RESEARCH-ARTICLE

Uncovering Meaningful Computing Contexts for Incarcerated College Students

EMMA HOGAN, University of California, San Diego, San Diego, CA, United States

JOHN DRISCOLL, University of California, San Diego, San Diego, CA, United States

ADALBERT GERALD SOOSAI RAJ, University of California, San Diego, San Diego, CA, United States

WILLIAM G. GRISWOLD, University of California, San Diego, San Diego, CA, United States

LEO PORTER, University of California, San Diego, San Diego, CA, United States

Open Access Support provided by:

University of California, San Diego

Published: 03 July 2024

ITiCSE 2024: Innovation and Technology in Computer Science Education
July 8 - 10, 2024
Milan, Italy

Conference Sponsors:
SIGCSE

https://doi.org/10.1145/3649217.3653632
ISBN: 9798400706004

Uncovering Meaningful Computing Contexts for Incarcerated College Students

Emma Hogan
emhogan@ucsd.edu
UC San Diego
La Jolla, CA, USA

John Driscoll
jjdrisco@ucsd.edu
UC San Diego
La Jolla, CA, USA

Adalbert Gerald Soosai Raj
asoosairaj@ucsd.edu
UC San Diego
La Jolla, CA, USA

William Griswold
bgriswold@ucsd.edu
UC San Diego
La Jolla, CA, USA

Leo Porter
leporter@ucsd.edu
UC San Diego
La Jolla, CA, USA

ABSTRACT

Higher education is expanding in United States prisons, with a growing demand for STEM offerings. Academics from other disciplines have stressed the importance of culturally relevant pedagogy (CRP) in prison higher education, and computing in context has shown major benefits in CS1— especially for women and nontraditional students. More work is needed to determine what contexts are relevant to incarcerated college students, and how to incorporate these into computing curricula. In this paper, we build on prior work on computing in context and culturally relevant techniques in computing. We analyze course data from a CS1 course taught in a college-in-prison program to answer the following research question: What contexts do incarcerated students in CS1 find relevant? We identify 24 topics pursued by students across 78 open-ended programming assignment submissions, the three most popular being business management, sports statistics, and physical health. These results offer insight into potential contexts that are meaningful to incarcerated college students to be incorporated into future computing curricula and interventions in prisons.

CCS CONCEPTS

• **Social and professional topics** → **Computing Education**; **CS1**; **Adult education**.

KEYWORDS

Adult Learners; Prison Education; Computing in Context

ACM Reference Format:
Emma Hogan, John Driscoll, Adalbert Gerald Soosai Raj, William Griswold, and Leo Porter. 2024. Uncovering Meaningful Computing Contexts for Incarcerated College Students. In *Proceedings of the 2024 Innovation and Technology in Computer Science Education V. 1 (ITiCSE 2024), July 8–10, 2024, Milan, Italy*. ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3649217.3653632

ITiCSE 2024, July 8–10, 2024, Milan, Italy

ACM ISBN 979-8-4007-0600-4/24/07
https://doi.org/10.1145/3649217.3653632

1 INTRODUCTION

The global prison population is currently at an all-time high, where "the poorest and most marginalised members of society continue to be over-represented" [2]. The United States has the highest incarceration rate in the world by a significant margin, fueling a national crisis of mass incarceration disproportionately impacting poor, Black and Latine communities [2, 24]. In the midst of a global shift towards rehabilitative as opposed to punitive approaches to criminal justice [2], higher education in prison (HEP) has proven particularly effective in reducing one's likelihood of returning to prison post-release [12]. HEP programs are expanding with a need for more STEM offerings [1, 9], while the computing field continues to suffer from a lack of diversity [30]. Improving computing education in prisons stands to benefit incarcerated individuals, computing, and society as a whole.

In their essay on racial inequity in HEP programs, Taylor et al. advocate for the use of culturally relevant teaching practices in prison education [29]. They state that culturally relevant pedagogy in higher education in carceral spaces facilitates "in-classroom opportunities for students to see themselves in every component of the course" [29]. This has positive consequences both inside and outside of the classroom: students make more personal connections to academic content leading to greater student investment, rigor, and motivation.

