-5_1.

[51] G. L. McDowell, *Cracking the Tech Career: Insider Advice on Landing a Job at Google, Microsoft, Apple, Or Any Top Tech Company*. John Wiley & Sons, 2014, ISBN: 978-1118968086.

[52] D. Ford, T. Barik, L. Rand-Pickett, and C. Parnin, "The Tech-Talk Balance: What Technical Interviewers Expect from Technical Candidates," in *2017 IEEE/ACM 10th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. IEEE, 2017, pp. 43–48, https://doi.org/10.1109/chase.2017.8.

[53] M. Behroozi, A. Lui, I. Moore, D. Ford, and C. Parnin, "Dazed: Measuring the Cognitive Load of Solving Technical Interview Problems at the Whiteboard," in *2018 IEEE/ACM 40th International Conference on Software Engineering: New Ideas and Emerging Technologies Results (ICSE-NIER)*. IEEE, 2018, pp. 93–96, https://doi.org/10.1145/3183399.3183415.

[54] G. L. McDowell and J. Bavaro, *Cracking the PM Interview: How to Land a Product Manager Job in Technology*. CareerCup, LLC, 2014, ISBN: 978-0984782819.

[55] D. S. Hunsinger, M. A. Smith, and S. J. Winter, "A Framework of the Use of Certifications by Hiring Personnel in IT Hiring Decisions," *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, vol. 42, no. 1, pp. 9–28, 2011, https://doi.org/10.1145/1952712.1952714.

[56] E. Giguère, J. Mongan, and N. Suojanen, *Programming Interviews Exposed: Secrets to Landing Your Next Job*. John Wiley & Sons, 2013, ISBN: 978-0470121672.

[57] K. Raaen and P. Lauvås Jr, "How companies find and evaluate graduate computer programmers," *Norwegian Informatics Conference (NIK) 2018 conference*, 2018.

[58] M. Craig, P. Conrad, D. Lynch, N. Lee, and L. Anthony, "Listening to early career software developers," *Journal of Computing Sciences in Colleges*, vol. 33, no. 4, pp. 138–149, 2018.

[59] M. Cerioli, M. Leotta, and F. Ricca, "What 5 Million Job Advertisements Tell Us about Testing: A Preliminary Empirical Investigation," in *Proceedings of the 35th Annual ACM Symposium on Applied Computing*, 2020, pp. 1586–1594, https://doi.org/10.1145/3341105.3373961.

[60] D. Akdur, "A survey on bridging the gap between software industry and academia: Preliminary results," in *13th Turkish National Software Engineering Symposium (UYMS), At İzmir*, August 2019.

[61] P. L. Li, A. J. Ko, and J. Zhu, "What Makes a Great Software Engineer?" in *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*, vol. 1. IEEE, 2015, pp. 700–710, https://doi.org/10.1109/icse.2015.335.

[62] Q. Burke, C. Bailey, L. A. Lyon, and E. Green, "Understanding the Software Development Industry's Perspective on Coding Boot Camps versus Traditional 4-year Colleges," in *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*, 2018, pp. 503–508, https://doi.org/10.1145/3159450.3159485.

[63] G. Matturro, "Soft skills in software engineering: A study of its demand by software companies in Uruguay," in *2013 6th international workshop on cooperative and human aspects of software engineering (CHASE)*. IEEE, 2013, pp. 133–136, https://doi.org/10.1109/chase.2013.6614749.

[64] C. Watson, K. Blincoe *et al.*, "Attitudes towards software engineering education in the New Zealand industry," in *28th Annual Conference of the Australasian Association for Engineering Education (AAEE 2017)*. Australasian Association for Engineering Education, 2017, p. 785.

[65] V. Garousi, G. Giray, E. Tuzun, C. Catal, and M. Felderer, "Closing the Gap Between Software Engineering Education and Industrial Needs," *IEEE Software*, 2019, https://doi.org/10.1109/ms.2018.2880823.

[66] E. K. Sharma, *Technical Aptitude For Interviews: Computer Science And IT*. PHI Learning Pvt. Ltd., 2014, ISBN: 978-8120352902.

[67] A. I. Schafer, "Diversity in Engineering: A new Approach and Awareness as a first Step to Change," in *4th ASEE/AaeE Global Colloquium on Engineering Education*. Australasian Association of Engineering Education, 2005, p. 1637.