Both culturally relevant techniques and computing in context engage and motivate students through making computing curricula relevant. On traditional campuses, significant prior work in the CS education literature have shown the benefits of computing in a relevant context [16, 18, 19]. For example, using media computation in CS1 increased retention for women in computing [18, 26]. However, incarcerated college students not only have vastly different experiences, but also face unique barriers in learning computing, such as a lack of access to or experience with technology [3, 6, 7]. Contexts that have shown to be successful for non-incarcerated college students may be less relevant to incarcerated college students, or difficult to implement due to technology limitations.

Learning from students about which topics resonate with them is part of culturally relevant pedagogy [20], but to our knowledge, has not been documented in prior work for incarcerated college students in computing. To fill this gap in the literature, we ask the following research question: *What contexts do incarcerated students in CS1 find relevant?* We present findings from a qualitative analysis

of open-ended assignment submissions from 20 incarcerated CS1 students. This paper is not intended to represent a complete list of topics of interest; rather, it serves as a starting point for expanding culturally relevant computing curriculum development for prison contexts.

2 BACKGROUND

To support the pursuit of meaningful computing context(s) for incarcerated college students, we draw on two existing frameworks in CS education: computing in context and culturally relevant education in computing. We also connect insights from these frameworks with andragogy theory, which stresses the importance of adult learners' ability to incorporate prior work and life experiences into the classroom.

2.1 Computing in Context

Cooper and Cunningham defined computing in context as "teaching computer science in a context in order to engage the student with computing in a concrete or personal way" [8]. In practice, this involves embedding course learning objectives in one coherent context (e.g., eTextile clothing design [5]) to motivate examples and assignments [8]. The two primary, proven benefits of contextualized computing are increasing relevance and retention [17].

Contextualizing computing courses through media computation [16] has been a significant innovation in computing education research [25]. Empirical evidence demonstrates that the media computation context dramatically increases retention in introductory programming, and also increases women's success and positive experience in the course [18].

Some prior works in contextualized computing have been aimed towards a specific population, particularly women in computing. Media computation was explicitly designed with female students in mind, focusing on concerns about applicability and creative expression [26]. Findings from interviews with women emphasized both the relevance and usefulness of the media computation course content [26]. LilyPad Arduino eTextile design is another example of a computing context successfully able to attract women to the field of computing [5]. More generally, contextualizing CS in a way that allows for creative expression has been effective in recruiting and engaging female students [4, 26, 28].

2.2 Culturally Relevant Education

The term culturally relevant education (CRE) encompasses the two most prominent frameworks related to cultural relevance in education: culturally responsive teaching (CRT) and culturally relevant pedagogy (CRP) [23]. CRT focuses on methods of teaching that empower students of diverse cultural backgrounds, such as acknowledging their lived experiences and setting high academic standards [15]. In contrast, CRP focuses on curriculum development, leveraging students' cultures in this process and transforming learning environments [20].

In their 2019 literature review of CRE in Computing, Morales-Chicas et al. synthesized 22 prior works on CRE in K-12 computing education [23]. The authors identified three primary theories/concepts related to CRE in computing: ethnocomputing, culturally responsive computing (CRC), and Culturally Situated Design Tools (CSDT). In addition, they identified six themes across the prior works. One of the themes, recognition of students' lived experiences, directly connects to both an original goal of CRE as defined by Ladson-Billings [20] as well as andragogy theory, discussed in Section 2.3.

While there are no large scale studies, to our knowledge, of CRE in computing outside of the K-12 context, Madkins et al. conducted a multi-case study of CRP in three K-12 CS education programs [21]. Participants included both teachers (N=15) and students (>1000), spanning multiple sites and two academic years. They found statistically significant increases in students' "understanding what CS is," including potential careers in computing, their "belief in the cultural relevance of CS," and access to CS support networks [21].

2.3 Adults and Incarcerated Students

Prior work has explored how principles of andragogy can be applied for adult learners in prison settings [7]. However, one criticism of andragogy is that it fails to address socio-political awareness [13]. French's literature review explores the influence and intersection of culturally relevant frameworks, including CRT and CRP, with adult education [13]. French found that these frameworks "did not neatly align with the andragogical model, partially due to the K-12 origination and intended purpose to support learner achievement," concluding that future work is needed to reframe andragogical assumptions to incorporate culturally relevant practices [13]. Below, we highlight works outside of computing education that have shown the benefits of culturally relevant techniques for adult students from racially diverse backgrounds and lower socioeconomic status in higher education, as well as incarcerated adult men in a violence prevention program.

Sealey-Ruiz studied the impact of a culturally relevant curriculum on 15 adult college students in a writing seminar course [27]. All students in the course were women, identified as Black or African American, and as poor or working class. Three major themes emerged from the qualitative study, all indicating a positive reaction to CRP from the students: language validation, fostering positive self- and group-identity, and affirmation of goals [27].

Daniels' dissertation research focused on the effects of CRP on 41 incarcerated adult men in a violence prevention program in a US prison [11]. Controlling for culturally relevant instructor (one with similar lived experiences and background), culturally relevant pedagogy was significantly related to improved communication and relationship skills, but not significantly related to violence prevention. A culturally relevant instructor had the opposite effect when controlling for pedagogy: there was a significant relationship with violence prevention, but not with communication and relationship skills. Despite the majority of participants in the program being from racial minorities in the US, the author stated that the existing program curriculum "framed violence from a domestic patriarchal point of view, but did not take into account the more complex trauma that men of color were expressing … and did not address their fear of losing their lives to gun violence" [11]. Implementing culturally relevant pedagogy meant changing the curriculum to use individual stories and give a voice to the participants' experience [11].

Table 1: Student Racial Demographics

Racial Identity	Percentage of Students
Chicanx or Latinx	30.0%
White or Caucasian	30.0%
African American or Black	30.0%
American Indian or Alaska Native	15.0%
Asian or Asian American	5.0%
Other: Mexican	10.0%
Other: African	5.0%
Other: Cuban American	5.0%

2.4 Relevance in CS Education

Relevance is a core component of intrinsic motivation, and central to all of the educational frameworks outlined above [28]. In contextualized computing, Guzdial writes that making CS relevant means students can more easily see value in what they are learning, which increases retention [17]. Similarly, andragogy theory emphasizes adult students' need to see value in what they are learning prior to investing effort, as well as the importance of incorporating and embracing adult students' breadth and diversity of past life experiences [7]. CRE frameworks go beyond the standards of relevance in contextualized computing and andragogy, additionally requiring the development of "broader sociopolitical consciousness" to critique societal norms and institutions that produce social inequality [13, 20] and connecting knowledge to "appropriate responsive and responsible action" [10]. Together, these frameworks provide a strong case for seeking to identify what computing contexts incarcerated students find most relevant.

3 METHODS

3.1 Course Context

This work is situated in a CS1 course taught through a college-in-prison program in an adult male prison in the United States. Unlike on traditional college campuses, all students in this bachelor's degree program take the CS1 course to fulfill the computing requirement for the only degree offered. The college-in-prison program was in its second year of operation at the time of this course offering, and we offered the same CS1 course during the fall term for both cohorts. As this was the second offering of the course, we had the valuable experience of having taught the course in a similar setting before and students had access to a previous cohort of students who had all successfully completed the course.

3.2 Study Population

In order to apply for the bachelor's degree program, all students had completed an associate's degree with a high GPA (most through a separate program offered by a community college on the same prison yard). Student racial demographics shown in Table 1[1] differed greatly from those on traditional college campuses in the US—especially within computing [22]. The demographics of the students in the college program have nuances related to the particular prison yard and facility: People incarcerated on this yard typically were serving long-term or life sentences, and had earned the relatively greater privileges and program access through "good behavior". All of the students self-reported their age group, with 85% age 40 or older. Although we did not collect information from students about their sentence length, both anecdotal evidence and the nature of the yard support the assumption that many had been incarcerated for multiple decades, meaning limited or no prior experience with computers or other modern technology.

[1]Students self-identified their racial identity(s) in the first Weekly Reflection assignment (see Section 3.3.2).

3.3 Data Collection

Assignment submissions from all 20 students in the course were included in our analysis. All of the programming assignment submissions and weekly reflection responses used for this study were anonymized by a third party before beginning our data analysis. This project was reviewed by the institutional Human Subjects Review Board under protocol #806658 and determined to be not human subjects research.

3.3.1 Programming Assignments: Four programming assignments (PAs) were assigned approximately bi-weekly during the course. Each of these PAs contained two parts: the first part was a predefined problem with detailed instructions and expectations; the second part was an open-ended prompt for students to write a program incorporating certain programming elements or concepts (e.g., boolean value, "for" loop, etc.). Students were given the option to collaborate in small groups on the first part of each PA, but were required to complete the second portion individually. In addition to their code, the open-ended portion of the assignment prompted students to describe the purpose of the program and define at least three test cases with expected outputs.