[68] E. N. Liu, "Access to Employment or Access to Employers: a Descriptive Study of Employers' Attitudes and Practices in Hiring Newcomer Job Seekers," Ph.D. dissertation, Ryerson University, 2006.

[69] M. Exter, S. Caskurlu, and T. Fernandez, "Comparing Computing Professionals' Perceptions of Importance of Skills and Knowledge on the Job and Coverage in Undergraduate Experiences," *ACM Transactions on Computing Education (TOCE)*, vol. 18, no. 4, pp. 1–29, 2018, https://doi.org/10.1145/3218430.

[70] N. E. A. M. Almi, N. A. Rahman, D. Purusothaman, and S. Sulaiman, "Software engineering education: The gap between industry's requirements and graduates' readiness," in *2011 IEEE Symposium on Computers & Informatics*. IEEE, 2011, pp. 542–547, https://doi.org/10.1109/isci.2011.5958974.

[71] C. Snell-Siddle, S. Snell, and A. Steele, "The first hurdle: Preparing students for industry interviews," in *Proceedings of the New Zealand Association for Cooperative Education Conference 2010*, 2014, pp. 27–30.

[72] A. Begel and N. Nagappan, "Pair Programming: What's in it for Me?" in *Proceedings of the Second ACM-IEEE international symposium on Empirical software engineering and measurement*, 2008, pp. 120–128, https://doi.org/10.1145/1414004.1414026.

[73] L. J. Charleston, "A Qualitative Investigation of African Americans' Decision to Pursue Computing Science Degrees: Implications for Cultivating Career Choice and Aspiration," *Journal of Diversity in Higher Education*, vol. 5, no. 4, p. 222, 2012, https://doi.org/10.1037/a0028918.

[74] W. DuBow, "Attracting and Retaining Women in Computing," *Computer*, vol. 47, no. 10, pp. 90–93, October 2014, https://doi.org/10.1109/mc.2014.272.

[75] J. Tyler, *Building Great Software Engineering Teams*. Springer, 2015, https://doi.org/10.1007/978-1-4842-1133-5.

[76] M. Sandgren, "When Glass Ceilings Meet Glass Walls," *Kennedy School Review*, vol. 14, pp. 39–47, 2014.

[77] G. Neville-Neil, "Interviewing Techniques: Separating the good programmers from the bad," *Queue*, vol. 9, no. 6, pp. 30–32, 2011, https://doi.org/10.1145/1989748.1998475.

[78] A. Calabrese Barton and E. Tan, "Designing for Rightful Presence in STEM: The Role of Making Present Practices," *Journal of the Learning Sciences*, vol. 28, no. 4-5, pp. 616–658, 2019, https://doi.org/10.1080/10508406.2019.1591411.

[79] S. Vakil, ""I've Always Been Scared That Someday I'm Going to Sell Out": Exploring the relationship between Political Identity and Learning in Computer Science Education," *Cognition and Instruction*, vol. 38, no. 2, pp. 87–115, 2020, https://doi.org/10.1080/07370008.2020.1730374.

[80] A. Calabrese Barton, E. Tan, and D. J. Birmingham, "Rethinking High-Leverage Practices in Justice-Oriented Ways," *Journal of Teacher Education*, vol. 71, no. 4, pp. 477–494, 2020, https://doi.org/10.1177/0022487119900209.

[81] E. P. Bettinger and B. T. Long, "Do Faculty Serve as Role Models? The Impact of Instructor Gender on Female Students," *American Economic Review*, vol. 95, no. 2, pp. 152–157, 2005, https://doi.org/10.1257/000282805774670149.

[82] L. J. Charleston, P. L. George, J. F. Jackson, J. Berhanu, and M. H. Amechi, "Navigating underrepresented STEM spaces: Experiences of Black women in U.S. computing science higher education programs who actualize success." *Journal of Diversity in Higher Education*, vol. 7, no. 3, p. 166, 2014, https://doi.org/10.1037/a0036632.

[83] A. D. Rittmayer and M. E. Beier, "Overview: Self-efficacy in STEM," *SWE-AWE CASEE Overviews*, vol. 1, p. 12, 2008.