3.3.2 Reflections: Reflection assignments intended to gather feedback for course improvement were completed weekly by students and graded for completion (amounting to 5% of the total course grade). Reflections included a mix of multiple-choice, Likert-scale, and open-ended questions pertaining to various topics such as study habits, confidence, and course policies. The questions varied each week, although some questions were asked at multiple points throughout the quarter.

3.4 Analysis Methods

Our analysis of the open-ended PAs included 78 total submissions from 20 students across four PAs throughout the course. Two students had no submission for the open-ended portion of one PA. These submissions included both the students' code as well as a description of the purpose of the code, although these descriptions were completed with varying levels of detail and 5 submissions had no description.

Qualitative Coding: Our qualitative coding process consisted of three rounds of open coding to establish thematic codes, followed by one additional round to establish a consistency score between the two coders [31]. We divided the coding process into three rounds (approximately one third of the submissions each), during which each of the two first authors independently coded a portion of the submissions and created or modified independent code books as needed. As some of the program submissions incorporated multiple themes, we coded the submissions with all codes that applied.

After each round, the two first authors met to compare codes from the previous round, resolving any differences by consensus, and updating a joint code book (adding, editing, or removing codes) so that we began the subsequent round with identical code books. After the third round, the two authors independently coded all 78 responses again using the final joint code book.

Inter-coder Reliability: Since it was possible for submissions to be assigned one or more codes, we first computed the Cohen's kappa value for each of the 24 individual codes (representing the consistency in applying each code between the two coders). We then calculated the average of these values for a final score of **ICR=0.84**, representing our average consistency across all of the codes.

4 RESULTS

We identified 24 unique topics or themes in students' open-ended PA submissions, listed in Table 2. In our attempts to group the codes into larger categories post-completion, we found too many overlaps between codes and ultimately decided to present each code without any broader categorization. Below, we provide examples and describe the most common topics in more detail.

4.0.1 Business Management. The most common topic amongst the open-ended PA submissions was business management. This label was applied to programs intended for business owners and/or tools for managing a business. Below is one such example from the second PA, including a snippet of code and the description provided by the student.

```
#The purpose of this program is to provide information
to the costumer based on the age they entered.
#The individuals who will use the program are the people
who shop at the liquor store.
#This will help the store determine which people to sale
alcohol to and who not to.
...
if age < 21:
    print("User is unable to consume alcohol")
    print("User must leave store")
elif age >= 65:
    print("costumer is a senior")
    print("Costumer can purchase alcohol and receive a
      free bag of chips")
...
```

Other examples under the business management category included programs that facilitated a customer checkout at a "prison store" which sold items such as candy and desserts, maintained a digital log of CD borrowers, and a "Community Center Tracking Program" that collected information about the customer experience.

4.0.2 Sports Statistics. The second most popular topic was sports statistics, which included programs that related to professional sports. Below is an example from the first PA[2] (Note: a *pod* refers to a branch of a housing unit in this prison):

```
# Purpose of this program is to simply calculate the
percent of NFL Team Fans per pod.
# Program to be used by Football enthusiasts.
# Program is a recreational tool used to make sports
more enjoyable.
...
number_fans1 = (input('enter number fans team1:'))
number_fans2 = (input('enter number fans team2:'))
...
pod_population = int(66)
percent_team1 = float(number_fans1 / pod_population * 100)
percent_team2 = float(number_fans2 / pod_population * 100)
...
```

Other examples included calculating baseball batting averages and base percentages, free throw percentages for players on a fantasy basketball team, and determining NFL draft pick order.

4.0.3 Physical Health. The third most prevalent theme was physical health. These programs were distinguished as physical healthcare related, including diet and heart rate. Several programs in this category warned users of potential health concerns depending on some input, including the program below locating the user's allergy in a list of ingredients:

```
# This program would make it more convenient to manage
food allergies by scanning the ingredients.
...
if ingredients.index(allergy)==-1:
   return "There were no allergens found."
else:
   locate=content.index(allergy)
   ...
   return warning
...
```

Other examples in this category determined whether a user's heart rate was in a dangerous zone, checked for concussion symptoms, and made dieting recommendations. During our analysis, we decided to distinguish *physical health* as a separate category from *mental health and addiction*, instead of having a single code for health. Programs in the mental health and addiction category included a program for triangulating emergency response for individuals in a mental health crisis, and a program recommending groups to attend based on a user's substance-use history.

5 DISCUSSION

5.1 Topic Variation by PA

During our qualitative analysis, we began to observe a pattern that certain topics were more popular in certain PAs. We decided to analyze how the chosen topics evolved through each PA, with results shown in Figure 1. Below, we highlight some of our takeaways based on these results.

5.1.1 Influence of Concepts Covered. The *writers* theme, encompassing programs that were tools for writers, were the most popular topic for a single PA and the fourth most popular overall. We hypothesize that this was influenced by the programming concepts covered in each of the PAs: PA3 covered string sequences, indexing/slicing, and using string methods. Anecdotally, some students

[2]Minor syntax errors, such as misplacement of quotation marks, were corrected by the author

Table 2: Student Topics for Open-Ended Programming Assignments

Code	%	Description
business management	14.1%	Tool for business owners/managing a business
sports statistics	11.5%	Compute sports player or team statistics;
physical health	10.3%	Physical healthcare related; warning / diagnosing of potential health issues
writers	10.3%	Tool for writers; correcting grammar
students	9.0%	Tool for students; study habits / time management for students; assess academic progress / standing
teaching	9.0%	Tool for educators / tutors; teaches a user about a topic
no discernible theme	6.4%	No discernible purpose or theme; random calculations
incarceration	6.4%	Related to prison reform; helps incarcerated people; improves life in prison
playing sports	6.4%	Tool for athletes; keeps score of a game / tournament; assesses athletic performance
activities	5.1%	Activities / hobbies (i.e., going to the movies, travel recommendations)
food	5.1%	Collecting food preferences; related to food quality
cars	3.8%	References car parts/fixing cars; recommendations on a car to purchase
commissary	3.8%	Related to prison commissary purchases; calculating commissary totals
outside occupation	2.6%	Tool specific to an occupation outside of prison (e.g., electrician)
budgeting	2.6%	Tool for budgeting money / saving money
prison work	2.6%	Program related to a work assignment in the prison
games	2.6%	Game simulation; game shows/reality TV games
mental health and addiction	2.6%	Tool for emergency response for mental health crises; help people struggling with addiction
family	2.6%	Tools for families / relationships between family members
social justice	2.6%	Educates / informs / collects information about a social justice topic
house	1.3%	Calculates some feature of a house (e.g., area of a room)
disabilities	1.3%	Intended to serve individuals with disabilities
artist	1.3%	Tool for artists
politics	1.3%	Political opinions or preferences

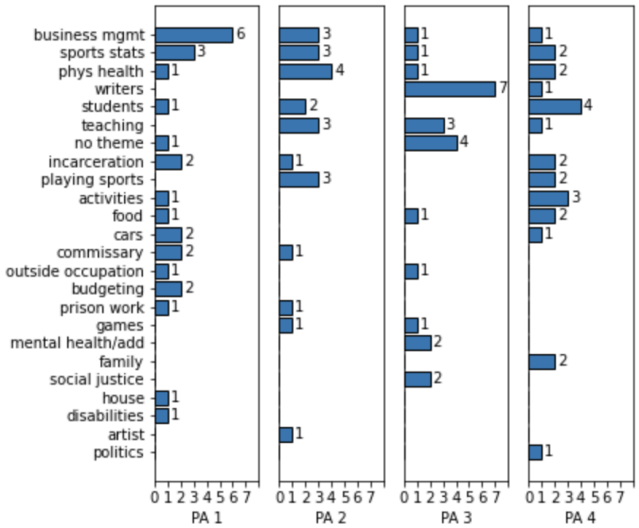

Figure 1: Counts of Topic by PA

expressed upon handing in their assignments that they had a particularly difficult time coming up with ideas involving string methods (as discussed in Section 5.2), which could explain the more significant showing of programs labeled *no discernible theme*. Similarly, topics covered by PA1 were limited to what was covered in the first lecture (e.g., variables, basic data types, expressions). Programs incorporating basic calculations, such as in the *budgeting* category, appeared in PA1 but not in later PAs.

5.1.2 Influence of Instructor-Conceived Examples. The results of our analysis of how topics progressed throughout the course led us to think that, especially in earlier PAs, topics chosen by students for open-ended programming questions might have been influenced by the examples the instructor chose to present in class. This makes sense: as students are exposed to an increasing variety of programming concepts and example applications, their own ideas for programs have more depth and variety. We also believe this is particularly true in an environment where students have limited or no access to the internet/outside resources (as is the case in nearly all prisons in the United States, and in many other countries including France and Canada [2]). For example, prior to the deadline for the first PA, we had presented an in-depth example program emulating a grocery store checkout process. It was designed by the instructor to incorporate the few topics covered so far and required on the PA: variables holding item prices, type conversion and user input to acquire the quantity of each item being purchased, simple mathematical expressions to compute the tax and total, and print statements with the resulting receipt. The most similar topics we identified as similar to this example were *business management* and *commissary*. As shown in Figure 1, despite the relative popularity of *commissary* as a topic for the first two PAs, it disappeared in the later PAs. Similarly, *business management* was more popular in the first two PAs, and tapered off. Although this potentially inflated the true interest in topics that were popular, especially in earlier

weeks, we see this as an opportunity to use our current findings to provide more relevant examples early on in future iterations of the course.

5.2 Student Motivations

One question that naturally arises is from where the students' topics are coming. We have some insight into this from student responses to an open-ended question on one of the Weekly Reflections, which asked students why they chose the topic they did for the open-ended question on the third PA. Seventeen of the twenty students responded to this question on the reflection, including two responses that did not specify a clear motivation (e.g., *"I can't remember"*). The first author categorized the remaining fifteen responses following a similar procedure to our qualitative coding of the open-ended PA submissions, applying multiple codes to each response as necessary. The two most common categories were as follows:

- **Interest/Enjoyment:** Four responses cited personal or group interest/enjoyment, such as *"I enjoy watching [and] playing soccer as a whole so I pick something I enjoy to apply a coding program to."*.
- **Meeting Criteria:** Four responses talked about meeting assignment criteria, including one student who wrote *"I choose something simple to get the assignment done while covering the assignment parameters..."*.
- **Difficulty Choosing:** Four students mentioned difficulty choosing a topic. As one student explained, *"trying to develop a program that is: 1) creative, 2) functional, and 3) features all of the assignments' requirements proved rather difficult."*

Other themes included **outside projects** (e.g., *"I was working on art proposal for the Warden at the time and I just combined my thoughts real quick"*) and **tools for students** (e.g., *"Because I thought about our grading and curriculum in fixing errors in our essays..."*), which both appeared in three student responses. These themes, along with **interest/enjoyment**, show students finding ways to apply computing to things they experience in their daily lives. As several students expressed difficulty in choosing a topic, we hope to create more of a balance in the future of more structure in a relevant context, while still allowing for creativity [28]. Notably, two students wrote their programs to **address a social issue**, such as *"mental health ... is a topic overlooked and just recently taken partly serious. I want to ... create a program to help people who suffer with depression and anxiety"*. These programs are particularly conducive to being incorporated into culturally relevant interventions and curriculum development in the future, as they take on broader social issues that are relevant to many incarcerated individuals [2, 20]. Another student wrote a program using his knowledge from his occupation prior to incarceration, which is noted as an important part of adult learning in andragogy theory [7].

5.3 Limitations

The contexts presented in this paper do not represent a comprehensive list of what is meaningful to all incarcerated college students, but are a step toward creating a relevant computing curriculum for prison contexts. Current data available on incarcerated college students in the United States is limited, making it difficult to assess the generalizability of our findings [14]. A 2019 report based on 2014 data found that incarcerated adults enrolled in higher education programs were 36% black, 26% hispanic, and 30% white [24]. In addition, the majority of those enrolled were between the ages of 25-44 and more than half were within 2 years of being released [24]. The racial demographics of students in our course were similar to those reported above. However, our students were generally older, and a larger proportion of the students were likely to have more time remaining in their sentences (as discussed in Section 3.2). We note that the age and race statistics above are nearly a decade old, during which time there have been shifts in policy shaping higher education in prison in the United States (for example, a lift on a 30-year ban on federal Pell Grant funding for incarcerated college students). As more information becomes available, and higher education becomes accessible to more incarcerated adults, we can build on the contexts presented here to better represent contexts relevant to incarcerated college students.

5.4 Future Work

Meaningful contexts identified for incarcerated college students can be used to create new innovations and interventions to improve the quality of computing higher education in prisons. In particular, the work in this study provides an initial basis for implementing contextualized computing and more culturally relevant techniques in teaching computing in prisons. For example, in future iterations of our CS1 course in prison, we plan to adapt lecture materials, office hours examples, assignments, and labs based on the contexts identified here. In addition, we will continue to learn from more students about what contexts are relevant to them and why.

6 CONCLUSION

Prior research on the benefits of computing in context in higher education motivates us to believe that creating a relevant context and curriculum can make a difference for computing students in prison. In this study, we present the following contributions:

- An approach to identifying relevant computing contexts for a specific population of students
- 24 programming assignment topics chosen by incarcerated college students, and their popularity across four PAs
- An initial basis for applying computing in context and culturally relevant pedagogies in prison contexts

Results of our qualitative analysis of open-ended programming assignment submissions from a CS1 course taught in prison found that the three most common topics chosen by students were business management, sports statistics, and physical health. The variety of topics chosen expanded throughout the course as students were exposed to more programming concepts and examples. Future courses and interventions can use these contributions to make computing more relevant to incarcerated students, promoting student engagement and success.

ACKNOWLEDGMENTS

This work is supported by the Ford Foundation Predoctoral Fellowship and NSF Award #2315909.

REFERENCES

[1] [n. d.]. STEM Opportunities in Prison Settings (STEM-OPS). https://stemforall2020.videohall.com/presentations/1801.html

[2] 2022. *Global Prison Trends 2022.* Technical Report. Penal Reform International. https://www.penalreform.org/global-prison-trends-2022/

[3] James R. Aman. 1990. Computer science in correctional education. In *Proceedings of the twenty-first SIGCSE technical symposium on Computer science education - SIGCSE '90.* ACM Press, Washington, D.C., United States, 147–151. https://doi.org/10.1145/323410.323436

[4] William H. Bares, Bill Manaris, Renée McCauley, and Christine Moore. 2019. Achieving Gender Balance through Creative Expression. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education.* ACM, Minneapolis MN USA, 293–299. https://doi.org/10.1145/3287324.3287435

[5] Leah Buechley, Mike Eisenberg, Jaime Catchen, and Ali Crockett. 2008. The LilyPad Arduino: using computational textiles to investigate engagement, aesthetics, and diversity in computer science education. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems.* ACM, Florence Italy, 423–432. https://doi.org/10.1145/1357054.1357123

[6] Erin L. Castro, Caisa E. Royer, Stephanie Gaskill, and Estefanie Aguilar-Padilla. 2022. *"It's Useless, to Put it Politely": Experiences with Technology Among Incarcerated Students Receiving Second Chance Pell at Four Institutions.* Brief 9. Collaborative for Higher Education Research and Policy, The University of Utah. https://cherp.utah.edu/projects/pell_is_not_enough.php#publications-slide

[7] Patrick Filipe Conway. 2022. Andragogy in Prison: Higher Education in Prison and the Tenets of Adult Education. *Adult Education Quarterly* 72, 4 (Nov. 2022), 361–379. https://doi.org/10.1177/07417136221100481 Publisher: SAGE Publications Inc.

[8] Steve Cooper and Steve Cunningham. 2010. Teaching computer science in context. *ACM Inroads* 1, 1 (March 2010), 5–8. https://doi.org/10.1145/1721933.1721934

[9] Trevor Craft, Nicholas Gonzalez, Kevin Kelleher, Miki Rose, and Ofu Takor. 2019. *A Second Chance: College-in-Prison Programs in New York State.* Technical Report. Nelson A. https://eric.ed.gov/?id=ED605777 Publication Title: Nelson A. Rockefeller Institute of Government ERIC Number: ED605777.

[10] A. Cheryl Curtis. 1998. Creating Culturally Responsive Curriculum: Making Race Matter. *The Clearing House: A Journal of Educational Strategies, Issues and Ideas* 71, 3 (Jan. 1998), 135–139. https://doi.org/10.1080/00098659809599344

[11] Reginold P Daniels. 2019. *Exploring Ways to Bring Culturally Relevant Pedagogy into Rehabilitative Programs.* Ph. D. Dissertation. The University of San Francisco.

[12] Lois Davis. 2019. *Higher Education Programs in Prison: What We Know Now and What We Should Focus On Going Forward.* Technical Report. RAND Corporation. https://doi.org/10.7249/PE342

[13] Patrice B French. 2019. Culturally Relevant Practice Frameworks and Application in Adult Education. *Adult Education Research Conference* (2019). https://newprairiepress.org/aerc/2019/papers/27

[14] Stephanie Gaskill, Mary R. Gould, Ved Price, Erin L. Castro, and Amy Lerman. 2023. *The Landscape of Higher Education in Prison, 2020-2021.* Technical Report. Alliance for Higher Education in Prison. http://higheredinprison.org

[15] Geneva Gay. 2018. *Culturally responsive teaching: Theory, research, and practice.* teachers college press.

[16] Mark Guzdial. 2003. A media computation course for non-majors. In *Proceedings of the 8th annual conference on Innovation and technology in computer science education.* ACM, Thessaloniki Greece, 104–108. https://doi.org/10.1145/961511.961542

[17] Mark Guzdial. 2010. Does contextualized computing education help? *ACM Inroads* 1, 4 (Dec. 2010), 4–6. https://doi.org/10.1145/1869746.1869747

[18] Mark Guzdial. 2013. Exploring hypotheses about media computation. In *Proceedings of the ninth annual international ACM conference on International computing education research.* ACM, San Diego San California USA, 19–26. https://doi.org/10.1145/2493394.2493397

[19] Mark Guzdial and Andrea Forte. 2005. Design process for a non-majors computing course. *ACM SIGCSE Bulletin* 37, 1 (Feb. 2005), 361–365. https://doi.org/10.1145/1047124.1047468

[20] Gloria Ladson-Billings. 1995. But That's Just Good Teaching! The Case for Culturally Relevant Pedagogy. *THEORY INTO PRACTICE* (1995).

[21] Tia C. Madkins, Alexis Martin, Jean Ryoo, Kimberly A. Scott, Joanna Goode, Allison Scott, and Frieda McAlear. 2019. Culturally Relevant Computer Science Pedagogy: From Theory to Practice. In *2019 Research on Equity and Sustained Participation in Engineering, Computing, and Technology (RESPECT).* 1–4. https://doi.org/10.1109/RESPECT46404.2019.8985773

[22] Jane Margolis, Rachel Estrella, Joanna Goode, Jennifer Jellison Holme, and Kim Nao. 2017. *Stuck in the Shallow End: Education, Race, and Computing.* The MIT Press.

[23] Jessica Morales-Chicas, Mauricio Castillo, Ireri Bernal, Paloma Ramos, and Bianca Guzman. 2019. Computing with Relevance and Purpose: A Review of Culturally Relevant Education in Computing. *International Journal of Multicultural Education* 21, 1 (March 2019), 125–155. https://doi.org/10.18251/ijme.v21i1.1745

[24] Monique O. Ositelu. 2019. *Equipping Individuals for Life Beyond Bars.* Technical Report. New America. http://newamerica.org/education-policy/reports/equipping-individuals-life-beyond-bars/

[25] Leo Porter, Mark Guzdial, Charlie McDowell, and Beth Simon. 2013. Success in introductory programming: what works? *Commun. ACM* 56, 8 (Aug. 2013), 34–36. https://doi.org/10.1145/2492007.2492020

[26] Lauren Rich, Heather Perry, and Mark Guzdial. 2004. A CS1 course designed to address interests of women. *ACM SIGCSE Bulletin* 36, 1 (March 2004), 190–194. https://doi.org/10.1145/1028174.971370

[27] Yolanda Sealey-Ruiz. 2007. Wrapping the Curriculum Around Their Lives: Using a Culturally Relevant Curriculum With African American Adult Women. *Adult Education Quarterly* 58, 1 (Nov. 2007), 44–60. https://doi.org/10.1177/0741713607305938 Publisher: SAGE Publications Inc.

[28] Sadia Sharmin. 2022. Creativity in CS1: A Literature Review. *ACM Transactions on Computing Education* 22, 2 (June 2022), 1–26. https://doi.org/10.1145/3459995

[29] Satra Taylor, Jay Holder, Bahiyyah Muhammad, Tiffany Jones, and Lawrence Haynes. 2021. Why Race Matters for Higher Education in Prison. *Peabody Journal of Education* 96, 5 (Oct. 2021), 588–597. https://doi.org/10.1080/0161956X.2021.1991704 Publisher: Routledge _eprint: https://doi.org/10.1080/0161956X.2021.1991704.

[30] V Taylor. 2016. Diversity gaps in computer science: Exploring the underrepresentation of girls, blacks and hispanics. *Google's Computer Science Education Research* (2016).

[31] Michael Williams and Tami Moser. 2019. The art of coding and thematic exploration in qualitative research. *International Management Review* 15, 1 (2019), 45–55.